Blockchain Financial Statements (BFS): A Transaction to Financial Statements Accounting System for Central Bank to Business Liquidity

A thesis submitted as partial fulfillment of the requirement of Doctor of Philosophy (Ph.D.)

by

Natalia Dashkevich

Computer Science Department

Brunel University London

—- —-

Abstract

This thesis introduces the concept of Blockchain Financial Statements (BFS), an innovative accounting system designed to bridge the liquidity needs between central banks and businesses through blockchain technology. The focus lies on streamlining the transaction to financial statement processes in a secure, transparent and efficient manner.

Context: Amidst evolving financial landscapes and the pressing need for more robust financial reporting mechanisms, the BFS framework emerges as a solution to enhance the liquidity management between central banks and businesses, underpinned by a more direct and verifiable approach to financial transactions and reporting.

Aim: The primary goal of this research is to design, develop and validate a blockchain-based accounting prototype - the BFS - that can transform traditional transaction data into comprehensive financial statements, facilitating central bank to business liquidity in a secure and efficient manner.

Methods: Employing a Design Science Research Methodology, this study constructs a BFS artefact, incorporating Domain-Driven Design to architect a system that harmonises with accounting standards, blockchain technology and business orchestration. The validation of the BFS artefact is conducted through a simulated environment, reflecting real-world transactional and accounting operations.

Results: The Java PoC implementation of BFS demonstrates a successful integration of blockchain technology into accounting practices, showing potential in real-time validation of transactions, immutable record-keeping and enhancing the transparency and efficiency of financial reporting.

Conclusion: The BFS framework and its PoC artefact, signifies an advancement in the application of blockchain technology for financial reporting and liquidity management. It offers a functional solution for enhancing transparency, accuracy and efficiency of financial transactions between central banks and businesses. The research underlines the necessity for further exploration into blockchain's potential within accounting systems, suggesting a promising direction for future innovations in financial reporting and liquidity management.

Acknowledgements

The journey of this research project and writing this Thesis has been a challenging yet profoundly rewarding adventure. It's an experience made possible by the unwavering support, encouragement and insight from numerous individuals, each of whom has contributed significantly to my academic and personal growth.

At the heart of this journey has been my supervisory team at Brunel University London. My profound gratitude extends to Professor Steve Counsell and Dr. Giuseppe Destefanis, whose unwavering support and belief in my abilities have been the bedrock of this project. I am deeply thankful for having the privilege of their mentorship. Their commitment to my growth have left an indelible mark on both my personal and academic development. Their insightful feedback, invaluable guidance, and continuous encouragement have illuminated my path, especially through the maze of administrative and health challenges I faced along the way. Their commitment to holding my hand throughout this process, uncompromisingly, have been a source of my strength and motivation, shaping not only this Thesis, but also my broader academic and professional outlook.

A special thanks goes to Allan Tucker, also at Brunel University London, for not only assembling an incredible and inspiring supervisory team, but also for being a source of inspiration and support numerous times throughout my research journey. I am also thankful to Dr. Stanislao Lauria, who recently joined my supervisory team as my Research Development Advisor, for his past insights that enriched my work. I am also eternally grateful to Crina Grosan for that pivotal phone call back in the spring of 2017, which set me on this enlightening path filled with challenges and discoveries and achievements.

Importantly, my heartfelt appreciation is extended to my family. Their unwavering support and encouragement have been invaluable throughout this journey. My PhD has been made possible through a scholarship, but the perseverance and dedication required were my own. To my husband, Anthony Richard Wardle, I am grateful for your encouragement and for the support you provided during my previous academic pursuits, which shaped my resilience and determination. To my daughter, Valeriia Dashkevich, your love and belief in me have been a source of strength, making this achievement even more meaningful

To all mentioned and to those who have walked with me throughout this process and have supported me in spirit, your contributions have left an indelible mark on my work and my life and I forever owe a debt of gratitude. Your collective wisdom, support and faith have been indispensable in this journey. Thank you.

Declaration

I hereby declare that all references to the Bank of England (or the Bank, or the BoE / BOE), the Asset Purchase Facility (or the APF), any aspects of central banking work or policy, as well as any use-case scenarios presented within this thesis, past or future academic publications related to this research project, are solely the product of my academic exploration. They represent my own thoughts and interpretations, which are entirely hypothetical, conceptualised, and constructed for the purposes of this research. These, or any other components, actors, elements, methodologies, processes, transactions or anything within this report, do not in any way represent or relate to my professional duties, insights, or experiences derived from my role or responsibilities at the Bank of England. All use-case scenarios discussed, along with their interpretations, are entirely hypothetical, crafted for academic purposes, and should not be construed as real or reflective of actual policies, strategies, or operational frameworks of the Bank of England or any related financial institutions. This declaration is made to clarify the independence of my academic research from my professional engagements and to affirm the speculative nature of the use-case scenarios and discussions presented in this work.

The source code for the BFS artefact is not publicly available. This decision is based on my current employment as a CBDC Architect at the Bank of England. Publicly releasing the code could lead to its misappropriation by media or the public, potentially resulting in incorrect assumptions regarding its relevance to the Bank's ongoing work on CBDC. Additionally, there is a significant conflict of interest, as the release of such code could create confusion or unintended consequences in relation to the Bank's current initiatives.

The following papers have been accepted or submitted/in preparation for publication as a direct result of the research discussed in this thesis:

- Natalia Dashkevich, Steve Counsell, Giuseppe Destefanis: Blockchain Application for Central Banks: A Systematic Mapping Study. IEEE Access 8: 139918-139952 (2020).
- Natalia Dashkevich, Steve Counsell, Giuseppe Destefanis: Blockchain Financial Statements: Innovating Financial Reporting, Accounting, and Liquidity Management. Future Internet 2024, 16(7), 244; https://doi.org/10.3390/fi16070244

Abbreviations

Abbreviation	Description
AI	Artificial Intelligence
AIS	Accounting Information Systems
AML	Anti-Money Laundry
APF	Assets Purchase Facility
BFS	Blockchain Financial Statements
BIS	Bank for International Settlement
BoE	Bank of England
BS	Balance Sheet
CB	Central Bank
CBDC	Central Bank Digital Currency
CFT	Commercial Paper Facility
CFT	Combating the Financing of Terrorism
COA	Chart of Accounts
CPI-1	Commercial Paper Issuer - 1
DDD	Domain Driven Design
DLT	Distributed Ledger Technology
DMO	Debt Management Office
DSRM	Design Science Research Methodology
DvD	Delivery versus Delivery
DvP	Delivery versus Payment
ECB	European Central Bank
FED	Federal Reserve
FinTech	Financial Technology
GFC	Global Financial Crisis
GJ	General Journal
GL	General Ledger
HMRC	His Majesty's Revenue and Customs
HMT	His Majesty's Treasury
HTLC	Hashed Time Lock Contract
I/S	Income Statement
IBPS	Interbank Payment Systems
IS	Information System
ISIN	International Securities Identification Number
IT	Information Technology
KYC	Know Your Customer
OECD	Organisation for Economic Co-operation and Development
OIS	Overnight Index Swap
OwP	One Way Payment
PCS	Payment Clearing and Settlement
PICO	Population Intervention Comparison Outcome
PoC	Proof of Concept
PoW	Proof-of-Work
QE	Quantitative Easing
RBAC	Role-Based Access Control
RQ	Research Question
RTB	Running Trial Balance
S&P	Standard & Poor's
SLR	Systematic Literature Review

SMS	Systematic Mapping Study
T/B	Trial Balance
TB	Treasury Bill
UTXO	Unspent Transaction Output
ZKP	Zero Knowledge Proof

Contents

List of Figures

List of Tables

Chapter 1

Introduction

An economic ecosystem is a complex network that consist of heterogeneous economic entities (i.e., businesses or business entities). These entities are interconnected *via* fundamental units of interaction - business transactions. In this complex and interdependent web of economic relationships, effective, accurate, timely, tamper evident and verifiable information flow is crucial to facilitate informed decision-making, creating feedback that can impact both the entities involved and the overall ecosystem. Financial statements provide a structured representation for the flow of economic activities and a current financial position report for each of these business entities. As such, they play critical role by provisioning a decision-making tool within this economic ecosystem by systematically presenting essential financial information. This information encompasses detailed insights into participants' financial outcomes, strategic approaches and future potential enabling ecosystem participants to pursue their individual economic objectives, such as maximising profits, reducing costs, or increasing market share. This standardised format ensures clarity and uniformity in conveying vital financial data. By doing so, financial statements foster transparency, accountability and contribute to effective decision-making, and the monitoring of these economic entities and overall ecosystem.

On the technological side, blockchain technology (or DLT) has emerged as a focal point of innovation across various sectors, drawing interest from a broad spectrum of communities. Blockchain, these days, is emerging as a truly disruptive force, extending its reach beyond Information Technology (IT) to encompass a multitude of other areas (Swan (2015), Beller & Hejderup (2019), Dashkevich et al. (2020)). Its potential applications span across law, real estate, banking, energy, and beyond, attracting attention from a multidisciplinary mix of academia, industry professionals, policymakers, regulators, public authorities, and financial market makers and participants (Judmayer et al. (2022), Besançon et al. (2019), Christidis & Devetsikiotis (2016), Lao et al. (2020)). This wide-ranging exploration reflects a collective effort to harness blockchain's capabilities to enhance operational processes, to develop coordinated solutions, and address existing challenges within various industries, by providing greater transparency and improving conduct (Ducas & Wilner (2017), Dashkevich et al. (2020)). Over recent years, the financial industry and wider diverse domains of the economy have started exploring various ways of leveraging blockchain.

The appeal of blockchain stems from its promise to provide a transparent, immutable, and cryptographically secure trail of digital events that can be shared, maintained, and verified by multiple participants and stakeholders. For example, the Bank for International Settlement (BIS (2017)) states that DLT adaptation could fundamentally change how assets are stored and their records maintained, obligations discharged, contracts enforced and risks managed (BIS (2017), Dashkevich et al. (2020)). The potential of blockchain to streamline business processes, foster transparency, and bridge trust gaps has ignited a reevaluation of traditional antiquated infrastructures and practices among economic actors (Hileman & Rauchs (2017), Ducas & Wilner (2017)). Central banks and financial institutions, alongside a broader community of researchers, are exploring how blockchain could redefine intermediated manual trust mechanisms, aiming to enhance financial data management (Dashkevich et al. (2020)). However, the overall potential of blockchain technology still remains to be fully unlocked (Ben Dhaou & Rohman (2018)), and there are various limitations to the existing purpose built blockchain architectures. Understanding the implications of such technology requires a multidisciplinary approach from the scientific perspective of academics together with policy-makers (Ben Dhaou & Rohman (2018)).

1.1 Problem Definition

The problem domain underpinning the work in this Thesis is situated within an economic ecosystem of heterogeneous entities, where transactions serve as the fundamental units of interaction amongst them. These transactions, while being the basic mode of such interaction, are part of a larger framework that includes individuals and institutions involved, adding layers of complexity and dynamism to the economic landscape. Financial statements play a central role in this context, offering a structured snapshot of economic activities and financial health report. That said, a retrospective (or historical) perspective provided by financial reports reflecting an entity's financial position and business performance up to a certain date in the past, or during some period in the past (Stolowy & Ding (2019)), may not consistently serve as a dependable source of the most current information about the organisation's financial "health", or its present state of the affairs. Financial statements often fail to incorporate economic changes that happen after the reporting period, i.e., following the conclusion of the latest accounting period. The lag in reporting can, in some situations can diminish their usefulness in providing a complete and up-to-date financial picture. Furthermore, despite existing accounting regulations and auditing procedures, the risk of intentional manipulation and fraudulent financial reporting remains. Through deliberate misrepresentation of the outcome of the transactional interactions or by misleading accounting disclosures, some adverse entities may resort to distorting transactional outcomes or engage in deceptive reporting practices, with intention of presenting a more favourable financial picture than is actually the case. The possibility of such fraud is amplified after the reporting period, by the absence of a reliable, verifiable, tamper-evident, and close to real-time auditability mechanisms that are focused on:

- ❖ a set of universally accepted verifiable authenticity controls that can guarantee integrity and this reliability of the aggregated history of accounting events, and
- ❖ a validity mechanism to verify financial drawdown requests or monetary claims made during transactional interactions.

An extensive example of such fraud can be illustrated using a following country-wide case in the United Kingdom (UK). The outbreak of the COVID-19 pandemic resulted in significant disruptions to financial conditions, necessitating urgent liquidity support for substantial businesses in the UK during periods of lockdown. In response, to support the flow of finance to UK companies, for example, the UK Government launched in excess of 150 financial support schemes (UK Government (2020)) in cooperation with the Bank of England (the BoE or simply the Bank) on the monetary side (Bank of England (2023*a*)), as "the Bank identifies its balance sheet as the most important tool for delivering monetary and financial stability through creation and management of public (central bank) money" (Bank of England (2023*a*)). To access this support, businesses were required to comply with pre-specified level for turnover or profit criteria, outlined in entity's financial statements, over a predetermined period time. Unfortunately, according to the Cabinet Office Minister Julia Lopez (UK Government (2021*b*)), the success of these initiatives was overshadowed by instances of accounting fraud. It was identified that such exploitation was driven by errors and evidence of gaps in provision of financial documentation, underpinned by challenges of verifying the legitimacy of claims made by the entities requesting funding support, as these requests were based on outdated or manipulated financial data. According to a HM Revenue and Customs (HMRC) report, the estimated value of fraudulent claims amounted to approximately £81.2 billion (UK Government (2021*a*)). The described real-world scenario underscores the urgent need for a system capable of bridging the divide between provision of public liquidity support and the assurance of data integrity and validity of the requests for such liquidity. Furthermore, these challenges highlight the critical need for overall improvement in how financial information's reliability and integrity are safeguarded.

The Design Science Research Methodology (DSRM) framework of Wieringa (2014) is an iterative, problem-solving process used widely in information systems (IS) and technology. It is particularly suitable for IS and software engineering, where innovation and practicality are key factors (Wieringa (2014)). This research approach is specifically driven by the need for creation and evaluation of an IT artefact intended to solve identified organisational problem (Wieringa (2014)).

Guided by the principles outlined in Wieringa (2014), the **Blockchain Financial Statements (BFS)** that this Thesis introduces, looks into the design and knowledge problems emerging from the current state of financial reporting and accounting practices. This research project is centred around confronting significant real-life challenges within the economic ecosystem: enhancing the trustworthiness and integrity of financial accounting and interactions between participants. It addresses the issues related to potential manipulation and fraudulent activities within financial statements, manual alteration of business data, leading to the misrepresentation of a company's financial condition. Furthermore, the study seeks to facilitate access to verifiable and secure financial information during the intervals between or after the reporting periods, thereby enhancing trust, transparency and reliability of financial information, whist maintaining confidentiality of business sensitive data. It tackles challenges of susceptibility of financial statements to manipulation and fraud, business data manipulation, misrepresentation of a company's financial state and enabling access to verifiable and tamper-evident financial data between reporting periods.

Furthermore, situated within a diverse social context, the research engages a broad spectrum of stakeholders, including economic entities, accountants and governmental bodies that can benefit from instantaneous, or on demand, auditability of financial data, ultimately contributing to a more transparent, accountable and efficient economic ecosystem. Enhanced timeliness and improved validity of financial reporting, will support the economic "well-being" for these stakeholders, impacting the overall ecosystem and advancing the state of knowledge and practice in IS and technology. Therefore, there are several critical problems that motivate the work in this Thesis:

Problem 1: In times of tightening of financial conditions, when fast liquidity support is required by heterogeneous economic entities, is it feasible to establish a direct and secure connection between these entities and public money liquidity providers (such as the government or a central bank (CB)) to facilitate automated, tamper-evident, and verifiable compliance with a set of arbitrary rules, designed for provision of such liquidity support? In this scenario, establishing a post-reporting period reconciliation mechanism for authentication of validity of financial claims made by a business entity is fundamental. This approach is required to guarantee a tamper-evident transactional data integrity, its chronological validity, and to provide specifications for its automation.

Problem 2: Given the crucial role of financial statements played in these circumstances, a second question emerges: Is it possible to establish direct connection between the financial statements of public money providers and diverse businesses entities? Furthermore, can this connection be enhanced with a mechanism that enables automated assurance of financial drawdown requests initiated by these businesses entities?

Problem 3: Investigating the role of blockchain within the BFS framework, highlights its distinctive technological attributes and capability to address systemic issues inherent in traditional financial accounting and reporting. The application of blockchain technology promises a substantial decrease in the risks associated with the manual methods prevalent in financial reporting, signalling a move towards more secure and dependable economic interactions. However, understanding the full scope of its capabilities, along with its current constraints, remains a subject of continuous inquiry (Dashkevich et al. (2020)). A critical question thus arises: How can blockchain technology be effectively utilised to design and implement a system for the distribution and validation of liquidity, leveraging on blockchain's data architecture and processes for the accurate recording and reporting of business activities?

Problem 4: The goal of creating a direct, secure, and tamper-evident connection between economic entities and public liquidity providers, such as governments or central banks drives this research. The exploration of innovative and practical adaptation of DLT underpins the presented Thesis' vision of a blockchain-enabled framework, does underscores the potential for such a system to automate compliance with financial support regulations, thus safeguarding the integrity and time-sensitive accuracy of transactional and accounting data. This introduces the core research question: Can a blockchain-based accounting system be designed and developed to facilitate secure and reliable liquidity distribution, utilizing the technology's strengths to log and disclose business operations in a way that addresses present challenges and risks associated with traditional financial accounting and transactional reporting of conventional systems?

1.2 Research Aim and Objectives

The BFS developed as an innovation against the backdrop of existing financial system inefficiencies is underscored by the COVID-19 pandemic's liquidity crisis. This section outlines the research's aim and objectives, underscored by a focus on leveraging blockchain technology to enable a direct, secure and tamper-evident channel for liquidity transmission from central banks to businesses. The research extends its focus to the challenges within accounting practices and system, particularly focusing on enhancing the efficiency, security and transparency of liquidity transmission from central banks to heterogeneous business entities, often required during times of recent financial crises.

The aim of the work in this Thesis can be summarised as follows: *The BFS aims to address accounting fraud, reduce data manipulation and the misrepresentation of company financial claims by enhancing the availability of real-time and tamper-evident accounting data.* The Thesis has five broad objectives:

Objective 1 ***Investigate the Impact of Blockchain on Business Models and Accounting.*** This objective involves exploration and understanding of the potential impacts from integration of blockchain technology with the conventional frameworks of financial accounting and the operational models of businesses, particularly how they transact, record and report economic activities. This objective involves examination of how blockchain's inherent characteristics, such as decentralisation, immutability and cryptographic security can innovate conduct of business operations and impact existing ways to conduct and track financial transactions. To understand the impact on the centralised nature of financial systems from central banks to the broader economic landscape. This objective will also involve a review of reported research effort on examination of capability of blockchain technology in solving issues of fraud within financial reporting. This examination intends to uncover how the technology's intrinsic properties can be leveraged to enhance the integrity and reliability of financial statements. This investigation will review existing literature, case studies and theoretical frameworks to assess the potential of blockchain for reducing fraudulent practices and ensuring the authenticity of financial data. This objective lays the groundwork for understanding the areas where blockchain technology is seen by the research community to make significant impact.

Objective 2 ***Explore the Application of Blockchain Technology in Central Banking.*** Given the key role of central banks in provision of public money, especially in times of crisis, this study will next analyse and map research effort between practitioners and academics on the potential of blockchain application for services, operations and functions performed by central banks. This exploration will have a specific emphasis on types of central bank use-cases considered for blockchain adaptation, with the goal of understand opportunities and challenges for central banks arising from blockchain adaptation and potential (if any) gaps for further exploration. By exploring how blockchain technology can be employed within central banking frameworks, this investigation will demonstrate the theoretical potential of blockchain to be employed as facilitator for secure, transparent liquidity distribution channel, utilised to support economic entities in times of need.

Objective 3 ***Design and Develop a Blockchain Financial Statements (BFS) Artefact.*** Drawing from the insights gained through the exploration of blockchain applications, this objective is to conceptualise, design and develop a BFS framework that demonstrates application of blockchain in creating a more reliable and verifiable financial reporting system. This involves the architecture for the BFS artefact and its technological implementation that demonstrates real-time or on demand verification of monetary claims underpinned by the tamper-evident controls designed for blockchain-based accounting. This addresses the problem of miss-allocation of financial support, from public money providers such as central banks and the government, driven by vulnerabilities to fraud in traditional accounting systems.

Objective 4 ***Contribute to the Body of Knowledge on Blockchain Applications in Finance and Accounting.*** By documenting the findings, challenges and successes encountered throughout the research project, this objective aims to contribute valuable insights to the academic and professional communities interested in blockchain technology's potential within finance and accounting. By investigating and demonstrating the BFS artefact's capabilities, the study contributes valuable insights into leveraging blockchain for financial reporting and transaction management. By answering knowledge questions and confronting the empirical reality with theoretical claims, the research aims to generate new knowledge and insights into the blockchain's potential in improving accounting practices.

Objective 5 ***Provide Practical Implications and Recommendations for Stakeholders.*** This objective is to offer actionable insights and recommendations for various stakeholders involved in the economic ecosystem, including businesses, financial institutions, regulatory bodies and FinTecs on adopting blockchain technology to enhance financial infrastructure. These recommendations will be grounded in the findings from the BFS artefact's design and development, aiming to inform future initiatives and innovations in the field.

Objective 6 ***Validate the BFS Artefact Through Simulation and Testing.*** The final objective focuses on the validation of the BFS PoC artefact to assess its operational capabilities and theoretical alignment with real-world scenarios. The aim is to ensure that the artefact meets stakeholder goals by critically examining its functionality, accuracy, and effectiveness in mitigating accounting fraud and enhancing transparency. The validation process will involve a series of controlled experiments, manual testing, scenario-based testing, and functional correctness tests. This objective aims to quantify the BFS artefact's performance against established benchmarks and to evaluate its ability to process financial transactions, execute smart contracts, and generate accurate financial reports. By accomplishing this objective, the BFS framework's theoretical design will be compared against practical results, establishing its effectiveness in addressing challenges in modern financial reporting. The insights gained from this validation exercise will be important in determining the BFS artefact's utility in real-world applications, ensuring its readiness for further refinement and eventual deployment in financial accounting practices.

1.3 Thesis Contributions

This section of the report aims to articulate the theoretical and practical contributions anticipated from the design and implementation of the BFS. The contributions of this research enhance the fields of blockchain technology, financial reporting and central banking, thereby setting a precedent for future studies and applications.

As the result of the research, this work sees three key ***theoretical*** contributions:

Contribution 1: ***Enhanced Understanding of Blockchain Applications in Central Banking.*** The BFS contributes to the theoretical framework by extending understanding of how blockchain technology can be applied to solve issues in financial reporting and central banking. It extends the current knowledge base, offering insights into the utility of blockchain for central banking for enhancing transparency, security and efficiency in financial transactions, reporting and expanding on use-cases for *Assets transfer and ownership (Assets)* and *Audit trail (Audit)* for central banks.

Contribution 2: ***Innovative Framework for Blockchain-based Accounting.*** By developing and demonstrating a framework for implementing accounting practices on BFS smart contracts for coordination of accounting processes, off-ledger repositories to store and access accounting ledgers and blockchain data structure to publish financial statements, the research introduces an innovative approach to automating and securing accounting processes using blockchain technology. This contribution furthers the discourse on blockchain applications in accounting and sets the groundwork for developing standardised blockchain accounting protocols. It provides valuable insights and guidelines for stakeholders interested in leveraging blockchain for financial management, reporting, and central banking functions.

Contribution 3: ***Exploration of Consensus Mechanisms for Financial Audibility.*** The introduction of a novel "funds verification" consensus mechanism within the BFS artefact offers a unique contribution to the theoretical exploration of consensus models tailored for financial auditing and real-time transaction verification. This advancement enhances the understanding of how blockchain consensus can be customised to meet specific industry needs.

From a ***practical*** perspective, this work sees four further contributions:

Contribution 1: ***Real-time Financial Reporting and Fraud Mitigation.*** The BFS artefact addresses the practical challenges of historical financial reporting driving accounting fraud by enabling real-time audibility and tamper-evident verification of transactional records. This practical solution contributes to a reduction in the risk of fraudulent financial reporting and improves the timeliness and reliability of financial information.

Contribution 2: ***Streamlined Liquidity Support Distribution.*** In response to the urgent need for efficient and secure mechanisms to distribute financial support, as highlighted during the COVID-19 pandemic, the BFS artefact offers a blockchain-based solution to automate and verify claims for provision of financial support from public sources of liquidity. This practical contribution ensures timely and accurate delivery of funds to eligible businesses, mitigating the risks of fraudulent claims.

Contribution 3: ***Facilitation of Secure Multi-Blockchain Transactions.*** Through the implementation of HTLC and atomic swaps and step-by-step granular demonstration of transactional and accounting flows, the BFS extends existing practices of secure and efficient cross-blockchain transactions. This contribution has practical implications for enhancing interoperability and flexibility in blockchain applications, allowing for seamless value transfer across different blockchain networks.

Contribution 4: ***Promotion of Transparency and Efficiency in Financial Management.*** By automating accounting entries and ensuring the tamper-evident recording of transactional data, the BFS artefact contributes to the practical advancement of financial management practices. By automating the creation of accounting entries, it reduces manual intervention, enhances processing efficiency and minimises the likelihood of errors. It enables organisations to achieve higher levels of transparency, accuracy and efficiency in their financial operations.

1.4 Ethical Issues

Ethical approval was sought from Brunel University's Research Ethics Committee and the research was approved - see Appendix A. This research notes that no data will be collected and analysed from humans as part of this research.

1.5 Thesis Structure

The thesis is delivered in nine chapters; there is a brief description of each chapter below.

Chapter 1: Introduction

Chapter 1 serves as the gateway to this Thesis, laying the groundwork for an in-depth exploration of the research problem, which focuses on enhancing the integrity and efficiency of financial reporting through blockchain technology. This chapter outlines the primary aims and objectives designed to guide the study, leading to the development of a Blockchain Financial Statements (BFS) PoC technological artefact. Additionally, the Thesis's contributions chapter highlights key insights this research offers. Ethical considerations are addressed, ensuring that the study's conduct and its implications adhere to the highest standards of academic and professional integrity. This introductory chapter sets the stage for a comprehensive investigation into the transformative potential of blockchain in financial reporting.

Chapter 2: Background and Literature Review

Chapter 2 provides into the foundational knowledge and academic research that underpin this study, offering a comprehensive overview of both traditional and innovative elements within the financial and technological realms. It begins by exploring the broader economic landscape, highlighting the roles of non-bank entities and central banks, with a focus on entities like the Bank of England and the Asset Purchase Facility. The discussion extends to foundational concepts in accounting, covering the accounting cycle, the structure and significance of financial statements, and the operational dynamics of blockchain technology, including a comparative analysis of Bitcoin and Corda architectures. Additional attention is given to the innovative financial mechanisms such as Atomic Swaps with Hash Time-Locked Contracts (HTLC), and their implications for transaction security and efficiency. The literature review chapter synthesises existing research on the transformative impact of blockchain on accounting and business models, and investigates blockchain's application within central banking contexts through a systematic mapping study. Finally, this chapter sets the stage for the BFS artefact's practical application, anchored by a guiding scenario that illustrates its intended use-case, thereby providing a structured foundation for the BFS implementation and validation explored in subsequent chapters.

Chapter 3: Research Methodology

Chapter 3 articulates the research methodology, employing the Design Science Research Methodology (DSRM) to steer the conceptualisation, development, and validation of the BFS architecture. This chapter systematically dissects the DSRM approach, highlighting its vital components from the design and development to the rigorous validation processes, ensuring the BFS architecture's alignment with real-world functional requirements and stakeholder expectations. It further describes practical application of DSRM within the BFS project, elaborating on the artefact's design, its operational context, and the dynamic interactions it facilitates. Essential to this methodology is the integration of a comprehensive knowledge base, detailing the artefact's functional requirements and identifying key stakeholders, thereby laying a foundation for the BFS framework's conceptual and technical blueprint. This approach not only guides the BFS development but also ensures its practicality and effectiveness in addressing the complex needs of blockchain-based financial reporting and transaction management.

Chapter 4: The BFS Architectural Approach

Chapter 4 privies the architectural framework for BFS's, employing Domain-Driven Design (DDD) to navigate the complex integration of blockchain technology with established accounting practices, reporting standards, and business orchestration. This architectural strategy segments the BFS into distinct yet interrelated domains: Financial Accounting, Blockchain, and Business Entity, where each designed to address unique aspects of the BFS prototype. By adopting DDD, the project ensures that business logic is not only encapsulated within domain-specific models but also reflected through granular understanding of the interactions between these. Moreover, the chapter explores the dual blockchain framework, highlighting how the BFS leverages both Bitcoin and Corda architecture and components to balance transactional integrity with operational efficiency, and privacy with accountability. This holistic approach underscores the project's commitment to creating a robust, scalable, and adaptable architectural foundation for the BFS, positioned to innovate financial reporting and management practices through novel blockchain adaptation.

Chapter 5: The Architecture of BFS

Chapter 5 unveils the architecture of the BFS, dissecting the frameworks that underpin financial accounting, blockchain technology, and business entity dynamics. Through a detailed exploration of each sub-domain, this chapter highlights the essential actors involved, the core elements that constitute the backbone of the BFS, and the mechanisms that facilitate seamless interactions across the system. By exploring the architecture's specifics, the chapter demonstrates how the BFS is specifically tailored to meet the integration demands of blockchain within the financial accounting realm, aligning closely with the operational requisites brought forth by the Asset Purchase Facility use-case - the APF::BFS. This analysis not only illustrates the BFS's conceptual design, but also showcases its capability to innovate the financial accounting landscape through blockchain adaptation, ensuring a comprehensive understanding of its operational efficacy and strategic value.

Chapter 6: BFS Implementation and Proof of Concept
Chapter 6 presents a granular practical implementation of the Blockchain Financial Statements through demonstration of key components of its overall architecture and their functionalities within designated contexts. This examination underlines the fusion of traditional financial accounting processes with innovative blockchain technology, within APF::BFS scenario. The discussion spans the comprehensive incorporation of financial accounting mechanisms, including the organisation of accounts, ledgers, financial reporting, and the role of accountant within the BFS ecosystem. Furthermore, it explores the blockchain sub-domain's integration, showcasing the BFS block and its filling history ledger structure, smart contract applications, and the novel approaches to ensuring transactional integrity and security. The entity sub-domain's implementation is articulated, covering company incorporation, business-specific component, and the facilitation of business transactions, thereby offering a holistic view of the BFS's operational capabilities. This chapter aims to provide a vivid illustration of how the BFS architecture not only aligns with but enhances the operational needs of integrating blockchain into financial accounting, emphasising its potential applicability and impact in real-world settings.

Chapter 7: BFS Smart Contract Data Flow

Chapter 7 navigates through the granular data flow within the BFS smart contract landscape. It begins by presenting the initial share settlement transaction, where tokenized shares are issued and sold by the APF to the BoE, employing Delivery versus Payment (DvP) with Hash Time-Locked Contracts (HTLC) for enhanced security and trust. The narrative progresses to articulate the integration of economic data from this transaction into the BFS's accounting framework, ensuring precise record-keeping within the General Journal and Ledger, laying the groundwork for transparent financial reporting. Additionally, the chapter describes a loan drawdown request mechanism within the BFS, a crucial step enabling the APF to engage in Quantitative Easing (QE) activities, facilitated through a streamlined, smart contract-based process. The subsequent discussion on the atomic four-party QE transaction underscores the BFS's capacity to automate asset acquisitions by the APF, showcasing the system's adeptness in handling complex financial transactions. This presentation not only demonstrates the BFS's practical application, but also aligns with established operational procedures of the APF, offering a comprehensive view of how smart contract technology can facilitate financial transactions, reporting, and QE implementation within the BFS ecosystem.

Chapter 8: BFS Validation

Chapter 8 provides validation of the BFS artefact, adopting a rigorous DSRM to validate its practicality and functionality in addressing real-world financial reporting and management challenges. It outlines the validation approach where the BFS is subjected to a simulated environment that reflects its intended use, ensuring the artefact's interactions and outcomes are thoroughly examined against stakeholder expectations and research objectives. It presents structured validation framework that encompasses design theory principles, interactions within the problem context, a set of validation criteria, and validation methods aimed at ensuring the BFS's alignment with both theoretical constructs and operational demands. Analysis of validation results offers discussion on the BFS's capability to innovate financial reporting through blockchain technology and meet stakeholder functional requirements.

Chapter 9: Conclusions and Further Work

The final Chapter 9 brings together insights and reflections derived from the research, focusing on the interpretation of findings within the context of the BFS. This concluding chapter synthesises the theoretical contributions, practical applications, and policy implications emerging from the

BFS implementation, highlighting the potential impact on the financial reporting landscape. It also addresses the study's limitations, proposing directions for future research that could further refine and expand upon the BFS concepts. The chapter wraps up with concluding remarks that underscore the significant strides made towards integrating blockchain technology into financial reporting, and also look forward to the evolving role of blockchain in reshaping financial practices. This forward-looking perspective sets the stage for ongoing exploration and innovation in the domain of blockchain-based accounting systems.

Chapter 2

Background and Literature Review

This chapter provides an exploratory journey through various foundational elements critical to understanding the complex interplay between blockchain technology and financial accounting systems. It is structured to provide a comprehensive background on several key areas, followed by a literature review that examines the nuances of blockchain's impact on accounting and business models, as well as its potential applications within central banking frameworks.

The chapter initiates with a foundational background on non-bank entities, offering insights into the wide range of organisations that operate within the financial ecosystem but outside the conventional banking framework. It proceeds to describe the function and impact of central banking institutions, with a particular focus on the Bank of England (BoE) and the Asset Purchase Facility (APF), highlighting their critical role in monetary and financial stability. Then it will examine accounting practices and provide insights into traditional financial reporting mechanisms, setting the stage for discussions on innovation. Next, an introduction to blockchain technology provides a foundational understanding of its principles, capabilities, and limitations. The chapter will contrast the architectural frameworks of Bitcoin and Corda, highlighting key differences and their implications for financial applications. The concept of atomic swaps and Hash Time-Locked Contracts (HTLC) is explored as well, demonstrating their utility in ensuring secure and conditional transaction execution in blockchain networks.

Building upon the background, the literature review chapter explore the state-of-knowledge first on the impact of blockchain technology on accounting and business models. This chapter assesses how blockchain's attributes such as immutability, transparency, and decentralisation could revolutionise traditional practices. Next, the theoretical work will provide a comprehensive report on: "Blockchain for Central Banks: A Systematic Mapping Study" presenting an overview of the scholarly discourse on the potential applications of blockchain technology in central banking operations, emphasising the exploration of Central Bank Digital Currencies (CBDCs), payment clearing and settlement systems, and regulatory compliance.

Lastly, this chapter introduces the Guiding Scenario for BFS Artefact Implementation - The Use-Case, providing a practical context that bridges theoretical insights with empirical application. This narrative thread not only enriches the discussion but also sets a foundational basis for the subsequent exploration of the Blockchain Financial Statements (BFS) artefact, demonstrating the practical application and implications of integrating blockchain technology within the realm of financial accounting and liquidity management.

2.1 Economic Entity

In the context of financial reporting and accountability, the concept of economic entities is central, acting as the foundation for understanding and categorising various organisational structures that participate in economic activities. Economic entities are organisations or units that independently engage in economic activities, make decisions and manage resources (Stolowy & Ding (2019)).

These entities are categorised into various types based on their roles and functions within the economy. For both central banks and non-bank entities, financial statements serve as fundamental tools for financial transparency and accountability. They provide critical information that helps stakeholders assess an entity's performance, financial position and future prospects (Stolowy & Ding (2019)). In doing so, financial statements contribute to the efficient allocation of resources, the stability of the financial system and the enhancement of investor and public confidence in the economic ecosystem. In summary, economic entities, whether central banks or non-bank entities are integral components of the economic landscape. Their activities and the subsequent reporting on these activities through financial statements play a vital role in maintaining transparency, fostering trust and ensuring the smooth functioning of the economy.

2.1.1 Non-Bank Entities

Non-bank entities encompass a broad spectrum of organisations engaged in economic activities outside the traditional banking sector. This category includes corporations, small and medium-sized enterprises (SMEs), non-profit organisations and government agencies. Non-bank entities participate in the economy by providing goods, services, or both and their operations directly influence economic growth, employment rates and market dynamics. Financial reporting is crucial for non-bank entities as it offers stakeholders a comprehensive view of the entity's financial health, operational results and cash flows (Stolowy & Ding (2019)). These entities are mandated to prepare and disclose financial statements in accordance with relevant accounting standards of their respective jurisdictions. The primary financial statements include the balance sheet, income statement, statement of cash flows and statement of changes in equity (Stolowy & Ding (2019)). These reports enable stakeholders (including investors), creditors, regulators and the public—to make informed decisions regarding their engagement with the entity (Stolowy & Ding (2019)).

2.1.2 Central Bank

Central banks, such as the Bank of England or the Federal Reserve in the United States, play a critical role in a country's economy. They are responsible for implementing monetary policy, issuing currency and overseeing the banking system to ensure financial stability.

In this Thesis, the focus is on central banks, but it is important to start by understanding the role of wider banking, as this should help to determine where and how innovative the blockchain technology can potentially fit (Dashkevich et al. (2020)). According to Dashkevich et al. (2020) and Casu et al. (2006) banks, as other financial intermediaries, play a pivotal role in the economy by channelling funds from units in surplus to units in deficit. They reconcile the different needs of borrowers and lenders who do not know and do not trust each other. They transform small-size, low-risk and highly liquid deposits into loans which are of larger size, higher risk and illiquid. The banking industry is broad and combines sectors related to central banking, investment, corporate, commercial, retail banking etc., differing by their business models and performance goals. More specifically, a central bank, a reserve bank or a monetary authority is a financial institution that manages domestic money supply, interest rates and oversees a country's broader banking system. According to Hayes (2016), some functional dimensions that set a central bank apart from other banks are that a central bank is a monopoly note issuer, the government's banker, the lender of last resort, and, in some cases, serves as a clearing house for settlement of payments - it is the banker's bank (Hayes (2016)). For example, as a clearing house, a central bank on a larger wholesale money market scale reconciles the funding needs of the commercial bank's participants, each of whom might have different business goals and do not trust each other. The other dimension is that a central bank must maintain a non-competitive stance and not seek profit maximisation. Most central banks also have supervisory and regulatory powers to ensure solvency of member institutions (Mills et al. (2016)) and are seen in many jurisdictions as the keeper of economic health, usually independent of the government and trusted to deliver public interest and overall economic welfare (Dashkevich et al. (2020), Chiu (2017)).

In broad terms, a central bank, reserve bank, or monetary authority is a financial institution responsible for managing a country's domestic money supply, interest rates, and overseeing the broader banking system. These institutions, considered financial intermediaries, play a crucial role in the economy by facilitating the flow of funds from units of surplus to units of deficit. It is important to note that during times of macroeconomic crises, these monetary authorities often assume a systemically critical role of the lenders of last resort. They may also employ unconventional monetary policy measures such as negative interest rates or Quantitative Easing (QE). In relation to the central banks' evaluation and research on application of blockchain to the accounting and auditing, Dashkevich et al. (2020) suggests that blockchain can provide trusted, time-oriented, immutable and shared databases. These enable regulators to monitor, supervise and audit reported transactional data, where a blockchain based global audit log ensures the integrity of records through the integrity of the ledger itself. Furthermore, by automating and streamlining blockchain enabled reconciliation processes for multi-party, multi-infrastructure interactions, central banks can reduce inter-mediation costs and risks by guaranteeing consistency across multiple data repositories (Dashkevich et al. (2020)).

Bank of England and Asset Purchase Facility

The Bank of England (BoE) and the Asset Purchase Facility (APF) are central to the financial infrastructure of the United Kingdom, playing critical roles in the country's economic stability and monetary policy implementation. Their roles in ensuring monetary and financial stability through liquidity provision are essential for the functioning of the economy. The research in the Thesis focus on these entities and the exploration of blockchain technology's application to their operations. It underscores these roles in facilitating financial support to economic entities and serving as a prime illustration of a direct top-down approach for liquidity provision from a central bank to the broader non-bank economic sector.

The BoE was established in 1694. It is tasked with a multitude of critical functions, including:

1. *Monetary Stability.* Responsibility for issuing the national currency (the British Pound Sterling) and managing the UK's monetary policy, primarily focusing on maintaining price stability.
2. *Financial Stability.* Overseeing the UK's financial system's stability, aiming to safeguard against systemic risks that could lead to financial crises.
3. *Regulator and Supervisor.* Acting as a regulator and supervisor for the banking sector, ensuring that financial institutions operate safely and soundly, protecting the economy and consumers.

Asset Purchase Facility (APF): A Tool for Monetary Policy - is a fund created by the BoE in January 2009 during the global financial crisis; it facilitates quantitative easing (QE) — a monetary policy used to stimulate the economy when standard monetary policy tools have become ineffective. The APF operates by purchasing financial assets, including government bonds and corporate debt from the private sector. These purchases inject liquidity directly into the economy, lowering interest rates on bonds, encouraging spending and investment. The APF's roles include:

1. *Stimulating Economic Activity.* By purchasing assets, the APF helps lower interest rates and increase the money supply, stimulating economic activity.
2. *Enhancing Liquidity.* The APF provides crucial liquidity to financial markets, ensuring that businesses and government can finance their operations and investments.
3. *Lowering Interest Rates.* By purchasing government bonds, the APF helps to lower the yields on those bonds, which typically results in lower interest rates across the economy, stimulating investment and spending.

The rationale behind inclusion of the BoE and its subsidiary, the Asset Purchase Facility in this research is driven by their instrumental roles in providing financial support to the UK's economic entities. This project focuses on exploring the BFS's potential to facilitate a direct, top-down approach for liquidity provision from the central bank to a wider array of non-bank economic entities. The research explores the potential for leveraging blockchain technology to streamline and enhance the efficiency of how central banks, like the BoE, or its subsidiary the APF, can pass financial support directly to non-bank economic entities. Specifically, the research in this Thesis aims to:

- *Illustrate Direct Financial Support Mechanisms.* By leveraging the BFS, the work highlights how central banks like the BoE can utilise advanced blockchain technologies to implement direct liquidity provision mechanisms to the economy.
- *Showcase Efficiency and Transparency.*The use of blockchain technology in the context of the BoE and APF operations can enhance the efficiency, transparency and traceability of financial support transactions, ensuring that funds reach intended recipients promptly and securely.
- *Innovative Financial Support Distribution.* The research proposes that the integration of blockchain with the operational frameworks of entities like the BoE and APF can revolutionise how financial support is distributed, ensuring that such support is timely, targeted and meets the liquidity needs of the economy more directly and efficiently.
- *Demonstrate a Novel Approach to Monetary Policy Implementation.* The project aims to show how the integration of blockchain technology with central banking functions can offer innovative approaches to monetary policy implementation, particularly in terms of liquidity provision and economic stimulation.

The BoE and the Asset Purchase Facility are the most important institutions in the UK's financial and economic landscape. Their specific roles in managing monetary policy, ensuring financial stability and stimulating economic activity through liquidity provision are crucial for the country's economic health. The focus of the Thesis on these entities underscores the potential for blockchain technology to revolutionise how central banks interact with and support the broader economy. By examining the direct top-down approach for liquidity provision facilitated by the BFS, the research contributes to the ongoing discourse on innovative monetary policy tools and the future of financial support mechanisms.

2.2 Accounting and Financial Statements

Accounting is a globally accepted standardised tool for accountability and reporting (Stolowy & Ding (2019)). It is a process of identifying, capturing, managing, analysing, interpreting and distributing transactional data; the economic activity of a business. Accounting records every even of the economic nature of a business life-cycle, in the form of economic variables, principally expressed in monetary units (Stolowy & Ding (2019)). It performs recording of each elemental transaction, classification and analysis its data, with the goal of chronological recording and periodic creation of reports/financial statements (Stolowy & Ding (2019)). These reports monitor what an economic entity does with its resources and what claims exist, at any point in time, on the resources of the business (Stolowy & Ding (2019)).

The domain of financial accounting is pivotal for capturing and presenting the economic activities and financial health of organisations through structured documentation known as financial statements. These documents serve as a collection of descriptors for various economic events, primarily business transactions, within a firm's operational life cycle (AccountingTools (2021), Corporate Finance Institute (2021)). Financial accounting involves a systematic process that leads to the preparation of four key financial statements:

- Income Statement - also known as the profit and loss statement, outlines the company's revenues and expenses within a specific period, leading to the calculation of net income or loss. It reflects the company's profitability by comparing earned revenues against incurred expenses.
- Balance Sheet - provides a snapshot of the company's financial position at a specific moment,

detailing assets, liabilities and equity. This statement showcases the solvency of the business at that particular point in time.

- Statement of Cash Flows - tracks the cash inflows and outflows from operating, investing and financing activities over a period. It offers insights into the company's cash generation and spending patterns, highlighting its liquidity and financial health.

These statements are essential for stakeholders, including but not limited to economic entities, company members, customers and accountants. They provide a comprehensive view of the company's financial status, supporting informed decision-making and strategic planning.

At the heart of financial accounting lies the concept of transactions, which are business events with a monetary impact (Stolowy & Ding (2019)). These transactions form the foundational blocks of financial records and are categorised into accounts for detailed tracking. This classification includes various common accounts like cash, accounts receivable, inventory and equity (among others), providing a structured overview of a company's financial standing (Stolowy & Ding (2019)). The process of financial accounting involves recording these transactions, classifying them into appropriate accounts, summarizing the data into financial statements and analysing these statements to interpret the company's financial health (Stolowy & Ding (2019)).

In this research, the exploration is centred on the demonstration of high-level process of execution of accounting. This abstracted scenario illustrates how the economic data generated from business transactions is anonymised and subsequently transformed into accounting data. This data transformation process is vital for the preparation of financial reports, specifically the Balance Sheet, which provides a snapshot of a company's financial position at a given point in time. The Balance Sheet, or statement of financial position, as a fundamental component of financial reporting. The approach to anonymizing and processing transactional data into a Balance Sheet is designed to showcase the potential of blockchain and smart contract technologies for enhancing the privacy, accuracy and efficiency of financial reporting. This exploration sheds light on the technical mechanisms behind the data transformation and emphasises the importance of maintaining data integrity and privacy in the financial reporting process. By focusing on the Balance Sheet, the research aims to demonstrate the practical application of these technologies in a critical area of financial management, laying the groundwork for future research.

The implementation of other financial statements, such as the Income Statement, Statement of Cash Flows and Statement of Changes in Equity, is acknowledged as an area for future optimisation. These documents are equally important in providing a comprehensive view of a company's financial health and performance over a time period.

2.2.1 Accounting Cycle

The accounting cycle is a fundamental concept in financial management, representing a series of steps undertaken to ensure accurate and comprehensive recording and reporting of an entity's financial activities (Stolowy & Ding (2019), NetSuite (2022)). Depending on requirements, one can find the accounting cycle described in 4 steps, 5 steps, or even 10 steps. However, the general consensus is that there are 8 steps. This process starts from the initial identification, collection and analysis of business transactions, leading to the recording of these in journals and posting to the general ledger etc. It progresses through the preparation of trial balance, adjustments and the creation of adjusted trial balances, culminating in the preparation of financial statements, communication of these and closing of the books for the accounting period (NetSuite (2022)).

The accounting cycle is a comprehensive process employed by businesses to ensure their business transactions are accurately recorded, processed and reflected in their financial statements. Such adaptation of the accounting cycle is illustrated in Fig. 2.1. The BFS cycle includes the following eight steps:

1. *Identifying and Analysing Business Transactions.* The financial accounting cycle only recognises transactions that have or will have monetary implications, i.e., which result in an exchange of funds at some point (Stolowy & Ding (2019)). These transactions are collected in

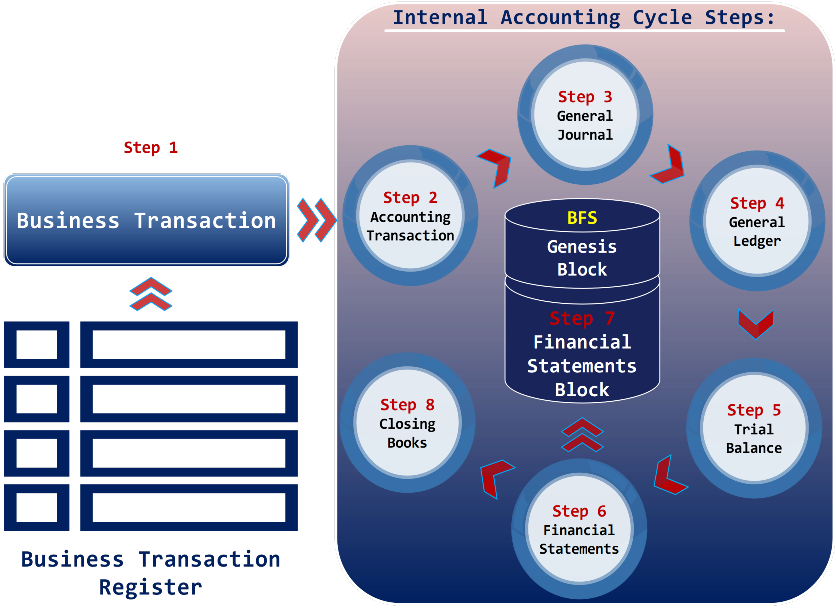


Figure 2.1: Steps of the Accounting Cycle

"Business Transaction Register" repository for further processing within the same accounting period, assessing their impact on the company's finances.

2. *Activation of Accounting Transaction Smart Contracts.* This step involves gathering economic descriptors about business transactions. This is facilitated through accounting smart contracts designed to enhance efficiency and reduce errors, making the whole process more streamlined and reliable. These accounting smart contracts define accounting rules to govern transformation of economic data produced by business transactions into an accounting data format. This process involves adherence to a set rules for recognition of which financial accounts will be impacted as the outcome of each business transaction.
3. *Recording Transactions in a General Journal.* After identification, the transformed data from business transactions is recorded chronologically as journal entries in the company's books, starting from the General Journal (GJ).
4. *Posting Transactions to the General Ledger.* Entries from the GJ are then posted to the General Ledger (GL), a master record that provides a complete overview of all financial accounts and the evolution of their values.
5. *Generating Entries Running Trial Balance.* At the end of the accounting period (continuously or on demand) ending balances from general ledger are automatically calculated, ensuring accuracy and reducing the potential for errors or manipulations. Only these final balances are posted to the running trial balance, illuminating details from the GL account. This step is necessary to ensure that debits equal credits. It is essential for analysis, error identification and corrections, ensuring accuracy of the financial records, before financial statement preparation. The resulting final trial balance is used to construct the financial statements - the end product of the accounting reporting.
6. *Generating Financial Statements.* Adapting this step to the BFS architecture involves automating the generation of the final balance sheet from the finalised running trial balance. The Balance Sheet is a critical financial statement summarizing a company's financial position at a specific point in time, detailing assets, liabilities and shareholders' equity. By automating the generation of the Balance Sheet, the BFS ensures accuracy and reduces the potential for human error, facilitating a more efficient and reliable financial reporting process.
7. *Publication of the Financial Statements on Blockchain.* The penultimate step in the BFS accounting cycle is the publication of the Balance Sheet into the blockchain financial statement data structure, referred to as the filing history of the BFS. This integration into the

blockchain ledger secures the financial information against unauthorised alterations and enhances the auditability of financial statements. By leveraging blockchain's inherent properties of immutability and transparency, this step ensures that financial data is readily verifiable and accessible, thereby providing confidence among stakeholders and regulatory bodies.

8. *Closing the Books.* Finally, temporary accounts are closed out to permanent accounts, like retained earnings and the company prepares for the next accounting cycle.

The adaptation of traditional accounting cycles to incorporate blockchain technology through the BFS addresses several critical problems within financial reporting and management. This innovative approach solves issues related to accounting fraud, data manipulation and the timeliness and reliability of financial data. By leveraging blockchain's inherent properties of immutability, transparency and security, the BFS ensures that all financial transactions are accurately recorded, processed and verified in real-time or on demand. This enhances the integrity and trustworthiness of financial statements and facilitates a more efficient and transparent audit process. Consequently, the adaptation contributes to a more stable, secure and equitable economic ecosystem, where stakeholders can confidently rely on the financial information presented, fostering a higher degree of accountability and trust among all parties involved.

2.3 Blockchain

In this Thesis, the terms *Blockchain* or *Distributed Ledger Technology (DLT)* are used synonymously. Although there is a thematic difference between those terminologies through their underlying architecture, it has become common practice in the industry to combine all those meanings under the same umbrella term. According to Hileman & Rauchs (2017) and Dashkevich et al. (2020), at its narrowest possible definition: "A blockchain is a special data structure - a database - that is composed of transactions, batched into blocks, that are cryptographically linked to each other to form a sequential, tamper-evident chain events that determines the ordering of transactions in the system. In this context, a transaction represents any change or modification to the database" (Hileman & Rauchs (2017)). More broadly, blockchain is a type of peer-to-peer (P2P) distributed network of independent participants that generally broadcasts all data to each other, each of whom may have different motivations and objectives. They may not necessarily trust one another, but reach a consensus (a consistent agreement about changes to the state of the shared database) on a linear history of operations of that shared database (Dashkevich et al. (2020), Hileman & Rauchs (2017)). A high-level ecosystem topology (workflow) of traditional blockchain is presented in Fig. 2.2 (Dashkevich et al. (2020)):

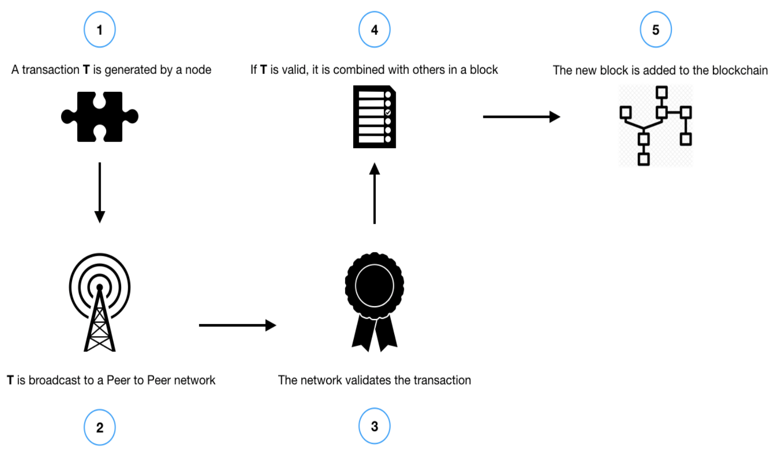


Figure 2.2: How a blockchain works

Some additional aspects of blockchain, that are of particulat relevance to this research project are the Unspent Transaction Output (UTXO) model and a Consensus mechanism.

UTXO is a fundamental concept in blockchain technology, used in cryptocurrencies such as Bitcoin. In its traditional UTXO model, transactions are represented by inputs and outputs, where the outputs of one transaction become inputs for future transactions. UTXOs refer to the portion of the cryptocurrency that remains unspent after a transaction and can be used in new transactions. This model ensures that no double-spending occurs, as each UTXO can only be spent once, and it serves as a traceable unit of account within the blockchain ecosystem (Nakamoto (2008)).

Furthermore, *consensus mechanisms* are trust facilitating protocols used in DLT networks, that are designed to achieve agreement on the validity of transactions and maintain a consistent state of the ledger across all nodes. The most common consensus algorithms include Proof of Work (PoW), as used in Bitcoin (Nakamoto (2008)), where miners compete to solve cryptographic puzzles to validate transactions. However, Corda, which is the basis for this research, takes a different approach, where it is designed to be more suited for business and financial applications. In Corda, consensus is focused on ensuring that only parties to a transaction agree on its validity, rather than requiring global consensus across the entire network, as in systems like Bitcoin. Each transaction must be valid according to the rules of the system. This is checked by ensuring that all required signatures are present, that the transaction follows the prescribed rules, and that all inputs are valid. Additionally, each transaction must be unique, meaning no double-spending or conflicting transactions. Uniqueness is guaranteed by notaries — trusted parties or nodes that confirm that a transaction has not been previously committed (R3 (2023), Hearn & Brown (2016)).

Other key advantages of blockchain, in comparison to existing distributed systems and database technologies, is in the use of a specialised data structure which bundles transactions into blocks and/or the broadcast of all data to all participants, in its automated reconciliation mechanisms, together with its resilience and transparent nature (Dashkevich et al. (2020), Hileman & Rauchs (2017)). Some of the main components of a blockchain are: cryptography, P2P networks, consensus mechanisms, the ledger, validity rules and access or permission types. There are general permission type distinctions for current blockchain architectures:

- *Permissionless, public or open* refer to blockchains where access is not restricted to a specific set of vetted participants (Dashkevich et al. (2020), Hileman & Rauchs (2017)). In these types of blockchain, participants do not know and trust each other, so "good" behaviour is incentivised through the existence of a native token;
- *Permissioned, private or closed* refer to blockchains where access is restricted to a specific set of vetted participants (Dashkevich et al. (2020), Hileman & Rauchs (2017)). These blockchains operate in an environment where participants are already known, vetted and there is a level of trust amongst them; this removes the need for a native token to incentivise good behaviour. Participants are held liable through off-chain legal contracts and agreements and are incentivised to behave honestly *via* the threat of legal prosecution in the case of misbehaviour (Dashkevich et al. (2020), Hileman & Rauchs (2017)).
- *Consortium or federated* refers to a blockchain where the architecture could be private or hybrid (public and private) (Dashkevich et al. (2020), Sun et al. (2017), Ducas & Wilner (2017)). This type of DLT uses features such as: permission restriction, multiple controlling authorities; they allow easy, yet controlled information sharing between various stakeholders and more.

Although the study have identified a small number of research studies on the potential application of permissionless blockchain for business of central banks (Dashkevich et al. (2020), Nguyen (2016), Didenko et al. (2020)), the predominant consensus amongst the research community is that the permissioned access model is the preferred type of blockchain by such institutions (Chiu (2017), Tsai, Deng, Ding & Li (2018), Chapman et al. (2017), Chiu & Koeppl (2019), Lipton (2018), Milne (2018), Priem (2020), Mills et al. (2016), Wang et al. (2018), Sun et al. (2017), Benos et al. (2017), BIS (2017), Chapman et al. (2017), Kumhof & Noone (2018), Barrdear & Kumhof (2016), Auer & Böhme (2020), Didenko et al. (2020), Dashkevich et al. (2020)). The Consortium or federated blockchain access type was not available in included peer-reviewed publications on DLT applications for the business of central banks (see later in this chapter for mapping study details).

2.3.1 Bitcoin vs Corda Architectural Differences

In the landscape of blockchain technologies, two significant frameworks stand out due to their distinctive approaches to security, privacy and scalability: Bitcoin (Nakamoto (2008)) and Corda (R3 (2023), Brown (2018), Brown et al. (2016)). These frameworks embody contrasting models within the blockchain ecosystem, each tailored to serve unique requirements within financial and transactional roles.

Bitcoin's Architecture. Bitcoin (Nakamoto (2008)), the pioneer of blockchain technology, utilises a decentralised, peer-to-peer architecture to facilitate digital transactions without the need for a central authority. Its design is centred around the concept of a fully public ledger, which records all transactions across all network participants in a transparent and immutable manner. This ledger, or blockchain, is maintained by a consensus mechanism known as Proof of Work (PoW) (Nakamoto (2008)), which requires miners to solve complex computational puzzles to validate transactions and create new blocks. Furthermore, although Bitcoin's architecture ensures anonymity of all participants in the network and full transparency, it does faces challenges with transactional privacy. Every participant has access to the entire transaction history, which raises these concerns about privacy, together with addition of issue of scalability. This process ensures the integrity and security of the network and also introduces challenges such as high energy consumption and scalability limitations due to the time and computational power required to process transactions.

Corda's Architecture. Corda, on the other hand, adopts a more tailored approach, as it is designed as a permissioned blockchain platform, focusing on the needs of businesses, particularly in the financial sector (R3 (2023), Hearn & Brown (2016)). It enables direct point-to-point messaging, to guarantee private transactions, ensuring transactional data remains confidential between parties involved with verified identities, thus addressing the privacy concerns inherent in public blockchains like Bitcoin (Hearn & Brown (2016)). Although it is not a traditional blockchain, because it does not maintain a global ledger of all transactions, it is still a distributed ledger technology (DLT) enabling businesses to transact directly and privately with each other, minimizing unnecessary data sharing (Hearn & Brown (2016), Brown (2018)). Instead, it allows only the parties involved in a transaction and those with a need to know to access the transaction's details. This is achieved through the use of Corda's "Flow Framework" for transaction processing, which facilitate direct communication between parties and complex transactional workflows (R3 (2023)). Corda records transactions in individual "vaults" - "Transaction Vaults", crucial for maintaining this transactional privacy while allowing traceability for compliance and transparency (R3 (2023), Brown (2018), Brown et al. (2016)). Furthermore, Corda's design supports the development of "CorDapps" (Corda Distributed Applications) and design and implementation of a wide variety of "smart contracts" that can be customised for various financial services, ensuring compliance and offering scalability and privacy (Brown (2018)) that the Bitcoin network cannot directly provide.

Integration into BFS Artefact. The BFS artefact leverages the strengths of both Bitcoin and Corda to address the challenges of financial reporting and transactional privacy within a unified innovative framework - "Dual Blockchain Framework" (see Section 4.3.1), to address the challenges of financial reporting and transactional privacy. This innovative integration utilises Bitcoin's robust, tamper-evident ledger design whereby the BFS can securely store financial statements in the BFS "Filing History" (see Section 4.3.1), ensuring the integrity, immutability and cryptographically linked chronology of financial data. Concurrently, the incorporation of Corda's privacy-centric transaction mechanisms and flow framework allows for confidential, direct transactions and efficient compliance verification, thereby addressing the critical need for transactional privacy and efficiency in a business context (see Section 4.3.1). This allows entities within the BFS ecosystem to engage in direct, secure exchanges without exposing sensitive business information to the entire network.

This dual-framework consolidation blockchain approach presents a comprehensive solution for the BFS that this study have develop for this Thesis, ensuring the secure, private and efficient management of financial statements and transactional data within a decentralised ecosystem. The nuanced integration of Bitcoin and Corda architectures within BFS underscores the potential of blockchain technologies to revolutionise financial processes. It highlights the importance of selecting the appropriate blockchain framework based on the specific requirements of privacy, security and scalability within the financial industry. The integration demonstrates a forward-thinking methodology for addressing the current limitations in blockchain applications for financial and transactional systems, providing ground for additional innovation and for further research and development in blockchain-based financial reporting and transaction management.

2.3.2 Atomic Swap and Hash Time-Locked Contracts (HTLC) in Transactions.

An *atomic swap* transactions, such as use-case based Delivery versus Payment (DvP) for share sales between entities such as the APF (representing the share issuer) and the BoE (representing the share purchaser) within the BFS's ecosystem involves *simultaneous exchange of shares for payment, ensuring that both sides of the transaction are fulfilled concurrently.*

In a traditional financial setting, DvP is a securities settlement mechanism where the delivery of securities occurs only if the corresponding payment is received. This ensures that the seller only delivers the securities if they receive the payment and the buyer only pays if they receive the securities. The atomic swap integrates this concept into the environment of blockchain and cryptocurrencies, allowing for a trustless exchange, meaning that the two parties do not need to trust each other or a third party for the transaction to occur. The key characteristics of an atomic swap in the BFS would be likely to include:

Smart Contracts. The transaction is facilitated by smart contracts, which are self-executing contracts with the terms directly written into lines of code. These smart contracts automatically enforce the conditions of the DvP.

Hash Time-Locked Contracts (HTLCs). A feature of atomic swaps, where a cryptographic hash function secures the transaction and a time lock ensures that the transaction is completed within a specific time-frame. If either party fails to confirm the transaction within the time lock, the transaction is automatically voided and assets are returned to their respective parties.

Interoperability Channel. If different blockchain platforms are used, the atomic swap must ensure a communication mechanism between the two. This concept aligns with the BFS, which is a multi-entity ecosystem of BFS, with each of these entities owing a separate BFS and business translations implemented between them.

Decentralisation. The swap occurs directly between the APF and BoE without the need for a centralised intermediary, reducing counterparty risk and enhancing transaction security.

Privacy and Security. Transaction details are verifiable and the resulting economic data generated by these transactions is immutable when posted within the BFS filling history data structure; privacy, but not the anonymity of transacting counterparties entities is maintained. Cryptographic methods ensure that sensitive financial details are not exposed publicly.

Transaction Verification. The BFS "funds verification" consensus validates transactions. This validation process ensures the verification of the availability of obligations, i.e., that the shares and funds are available and locked in until both parties fulfil their obligations, at which point the swap occurs instantaneously.

Finality and Irreversibility. Once the atomic swap is completed, it is final and irreversible, providing certainty of settlement to both the APF and the BoE. Implementing such a transaction within the BFS would require a robust blockchain infrastructure capable of supporting these features, ensuring that all regulatory and security standards are met. This approach to

DvP would significantly reduce the settlement time, lower transaction costs and potentially eliminate the need for traditional clearing-houses.

2.4 Guiding Scenario for BFS Artefact Implementation - The Use-Case

This section of the research explores the specific real-world scenario that informed and guided the design and implementation of BFS artefact. It is anchored in a key phase of recent global economic developments, the narrative focuses on measures adopted in response to the Global Financial Crisis (GFC) and the subsequent economic impacts of the COVID-19 pandemic. The establishment and operational evolution of the Bank of England Asset Purchase Facility Fund (APF) plays a central role in this discourse, illustrating the dynamic landscape of economic intervention mechanisms.

The use-case for the BFS project is grounded in a real-life, chronological, and sequential flow of events, specifically reflecting the establishment of the Asset Purchase Facility (APF). This approach ensures that the BFS framework reflects realistic and practically relevant processes, providing an accurate representation of business activities within a blockchain-based financial system. Specifically, all steps and business processes in the use-case are adapted from the actual incorporation of the APF, starting with the formal establishment of the APF as a business entity. This includes the issuance of shares as part of its initial capital structure. The sequence proceeds to replicate the first quantitative easing (QE) transaction conducted by the APF. The QE process, in which the APF purchased assets such as commercial paper from the private sector, is abstracted within the BFS framework. Each stage of the QE transaction — from asset acquisition to the flow of funds between counterparties — has been modelled to demonstrate the BFS's capabilities in managing real-life complex liquidity flow, processing business transactions, and integrating blockchain technology into traditional accounting activities. This mirrored adaptation allows the BFS to demonstrate how blockchain can facilitate financial reporting, provide transparency, and reduce fraud risks in processes closely aligned with those of the APF's actual operations.

The backdrop of this use-case that is marked by GFC leads to the incorporation of the APF on 30th January 30, 2009 as a direct subsidiary of the BOE. This strategic establishment was geared towards executing Quantitative Easing (QE) measures on behalf of the BoE, aiming to mitigate the economic downturn(Bank of England (2009*b*)). The APF's creation signified a decisive step towards injecting public liquidity into the economy to stabilise fluctuating markets. As the global economic landscape evolved, particularly with the advent of the COVID-19 pandemic, the APF's operations adapted and evolved to meet new emerging economic challenges. This adaptability underscores the APF's continuous relevance in economic stabilisation efforts and public liquidity distribution.

The foundational role of the APF, acting as the BoE's legal market counterparty for transactions, facilitated the acquisition high-quality private sector assets, such as sterling investment-grade commercial paper (Bank of England (2009*a*)). To support these APF's activities, funding was essential. The program's early stages were funded through the issuance of Treasury Bills(Bank of England (2009*b*)), which were then lent to the APF through the BoE (Bank of England (2010*b*)). This critical mechanism enabled the APF to manage its "near-term cash flow requirements" efficiently, underlining the integrated approach between the BoE and the APF. Treasury Bills were advanced (deposited) to the APF by the BoE upon receiving a notification of the APF's intention to make a draw down under the loan (Bank of England (2010*b*)). Such arrangements enabled the APF to perform effectively in operational activities by addressing public liquidity requests from the market participants.

2.4.1 Simulation Scenario for BFS Artefact

The Blockchain Financial Statements (BFS) artefact's design, development and implementation are significantly informed by a detailed simulation scenario. This scenario draws heavily on the operational backdrop and foundational mandate of the APF, particularly its role in executing market transactions and facilitating QE on behalf of the BoE.

As the contextual foundation for the BFS project, at the heart of this simulation scenario is the APF's establishment and its operational dynamics, where the APF engages in the purchase of high-quality assets to inject liquidity into the financial system. This strategic operation forms the core narrative for the BFS artefact simulation, providing a tangible, real-world process around which the blockchain technology's application is explored and assessed. By simulating the APF's activities, the BFS artefact is tested against the complexities of real-life financial operations, offering insights into how blockchain can be effectively harnessed to:

- ❖ Enhance the reliability and accuracy of financial statements.
- ❖ Minimise the risks associated with manual and conventional methods of financial reporting.
- ❖ Provide a verifiable, secure, and direct channel between liquidity providers and economic entities.

Central to this use-case is the detailed demonstration of transactional dynamics among three key economic entities: the APF, the BoE, and a hypothetical market participant referred to as "Commercial Paper Issuer - 1" (CPI-1). This setup aims to showcase the capability of the BFS in facilitating and recording of processes for business transaction and an accounting transaction, facilitated by adaptation of blockchain's smart contract framework to ensure data integrity from transaction generation through to financial statement consolidation:

1. **Business Transaction** *Initial Share Settlement Business Transaction (DvP with HTLC)*: Initiated immediately upon the BFS's instantiation, this transaction involves the APF issuing shares to the BoE. An exploration of the first business transaction smart contract executed between the APF and the BoE, showcasing how the APF issues and sells tokenized Shares to the BoE (see Section 7.2).
2. **Accounting Transaction** *Accounting Transaction Smart Contract*: This process automates the flow of transaction-generated data through the traditional accounting cycle, culminating in the real-time publication of financial statements on the blockchain ledger.Details on how economic data from the Initial Share Settlement Business Transaction is recorded into the APF::BFS's accounting systems, ensuring accurate documentation in the General Journal and General Ledger(see Section 7.3).
3. **Business Transaction** *Drawdown Request on Loan Agreement with HTLC*: Although not a stand-alone transaction, this part illustrates how a drawdown request can be implemented within the BFS framework to facilitate subsequent QE activities by APF. It is a One-Way-Payment, execution of which is implemented in as a sequential step of the complex atomic three-party QE business transaction, described next(see Section 7.4).
4. **Business Transaction** *Atomic Four-Party QE with HTLC*: A section on the implementation of the Quantitative Easing (QE) process, highlighting how the BFS facilitates asset acquisitions by the APF through smart contract automation. It is a Delivery vs. Delivery (DvD) and One-Way-Payment transaction, demonstrating the BFS's smart contract's capability to streamline automate QE process, especially in asset acquisitions by the APF (see Section 7.5).

The BFS architecture, demonstrated through the lens of the APF, positions it as the primary entity engaging in transactions with the BoE. This exploratory framework is restricted to a select nsactions within a specified timeframe, treated as discrete events within distinct accounting periods. This meticulous approach provides a comprehensive examination of the BFS functionality, from the genesis of transactions to the culmination in financial statement generation.

The BFS artefact utilises this adopted scenario to showcase the practicality of blockchain in addressing current challenges in financial accounting and reporting. It leverages components, data structure and some operational mechanisms of DLT to create a framework where every transaction and accounting entry is instantaneously recorded in a manner that can be immutable, secure, and easily verifiable. This not only aims to reduce the incidence of fraud and misreporting but also enhances the overall trust in financial disclosures.

The centre of this project's exploration is the development of a JAVA Proof of Concept (PoC) technological prototype of the BFS. This prototype examines how blockchain technology can be utilised to minimise fraudulent activities within the sphere of accounting, by simulating a structured economic environment that mirrors real-world financial interactions.This prototype encapsulates the flow of liquidity through a network of business transactions, each marking a significant point of interaction within the economic fabric. To bring this vision to together, the BFS mirrors the automated journey of transactional data through the accounting cycle, culminating in the generation of financial statements that are published in blockchain data structure. This is achieved through the innovative use of smart contract technology, which enables automated flow of data generated from business interactions directly into the BFS filing history.

In conclusion, the BFS artefact's design and implementation are rooted in the operational specifics of the APF, serving as a narrative for exploring blockchain's transformative potential in the financial and accounting domains. It provides a foundation for evaluating how a blockchain-enabled accounting system can address the challenges of traditional financial reporting and pave the way for more secure, transparent, and accountable financial ecosystems. Through this simulation scenario, the BFS artefact emerges not just as a theoretical innovation but also illustrates the practical application of blockchain in enhancing the integrity and reliability of financial transactions and reporting, marking a significant step towards the future of financial accounting.

2.4.2 Use-Case Steps

The BFS unfolds within a modelled ecosystem, encapsulating the APF and the BoE, charting the operational chronology of the use-case. The journey begins with the establishment of the APF, moving through the progressive structuring of the BFS framework to the directors' appointments, underscoring the APF's operational autonomy despite its close ties with the BoE.

The BFS artefact's architecture and its practical demonstration are presented through the lens of the APF, which assumes the role of a principal business entity engaging in transactions with counterpart entities, specifically the BoE and CPI-1. Such use-case adaption underscores the BFS artefact's ability to automate the compliance with regulatory and financial support measures, ensuring the integrity and timely accuracy of both transactional and accounting data. By demonstrating a blockchain-based system that can efficiently manage liquidity distribution and validation, the BFS artefact aligns with the broader goal of this research to innovate financial reporting. For the sake of clarity and simplicity in this demonstration, the research concentrates on minimal number of transactions. Each is treated as the sole transaction within its own distinct accounting period, despite the brevity of the time frame considered. This approach is adopted purely for illustrative purposes, to ensure clear and focused demonstration and validation of the functionality of BFS artefact through simulated lab modelling.

The definition of the behavioural diagram is a necessary first step in the design of the architecture for a complex system, because it enables identification of the key elements for the system design.

In Fig. 2.3, a behavioural diagram that represent primary actors (as different players in the project's scenario), elements and their use-case related actions are illustrated. The system of the use-case is defined as the middle ground (what one is building) across which the actors interact. These are:

Actors: APF directors, the Registrar, the BOE, the APF, and the CPI-1;

Elements: all issued wallets representing business roles of wallet owners; the APF::BS.

Textual use-case description for the behavioural diagram in Fig. 2.3 are described next. The

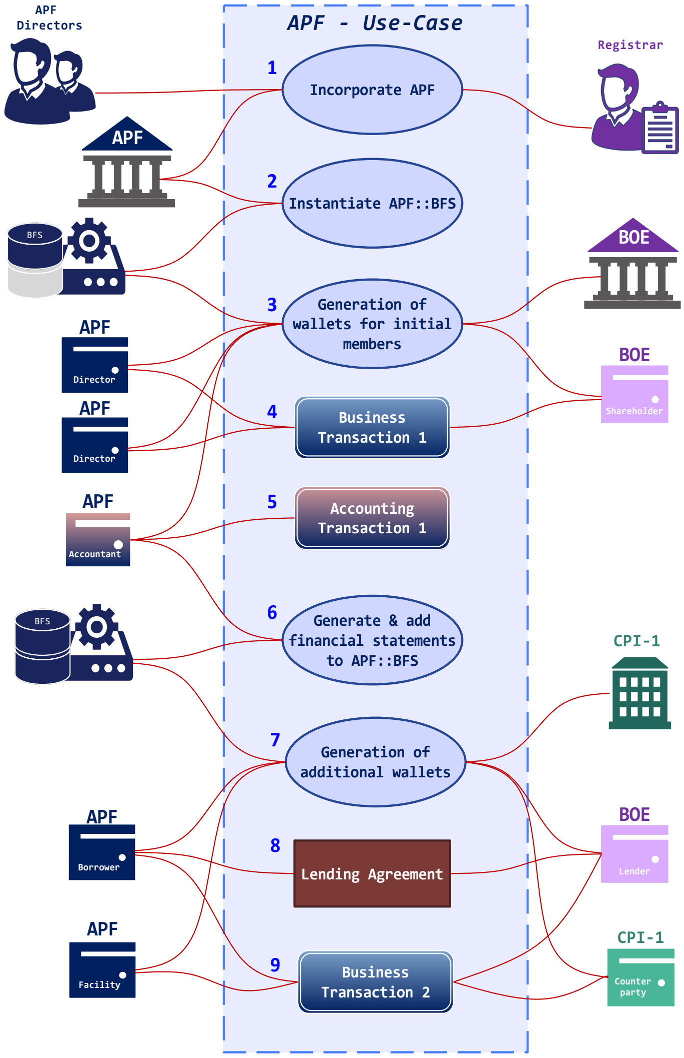


Figure 2.3: Behavioural diagram for the APF use-case step.
The figure illustrates actors, elements, and sequential interactions involved in the APF Use-Case within the BFS system. The diagram is structured to show key actors on either side of the system and their interactions through various transactional and accounting activities. Actors on the left include APF Directors, while the Registrar, the BoE, and the CPI-1 are on the right. The placement of actors on either side of the system does not signify a difference in hierarchy, but rather their involvement in distinct roles within the system. The colours of the entities mapped with corresponding roles and activities. The centre section (contained within the central dashed rectangle) represents interactions, such as business and accounting transactions and activities for the sequential flow of the use-case. Rectangular boxes indicate business and accounting smart contracts. Oval boxes represent all other sequential business activities of the practical BFS use-case demonstration. Sequential numbering aligns with key steps in the use-case (e.g., APF incorporation, share issuance, and commercial paper transactions), which follow real-world chronological events of APF's creation and operations.

sequential numbers within the system body of the APF use-case Fig. 2.3 represent the top level sequential points from 1 to 9 in this textual description.

1. APF incorporation - 30 January 2009 APF is incorporated by providing required documentation to the registrar in "Companies House"
   → As a fully owned subsidiary of the BoE but has its own financial statements;
   → To incorporate APF, as a private company limited by shares, two perspective directors of the APF filled-out electronic version of form `IN10` and submitted this form to the

`Registrar` - a hypothetical officer of the “Companies House” UK.
  - → From the registrar, the directors of APF receive its incorporation package with new company number and all relevant documentation;
2. APF::BFS instantiation:
   - → Upon receiving the `PrivateRegister` instance from said `Registrar`, containing incorporation documentation, the directors are able to instantiate their `APF::BFS` as a “private company limited by shares” Companies House (2023*a*), and commence their economic activity.
   - → Ordinary Shares are issued as tokens at instantiation of APF::BFS and 100% of these will be purchased by the BOE during the “first business transaction of the APF”;
3. Generation of wallets for initial members of the APF
   - → two director wallets
   - → a shareholder wallet
   - → a hypothetical accountant wallet
4. First business transaction of the APF - sale of the APF tokenized shares to the BOE for the payment of £100.00 of digital funds.
   - → The transaction is between the APF (the share issuer), and the BoE (the shareholder);
   - → The transaction is an atomic swap of shares vs payment (DvP), a delivery vs payment transaction type.
5. First accounting transaction:
6. Generation of the accounting Block and appendage of this Block to the BFS ledger (filing history):
   - → The concluding data from accounting ledgers is packaged in the data structure of the accounting block type of the BFS;
   - → This block is appended to the APF::BFS::filingHistory ledger.
7. Generation of additional wallets to enable diversity of business roles
   - → Borrower - the APF’s member - a hypothetical employee of the APF;
   - → Lender - the BoE as the counterparty to the APF.
   - → Commercial Paper Facility - commencement of the “Commercial Paper Facility” (CPF) with first transaction on 13 February 2009. The initial goal of CPF is to purchase investment grade CP from its issuer, and finance this purchase with TB.
   - → Hypothetical market participant - a counterparty to the APF: “Commercial Paper Issuer - 1” (CPI-1). An entity in need of liquidity support from public money provider. Issues Commercial Paper (CP) with the goal to sell it to the APF as the collateral for liquidity support.
8. Establishment of the lending agreement between the APF and the BOE to enable provision of funding for QE implementation by the APF. All market transactions of the APF are financed by the loan (deposit) from the Bank. Demonstration of liquidity claim by the APF to the BoE, its verification and fulfilment by the BoE, with the subsequent transfer of this liquidity (the TBs) from the BoE to the APF.
   - → Inception of the lending agreement between the APF and the BoE to borrow Treasury Bills (TBs);
   - → Establishment the ability to make draw downs on that loan based on the liquidity requirements of the APF.
   - → Liquidity claim - APF requires funds to pay for CP;
     - → Claim verification - APF demonstrates using its financial statements that they do not have liquidity required to pay for CP;
     - → Fulfilment of liquidity claim - the BoE transfers TB to the APF, which in turn allows APF proceed and successfully execute first market transaction of purchasing CP;
   - → Although in the real use-case, the TB are issued by the HMT, lent to the BoE, who then lends it to the APF, for simulation model, a simplified adaptation was modelled,

where it was assumed that the TBs were "issued" and provided by the BoE.

9. The second business transaction between the APF and the CPI-1 and the APF and the BoE - acquisition of commercial paper from the CPI-1 by the APF with the simultaneous provision of Treasury Bills (TBs) from the BoE to the APF to finance this purchase of commercial paper. This market transaction between the APF and the CP issuer is executed as a DvP atomic swap;
   - → The APF, using atomic swap for delivery vs delivery (DvD) transaction type, purchases CP in the primary market (at issue). CPI-1 issues CP, sell it to APF and APF pays for CP with request of simultaneous issue and transfer of TBs.
   - → This purchase is financed by the Treasury Bills (TBs) issued by BoE;
   - → These TBs are acquired by the APF through drawdown request from the APF to the BOE based on the lending faculty agreed prior between them. This transaction matches: i. Issue and maturity days; ii. Nominal values; iii. Discount rates applied in accordance with market notice 2009; iv. Face values of these securities.
   - → BoE lends (transfers) requested TBs to the APF - executing one-way-payment type of the transaction;
   - → When in possession of the APF, these TBs are swapped for illiquid Commercial Paper (CP) issued by hypothetical market participant CPI-1.

Each step of this process demonstrates a near-instantaneous disclosure and it is not merely procedural but forms an integral component of the BFS's technological validation provided in Chapters 6 and 7, showcasing the practical and real-time benefits of blockchain application in the domain of financial reporting.

2.4.3 Sourcing the Data for Current BFS Implementation

For the purpose of accurately replicating the progression of events and the interplay of transactions within BFS use-case, a chronological sequence was reconstructed based on historical accuracy.The design and implementation of the BFS is built upon publicly available information published by the BoE (Bank of England (2023*b*), Bank of England (2010*a*)) and the Asset Purchase Facility Fund (APF) (Companies House (2023*a*), Companies House (2023*b*), Bank of England (Year)),

All designed and implemented transactional components and operational processes, demonstrated within complex payment chains for this BFS business transactions, together with relationships between them and operational processes to orchestrate these are modelled on the APF's *Operating Procedures for the Asset Purchase Facility* (Bank of England (2024*a*)), *Terms and Conditions of the Asset Purchase Facility* (Bank of England (2024*b*)), *Bank of England: Settlement Procedures for the Asset Purchase Facility* (Bank of England (2010*c*)), and *Bank of England: Market Notice 24 April 2009* (Bank of England (2009*d*)).

2.4.4 Practical Implementation Considerations for a Potential User of the BFS

This section outlines for of the key considerations for practical implementation. While the current BFS is at the proof-of-concept stage, understanding how a user could eventually adopt the system is important for framing its future development.

The first step for any potential user, such as a financial institution, company, or government body, would be conducting a thorough feasibility study. This involves assessing the organisation's current financial reporting processes and determining how blockchain-based systems such as BFS could enhance existing business and accounting process. The study should additionally evaluate technical and operational compatibility of BFS with the organisation's existing systems.

The BFS framework, as developed in this thesis, offers a generic structure. To implement it, as the next step, a user would be required to customise BFS based on specific regulatory, operational, and financial reporting requirements of their jurisdiction. This may include adjusting the data structures, modifying smart contracts to fit local legal frameworks, and tailoring existing financial reporting templates to meet industry-specific standards. Moreover, integrating BFS with existing legacy systems (such as enterprise resource planning (ERP) systems, accounting software, or financial reporting tools) would require development of APIs and/or middleware to facilitate data exchange between BFS and traditional databases.

Once the system's customisation is complete, the next step involves extending and setting up required blockchain infrastructure. Depending on the scale of implementation, users would decide whether to adopt a fully private blockchain (as the BFS prototype uses), or explore hybrid approaches. This step includes setting up BFS Filing History blockchain, nodes, managing BFS wallets for stakeholders, adopting consensus, and configuring security protocols.

Additionally, the choice between platforms (e.g., Corda, Hyperledger, Ethereum, or novel architecture of the time) will impact implementation, as these platforms differ in terms of privacy, scalability, and functionality. BFS is built with Bitcoin Corda in mind, but a potential user will need to consider platform interoperability or compliance with the existing technical environment.

At this stage it is also important to consider security and compliance. Security is a central aspect of any blockchain implementation. The BFS adaptation will need to ensure the encryption of sensitive business and accounting data, as well as establish extended identity management systems to control access to business sensitive information of an economic entity. This would involve extend implementation of existing role-based access control and implement multi-factor authentication for users, along with auditing trails to prevent fraud.

A potential user must also ensure that BFS complies with local and international data protection laws, such as GDPR in Europe. This includes addressing concerns about the immutability of blockchain data, which could conflict with the "right to be forgotten" or similar regulations.

Blockchain-based financial reporting systems introduce new concepts and workflows to traditional accounting processes. To implement BFS successfully, organisations will need to conduct comprehensive training sessions for financial professionals, accountants, and auditors. This ensures that stakeholders understand how to use BFS smart contracts, interpret blockchain-based financial statements, and handle the decentralised nature of the system.

Additionally, before full-scale deployment, extensive testing of the BFS in a sand-boxed environment is essential. This stage includes stress-testing the system for scalability, conducting mock transactions to validate accounting outputs, and ensuring that BFS smart contracts perform as expected. Potential users would also need to work closely with auditors to confirm that the BFS generated financial reports meet regulatory and compliance standards.

Lastly, once testing is successful, the BFS system can be deployed across an organisation. Continuous monitoring and maintenance would be necessary to ensure system performance, identify potential bottlenecks, and apply patches or updates to BFS smart contracts as legal and regulatory frameworks evolve. Regular system audits would also help maintain data integrity and compliance.

While the BFS framework is currently at the proof-of-concept level, its future implementation requires careful planning, customisation, and security considerations. Potential users will need to integrate it into existing infrastructures, ensure compliance with regulatory standards, and prepare for training and adoption challenges. Practical implementation of BFS presents an opportunity for organisations to enhance the accuracy and transparency of their financial reporting, but it must be approached with the necessary preparatory and testing phases before full-scale adoption can be achieved.

2.5 Impact of Blockchain on Accounting and Business Models

A Business Model (BM) is a concept in management studies (Dashkevich et al. (2020), Holotiuk et al. (2017), Al-Debei & Avison (2010)). Although a specific definition has still to be found (Holotiuk et al. (2017), Wirtz et al. (2016)), a BM has been identified as the "story" that explains how an enterprise works (Holotiuk et al. (2017), Magretta (2002)) and also as the way firms do business – i.e., the rationale of how an organisation creates, delivers and captures value (Holotiuk et al. (2017)). A BM represents an intermediate layer – the link between a firm's strategy, processes and information technology (IT) (Dashkevich et al. (2020), Holotiuk et al. (2017)). The major cornerstone of any business's operations is its business model as it is a foundational aspect of a business's functionality, particularly the processes surrounding payments, transactions and the conversion of an entity's economic activity into financial statements, serving as a critical tool for analysis and decision-making within the business environment (Dashkevich et al. (2020), Holotiuk et al. (2017)). Financial transparency and a sound reliability on financial reporting are increasingly crucial, especially in fragile economic climates where businesses compete for funding. The significance of understanding these processes lies in their ability to provide stakeholders with essential insights into the entity's financial health, guiding informed decision-making and strategic planning.

Blockchain technology is increasingly recognized as a transformative force in accounting and business operations, with research indicating its potential to enhance transparency and security in financial transactions. Blockchain innovation has the potential to enhance legacy infrastructures (Dashkevich et al. (2020), Rio & César (2017)) that surround economic activity, by creating new and improved blockchain-based business models, which in itself is believed to be one of the major factors behind the push for DLT adoption by industry (Dashkevich et al. (2020), Rio & César (2017)). These will allow for a fundamentally different way of conducting and tracking financial transactions and could thus challenge the centralised nature of existing financial systems, starting from central banks (Dashkevich et al. (2020), Rio & César (2017)) and down to diverse participants in the economy.

Blockchain-based accounting is gaining increasing attention from both industry and academia (Nadine (2017), Hans (2019), Peters & Panayi (2016), David (December 2015)). According to Nadine (2017), accounting emerges as a significant sector standing to benefit from the adoption of blockchain technology. The inherent immutability feature of blockchain is expected to facilitate tamper-evident creation and maintenance of permanent and timely records of transactional data (David (December 2015)), thus enhancing reliability of record-keeping. This attribute of blockchain also offers a robust capability of detection to the network participants of any manipulation or alterations of recorded transactional data (Andersen (2016)), thus discouraging improper accounting practices, transactional data manipulation and mitigating fraud (Andersen (2016), Mei et al. (2011)). The immutability and decentralisation of transparency (Centobelli et al. (2022)) afforded by blockchain presents opportunities for verifiable data access and sharing, close-to-real-time reporting and transactional history verification. Improved transparency minimises data asymmetry between stakeholders (Centobelli et al. (2022), Bonson & Michaela (2019)). These promote advancing of the integrity and reliability of accounting processes and offer the establishment of more secure and transparent blockchain, underpinned accounting controls to counteract the likelihood of such unethical conduct (Nadine (2017)).

Studies such as those by McCallig et al. (2019) and Moll & Yigitbasioglu (2019) look into how blockchain can secure data transmission in reporting and auditing processes. They highlight the importance of cryptography in strengthening the trustworthiness of financial information (Garanina et al. (2022)). Emerging research also points to the role of blockchain in reshaping future decision-making processes by integrating advanced technologies like AI and Big Data analytics. As reported by Dashkevich et al. (2020), relevant business processes related to current Big Data analytics, the importance of filtering and signal extraction utilised by industry grows (Hassani et al. (2018), Guo & Liang (2016), Tinn (2018)). The opportunity here is to improve current limitations in the transaction processing life cycle, such as problems of quality and completeness of messaging between systems, lack of reference data systems, various problems with book-keeping, manual or even paper-based confirmations (Walker (2017)). This integration could revolutionise the accounting profession, altering the traditional roles of accountants and auditors as identified by Schmitz & Leoni (2019) and Garanina et al. (2022).

Lastly, the effectiveness of blockchain in fraud detection has been studied across various business domains, including insurance, banking and real estate. The literature on the application of blockchain technology to mitigate accounting fraud reveals a consensus that blockchain provides a secure and transparent platform that various stakeholders can trust to prevent fraudulent activity (Mohanty et al. (2023)).

Rückeshäuser (2017) identifies one of the core problems with existing accounting practices as the ability to conduct fraud by use of accounting manipulation and concealment techniques. It defines accounting fraud as the deliberate preparation and dissemination of accounting records by direct or indirect involvement of top management of an organisation. Traditional accounting systems rely heavily on centralised authority and are susceptible to the risk of management override - a significant existing concern in accounting fraud. Rückeshäuser (2017) concludes that blockchain's distributed ledger technology (DLT) ensures that transactions are not unilaterally recorded or altered without consensus, thereby offering a structural resistance to fraudulent activities. The same study also states that blockchain-based accounting systems could enable decentralised consensus mechanisms which may act as a barrier against the manipulation of financial data.

The role of blockchain in accounting is also viewed through the lens of it being a foundational technology rather than a disruptive one, with the potential to transform economic and social systems fundamentally (Garanina et al. (2022)). Blockchain's decentralised and immutable ledger offers a reliable framework for transparent and reliable transactions, visible to all network participants key to deterring potential manipulative actions (Garanina et al. (2022), Oladejo & Jack (2020)). Moreover, blockchain's attribute of immutability can significantly aid in fraud detection since once data is recorded on the blockchain, it cannot be altered or deleted, thereby creating a verifiable and permanent record of transactions (Oladejo & Jack (2020)).

Mingming (2020) offers more relevant insights into the transformative role that blockchain can play in modernizing Accounting Information Systems (AIS). The work emphasises the importance of integration of blockchain technology into existing accounting frameworks, to enable improvements in the auditing process making it more efficient and less prone to errors. The same study states that inherent characteristics of DLT, such as immutability of records and transparency of transactions, make it an excellent tool for building robust accounting information systems, aligning with core requirements of reliable financial reporting and fraud prevention. By ensuring that financial entries cannot be tampered with post-confirmation, blockchain creates an environment where fraud is not only difficult to commit but also easier to detect (Mingming (2020)). One of the revolutionary aspects of blockchain in AIS, as highlighted by Mingming (2020) is the potential for real-time auditing. This can ensure that anomalies are quickly identified and addressed, thereby maintaining the system's integrity.

Bonsón & Bednárová (2019) explore implications of blockchain for auditing and accounting within the context of the emerging digital economy, by evaluating how blockchain can strengthen the trustworthiness of financial statements and reduce the occurrence of accounting fraud. Theoretical insights of Bonsón & Bednárová (2019) underscore transformative potential of blockchain in accounting and auditing, advocating for a future where financial reporting is more secure, transparent and efficient. The findings of Bonsón & Bednárová (2019) provide a tangible model for the application of blockchain in combating accounting fraud. The research is predicated on:

- a permanent and unalterable record of all transactions provide by blockchain ledger safeguards against fraudulent alterations (Bonsón & Bednárová (2019));
- decentralisation illuminating the potential for single point of failure or control, which is often exploited in fraudulent scheme (Bonsón & Bednárová (2019));
- blockchain transactions being transparent and verifiable by any allowed participant in the network, validating the accuracy of financial information (Bonsón & Bednárová (2019));
- blockchain-based smart contracts being able to automate and enforce compliance with accounting standards and policies (Bonsón & Bednárová (2019));
- the real-time and comprehensive nature of blockchain records being able to streamline the auditing process, making it more efficient and less susceptible to human error or intentional oversight (Bonsón & Bednárová (2019)).

However, blockchain's advantages in fraud prevention are balanced by challenges related to scalability, energy consumption, regulatory uncertainty and the permanency of its records (Oladejo & Jack (2020)). There is a recognized need for accounting professionals, especially forensic accountants to adapt and develop technical skills suitable for detecting fraud within blockchain systems, as traditional methods may not suffice (Garanina et al. (2022)). These forensic accountants, for instance, face the task of navigating through vast and complex databases to detect patterns of fraud. Blockchain's distributed data organisation can pose both opportunities and challenges in this regard (Secinaro et al. (2021)). Additionally, Rückeshäuser (2017)) also notes concerns surrounding the adaptation of existing legal frameworks, the technical complexity of blockchain systems and the need for widespread understanding and trust in its mechanisms. These studies suggest that while blockchain has a significant role to play in mitigating fraud, there is still much to be explored, particularly in terms of practical applications, regulatory responses and integration of this technology within existing financial systems

By leveraging blockchain technology, the BFS developed in this Thesis introduces a novel approach to capturing, processing and reporting economic activities. The BFS adopts the principle of decentralisation of the single point of control of Mingming (2020) by distributing the BFS ledger across a network of entity members, ensuring that no single entity has the authority to alter financial records unilaterally, thus upholding data integrity. The BFS artefact also aligns with Rückeshäuser (2017) on decentralised accounting systems. By implementing a blockchain-based accounting framework, the BFS artefact provides a transparent, secure and immutable record of financial transactions. This framework reduces the risk of fraud and streamlines the auditing process, as each entry is verifiable and traceable to its origin.

The research presented by Rückeshäuser (2017) provides a foundational understanding of the potential of blockchain in revolutionising accounting practices and where BFS serves to illustrate the tangible application of these concepts in a real-world context. By leveraging blockchain for the BFS, the following attributes discussed by Rückeshäuser (2017) are introduced to combat accounting fraud:

- ❖ Immutability: Once a transaction is recorded on the blockchain, it cannot be altered retroactively. This feature is instrumental in preventing the tampering of financial data, a common tactic in fraud schemes (Rückeshäuser (2017)).
- ❖ Transparency: Blockchain's transparency ensures that all transaction records are accessible to authorised parties, providing a clear audit trail that deters fraudulent behaviour (Rückeshäuser (2017)).
- ❖ Decentralisation: The absence of a single point of control in blockchain architecture disperses the power to authorised transactions, reducing the risk of management override and unauthorised alterations (Rückeshäuser (2017)).
- ❖ Smart Contracts: Autonomous and self-executing contracts with predefined rules can enforce compliance and internal controls without human intervention, thereby minimizing the scope for fraudulent overrides (Rückeshäuser (2017)).

The potential capabilities for innovation from blockchain integration with existing models of business operations, together with improved efficiency of accounting implementation and enhanced mitigation of accounting fraud have the possibility to optimise financial infrastructure. These improvements can play an important role in sustainable development of the global economy by creating shared value systems and improving cooperation among banks, technology companies, regulatory agencies, customers and the overall economy (Dashkevich et al. (2020)).

2.6 Blockchain for Central Banks: A Systematic Mapping Study

2.6.1 Declaration

The literature review incorporated into this thesis, which was published as part of a systematic mapping study in the IEEE Access in 2020, under the title Blockchain Application for Central

Banks: A Systematic Mapping Study (Dashkevich et al. (2020)), was designed to provide a comprehensive exploration and analysis of the peer-reviewed research on blockchain applications in central banking. It highlights that industry is leading research efforts on use-cases such as CBDC, PCS, and regulatory compliance, and payment clearing and settlement systems, while areas such as assets transfer and audit trail indicated a gap in academic engagement. The scope of this review was intentionally limited to cover publications up to 2020 for several important reasons. Firstly, the primary focus of the study was to capture and synthesise the most relevant and foundational research on blockchain and its potential applications in central banking systems during the early years of its development. Given the evolving nature of blockchain technology and its potential applications, the literature review included all available academic work on this starting from formative years of blockchain development in finance, including the early interest from central banks. This study provides a comprehensive foundation for understanding how blockchain could be integrated into the financial ecosystem. Additionally, as I am currently employed by the Bank of England (BoE) as a Central Bank Digital Currency (CBDC) Architect, there are potential conflicts of interest concerning public elaborations on the subject beyond the 2020 time-frame. Engaging in further discussions or reviews of post-2020 developments related to blockchain applications within central banking will overlap with my professional responsibilities and ongoing work at the BoE. Therefore, to maintain professional integrity and avoid any conflict of interest, I have chosen to limit the scope of the review to the period that is already communicated by me in the public domain. Lastly, in the Mapping Study, there were three authors involved: myself, as the primary researcher and writer of the paper, and my two supervisors who provided guidance and refinement of the research approach. I conducted the majority of the analysis and writing, while my supervisors contributed to the methodological rigour, reviews, and crucial feedback for refinements. When differences in interpretation or approach arose, we resolved them through collaborative discussions, with a focus on aligning the methodology and thematic findings with the research objectives, ensuring that the final paper reflected a consensus based on evidence and academic rigour.

2.6.2 The Mapping Study

Blockchain is a novel technology capturing the attention of Central Banks and a technology with significant disruptive potential. However adaptation of the scientific community to this topic has been comparatively slow and resources have been limited to Bitcoin source code, blog and forum posts, mailing lists and other online publications (Judmayer et al. (2022)). Following the work of the Bitcoin White Paper (Nakamoto (2008)), the majority of blockchain-based innovation was provided not by peer-reviewed scientific publishing but by directly interested industries (Judmayer et al. (2022)). Although this has reduced time-to-market for blockchain, it has also lead to deficits in systematisation and a gap between practice and the theoretical understanding of this novel field (Judmayer et al. (2022)). The purpose of the mapping study presented in the following sections is to reduce that gap by presenting a thematic overview of peer-reviewed publications on potential application of blockchain technology to the functions performed by central banks.

The objective of the mapping study is to find and systematically map all available scientific papers to empirical and non-empirical research approaches. Identification of the scope for blockchain use-cases, applicable to the business of central banks, allows us to determine what problems have already been investigated, yielding a theoretical understanding or practical contribution. Furthermore, it provides narrative summaries of opportunities and challenges to businesses and operational performance of central banks from hypothetical adaptation of blockchain for each of the identified use-cases:

1. *Central Bank Digital Currency (CBDC)*;
2. *Payment Clearing and Settlement (PCS) systems operated by central banks*;
3. *Assets transfer and ownership*;
4. *Audit trail*;
5. *Regulatory compliance (Regulation)*.

The mapping study does not aim to promote or highlight any particular approach, a benefit or a challenge, but merely to help academics and practitioners identify where the greatest or least effort has been directed by the research community, understand where the gaps for future exploration could be and provide a starting point for further systematic discussion. To achieve those goals, this study adopts a Systematic Mapping Study (SMS) research methodology that follows the guidelines of Petersen et al. (2008, 2015).

CBDC models are often seen as the next milestone in the evolution of money. Academic publications focus on design characteristics and country-specific requirements of CBDC to guide its potential application and adaptation. Overall, CBDC promises to provide central banks with a reliable close to real-time ‘window’ on economic activity to guide monetary policy. However, the trade-off between the risks and benefits of such systems are still unclear, because, despite the promises of various benefits and reduction of particular risks, other new unknown risks could emerge, some of which could stem from immature blockchain technology and/or lack of empirical research; some could also arise from operational or security risks stemming from technological disruption.

In relation to hypothetical blockchain underpinned by the *Payment Clearing and Settlement (PCS) system, operated by a central bank*, researchers predict that such a system could generate value by improving efficiency through modernisation of underlying technology of financial markets infrastructure. These present the possibility of reduction of costs for transactions, reconciliation, clearing and processing, together with reduction of legal, settlement, operational and financial risks. On the other hand, researchers seem sceptical about the full substitution of well-established, collective infrastructure and processes, built by banks with currently available blockchain protocols. The lack of incentive for alternative systems is driven by inefficiencies arising from high set-up costs and already existing network effects. Additionally, a one-size-fits-all approach to blockchain application to PCS activities raises a broad range of further challenges.

Transfer and ownership of assets through central bank-maintained systems has also been claimed as a hypothetical beneficiary from blockchain adaptation. Researchers insist that the assets-agnostic nature of DLT can provide trusted, time-oriented, immutable, shared databases for recording transfer of assets and change of ownership, without relying on numerous specialised third-party infrastructures and intermediaries, reducing intermediation costs and risks. On the other hand, serious outstanding questions are raised by some researchers. Current laws do not define DLT-based proof of ownership and overall legal validity of financial instruments issued on the blockchain.

Small numbers of research studies have been devoted to the enhancements of the *regulatory audit trail* from blockchain application. Regulators could attain a real-time opportunity to monitor, supervise and audit trades through a blockchain-based “global audit log” which promises to ensure integrity of records through the integrity of the blockchain ledger itself. Furthermore, such a system could promote the reduction of multiparty multi-intermediated reconciliation costs and risks, by automating and streamlining it. However, some researchers highlight issues of ensuring the validity and reliability of transactional records, because a DLT system does not provide a mechanism for guaranteeing that the added information is correct. Blockchain application for *regulatory compliance* has also been extensively covered in peer-reviewed literature. Researchers suggest that financial regulation could be improved by automating mandatory regulatory reporting or through the creation of an algorithmic rule-following monetary authority on blockchain. That would facilitate embedded supervision, thus reducing some legal risks and deterring avoidance of the regulatory arbitrage. Traceability characteristics of blockchain can promote the reduction of risk of fraud through automation of Know Your Customer (KYC), Combating the Financing of Terrorism (CFT), Anti-Money Laundry (AML), tax misreporting etc. On the other hand, researchers also discuss a number of regulatory friction points to blockchain adaptation. The effects of blockchain application for central banking are not currently covered by the existing regulatory framework, thus spanning new legal issues. Current blockchain architectures provide limited access to the regulators, leaving governance, risk allocation and consumer protection in the hands of the coding experts, who might lack legal and/or financial expertise. Furthermore, blockchain promising information transparency could cause confidentiality and privacy loss leading to competition issues.

The notion of a "Technical Argument" (Bains (2019)) is also relevant to the work presented in this mapping study and Thesis more generally and allows a dissection of the different elements of why to undertake studies and the motivation for doing such studies. Such an argument has several components. The first is "a vision" for the work. From the point that started this mapping study, the study envisaged the work as potentially seminal and that it would be a source of reference for central banks to use for understanding the state-of-knowledge in blockchain utilisation. The second component of a technical argument asks "why progress is needed" in the area. So, central banking is seen as a fundamental part of society's fabric. Understanding how disruptive technology of blockchain could influence practices of central banks has the potential in the future to shape those banking practice and the implications of these factors is essential for highlighting problems and areas for progress in this domain. The third component of a technical argument is "prognosis". Although it is difficult (as for most things in life) to predict the likely outcomes of blockchain use in central banking, not least because the field is advancing so quickly, work highlights throughout this mapping study the areas that could be exploited, the areas that come to the fore and those that present new challenges and that can be extended. The final component is an explanation of "why the status quo is not good enough". Blockchain provides a wealth of opportunities for the banking sector and the impact of exploiting those opportunities is extensive. As such, the inadequacies of current systems should not be seen as problems necessarily reflecting a poor situation, but as exciting ideas for the future. The work in this mapping study brings these ideas to the fore through a complete study of industrial and academic work thus far.

2.6.3 Banking and Central Banking

This mapping study focuses on central banks, but it is important to understand the role of wider banking, as this should help to determine where and how innovative the blockchain technology can potentially fit. According to Casu et al. (2006), banks, as other financial intermediaries, play a pivotal role in the economy by channelling funds from units in surplus to units in deficit. They reconcile the different needs of borrowers and lenders who do not know and do not trust each other. They transform small-size, low-risk and highly liquid deposits into loans which are of larger size, higher risk and illiquid. The banking industry is broad and combines sectors related to central banking, investment, corporate, commercial, retail banking etc., differing by their business models and performance goals. More specifically, a central bank, a reserve bank or a monetary authority is a financial institution that manages domestic money supply, interest rates and oversees a country's broader banking system.

2.6.4 Central Banks: Overview of Problems

These days, global central banks vary substantially in their structure and purpose (Ortiz (2009)). They face complex issues in designing effective governance policies for each of their major functions and to accommodate their many differences (Ortiz (2009)). As a monetary authority, they sometimes fail to contain macro-economic crises (Hayes (2016)) that could stem from incentivised excessive risk-taking e.g., via unconventional monetary policy tools such as negative rates or Quantitative Easing (QE). These, in times of financial distress and high volatility, exacerbate negative outcomes (Hayes (2016)). Further problems result from large numbers of financial intermediaries (Tsai, Deng, Ding & Li (2018)). In addition to high fees, service charges paid for financial intermediation and cost of regulatory compliance, there are delays, onerous paperwork and opportunities for fraud and crime (Ozili (2019)). Multifaceted linkage between banks and a variety of central intermediaries adds to a incomplete understanding of the post-crisis financial system; in particular, this relates to the concentration of the risk management of credit and liquidity risks in those intermediaries and the impact on systemic risks (Golub et al. (2018), Domanski et al. (2015)).

Ben Dhaou & Rohman (2018) suggests that there are issues with the banknote creation functionality of central banks, when used as a main instrument of tax evasion, money laundering and the financing of illegal activities. Cash also limits the scope for monetary policies based on negative interest rates, since it provides a zero-rate alternative that can be stored (Ben Dhaou & Rohman (2018)) and it deteriorates rapidly, especially in high inflation countries (Ben Dhaou & Rohman (2018)). The current set-up of the European post-trade market is still a legacy of earlier domestic market infrastructures (Pinna & Ruttenberg (2016)). The problems stem from the lack of interoperability between centralised proprietary databases and that often restricts straight-through processing for a range of non-vertically integrated financial institutions (Pinna & Ruttenberg (2016)). This prolongs ongoing use of siloed digital records of ownership and requires manual updating to be reconciled with any change that occurs in the records of counterparties at different levels of the post-trade value chain (Pinna & Ruttenberg (2016)). These escalate the cost of back-office procedures and inflate certain risks such as: operational risk, chains of settlement failures (as delayed settlement of one transaction may affect the settlement of trades with third parties), human errors (the system being reconciled manually) and limited collateral fluidity (Pinna & Ruttenberg (2016)). Overall, all payment systems suffer from settlement or payment risks for technical or financial reasons, such as settlement, credit and market risks (Didenko et al. (2020)). This and the aforementioned challenges have attracted the attention of the financial regulators and provide the context and opportunities for modernisation and improvements.

2.6.5 Related and Excluded Surveys

Four existing surveys discuss literature in the area of application of blockchain as financial technology (FinTech) for the central banking business. However, none of those surveys focus solely on peer-reviewed publications about utilisation of DLT by central banks. This is the research gap that is identified of current mapping study. Firstly, the work of Rio & César (2017) reviewed stages of acceptance of DLT by central banks between 2016 and 2017 for their various systems and functions. The review was based on grey literature, i.e., on a central bank's own available publications, reports and press releases. The subset of utilised countries were those that belonged to the Organisation for Economic Co-operation and Development (OECD) and to the G20 organisations, including the Bank for International Settlement (BIS) and the European Central Bank (ECB), but excluded the European Union (EU) and countries outside the OECD. The work concluded that, despite all central banks used in the study expressing interest in DLT, not one had an operational DLT-based system (Rio & César (2017)). The reasons for the current unavailability of live blockchain applications were due to issues with: "Speed, cost of processing, security, transparency and privacy, legal settlement finality, scalability, network effects and immature technology" (Rio & César (2017)). The same research did not go into the specifics of research trends and thus differs from the research approach and results of this mapping study.

Secondly, in a systematic literature review, Lutz (2018) examined financial literature on the topic of: "dual or multiple currency scenarios for privately issued cryptocurrencies" coexisting or competing with the central bank issued fiat currency and suggested a coexistence theory. The review was limited to a financial/economic perspective and excluded ethnological aspects of blockchain as well as its legal contributions. The work provided a comprehensive, detailed overview and analysis of the relevant contributions on currency coexistence, competition and developed a theoretical framework of the main ideas and functions of cryptocurrencies. The work concluded that: "little academic research looks closer on the existence, interaction and consequences, as well as on a possible set up of coexisting private cryptocurrencies and central bank issued fiat currencies". This survey is different from this research since it focused on privately issued cryptocurrencies as competing and coexisting with fiat currencies.

Thirdly, the work of Thakor (2020) summarised theoretical and empirical literature on the interaction between novel financial technologies such as blockchain, its cryptocurrencies and the banking industry. The study considered: "Innovations in payment systems (including cryptocurrencies), credit markets (including P2P lending) and insurance, with blockchain-assisted smart contracts playing a role". The work debated the consequences for central banks, its payments, clearing and settlement systems (PCS) from cryptocurrency, created privately or by the banks themselves as a competitor to fiat money. The survey focused on cryptocurrencies and wider financial markets and is thus different from this work.

Lastly, Hassani et al. (2018) presented an example of a comprehensive overview of increasing interest from a global banking industry towards the adoption of blockchain and presented a wide-ranging taxonomy of existing applications and relationships between blockchain and the wider banking sector. The work summarised the opportunities and challenges from a banker's perspective on blockchain adaptation. Furthermore, they elaborated on what future impact from Big Data generated on blockchain could have towards existing practices of data analytics in banking. They highlighted the increasing importance of filtering and signal extraction for the banking industry and also highlighted the lack of academic interest in this subject area (Hassani et al. (2018)). This work is different to this research, because it covered research into wider the banking business and blockchain adaptation, without specific focus on central banking and only peer-reviewed research; in addition to academic publications, they also included industry wide reports, blogs and wider media sources on blockchain applications.

Surveys excluded from this study focused on wider applications of blockchain other than those for central banks. More specifically, on economic aspects of cryptocurrency (without interaction with fiat currency), blockchain evolution and technological concepts, surveys that did not focus solely on the application of DLT for central banking or financial services and surveys on the application of blockchain by industries other than banking or financial services.

2.6.6 Research Methodology

Motivation and Research Questions

The Systematic Mapping Study (SMS) research methodology was selected and applied for this mapping study, with the aim of describing the wide state of knowledge about the interest in blockchain technology for, and by, the central banking system. An SMS is a form of Systematic Literature Review (SLR), described by Keele et al. (2007) and it aims to give a broader examination of a researched topic than an SLR. It is motivated by the need to understand trends through thematic categorisation, a spectrum of publications and common or important topics and gain an understanding of the evolution of the field. The objective of this SMS was to find and map all empirical and non-empirical peer-reviewed research on DLT to the various areas of central banking. The outcome of this study provides an overview of the scope of the researched area; this will allow identification of research gaps that could be considered for further examination. The study follows the guidelines of Petersen et al. (2008, 2015), utilising steps of the Systematic Mapping Process (SMP). The high-level steps for the review were as follows: 1) define research questions; 2) conduct a pilot search for primary studies; 3) construct search string; 4) search for all relevant papers; 5) process keyword/s for all abstracts; 6) extract and classify data; 7) analyse the results.

The first step of the SMP was to define research questions (RQs), which, according to Petersen et al. (2015) and Kitchenham, Brereton & Budgen (2010) allow for a wide overview of the available topics related to blockchain for central banks. The research questions outlined next were motivated by the focus of this study - in other words, to review all peer-reviewed research available on the intersect of blockchain and central banks:

RQ1 *What are the trends in research on blockchain application for central banks?* This research question is motivated by the need to understand the comparative maturity of the topic, by examining where, when, how and by whom the research was communicated.

RQ2 *What potential blockchain-based use-cases for central banks are addressed by the research community?* This research question is motivated by the need to understand where DLT is seen to be suitable for application for the central banking.

RQ3 *Why or why shouldn't blockchain be considered?* This question is motivated by the need to understand why DLT was considered for each of identified use-cases and what challenges the application of blockchain poses, but not to highlight or promote any specific approach.

RQ4 *What is the depth/breadth of the research for identified use-cases?* This research question is motivated by the need to understand the comparative maturity and application specifics of each separate use-case.

Primary Study Search and Search String

To develop a rigorous search strategy, the next step of the mapping study was to search for all relevant papers. A *pre-defined search protocol* that specified methods of undertaking the search for the literature was established, to reduce the possibility of researcher bias and to allow for subsequent validity evaluation (Petersen et al. (2008, 2015)). The final search was conducted in 2020 and included years between 2008 and 2020. The current study used two common *search strategies* (Petersen et al. (2008, 2015)): database search and manual search. Leading *academic databases* were searched to obtain the literature for the study, namely: IEEExplorer; ELSEVIER: Scopus, SSRN (including JEL - Journal of Economic Literature), ScienceDirect; arXiv.org; Web of Science; ACM.

The *steps of the search* were as follows:

1. following the guidelines of Petersen et al. (2008, 2015) an initial set of keywords was identified from the study title: "blockchain" and "central bank";
2. a pilot manual database search was first conducted using those keywords, where additional keywords were derived from the known papers (Petersen et al. (2015)) and categorised based on James et al. (2016). A Population Intervention Comparison Outcome (PICO) approach allowed the creation and structuring of the search string (Petersen et al. (2008, 2015));
3. improvements in the search were implemented to find more relevant papers *per* iteration (Petersen et al. (2015)) and update the search string.

According to Petersen et al. (2015), Population (P) and Intervention (I) are the most relevant for a mapping study, since the other dimensions may restrict the search too much and remove relevant articles. As a result, only P and I dimensions were applied for search string composition. In the current research context, those elements are defined as: *Population*: an industry group comprising a central banking business and its underlying products and services; *Intervention*: blockchain technology as a software engineering tool considered for the application and adaptations for central banking functions. The decision not to use "cryptocurrency" and "Bitcoin" as keywords for the search string was based on this pilot search results. Papers collected by the search tended to be related to the economics of publicly issued cryptocurrencies such as Bitcoin rather than aspects of underlying blockchain technology and its applications. The steps of composing a search string and applying a database suitable variation of it, using the P and I dimensions were as follows (Petersen et al. (2015)):

> Step 1: Scope the search for banking industry related publications: *("banking" OR "bank" OR "central bank" OR "reserve bank" OR "monetary authority" OR "monetary" OR "financial Intermediary" OR "financial Intermediation" OR "clearing" OR "clearinghouse" OR "settlement" OR "financial institution" OR "FinTech" OR "financial technology" OR "inter-bank" OR "IBPS" OR "real-time gross settlement" OR "RTGS" OR "payment settlement" OR "CBDC" OR "money supply" OR "monetary policy" OR "technocracy")*
> *AND*
> Step 2: Search further in the population of papers obtained by the Step 1 for reference to blockchain technology - intervention: *("blockchain" OR "distributed ledger technology" OR "DLT" OR "smart contracts")*

Search for Relevant Papers

Not all identified papers were relevant to the topic, so the next phase was to evaluate the actual relevance of obtained articles against what was known about the population of the topic of interest (Petersen et al. (2015)). The study achieved this by defining rigorous inclusion and exclusion criteria. Those criteria were applied to all titles, abstracts and keywords of articles obtained earlier with the goal of identifying papers clearly in or out of the scope of the mapping study (Petersen et al. (2015)). Grey literature such as relevant government project reports, working papers and evaluation documents available through earlier pre-specified databases was also included. (Garousi et al. (2019, 2016)) underlined the importance of such literature to be used as an additional source for understanding the area of novel research. The topic of development, application and evaluation of blockchain technology for central banks is a novel research domain and inclusion of grey literature broadens the outlook for both the state-of-the art and the state-of-practice in the area (Garousi et al. (2016)) by including wider research sources.

Inclusion criteria:

1. English scientific and grey, empirical and non-empirical, peer-reviewed articles, conference papers, available through pre-specified databases;
2. publications between 2008 - 2020 inclusive;
3. papers with research scope of blockchain technology and sub-scope - the application of that technology for the domain related to the central banking business.

Exclusion criteria:

1. papers without full text availability;
2. papers that were not written in the English language;
3. studies that were duplicates of other studies;
4. studies that were an older version of studies already considered;
5. the study was not a scientific study, such as editorials, summaries of keynotes, workshops and tutorials;
6. studies that were book chapters;
7. papers that had some other meaning other than one relevant to the application of blockchain technology for central banking.

The final “Database Search Results” with the database specific search strings and automated (if database functionality permitted) or manual application of inclusion/exclusion criteria on title, keywords and abstract is provided in Appendix B.1. For *borderline papers* deemed relevant during the inclusion and exclusion, based on their title, abstract and keywords, further reading of introduction, conclusion and, if the decision was still unclear, full text reading was conducted to establish relevance to the research questions. Excluded borderline papers had a primary focus on 1) blockchain application for the wider financial sector other than central banks, i.e.,: commercial banking, financial trading and/or exchanges (excluding Payment Clearing and Settlement (PCS) infrastructure operated by central banks), general economy, unbanked; 2) papers that provided publicly issued cryptocurrency economics and solutions, i.e., that described it as a digital asset or private sector money, such as Bitcoin, not issued by the central bank; 3) wider FinTech and blockchain regulation and legal implications for blockchain and cryptocurrency other than those concerned with central banking activity. This study also performed a *forward snowballing* sampling technique on the most cited papers (Wohlin (2014)). Citing metadata is available through the majority of the databases. A further 13 studies were added through this technique (Wohlin (2014)). The decision to use forward snowballing was underpinned by the focus on more recent and novel publications and to allow for theoretical validity evaluation. Final *quality assessment* was performed on the set of 72 primary studies. According to Petersen et al. (2015) and Kitchenham, Budgen & Brereton (2010), for a mapping study: “Quality assessment should not pose high requirements on the primary studies, as the goal of the mapping is to give a broader overview of the topic area” (Petersen et al. (2015), Kitchenham, Budgen & Brereton (2010)). The criteria for paper evaluation was whether the knowledge claims made by the paper were interesting and justified by the research method (Wieringa et al. (2006)). Fig. 2.4 represents the final results for each step of the mapping study.

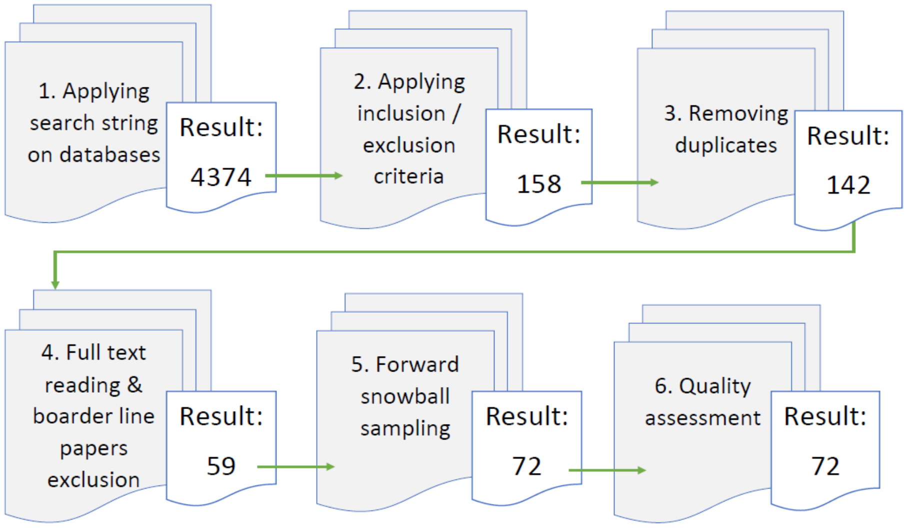


Figure 2.4: Number of included articles during the study selection process.

Keywording of Abstracts

The next stage of the SMS was the keywording of abstracts of the final set of relevant papers (Petersen et al. (2008)). Keywording is a way to reduce the time needed for developing the classification schema and to ensure that the schema takes the existing scope of studies into account. To build the current classification schema, again followed the guidelines of Petersen et al. (2008), conducted through the following steps:

1. Abstracts were read and searched for keywords and concepts that reflected the contribution of the paper; while doing so, the context of the research paper was identified. When the abstracts provided no meaningful category of keywords, the paper's introduction and conclusion were also read;
2. Sets of keywords from different papers were combined to develop a high-level understanding about the nature and the contribution of published research. This process produced a set of categories representative of the underlying included studies;
3. All selected papers were then read fully. If a paper revealed some new important keywords in the text, existing categories were updated (Petersen et al. (2015));
4. The final set of keywords was then clustered and used for categories of the current mapping study (Petersen et al. (2008)).

Data Extraction, Analysis and Classification

The aim of this step was to collect all the information required to structure the literature for this study in order to map it and to answer research questions. Following the guidelines of Petersen et al. (2015), this work developed a data collection form to enable data extraction from the included publications. Each data collection field was populated with a data item (the column header – a category) and its corresponding values. This allowed a check for the correctives of extracted data in the collection form by tracking it back to its original paper. The development of the form was achieved in two stages:

1. The study itemised basic metadata available through pre-defined databases and populated the data form with corresponding values. The added data items were: *document title, authors, publication year, publication venue, publication type and publisher*. After reading all papers,

further fields were added, such as *research type, research contribution.* This step allowed development of the facets for: *"Topic-Independent Classification Schema"* (Petersen et al. (2015)). These facets enabled us to firstly, answer RQ1 and RQ4 and, secondly, to facilitate comparison of the similar or same research in the different fields (Petersen et al. (2015)). This allowed us to gain insights into the comparative maturity of the study area (Petersen et al. (2015)) and helped to improve and clarify classification (Petersen et al. (2015)).

2. Further categories were then added to the data collection form headers that emerged from keywording of the abstracts. This stage developed a schema representative of the underlying publications: *"Topic-Specific Classification Schema"* (Petersen et al. (2015)). This provided study specific categories (Petersen et al. (2015)) allowing us to answer RQ2 and RQ3 and to map findings against the facets identified in the previous stage.

The topic-specific classification schema developed could be considered as an additional contribution on its own, since it provides a framework for categorising and describing the blockchain-based interest and application for central banking business in peer-reviewed literature. The full list of headers of the data collection form is provided in Appendix B.2. Categorised data then was used to visualise, summarise, analyse and draw conclusions in relation to the research questions, to satisfy the aim of the research.

2.6.7 Mapping Results

A total of 72 papers were used in the completed review, with three categories for topic independent classification schema and five categories for topic specific schema defined for each paper. The complete list of all included papers is provided in Appendix B.3. It is important to note that the current study does not represent a full and comprehensive review of how all central banks explore blockchain technology today. Such a review would require us to consider, in addition to academic publications: industry reports, press releases, white papers etc., with emphasis primarily on grey literature sources. A good example of one such review is in the work of Hassani et al. (2018), where the authors summarised blockchain adaptation for the wider banking community largely utilising industry and media reports. The focus of current study is to report the state of academic research.

Topic-Independent Classification Schema

This section provides an overview of the data from included literature allowing us to answer RQ1. This question is motivated by the need to provide a comparison of similar research in different fields (Petersen et al. (2015)).
RQ1: What are the trends in research on blockchain application for central banks?

Frequency of Publications and Literature Types

Fig. 2.5 represents numbers for all publications identified between the beginning of 2008 and June of 2020. The colour categorisation communicates the type of the peer-reviewed literature published, distinguished between: 1) conference proceedings, 2) grey literature and, 3) journal and magazine academic articles. The bubble plot in Fig. 2.5b shows the count and percentage of the total for each of those literature types. The data reveals that peer-reviewed grey literature contributed the most research to this topic - 31 papers (or 43.06% of total), with academic articles being a close second at 27 papers (or 37.5% of total), leaving just 19.44% (or 14 publications) for Conference proceedings (Fig. 2.5b). Although the search included years 2008 - 2020, the data shows that, across all pre-specified databases and including manual search and forward snowball sampling, there were no publications available reflecting the interest of the research community in application of blockchain for central banks until 2016 (bar chart of Fig. 2.5a). During that year, a total of 11 publications (or 15.28% of total available literature) were shared, with almost half provided by industry (grey literature types), only two being a pure academic article and four communicated as Conference

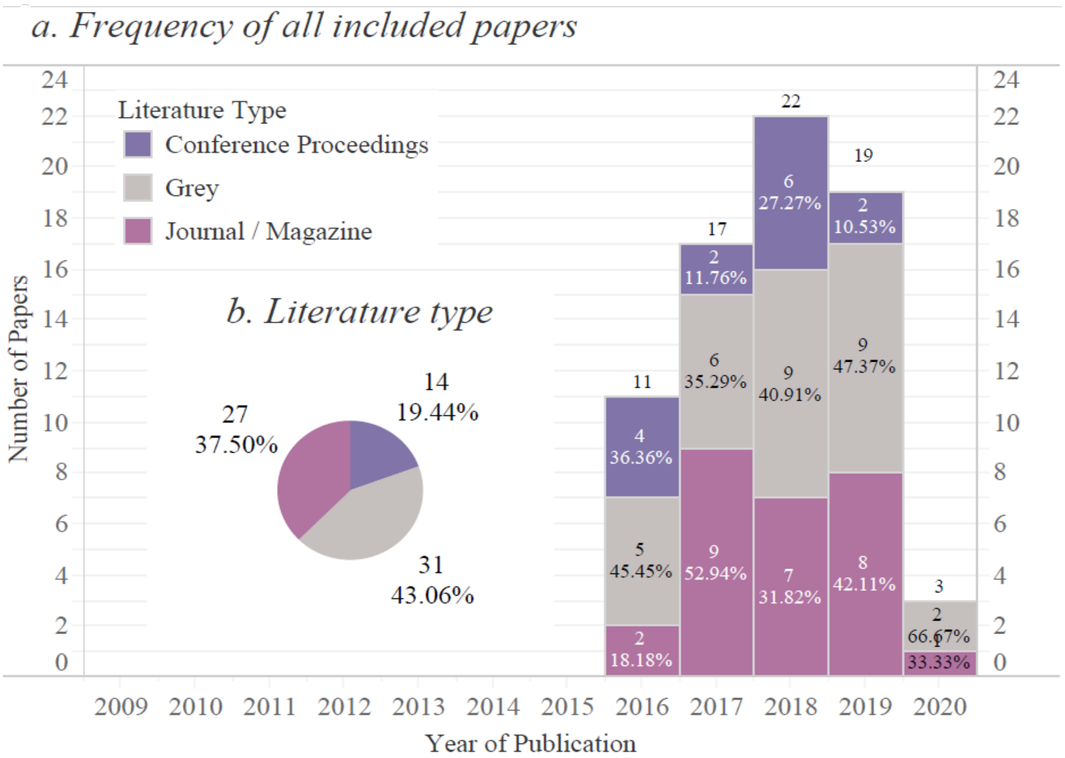


Figure 2.5: Frequency of all included publications and literature type

proceedings. Over the following two years, the overall number of available papers steadily grew and peaked in 2018 at 22 papers (or 30.56% of total). For that year, grey literature provided a slight majority of the research (nine papers or 40.91% for that year), closely followed by academic articles (seven papers or 31.82% of 2018). Availability of Conference proceedings fluctuated over the years, peaking in 2018 to 6 papers. The greatest number of academic articles was found in 2017 (9 papers), showing a steady decrease thereafter. Academics provided more than half of all research for that year. For 2019, the results showed a total of 19 papers, almost half of which were grey literature sources (nine or 47.37% of 2019). (In relation to 2020, only two papers were found as grey literature and one as an academic article, although it is difficult to judge with confidence about the final trend for 2020 since there was still a considerable amount of time left in the year when the mapping study was completed.) Overall, these results show, firstly, that the interest of the research community in the application of blockchain for central banks is a very young; secondly, that the overall trend of interest in this topic is potentially growing, and, finally, that there is a strong industry presence providing and potentially guiding such research, although participation of academics and industry practitioners in research is somewhat balanced.

Frequent Publication Venues and Publishers

Fig. 2.6 shows the most frequent venues (Fig. 2.6a) and publishers (Fig. 2.6b) for sharing peer-reviewed publications. Additionally, the bars are colour-coded to demonstrate what type of literature was available through each of those sources.

The data indicates that publications related to examination of DLT for (and by) central banks have been published in a very broad range of venues and by a wide variety of publishers. This study includes literature from 57 different publication venues, including 48 venues that have only provided a single paper. Furthermore, the research was published through 31 distinct publishers and 19 of those had only published one paper on this topic. The most frequently targeted journal that published both academic articles and grey literature was the SSRN Journal of Economic Literature (JEL) – totalling 4 publications overall (Fig. 2.6a). No pattern for conferences was

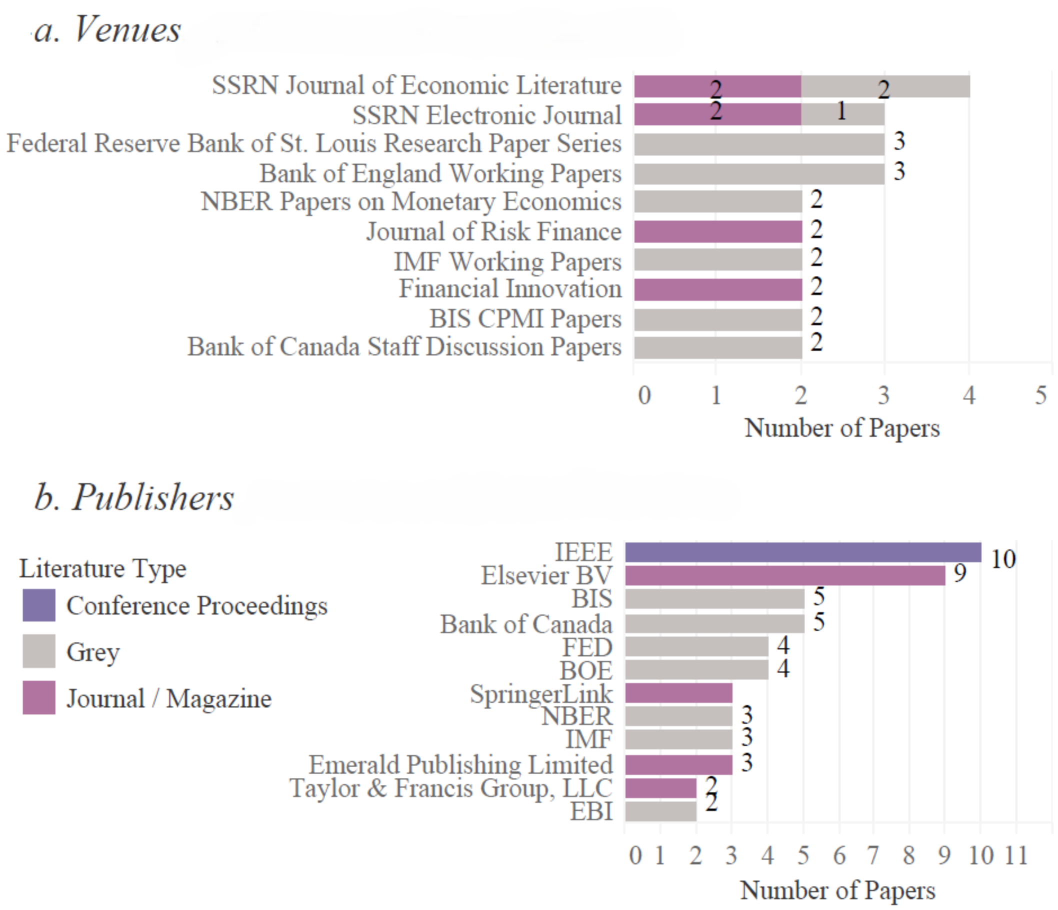


Figure 2.6: Frequency of all included publications and literature type

found, as all 13 provided one publication each.

Fig. 2.6b shows that IEEE and Elsevier BV were the most popular publishers; the former only focuses purely on conference proceedings and the latter only on academic articles; a total of 10 and 9 papers were found over the period, respectively. For the grey literature, the most frequently used channels for research outputs were the central banks themselves - Bank of Canada, FED, BoE, BIS and so on. The full list of venues and the publisher is provided in Appendix B.4.

Research Type and Contribution

This study identified the *research type facet* that reflected classes of non-mutually exclusive research approaches (or types), to which all primary studies could be mapped. As this facet is general and independent of specific focus area (Petersen et al. (2008)), it allows for comparison with other fields. This work utilised the research type categories from Petersen et al. (2008) and Wieringa et al. (2006) and this facet captured six categories in total, further grouped into two broader categories:

Empirical Research Types:

1) Validation Research;
2) Evaluation Research;

Non-Empirical Research Types:

3) Solution Proposal;
4) Philosophical Paper;
5) Opinion Paper;
6) Experience Paper.

Another facet was the *research contribution facet*, which represented non-mutually exclusive types of novel contributions provided by the included papers to the research field and captured six categories in total. Those categories could be more broadly divided into:
Practical (or technological) contribution:
1) Model;
2) Method;
3) Proof of Concept (PoC);

Theoretical (or knowledge) contribution:
4) Conceptual Framework;
5) Taxonomy;
6) New Knowledge.

This facet allows for comparison of papers with similar objectives.

As one paper can use more than one Research Type and provide more than one Contribution to communicate the work of its authors, overall numbers for each of those facets are greater than the number of included papers. An important distinction for Evaluation research is that it involves industry cooperation (Petersen et al. (2008)). The contributions from Evaluation and Validation research types, in addition to new knowledge, could also include a novel technique, such as a model or a protocol. Although a Solution proposal is a non-empirical research type (Petersen et al. (2008), Wieringa et al. (2006)), in addition to new knowledge, it sometimes provides a technological contribution in the form of Proof of Concept (PoC) – a model or protocol - but without "full-blown" validation (Wieringa et al. (2006)). Contributions of a Philosophical paper can be a new conceptual framework (Petersen et al. (2008), Wieringa et al. (2006)) and/or taxonomy (Petersen et al. (2008)), both of which are theoretical contributions. Opinion papers and Experience papers both contribute to knowledge, but, in contrast, Experience papers can involve experience reports from industry practitioners (Wieringa et al. (2006)), so often utilised for grey literature.

Fig. 2.7a represents the frequency of all contributions (column headers), provided by each research type (colour coded pie charts). Results show that, overall, the dominating contributions to this topic are knowledge, framework and taxonomy. It is seen that, overall, there are 10 novel models provided, five of those are communicated using Evaluation research type, the other two from Validation research and three are Solution proposals. Evaluation research added the most novel protocols (three out of five). Solution proposals added three models, one protocol and six PoCs as practical contributions, although those were not empirically validated. Philosophical papers added 24 new frameworks and 18 new taxonomies. The data shows that theoretical contributions dominate the field with technological artefacts appearing less frequently and predominantly provided with industry cooperation.

Fig. 2.7b shows the distribution of identified Research Types, represented as colour coded pie-bobbles, for each year. The size of each bobble shows a total count of papers and the colour of the pie is relevant to different research types. For example, in 2016 and 2019, Validation research type was utilised once in each of those years. Uses of Evaluation type peaked to five in 2018 from three in 2017. Opinion papers peaked in 2017 at 11 and were utilised six times in 2018 and nine in 2019. Philosophical papers were used the most in 2018 (13 times) to communicate new findings, nine times in 2019 and twice in 2020. Experience papers were utilised the least overall. The data potentially points out that, because the topic of this study is young, there is still little practical experience to report on - the majority of work is still theoretical. Communication of empirical findings, on the other hand, although significantly lagging, seems to be slowly but steadily increasing over time.

In Fig. 2.7c, a distribution of the cohorts of Literature Types for each Research Type is given. Results show that, overall, the authors preferred to use non-empirical research types to communicate their findings. Only 13 times out of total (or 13%), was Empirical research used (11, or 11% of total, for Evaluation research and two, or 2% of total, for Validation Research). The other times a non-empirical research was utilised, with 38 (38%) for Philosophical, 34 (34%) for Opinion Papers, eight (8%) for Solution Proposals and seven (7%) for Experience papers. Evaluation and

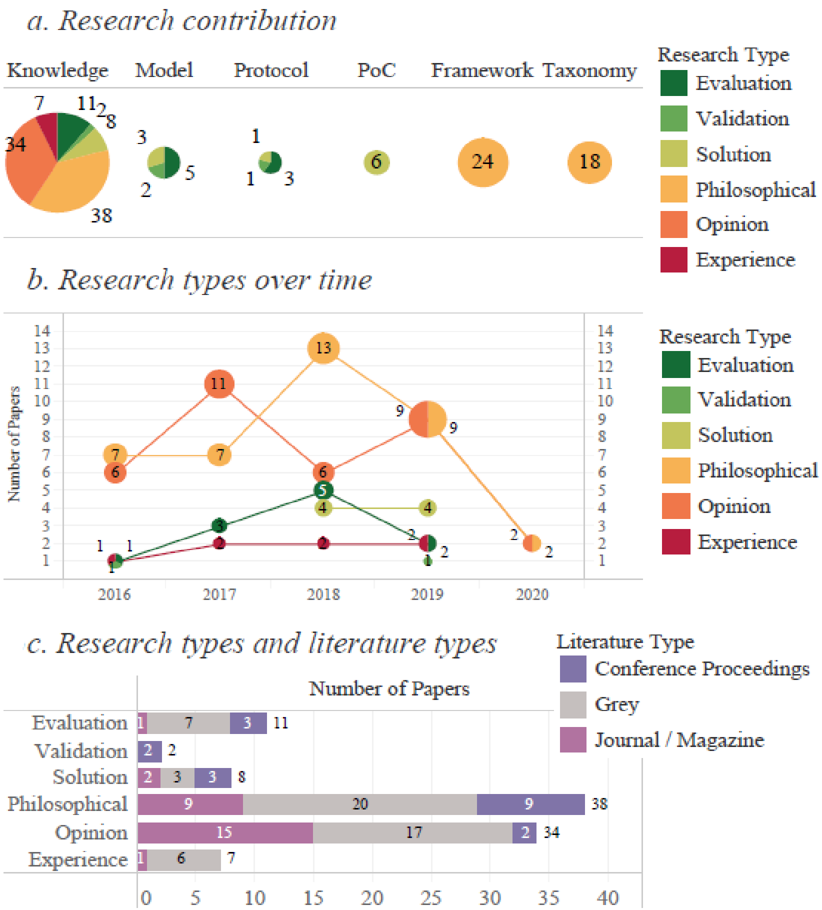


Figure 2.7: Frequency of Publications.

Experience research were mainly communicated through grey literature and the same applied to Opinion and Experience papers. Half of the Philosophical papers were provided by industry (grey literature). Solution Proposal cohorts of the researchers appears balanced. All Validation Research is available as conference proceedings. This data suggests that empirical research was mostly used by industry participants to communicate their findings. Non-empirical research was also noticeably dominated by grey literature, making practitioners into prominent debate contributors.

Topic-Specific Classification Schema: Blockchain-Based Use-Cases for Central Banks

This part of the mapping study introduces a classification schema that emerged from reading all paper keywords, abstracts and full text (Petersen et al. (2008, 2015)). This classification is specific to the underlying research topic and maps the interest from the academic circles to utilisation of blockchain for services and operations of a central bank. The schema allows us to answer RQ2 by structuring the researched topic in terms of variability of themes in relation to the application of blockchain for central banks in general.

After reading all included papers, it was evident that they fell into the five following categories for DLT-based use-cases for central banks:

1. *Central Bank Digital Currency (CBDC)*;
2. *Payment Clearing and Settlement (PCS) systems operated by central banks*;
3. *Assets transfer and ownership (Assets)*;
4. *Audit trail (Audit)*;
5. *Regulatory compliance (Regulation)*.

These use-cases provided another facet of this classification to map all included primary studies against earlier identified facets of: *research type*, *research contribution* and *literature type* and to answer RQ2 - RQ4. Furthermore, after reading all papers, an additional two sets of information emerged for each of the identified use-cases; these broadly answered the questions of:

1. *Why DLT was considered for each of those use-cases?* and
2. *What challenges the application of blockchain posed for those use-cases?*

After reading all papers, it became evident that a single paper could span multiple use-cases, could utilise more than one research type and provide more than one contribution. Therefore, the overall number of studies across all categories is larger than the total number of publications. Lastly, several technical variables emerged, most prominently discussed in the included papers. Tables C.1 and C.2 in Appendix C summarise positive and negative opinions of the researchers about application of those variables in the central bank settings. *RQ2: What potential blockchain-based use-cases for central bank are addressed by the research community?*

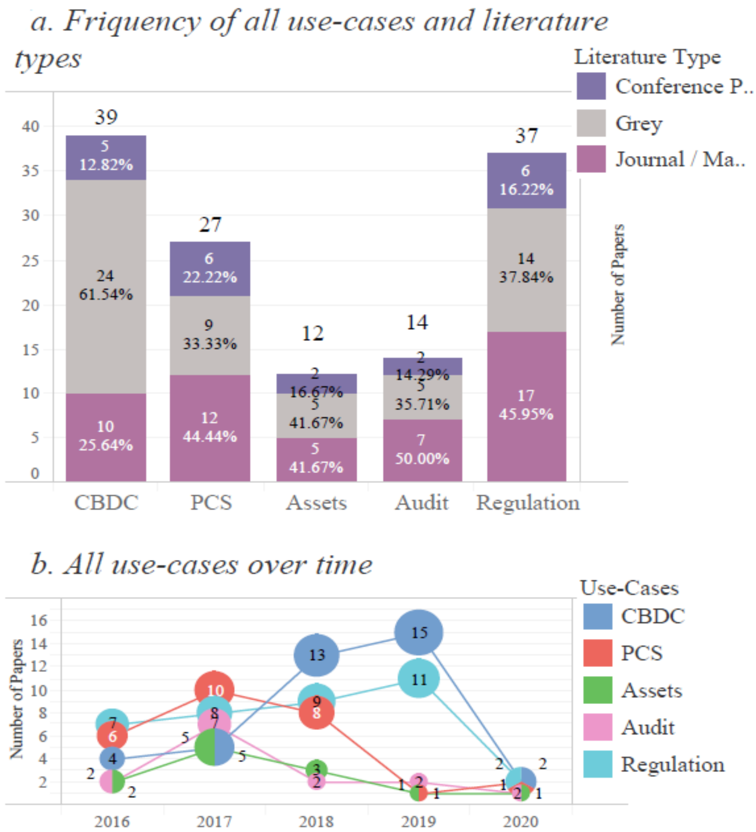


Figure 2.8: All Blockchain-Based Use-Cases for Central Banks.

Fig. 2.8 represents use-cases available in the included literature. Fig. 2.8b shows a distribution of identified use-cases for each year, represented and colour-coded in pie-bobbles for each use-case. The size of each bobble reflects a total count of use-cases in that bobble. It is evident from the data (Fig. 2.8a) that *CBDC* is the most widely investigated and reported central bank use-case for blockchain, with 39 papers (or 30.23% of total) examining it. The number of those use-cases available in the included literature has been steadily growing over the years (Fig. 2.8b), with four papers discussing it in 2016, five in 2017, 13 papers in 2018, peaking to 15 in 2019 and twice in 2020 so far. Over half of overall research on CBDC was provided through grey literature (Fig. 2.8 a) – 24 publications (or 61.54% of all CBDC research). The data indicates that interest in CBDC is growing and industry is leading and influencing the trend.

The second most popular use-case was *Regulation*, with 37 publications (or 28.68% of the total). The largest proportion of that research was communicated *via* academic journals (17 articles or 45.95%), with grey literature a close second. Over the years, availability of information on regulation in academic print was consistently growing, peaking to 11 in 2019, two more for the first half of 2020. The data suggests that interest in regulatory compliance is expanding and participation between industry and academia is more geared towards the academic side. *PCS* was researched 20.93% of the time, totalling 27 publications, where academic articles were leading the general trend (12 articles or 44.44%), with grey literature not far behind (9 papers or 33.33%). For this use-case, conference proceedings appeared to be proportionally popular, compared to some other use-cases, although still the least frequent venue for research communication. The year-on-year change for this use-case revealed that interest in this topic initially almost doubled from six papers in 2016 to 10 in 2017. 2018 provided eight papers, in 2019 there was only one publication available and two in 2020. The data indicates that, although PCS was a popular topic, participation of researchers is subsiding, potentially indicating underlying lack of interest and/or development of the gap in knowledge. Two further categories had the least overall coverage in the literature, with 9.30%, or 12 papers, for *assets transfer and ownership* and 10.85%, or 14 papers, for *audit trail*. Research on assets was evenly divided between academic articles and publications from industry, with little input from conference proceedings. The majority of information on the topic of audit trail was available through academic articles. The year-on-year trend for both of those use-cases was similar, peaking for both in 2017 and slowing down thereafter. The data indicates that although these two topics show some interest from researchers, interest seems to be lagging behind, unable to sustain an upward trend and potentially indicating another gap.

2.6.8 Opportunities and Risks From Blockchain Adaptation

This section provides a narrative summary of discussions in the included literature for each of the earlier identified use-cases answering questions of: 1)*why DLT was considered for each of those use-cases?* and 2) *what challenges the application of blockchain posed for those use-cases?*

RQ3: Why or why shouldn't blockchain be considered?

Central Bank Digital Currency (CBDC)

"Banknotes" and "commercial bank reserves/deposits" - are both a form of central bank money, which is the main form of the central bank's liability and underpin nearly all other forms of money in the economy. In the UK, over 98% of sterling payments, by value, are made electronically, with less than 2% made by banknotes, coins or cheques (Engert & Fung (2017)). BIS (Bech & Garratt (2017)) states that the only way for the general public to own central bank money is through physical cash. "If someone wishes to digitise that holding, they have to convert the central bank liability into a commercial bank liability (commercial bank money) by depositing cash in a commercial bank" (Bech & Garratt (2017)).

Why was blockchain considered for CBDC?
Currently, CBDC models receive more serious consideration (Murray (2019)) from the research community and central banks themselves. 28 included papers argue about the potential benefits from its hypothetical introduction, because CBDC is seen as a potential next milestone in the evolution of money (Griffoli et al. (2018)); it is believed to provide a more stable unit of account, a more efficient medium of exchange and a more secure store of value (Murray (2019), Bordo & Levin (2017)). The focus of many researchers is on its application to the domestic economy (Griffoli et al. (2018)), monetary supply-side considerations (Khiaonarong & Humphrey (2019)) and for promoting financial inclusion (Engert & Fung (2017), Andolfatto (2018), Koumbarakis & Dobrauz-Saldapenna (2019)) or as an enabler of cross-border payments (Han et al. (2019), Nabilou & Prum (2019)). Some explore whether the introduction of CBDC could improve the efficiency of fiat currency function (Fung & Halaburda (2016)) by providing a way to directly transfer central bank funds to households and firms (Engert & Fung (2017), Didenko et al. (2020)). Didenko et al. (2020) argue that the replacement of cash with a cash-like CBDC can lower the cost of maintaining

the supply of physical currency and protect it against counterfeiting (Didenko et al. (2020)). Thus, the social value of CBDC is believed to be in its ability to bring some of the anonymity of cash into the digital realm (Didenko et al. (2020)) or even blend the features of cash and deposits together (Agur et al. (2019)). In another example, after studying the macroeconomic consequences of issuing CBDC, the BoE states that its introduction could promote financial stability by permanently raising "GDP by as much as 3%, due to reductions in real interest rates" (Barrdear & Kumhof (2016)).

A large proportion of included papers focus on design features of a hypothetical CBDC because, from the perspective of central banks, the impact of CBDC introduction hinges on its design, country-specific economic and financial characteristics (Griffoli et al. (2018), Fung & Halaburda (2016), Didenko et al. (2020)) and reasons for its introduction (Meaning et al. (2018), Fung & Halaburda (2016)). For example, a CBDC designed to provide a secure payments service could serve a different core purpose to the one used "as an instrument of monetary policy" (Meaning et al. (2018)), or it could be designed in such a way as to blend a monetary and a payment systems into one (Didenko et al. (2020)) and it could have a "separate operational structure to other forms of central bank money", (BoE) (Kumhof & Noone (2018)).

In terms of payment economics, an important design consideration is what is verified on the blockchain – a token (an individual receiving a token will verify that the token is genuine (Barontini & Holden (2019))), or an account (an intermediary verifies the identity of an account holder (Barontini & Holden (2019))), i.e., an *account-based CBDC* versus *token-based CBDC*. A token-based CBDC could extend some of the attributes and functionality of cash for retail (transactions (Griffoli et al. (2018), Kumhof & Noone (2018), Auer & Böhme (2020), Didenko et al. (2020)) and could be made widely available to the public as a general-purpose currency (Murray (2019), Auer & Böhme (2020)). Universal access to this CBDC could be obtained through a digital signature and privacy will be ensured by default (Auer & Böhme (2020)). Khiaonarong & Humphrey (2019) believe that the role of cash in the economy should be maintained. However, CBDC could reduce the demand for cash (Hileman & Rauchs (2017)) or facilitate the gradual obsolescence of paper currency (Bordo & Levin (2017)), effectively reducing costs associated with maintaining a cash-based system (Hileman & Rauchs (2017), Koumbarakis & Dobrauz-Saldapenna (2019), Didenko et al. (2020)). These would be helpful in discouraging tax evasion, money laundering and other illegal activities. An account-based CBDC could be utilised with payments through the transfer of claims recorded on an account (Griffoli et al. (2018)). It is the preferred design choice of central banks (Griffoli et al. (2018), Bordo & Levin (2017), Murray (2019), Didenko et al. (2020)), because it could provide them with a more reliable real-time window on economic activity to guide monetary policy (Murray (2019), Koumbarakis & Dobrauz-Saldapenna (2019)).

Academics further categorise account-based design depending on who has access to CBDC. The difference here is between *retail CBDC* which is issued for the general public and *wholesale CBDC*, issued by financial institutions holding reserve deposits with a central bank (Shirai (2019), Murray (2019), Pfister (2019)). If anonymity is not seen as an issue, a central bank could provide bank accounts for the general public (Bech & Garratt (2017), Griffoli et al. (2018), Cœuré & Loh (2018)), in the same way deposit accounts are today (Andolfatto (2018)) – the retail CBDC. Those types of accounts could be made available through public-private partnerships with commercial banks or could be held by private individuals directly at the central bank itself (Bordo & Levin (2017)). "This is something that has been technically feasible for a long time, but which central banks have mostly stayed away from" (Andolfatto (2018)). This type of a central-bank-run system would provide convenience, resilience, accessibility (Auer & Böhme (2020)), opportunity to better track payments, making CBDC widely accessible, held by anyone for any purpose (Meaning et al. (2018)), with ease of use to peer-to-peer cross border payments (Han et al. (2019), Auer & Böhme (2020)). On the other hand, a wholesale CBDC issued for large-value wholesale interbank payments (Fung & Halaburda (2016), Cœuré & Loh (2018)) is considered when its design implementation can guarantee anonymity (Borgonovo et al. (2018)), provide restricted access to a predefined group of economic agents and is applicable to a limited range of purposes (Cœuré & Loh (2018), Shirai

(2019), Meaning et al. (2018), Pfister (2019)). This type of CBDC design could facilitate faster or immediate settlement (Griffoli et al. (2018), Fung & Halaburda (2016)) or extended settlement hours (Fung & Halaburda (2016)) and could be accessed more broadly than central bank reserves (Kumhof & Noone (2018)). Khiaonarong & Humphrey (2019) provides an example of the Bank of Canada exploration of DLT for digital representations of the Canadian dollar (called a digital depository receipt), used for wholesale payments. By improving efficiency and safety of both retail and large-value payment systems (Fung & Halaburda (2016), Nabilou (2019)), CBDC could aid central banks in easing liquidity pressures and potentially help to curtail bank runs (Griffoli et al. (2018), Brunnermeier & Niepelt (2019)).

Another reported design feature of an account-based CBDC was that it could allow for interest payments - the *interest-bearing CBDC*, supplied by a central bank under either a monetary quantity rule or a monetary price rule (Kumhof & Noone (2018)). When CBDC is designed as non-interest-bearing, its similarity to cash becomes the sole design choice (Agur et al. (2019)). If a return could be paid/earned on CBDC, the overall probability of its introduction increases (Borgonovo et al. (2018)), because an optimally designed interest-bearing CBDC could safeguard bank intermediation and protect the variety of payment instruments against network effects (Agur et al. (2019)). Furthermore, being a liability of a central bank (Shirai (2019)), CBDC could be backed on the asset side of the central bank's balance sheet by liquid federal government risk-free assets (Murray (2019)), thereby serving as a secure store of value with a rate of return (Bordo & Levin (2017), Pfister (2019)) different to the rate on reserves (Kumhof & Noone (2018)). By facilitating access to the balance sheet of a central bank, CBDC could promote contestability for banks and non-bank financial institutions (Engert & Fung (2017)). It does not have to dis-intermediate banks in any way (Andolfatto (2018), Brunnermeier & Niepelt (2019)). If an account-based interest-bearing CBDC is used by the general public as a viable option to bank deposits (Chiu et al. (2019), Pfister (2019), Didenko et al. (2020)) it could discipline behaviour of commercial banks (Chiu et al. (2019)), address competition problems in the banking sector (Kahn & Wong (2019), Koumbarakis & Dobrauz-Saldapenna (2019)) and compel commercial banks to raise their deposit rates (Andolfatto (2018)). If CBDC is used as reserves, it can increase overall lending by reducing a banker's costs of holding those reserves in central banks through increases in the CBDC rate paid on those reserves (Chiu et al. (2019)). As CBDC could also pay positive, zero or even negative rates at various points in the economic cycle (Andolfatto (2018)), it could be utilised as a tool for conducting monetary policy (Bordo & Levin (2017), Koumbarakis & Dobrauz-Saldapenna (2019)).

Lastly, the underlying architecture of CBDC could differ between *centralised*, *fully decentralised* and a *hybrid* system (Didenko et al. (2020)). A centralised system would be characterised by a permissioned blockchain, be account-based and provide direct access to a central bank, but lack cash-like qualities such as anonymous exchange. A decentralised CBDC could be based on a permissionless blockchain where full decentralisation is achievable through tokenisation and could offer cash-like features. A hybrid architecture is a blend of a centralised and decentralised CBDC. It may provide central bank accounts for financial intermediaries, where other participants could use intermediary services to access CBDC-takens; these could represent the drawing rights on the funds stored in the central bank accounts.

What challenges could the introduction of CBDC pose?

A total of 32 papers discuss the potential negative side of CBDC introduction, as a large proportion of researchers agree that its net benefits for financial stability are not as clear cut. This is because, while its adaptation could reduce some of the existing risks, other novel and unknown risks could emerge and it is not certain which would be greater (Barrdear & Kumhof (2016)).

Despite a range of pilot CBDC projects and theoretical studies Didenko et al. (2020), a high price volatility Ben Dhaou & Rohman (2018), Ducas & Wilner (2017), Koumbarakis & Dobrauz-Saldapenna (2019) and low level of acceptance of cryptocurrencies demonstrated that they fail to satisfy full requirements of fiat money in their current form. If a hypothetical CBDC is to be introduced, it is still unclear what role should be taken by a central bank and eventual intermediaries (Hileman & Rauchs (2017)). There is a possibility that CBDC introduction could create a parallel monetary system (Ben Dhaou & Rohman (2018), Nabilou & Prum (2019)), which could pose risks to the central bank monopoly over issuing base money (Nabilou (2019)). It is also uncertain whether CBDC should complement or serve as a substitute for existing central bank money (Hileman & Rauchs (2017)). If CBDC were to be designed as cash-like (Agur et al. (2019)), it may lead to the reduction in demand or disappearance of cash, thus lowering the variety of payment instruments available to households with diverse needs. Additionally, such a system creates a risk of permanent loss of funds if end users fail to keep their private key secret secure (Auer & Böhme (2020)). Furthermore, there are risks to price stability (Nabilou (2019)), to smooth operation of payment systems (Tsai et al. (2016), Nabilou (2019)) and to the conduct of monetary policy (Nabilou (2019), Nabilou & Prum (2019)). CBDC also could have a negative effect on seigniorage, the interest rates (Engert & Fung (2017)) as well as face numerous legal challenges (Nabilou (2019), Nabilou & Prum (2019)).

The next important challenge, reported in the included papers was to establish who should get access to CBDC in the first place: only commercial banks, financial institutions in general or even citizens (Hileman & Rauchs (2017)). If CBDC was to be issued to the general public, who should run the nodes (end users or money providers) and how should off-line payments be processed. Some researchers believe that disintermediated public access to the central bank balance sheet *via* interest-bearing CBDC could result in destabilising consequences for the banking sector (Nabilou (2019), Murray (2019), Agur et al. (2019)). As a competitive and safe and convenient alternative to commercial bank deposits (Griffoli et al. (2018), Murray (2019), Koumbarakis & Dobrauz-Saldapenna (2019), Pfister (2019), Agur et al. (2019)), CBDC would likely to have a disruptive effect on the financial stability of credit institutions and key financial market infrastructures (Murray (2019), Cœuré & Loh (2018)) with contagion to the overall financial system (Nabilou (2019)). Disruption of a commercial bank's business model (Murray (2019) could lead to adverse consequences for the real economy (Seretakis (2019)). With a sufficiently high CBDC interest rate (Chiu et al. (2019)) commercial bank reliance on customer deposits as a major source of their funding may become less stable (Griffoli et al. (2018), Agur et al. (2019)) and more expensive (Chiu et al. (2019), Griffoli et al. (2018), Koumbarakis & Dobrauz-Saldapenna (2019)), leading to additional reductions in lending activity (Chiu et al. (2019), Seretakis (2019), Koumbarakis & Dobrauz-Saldapenna (2019), Agur et al. (2019)) or increased lending rates to general public (Griffoli et al. (2018), Koumbarakis & Dobrauz-Saldapenna (2019)). Since, in times of financial distress, commercial customer deposits could far more easily take flight to a central bank (Cœuré & Loh (2018), Nabilou (2019), Koumbarakis & Dobrauz-Saldapenna (2019), Agur et al. (2019), Nabilou & Prum (2019)), CBDC could act as an accelerant of bank runs (Cœuré & Loh (2018), Murray (2019), Griffoli et al. (2018), Koumbarakis & Dobrauz-Saldapenna (2019), Nabilou & Prum (2019), Didenko et al. (2020)), "transforming an isolated concern about one bank's solvency into a system-wide crisis" (Murray (2019)).

Key characteristics of current blockchain architectures, i.e., the anonymity of a beneficial owner of CBDC (Engert & Fung (2017), Berentsen & Schar (2018)) – is another reported red flag of CBDC design. If there were to be interest payments/charges on CBDC holdings (Dow (2019)), it would be impossible for a central bank to sustain that owner's anonymity, because the holder would need to be identified for income tax purposes. Overall, a central bank cannot issue CBDC in the sense of a truly decentralised and permissionless asset that permits its users to remain anonymous (Berentsen & Schar (2018), Didenko et al. (2020)), because anonymous CBDC would facilitate criminal activity (Engert & Fung (2017)) leading to high reputational risks for central banks. Additional restrictions and compliance costs would have to be imposed (Engert & Fung (2017), Koumbarakis & Dobrauz-Saldapenna (2019)), such as KYC (Know-Your-Customer), AML (Anti-Money-Laundering) and CFT (Counter-Terrorist Financing) (Berentsen & Schar (2018), Koumbarakis & Dobrauz-Saldapenna (2019)). Another issue is that novel CBDC system will have to contend with operational and security risks arising from technological disruptions (Griffoli et al. (2018), Mills et al. (2016), Ben Dhaou & Rohman (2018), Benos et al. (2017), BIS (2017), Berentsen

& Schar (2018), Didenko et al. (2020)). Overall, there is an agreement in the research community that there is a need for more research on the impact of a potential deployment of CBDC on monetary policy and financial stability (Hileman & Rauchs (2017), Didenko et al. (2020)). More work is required to assess the full potential of CBDC (Cœuré & Loh (2018)), its technological feasibility and operational costs (Griffoli et al. (2018), Koumbarakis & Dobrauz-Saldapenna (2019), Didenko et al. (2020)) and country-specific circumstances (Griffoli et al. (2018)). There is a growing consensus amongst researchers that, due to outstanding uncertainties regarding the design and architecture of the CBDC systems (Hileman & Rauchs (2017), Didenko et al. (2020)), to technical constraints Shirai (2019) of current blockchain architectures and maturing technology (Barrdear & Kumhof (2016), Berentsen & Schar (2018), Hayes (2016), Pinna & Ruttenberg (2016), Chiu (2017), Mills et al. (2016), Ben Dhaou & Rohman (2018), Hileman & Rauchs (2017), Bordo & Levin (2017), Benos et al. (2017), Kumhof & Noone (2018), Meaning et al. (2018), Griffoli et al. (2018), Cœuré & Loh (2018), Murray (2019), Arthur et al. (2018), Engert & Fung (2017)), it is too early to draw firm conclusions on the real benefits of CBDC (Griffoli et al. (2018), Engert & Fung (2017), Didenko et al. (2020)). Today, the general view remains that such a move towards CBDC adoption would be premature (Murray (2019)) and the risks connected with issuing CBDC would outweigh the potential benefits for society (Khiaonarong & Humphrey (2019)); currently, no central bank has a live and operating CBDC system (Hileman & Rauchs (2017), Kumhof & Noone (2018), Shirai (2019)).

Payment Clearing and Settlement Systems Operated by Central Banks

Central banks play a fundamental role in supporting, regulating and supervising payment systems, because such infrastructure stands at the core of monetary and financial systems by creating a linkage between them (Didenko et al. (2020)). "In its simplest form, the PCS of a financial transaction, regardless of the asset type, requires: 1) a network of participants, 2) an asset or set of assets that are transferred among those participants, and 3) a transfer process that define the procedures and obligations associated with transactions" (Mills et al. (2016)). Central banks facilitate settlement using central bank accounts to ensure finality (Didenko et al. (2020)).

Why was blockchain considered for PCS systems of central banks?

Overall, 31 papers argue about the potential benefits of hypothetical blockchain-based PCS systems, operated by central banks. Some researchers believe that DLT has the potential to improve efficiency (Benos et al. (2017), Didenko et al. (2020)) and bring greater value (Cocco et al. (2017)) to PCS (Raskin & Yermack (2018)) by modernising financial market infrastructure (Seretakis (2019), Didenko et al. (2020)) and revolutionising the underlying technology (Guo & Liang (2016)) underpinning those processes today.

Researchers argue that DLT is capable of enhancing service and overall operational efficiency, safety (Guo & Liang (2016), Didenko et al. (2020)) and global reach (Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)) of *Interbank Payment Systems (IBPS)* (Fung & Halaburda (2016), Wu & Liang (2017), Hileman & Rauchs (2017), Yoo (2017), Zhai & Zhang (2018), Wang et al. (2018)) for large-value wholesale payments (Fung & Halaburda (2016), Bech & Garratt (2017), Kang & Lee (2019), Didenko et al. (2020)). By allowing point-to-point transmission (Guo & Liang (2016)) and straight-through processing of global (Yoo (2017), Lipton (2018)) financial transactions (Chiu (2017)), DLT could reduce complexity for multiparty, cross-border (Mills et al. (2016)) inter-bank payments and settlements (Khiaonarong & Humphrey (2019)) and allow for transfers in multiple currencies with the use of a single transaction system (Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)). Blockchain could enhance efficiency of IBPS for cross-bank money transfers (Zhai & Zhang (2018)) by speeding them up (Ducas & Wilner (2017), Yoo (2017), Didenko et al. (2020)) to near-real-time updating and 24x7x365 processing (Milne (2018), Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)). Also, for general application of Real Time Gross Settlement (RTGS), CBDC could be utilised to make it open access, allowing any financial agent to settle large value payments, achieving finality in virtually real-time (Engert & Fung (2017), Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017), Milne (2018), Didenko et al. (2020)). For the inter-bank market, each bank can be a participant in DLT-PCS and take part in the consensus process (Wu & Liang (2017)), thus eliminating the intermediary link of third-party financial institutions (Guo & Liang (2016), Hileman & Rauchs (2017), Sun et al. (2017), Didenko et al. (2020)). Blockchain can also eliminate the need for centrally maintained back-up systems (Benos et al. (2017), Didenko et al. (2020)), by creating decentralised, technology-led, automated IBPS (Chiu (2017)) that no longer require reconciliation between different databases (Wu & Liang (2017), Didenko et al. (2020)). By tracking on blockchain (Zhai & Zhang (2018), Didenko et al. (2020)), a central bank can oversee (Hayes (2016)) payments, settlement and remittance transfers (Hileman & Rauchs (2017), Yoo (2017)) of inter-bank cash flow Zhai & Zhang (2018) and ensure a delivery-vs-payment (DvP) by linking transfers of assets with payments (Chiu & Koeppl (2019), Benos et al. (2017)).

Another potential benefit of DLT cited by researchers is that it could streamline a post-trade value chain (Pinna & Ruttenberg (2016)) by simplifying and automating many of the processes currently involved in the post-trade cycle (Benos et al. (2017)) such as clearing and settlement. For clearing, there is an opportunity to speed it up to almost immediate (Bech & Garratt (2017)), where a collection of DLT nodes could clear payments on a continuous basis (Tsai et al. (2016)). As an example, Tsai, Deng, Ding & Li (2018) propose a framework for a permissioned multi-blockchain clearinghouse that could be shared with exchanges, banks and regulators, thus providing redundancy, high speed processing and scalability. In relation to the inter-bank settlement of assets, issued and controlled by a central bank, DLT could reduce back-office costs by automating various settlement processes (Chapman et al. (2017)). It could enhance settlement efficiency (Cœuré & Loh (2018), Bech & Garratt (2017), Benos et al. (2017)) and simplify procedures by reducing the number of intermediaries (Priem (2020)) because blockchain is capable of facilitating direct connection (Ducas & Wilner (2017)) between transacting parties. DLT can enable settlement to occur through consensual reallocation of the balances (Tsai et al. (2016)) as a decentralised settlement of a transaction could be simultaneous with the validation process (Priem (2020)). Furthermore, blockchain can improve end-to-end duration of the settlement cycle (Mills et al. (2016), Benos et al. (2017)), where transactions could happen in almost real-time and peer-to-peer (Wu & Liang (2017), Didenko et al. (2020)). It could additionally provide a more flexible settlement (Chiu & Koeppl (2019), Benos et al. (2017)), by extending settlement hours (Fung & Halaburda (2016)) or shortening settlement periods (Priem (2020), Ducas & Wilner (2017), Benos et al. (2017), Engert & Fung (2017), Bech & Garratt (2017), Milne (2018), Chiu & Koeppl (2019), Fung & Halaburda (2016)) from the current standard of 'trade date plus three days' (T+3), to near instantaneous settlement (T+0) (Ducas & Wilner (2017), Benos et al. (2017), Bech & Garratt (2017), Priem (2020), Chapman et al. (2017)). Faster transfer will allow participants of inter-bank market to receive funds and securities more quickly, freeing up liquidity that could be tied up in collateral (BIS (2017)). Improved availability of assets and funds (Mills et al. (2016)), could illuminate the shortcomings of fractional reserve banking (Seretakis (2019)), by facilitating more effective use of collateral and regulatory capital (Pinna & Ruttenberg (2016)), such as central bank reserves used for settlement of inter-bank payments (Milne (2018)). These provide a real opportunity to address the separation between transactions (such as securities or derivatives transactions) and payment for those transactions, particularly at the wholesale level (Didenko et al. (2020)).

The challenges that blockchain-based PCS systems could face?

The multitude of possible designs for DLT is an indication that a one-size-fits-all approach is not appropriate for addressing the broad range of challenges in payment, clearing and settlement (BIS (2017), Didenko et al. (2020)). 15 papers discuss the potential negative implications of blockchain adaptation for PCS. Researchers are sceptical about full substitution (Chiu (2017)) of existing and well-established PCS process by currently available DLT architectures and protocols. High barriers to entry were explicitly recognised as a critical factor influencing the adoption of DLT for PCS processes (Priem (2020), Mills et al. (2016)). In jurisdictions where banks have already built collective infrastructure (Chiu (2017)) for PCS, the lack of incentive for alternative equivalent systems arises due to the inefficiencies of high set-up cost, duplication (Chiu (2017)) and because of its already existent network effects (Raskin & Yermack (2018), Mills et al. (2016), Hileman & Rauchs (2017)). Building these new networks through an alliance of incentives of different participants is a challenging task (Hileman & Rauchs (2017)). The rationale for it is that the creation of new networks of participants in such settings requires each party to give up some amount of existing control, combined with an unwillingness to change well-established business processes at their respective institutions (Hileman & Rauchs (2017)). The banking industry will require an uptake from a critical mass of those participants for any application of new technology to be successful (Mills et al. (2016)).

Furthermore, faster blockchain-based PCS processing, reduced reconciliation work and real- or near-real-time transaction time (BIS (2017)) will remove net benefits that clearing provides, thereby increasing the 'spot liquidity' demand for settlement (Benos et al. (2017)). A real-time (T+0) cycle would require prepositioning of cash or securities (collateral) in advance of a trade (Benos et al. (2017)), thus increasing credit and liquidity needs associated with payment, clearing and settlement activity (BIS (2017), Benos et al. (2017)). Another important issue is that an ultimate settlement of sovereign-backed currency, in accounts held at a central bank is fundamental to social confidence and trust (Chiu (2017)). A blockchain-based settlement is probabilistic (BIS (2017), Chapman et al. (2017)) - in other words, the payment is therefore never fully settled because there is always a small probability that the payment could be reversed (Chapman et al. (2017)) due to forking (Benos et al. (2017), Chiu & Koeppl (2019), Raskin & Yermack (2018)).

It is also suggested by some researchers that operational capacity and performance-based scalability of current blockchain designs is a further concern (Hileman & Rauchs (2017), Ben Dhaou & Rohman (2018), BIS (2017)). This is based on limits of the size of the blocks in a blockchain (Geva (2018)). As only a limited number of simultaneous transactions can be written into the blockchain at any given time, a block's capacity to grow and accommodate more interactions is not promising (Ducas & Wilner (2017)). Current PCS systems are capable of handling a significant fluctuation in volume of transactions, which impose a requirement on blockchain-based PCS systems to be operationally scalable to accommodate processing large daily volumes and peak volumes in times of market distress or volatility (BIS (2017)). As hundreds of millions of daily transactions are processed through current PCS (Mills et al. (2016)), any novel system that fails to meet these requirements will weaken the safety of PCS system activity (BIS (2017)). Moreover, when DLT-based settlement is compared with existing centralised RTGS, BIS highlights that it may take longer to achieve settlement on blockchain, thus actually decreasing the speed of transactions (Sun et al. (2017), Benos et al. (2017)). This is because technically, to update and synchronise state changes to a ledger, the process for validating a transaction and reaching a consensus across all nodes in DLT is potentially more complex than with a centralised entity (BIS (2017)). Combined with cryptographic verification, such settings introduce latency and limit the number of transfers that DLT can process concurrently (Mills et al. (2016)).

Another essential requirement of any PCS system is trade matching of transactions over a large number of attributes with complex rules and cross-dependencies (Benos et al. (2017)). Blockchain does not necessarily have the functionality to compare different data domains, to address contract mismatches or to process exceptions (Benos et al. (2017)). Furthermore, operational settlement becomes even more complex if it involves delivery-vs-payment (DvP), payment-vs-payment (PvP) systems (Geva (2018)) or delivery of one asset against another (BIS (2017)) and so on. "Central matching may continue to be required as pre-ledger processing" (Benos et al. (2017)), because in arrangements involving an exchange of value, multiple financial market infrastructure is typically involved (BIS (2017)). Hence, certain processes of the post-trade cycle in the securities markets will still require involvement of intermediary institutions, irrespective of the market players involved and technology used (Pinna & Ruttenberg (2016)). Despite the need for immutability that stems from irreversibility of a blockchain, there are further issues identified with self-executing code (BIS

(2017), Benos et al. (2017), Ben Dhaou & Rohman (2018)) where mistakes in coding may need to be corrected (BIS (2017), Didenko et al. (2020)). In PCS systems, there is a requirement for error management (Benos et al. (2017), BIS (2017), Mills et al. (2016), Didenko et al. (2020)) in circumstances such as inadvertent errors (BIS (2017)), e.g., mistaken or unauthorised payments (Geva (2018)). Also, there are requirements for maintenance (Chiu (2017)) of PCS, management of technological failures or misuse (Chiu (2017)) and fraud (BIS (2017), Mills et al. (2016), Didenko et al. (2020)), as currently existing and well-established PCS systems secure public interest objectives in stability and anti-abuse and are subject to regulation as a critical financial market infrastructure (Chiu (2017)).

Asset Transfer and Ownership

Comparatively smaller numbers of papers available in the included literature discuss how introduction of blockchain could affect current processes for asset transfer and ownership in central banks.

Why was blockchain considered for asset transfer and ownership of central banks?
Nine publications deliberate on the potential improvements from blockchain. The capabilities of DLT such as its ability to provide record-keeping, storage and transfer of any type of asset (such as securities, commodities, derivative transactions and so on), make it asset-agnostic (Mills et al. (2016)). A key innovation of blockchain is that it can offer, *via* a shared database (Chiu & Koeppl (2019)), a time-ordered and immutable record of transactional history (Milne (2018), Priem (2020)), security ownership (Priem (2020), Mills et al. (2016), Hileman & Rauchs (2017)) and all transfers among all participants in the payment system (Ducas & Wilner (2017), Mills et al. (2016)), which can be updated without relying on multiple, specialised intermediaries or a third-party infrastructure (Chiu & Koeppl (2019)). When financial institutions trade with each other through IBPS, all relevant counterparties would have a copy of that ledger. These could also involve asset issuance and servicing (Tsai et al. (2016), Ducas & Wilner (2017)) such as creation of assets, enablement of trading between partners and liquidation of positions (Tsai et al. (2016)). For example, Chen et al. (2018) outlined a blockchain-based financial product information management platform that allowed for multi-institutional update of multi-dimensional and diversified financial product information.

There are also implications when protection of business sensitive information, such as the appropriate level of information is shared on the ledger and which participants have the ability to read or write to (Mills et al. (2016)). Even if all nodes have a complete copy of the ledger, it is technologically possible that some of the data on the ledger is encrypted so that only authorised participants can decrypt and read the underlying information. This way, the system could facilitate a tamper-resistant (Chen et al. (2018)) direct ownership (Mills et al. (2016)), reducing intermediation costs for investors, together with legal, operational and overall systemic risks (Milne (2018)). "This will improve accounting, auditing and regulatory supervision functions while increasing transparency of ownership" (Ducas & Wilner (2017), Benos et al. (2017), Chen et al. (2018)).

What challenges could blockchain-based asset transfer and ownership impose?
There are also four papers that discuss potential issues with utilising DLT for assets transfer and ownership. The concern is whether a DLT entry legally constitutes a proof of ownership. For ultimate and legal settlement, there must be a formal, i.e., a legally defined indication of transfer of ownership, once securities and cash have changed hands (Benos et al. (2017)). There is uncertainty in regards of legal validity of financial instruments issued on a DLT, because such legal ownership (Seretakis (2019)) is not defined and elaborated on by law (Geva (2018)) and not assured by the regulators and supervisors (Seretakis (2019)). Proprietary rights (BIS (2017), Geva (2018), Seretakis (2019)) and obligations, associated with DLT representation of assets (Seretakis (2019)), as well as the liabilities and enforceability (BIS (2017)) of the rights of transacting parties are unclear (Geva (2018), BIS (2017)). However, it is a legal requirement for those to be articulated

clearly, understood by all participants and supported by applicable law (BIS (2017)). "As things stand now, there is not even a standard satisfactory definition as to what constitutes a digital asset, not to mention an elaboration of its relationship to the physical asset it represents" (Geva (2018)). Furthermore, for transactions that take place across borders or in multiple jurisdictions (BIS (2017), Benos et al. (2017)), there are currently no laws that underpin the activity "in ways that are mutually compatible" (BIS (2017)). "Decentralisation further challenges traditional methods of the enforcement of ownership judgment, as well as of a security interest, because, without the cooperation of the owner of an asset, placed on the blockchain, the asset may not be accessible" (Geva (2018)).

Audit Trail

There are several papers that provide a high-level discussion on potential improvements or limitations to current auditing practices from blockchain innovation in central banking settings.

Why was blockchain considered for audit trail of central banks?
Thirteen publications mention some potential improvements. Researchers argue that blockchain can enhance audit and regulatory functions (Ducas & Wilner (2017)) by providing the opportunity to monitor, supervise and audit trades and agreements in real-time, which drastically improves regulatory systems in place today (Hileman & Rauchs (2017)) and assists central banks with their supervision role (Hileman & Rauchs (2017), Kavassalis et al. (2018)). The global shared audit log (Mills et al. (2016), Hileman & Rauchs (2017)), provided by the use of a DLT ensures the integrity of records through the integrity of the ledger itself (Benos et al. (2017)). Another cited advantage is reduction of reconciliation cost (Hileman & Rauchs (2017), Benos et al. (2017), Pinna & Ruttenberg (2016), Mills et al. (2016)). The majority of back office costs are tied to manual reconciliation of conflicting trade data (Priem (2020)). Blockchain promises to eliminate manual reconciliation processes (Benos et al. (2017), Grody (2018)) across multiple record-keeping infrastructures (Mills et al. (2016)) of many of the hundreds of data intermediaries (Grody (2018)) that play a significant role in reconciling costly and potentially conflicting, risk prone non-standard data (Grody (2018), Priem (2020)) in different locations by automating that reconciliation. Moreover, the immutable (Mills et al. (2016), Benos et al. (2017), BIS (2017)), tamper-resistant (Mills et al. (2016), Wu et al. (2019)) nature of the DLT enables greater transparency (Hileman & Rauchs (2017), Kavassalis et al. (2018), Benos et al. (2017)) and traceability (Hileman & Rauchs (2017), Wang et al. (2018), Kavassalis et al. (2018), BIS (2017), Benos et al. (2017), Chapman et al. (2017), Wu et al. (2019)) of history of any flow of funds or securities (Benos et al. (2017), Wang et al. (2018)), where data cannot be unilaterally changed once recorded. Immutability is crucial for safety as it relates to data integrity (BIS (2017)) and gives participants the assurance that everyone is storing, seeing, using and processing the same data as everyone else (Hileman & Rauchs (2017), Ducas & Wilner (2017)). As any amendments to the ledger are traceable (Benos et al. (2017)), there is a possibility of reduction of data falsification and manipulation (Ben Dhaou & Rohman (2018)) resulting in reduction of the risk of fraud (Hileman & Rauchs (2017), Ducas & Wilner (2017)). While this refers mostly to the payment systems currently operated by central banks, it could, in theory, be extended to any DLT-based system to which central banks would be granted access to, such as internal bank ledgers (Hileman & Rauchs (2017)).

What challenges could blockchain-based audit trail impose?
Four publications also mention issues associated with DLT and audit. Although it would be expected that integrity of records in the ledger is ensured by the integrity of the ledger itself, a trusted body may still be needed to guarantee the validity (Benos et al. (2017)) of that data. The reason for this is because the existing legal regime cannot assure the reliability of those records (Seretakis (2019)) and that the entered common information is correct (Mills et al. (2016)) when large number of participants have an ability to write to the ledger without some kind of supervision.

The decisions on who provides and how to provide accuracy checks on information stored in the system (Mills et al. (2016)) still requires regulation to accommodate record-keeping and to provide for the reliability and authoritativeness of those records (Seretakis (2019)) on blockchain.

Regulatory Compliance

A comparatively larger proportion of publications provide a discussion on various aspects of regulatory compliance of blockchain adaptation for central banks. More specifically, on blockchain's intersection with regulatory compliance, CBDC impact onto Monetary Policy and regulation of blockchain-based PCS systems of central banks.

Why was blockchain considered for regulatory compliance?
Overall, 25 papers present thoughts on general aspects of regulatory improvements through blockchain. Adoption of blockchain for central banking business depends on its ability to comply with the existing regulatory framework (Seretakis (2019), Nabilou & Prum (2019)); therefore wider financial industry participants ask for updates in regulatory guidance and legal structure (Priem (2020), Dow (2019)). Researchers debate on how to facilitate "embedded supervision" (Auer (2019)) by automating mandatory regulatory reporting (Micheler & Whaley (2019)), a process which is currently complex and tedious (Hileman & Rauchs (2017), Auer (2019)). Central banks foresee the potential of DLT to ease regulatory compliance (Micheler & Whaley (2019)), e.g., automatically enforce market regulation (Hileman & Rauchs (2017), Ducas & Wilner (2017)). To create an algorithmic, rules-following monetary policy (Hayes (2016)) regulators could participate as a node in DLT (Tsai, Deng, Ding & Li (2018)) and have full authority to set initial blockchain rules, the right to veto against existing blockchain codes and the power to enforce, update and change rules when necessary (Ozili (2019)).

Automation of 'terms and conditions' of legally binding agreements could reduce some legal risks (BIS (2017)). To achieve those goals, researchers argue for development of shared technical interoperability standards (BIS (2017), Guo & Liang (2016), Mills et al. (2016), Benos et al. (2017), Priem (2020), Hileman & Rauchs (2017), Grody (2018), Ducas & Wilner (2017), Kavassalis et al. (2018), Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017), Caytas (2016)) which could provide a base layer of connectivity; this could help lower implementation and integration costs (BIS (2017)), halt avoidance of regulatory arbitrage (Ducas & Wilner (2017)) and provide access to more granular standardised transactional data (Grody (2018), Kavassalis et al. (2018), Micheler & Whaley (2019)). A current absence of standardization still makes necessary and important the manual post-trading validation processes (Priem (2020)). Overall, establishment of technical standards may encourage broader adoption of DLT in the financial system that could potentially bring network scale efficiencies (BIS (2017)). In combination with cost-effective and secure data storing solutions (Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)), there is an opportunity to facilitate quicker reconciliation, reduce data discrepancy and demanding back office activities (BIS (2017)) important for regulatory reporting. Additionally, some papers propose establishment of a regulatory "sandbox" model (Guo & Liang (2016), Tsai, Zhao, Zhang, Yu & Deng (2018), Ducas & Wilner (2017)) as a facilitative approach to FinTech; this eases regulation in the testing, development and partial delivery to the public of new technologies, promoting the most suitable approach to regulating blockchain technologies (Ducas & Wilner (2017)).

The traceability feature of blockchain could potentially reduce the risk of fraud (Ducas & Wilner (2017)) by designing a legal framework (Nguyen (2016), Ducas & Wilner (2017)) for automating the connection of real-world identities to cryptographic identities in a database (Ducas & Wilner (2017)) for customer protection, KYC rules (BIS (2017), Cœuré & Loh (2018), Ducas & Wilner (2017), Lipton (2018), Guo & Liang (2016)), AML (BIS (2017), Cœuré & Loh (2018), Ducas & Wilner (2017), Lipton (2018)), CFT regulations (BIS (2017), Cœuré & Loh (2018), Ducas & Wilner (2017)), tax, capital and credit management (Nguyen (2016), Cœuré & Loh (2018), Ducas & Wilner (2017), BIS (2017), Guo & Liang (2016)) and overall monetary policy (Nguyen (2016), Cœuré & Loh (2018), Ducas & Wilner (2017), BIS (2017)). This would remove duplication effort in identification across institutions and enable encrypted sharing (Ducas & Wilner (2017), Nguyen (2016), Shah & Jani (2018), Tsai et al. (2016)).

What challenges could blockchain pose to regulatory compliance?

A total of 18 papers discuss some regulatory frictions from DLT. First, traceability should be weighed against privacy and the need to keep certain information confidential (BIS (2017), Cœuré & Loh (2018), Ducas & Wilner (2017)). On a blockchain, all information in the ledger is typically observed by all participants (Priem (2020), Chapman et al. (2017)). When such arrangements are applied to financial markets, this information transparency might cause privacy loss, confidentiality or competition issues (Priem (2020), Murray (2019), Ben Dhaou & Rohman (2018), Hileman & Rauchs (2017)) and should be balanced against data protection and applicable privacy laws (BIS (2017), Priem (2020), Mills et al. (2016), Benos et al. (2017), Chapman et al. (2017), Seretakis (2019)), such as the General Data Protection Act (GDPR), the Bank Secrecy Act (BSA) or others. Furthermore, blockchains of today are incapable of being influenced by governmental controls and provide limited access to regulators - read-only mode (Ben Dhaou & Rohman (2018)). In such a setting, the governance and regulatory enforcements are solely concentrated in the hands of coding experts who do not usually possess governance expertise in areas of risk location and determination, consumer protection rights, financial and legal expertise (Chiu (2017)) etc.

Blockchain application for central banking business potentially generates new services and involves new players (Geva (2018), Mills et al. (2016)) and therefore creates new legal issues (Nabilou (2019), Benos et al. (2017), Hileman & Rauchs (2017), BIS (2017), Caytas (2016), Nabilou & Prum (2019)) that require additional supervision (Sun et al. (2017)). Current regulation and supervisory policies that govern financial systems and the prevailing financial market architecture are not generally intended to favour a particular electronic technology (Mills et al. (2016)); unclear regulatory environment (Hileman & Rauchs (2017)) is one of the important reasons preventing blockchain from adoption (Nguyen (2016)). Another important issue is that today, the interdependence of existing financial systems suggests that issues arising in any one area of the wider banking ecosystem could result in the transmission of risk to other financial market infrastructures, leading to systemic damage at national and even international levels (Ducas & Wilner (2017)). There is a diverse set of participants interacting within a single financial market or across different financial markets (Mills et al. (2016)). Because of this interdependence of legacy payment systems, adaptation of blockchain-based solutions for one area of central banking business could interrupt existing processes (Nguyen (2016)) and drastically affect a wide range of interconnected financial markets and infrastructures, including payment systems, stock exchanges, central securities depositories, securities settlement systems, trade repositories and others (Ducas & Wilner (2017)). Moreover, interoperability across blockchains (Mills et al. (2016), Hileman & Rauchs (2017)) or between DLT and legacy systems (Mills et al. (2016), Priem (2020), Benos et al. (2017), Chiu (2017)) is crucial to the efficient functioning of the wider financial system (Mills et al. (2016)). Currently, interoperability is still in its infancy (Hileman & Rauchs (2017)) and the risks are further enhanced by the technological complexity of blockchain systems, including use of strong encryption, decentralised governance structures and its status as software (Ducas & Wilner (2017)). Furthermore, as market participants are developing their own niche DLT systems (Priem (2020)), the current landscape is fragmented and comprises a variety of incompatible protocols (Hileman & Rauchs (2017)) leading to additional complexity, costs (Mills et al. (2016)) and operational risks, due to incompatibility issues (Priem (2020)). Should widespread implementation of these systems occur, the International Monetary Fund (IMF) (Ducas & Wilner (2017)) warns, scenarios where blockchain technologies become simultaneously “too big to fail, yet too complex to resolve”, could potentially arise (Ducas & Wilner (2017)).

Monetary Policy and CBDC

Monetary Policy is the macroeconomic policy laid down by the central bank. It involves management of money supply and interest rates and is used by the government of a country to achieve macroeconomic objectives like inflation, consumption and liquidity growth. In total, 14 papers elaborate on positive implications from introduction of CBDC onto Monetary Policy operations. Overall, CBDC is seen by researchers as an appropriate policy response to payment innovations (Fung & Halaburda (2016), Koumbarakis & Dobrauz-Saldapenna (2019)), because a CBDC-based monetary policy framework could foster true price stability (Bordo & Levin (2017), Nabilou & Prum (2019)) by simplifying (Berentsen & Schar (2018)) and facilitating systematic and transparent conduct of it (Bordo & Levin (2017), Berentsen & Schar (2018)). CBDC could be utilised as an additional monetary policy tool (Cœuré & Loh (2018), Dow (2019), Barrdear & Kumhof (2016)) that could strengthen monetary transmission mechanisms and simplify conduct of monetary policy (Nabilou & Prum (2019)), because a central bank could use it as a transmission channel and directly manipulate account holder balances (Seretakis (2019), Murray (2019), Cœuré & Loh (2018), Meaning et al. (2018), Koumbarakis & Dobrauz-Saldapenna (2019)). Account-based CBDC could support unconventional monetary policy (Engert & Fung (2017)) such as Quantitative Easing (QE) (Meaning et al. (2018)), contribute to the stabilisation of the business cycle (Barrdear & Kumhof (2016)) or bring fiscal advantages relating to seigniorage (Dow (2019), Engert & Fung (2017)). Another example: a central bank could commit to an algorithmic rate of money creation (Cœuré & Loh (2018), Koumbarakis & Dobrauz-Saldapenna (2019), Pfister (2019)) by directly manipulating account balances of electronic central bank money and/or the aggregate quantity of that money (Meaning et al. (2018)) through precise control over interest rates (Raskin & Yermack (2018), Murray (2019), Berentsen & Schar (2018), Barrdear & Kumhof (2016)) or overnight inter-bank rates (Murray (2019)), thus addressing or removing the limitations of the Zero Lower Bound (ZLB) on those rates (Griffoli et al. (2018), Cœuré & Loh (2018), Dow (2019), Engert & Fung (2017), Pfister (2019), Nabilou & Prum (2019)).

In addition, 23 papers debate the range of challenges to architectures and operations of Monetary Policy from CBDC introduction. Because a monetary regime with CBDC has never existed (Barrdear & Kumhof (2016)) and technology to make it feasible and resilient have not been available (Barrdear & Kumhof (2016), Berentsen & Schar (2018), Hayes (2016), Pinna & Ruttenberg (2016), Chiu (2017), Mills et al. (2016), Ben Dhaou & Rohman (2018), Hileman & Rauchs (2017), Ducas & Wilner (2017), Benos et al. (2017), Kumhof & Noone (2018), Meaning et al. (2018), Griffoli et al. (2018), Cœuré & Loh (2018), Murray (2019), Arthur et al. (2018)), it is difficult to predict an impact of CBDC (Engert & Fung (2017), Griffoli et al. (2018), Pfister (2019)) on the monetary transmission mechanism (Meaning et al. (2018)). From a monetary policy perspective, CBDC could provide a dangerous widespread balance sheet exposition of an economy (Meaning et al. (2018)). Also, its introduction might unexpectedly affect the size and composition of the balance sheets of central banks, commercial banks, non-bank financial institutions, households and firms (Kumhof & Noone (2018), Meaning et al. (2018)). It is also unclear how CBDC could affect a money supply and which algorithm or regulator/authority/group of entities would control the issuance of CBDC (Didenko et al. (2020)). A central bank introducing CBDC would additionally face legal challenges (Nabilou (2019), Nabilou & Prum (2019), Didenko et al. (2020)) and have to ensure the fulfilment of AML/CFT requirements, as well as satisfy the public policy requirements of other supervisory and tax regimes (Cœuré & Loh (2018), Koumbarakis & Dobrauz-Saldapenna (2019), Didenko et al. (2020)). Every jurisdiction considering a CBDC should carefully consider the implications before making any decision (Cœuré & Loh (2018), Didenko et al. (2020)). "There is very little historical or empirical material that could help understand the costs and benefits of transitioning to such a regime, or to evaluate the different ways in which monetary policy could be conducted under it" (Barrdear & Kumhof (2016), Arthur et al. (2018), Cœuré & Loh (2018), Didenko et al. (2020)). A move towards CBDC adoption would be premature (Murray (2019), Berentsen & Schar (2018), Meaning et al. (2018), Engert & Fung (2017), Griffoli et al. (2018)), as further analysis of technological feasibility and operational costs/benefit is required (Griffoli et al. (2018)). So far, no central bank has a live operating CBDC system (Shirai (2019), Kumhof & Noone (2018), Hileman & Rauchs (2017)).

Regulation for Blockchain-Based PCS System of Central Banks

Central banks have an objective of maintenance of public policy interests through regulation of both large value and retail payment systems innovation (Chiu (2017)). There are some papers in the included literature that discuss regulatory approaches to DLT adaptation to central bank operated PCS systems. Only six of those outline benefits to regulators from blockchain-based PCS. For example, PCS system implemented on DLT could provide a central bank with an enhanced regulatory audit function, as information is more easily tracked and visible to all parties, enhancing resolution management capabilities (Ducas & Wilner (2017), Wu & Liang (2017), Tsai et al. (2016), Didenko et al. (2020)). Furthermore, the laws and regulations applicable to DLT-based PCS can affect the manner, speed and extent to which any implementation or configurations of DLT can be adopted (Mills et al. (2016), BIS (2017)) by financial services. A further 11 papers offer a deliberation on legal challenges and risks from hypothetical DLT-based payment clearing and settlement. Application of blockchain technology to PCS activity is a new (BIS (2017)) paradigm, contrasting with current legal frameworks, e.g., statutes, regulations, policy and supervision that are well established (Mills et al. (2016)) and have specifically been drafted to accommodate existing architectures of the system and hence the requirement for legislative adaptation to cover DLT-based PCS (Geva (2018)).

When entering into any financial transaction, the key risk is that the final/legal settlement will not materialise as expected (Mills et al. (2016)). "Settlement finality (or legal settlement) for post trade clearance and settlement is a legally defined moment in time at which the transfer of an asset, a financial instrument, or the discharge of an obligation is irrevocable and unconditional and not susceptible to being unwound following the bankruptcy or insolvency of a participant" (Mills et al. (2016), BIS (2017), Chapman et al. (2017), Geva (2018)). It is typically supported by a statutory, regulatory and/or a contractual framework underlying a given financial transaction (Mills et al. (2016)). Parties to a transaction and their intermediaries rely on that definition and timing of settlement finality when they update their own transactional ledgers to measure and monitor various risks and determine the ownership of assets (Mills et al. (2016), Seretakis (2019)). For a settlement to be achieved on blockchain, legal settlement finality may not be as clear. First, in arrangements that rely on a consensus algorithm to effect settlement finality (Geva (2018), Mills et al. (2016)), there may not necessarily be a single point of settlement finality, as there can be a gap between the period in which new additions to the ledger are made and later confirmed into blocks (Ducas & Wilner (2017)). Second, consensus protocols are probabilistic (Seretakis (2019), Mills et al. (2016), Chapman et al. (2017), Benos et al. (2017)), i.e., the payment is never fully settled because there is always a small probability that the payment could be reversed (Chapman et al. (2017)) due to forking (Benos et al. (2017), Chiu & Koeppl (2019)). The existence of forks brings into question the nature of any claims and rights that depend on the ledger records for their proof and can pose serious legal risks for users (Raskin & Yermack (2018)). If a group of nodes have a fundamental disagreement in the history of events and decide to create an alternative ledger causing a fork, it undermines the assumption that there always will be only one reliable and authoritative ledger (Chiu (2017)), failing the settlement. Even though the settlement becomes increasingly certain as recorded transactions become immutable over time, it never reaches the point of being irrevocable (Chapman et al. (2017), Mills et al. (2016), Benos et al. (2017)). The applicable legal framework does not support a legal settlement in such cases (BIS (2017), Geva (2018)). As it is a critical element of risk management, a legal basis is required to clarify when settlement finality happens. This allows definition of the key financial risks and obligations in the system, including the point at which transactions become irrevocable (Benos et al. (2017), BIS (2017), Geva (2018), Seretakis (2019)).

Furthermore, DLT-based PCS systems are exposed to being hosted in multiple jurisdictions simultaneously (Ducas & Wilner (2017)) which opens them up to the risk of regulatory arbitrage, whereby participatory nodes become concentrated in jurisdictions with loose regulatory controls (Ducas & Wilner (2017), BIS (2017)). Additionally, for DLT-based PCS systems, compliance with the Bank Secrecy Act (BSA) (Mills et al. (2016), Geva (2018)), AML (Chiu (2017), Mills et al. (2016), Seretakis (2019), Geva (2018)), KYC (Mills et al. (2016), Seretakis (2019)), transaction monitoring and reporting of suspicious activity (Mills et al. (2016)) is not currently provided. For a large value wholesale payment system, the need to keep transactional data private from other parties is fundamental (Chapman et al. (2017)) (Benos et al. (2017)). "This is necessary to prevent other participants from being able to take advantage of this information. A participant's clients may also prefer or require this privacy" (Chapman et al. (2017)). Lastly, as these technologies are not fail-safe, further risk of greater expense in recovery or litigation if such technology fails (Chiu (2017)) cannot be overlooked. Decentralised systems do not provide an independent regulatory party that can facilitate dispute resolution (Ben Dhaou & Rohman (2018)) functionality, raising questions about conflict of laws and jurisdictions (Geva (2018)) that determine the nature and extent of rights and claims (Chiu (2017)).

All of above-mentioned issues are costlier in a distributed (no governing jurisdiction) and permissionless (no identifiable responsible party) environment (Geva (2018)). The questions of how and when transactional certainty and security is achieved, as well as responsibility and risk allocation among participants need to be considered prior to blockchain adaptation. By itself, this is a costly operation. One path to manage those risks, could be an incremental adaptation of blockchain for PCS.

2.6.9 Statistical Analysis of Research trends for Blockchain Use-Cases

This section provides a narrative summary and further statistical insight into each separate use-case. Appendix B.5 of this study provides a matrix of this research.

RQ4: What is the depth/breadth of the research for identified use-cases?

Central Bank Digital Currency

According to a generalised definition, a Central Bank Digital Currency (CBDC) is an electronic, 24x7, fiat liability of a central bank that can be used as a digital account or as an electronic token (Engert & Fung (2017)) to settle payments or as a store of value (Meaning et al. (2018), Kumhof & Noone (2018), Barrdear & Kumhof (2016), Engert & Fung (2017), Griffoli et al. (2018)) and could provide access to a central bank's balance sheet (Barrdear & Kumhof (2016)). It is an electronic central bank or narrow money (Meaning et al. (2018), Kumhof & Noone (2018), Barrdear & Kumhof (2016)), intended as legal tender (Griffoli et al. (2018), Cœuré & Loh (2018)) which can be exchanged (Bordo & Levin (2017)) in a decentralised manner, known as peer-to-peer (P2P). This means that all transactions occur directly between the payer and the payee, without the need for a central intermediary (Bech & Garratt (2017)). Fig. 2.9 shows representation of CBDC use-cases in the included literature. Out of 39 publications (Fig. 2.8a) describing CBDC, 11 publications employ Empirical research types with nine publications using Evaluation and two Validation Research approaches (Fig. 2.9c). As contribution to techniques (Fig. 2.9a), Evaluation papers add four models (Sun et al. (2017), Tsai, Zhao, Zhang, Yu & Deng (2018), Chiu et al. (2019), Agur et al. (2019)) and two protocols (Sun et al. (2017), Tsai, Zhao, Zhang, Yu & Deng (2018)); Validation research adds two models (Danezis & Meiklejohn (2015), Kang & Lee (2019)) and a protocol (Danezis & Meiklejohn (2015)). Interestingly, the majority of Evaluation research was provided *via* grey literature and all Validation research was communicated during conferences (Fig. 2.9c). Over time (Fig. 2.9b), the bulk of industry and data driven papers (Evaluation) were available in 2018, totalling 4. Validation research on CBDC was only published in 2016 and 2019 - a single paper for each year. Amongst Empirical papers, the research of (Hileman & Rauchs (2017)) provides a wide global benchmarking study on blockchain current areas of focus, attitudes toward the technology and outstanding questions. Researchers use surveys and focus groups to identify which overall blockchain use-cases were investigated by central banks (and a wider community of

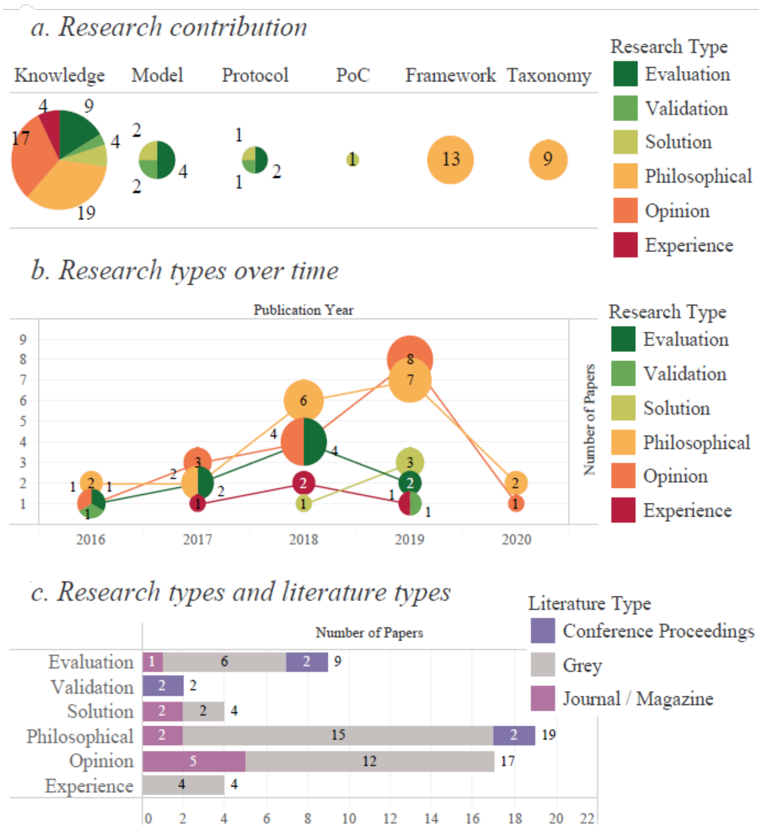


Figure 2.9: Central Bank Digital Currency (CBDC).

practitioners), maturity and future roadmap of that research. The authors establish that 82% of central banks were investigating DLT as a platform to launch CBDC (Hileman & Rauchs (2017)). Agur et al. (2019) analyse the optimal CBDC design that maximises social welfare by comparing non-interest-bearing versus interest-bearing CBDC and the degree to which the CBDC resembles cash (Agur et al. (2019)). Researchers evaluate impact of those design choices onto cash, bank deposits and bank intermediation. The network effect lies in the core of their model. They show that when CBDC is designed as non-interest-bearing, its similarity to cash becomes the sole design choice (Agur et al. (2019)). If CBDC is designed as interest-bearing, it safeguards bank intermediation and provides households with a variety of payment instruments.

Chiu et al. (2019), with the cooperation of the Bank of Canada address implications of CBDC issuance for monetary policy and banking. They built a "tractable model" to represent imperfect competition in the deposits markets of the banking sector. Using quantitative analysis to demonstrate that an interest-bearing CBDC could promote bank intermediation, increase lending and aggregate output, they showed that the design choice of CBDC, competition level in the deposit market and the interest rate on CBDC does affect the banking system and real economy. Kang & Lee (2019) develop a "search theoretical model", where public cryptocurrency is used as a medium of exchange and coexists in an equilibrium and competes with central bank issued fiat money, thus affecting monetary policy, overall economic activities and welfare. Their quantitative analysis showed that, provided there is a sufficiently high inflation rate (to justify cryptocurrency mining fees), public permissionless cryptocurrency is able to compete with fiat money. However, due to the inefficient cryptocurrency mining process, the welfare in economy with both fiat money and cryptocurrency is lower than that in a money-only economy. The rest of the Empirical papers use permissioned blockchain as a platform to launch CBDC, controlled by a central bank (Sun et al. (2017), Tsai, Zhao, Zhang, Yu & Deng (2018), Danezis & Meiklejohn (2015)). Two studies (Sun et al. (2017), Tsai, Zhao, Zhang, Yu & Deng (2018)) propose multi-blockchain models and evaluate their feasibility and scalability. Danezis & Meiklejohn (2015) propose two "thread models", where transactions were processed with and without minters. Two papers use experiments (Sun et al. (2017), Danezis & Meiklejohn (2015)); one uses simulation (Tsai, Zhao, Zhang, Yu & Deng (2018)). Sun et al. (2017) propose a protocol for "inter-blockchain transactions", design of which was influenced by the Practical Byzantine Fault Tolerance (PBFT) (Castro et al. (1999)) algorithm and Bitcoin blockchain (Nakamoto (2008)); Tsai, Zhao, Zhang, Yu & Deng (2018) provide a consensus protocol for two types of blockchain - a "trading blockchain" and an "account blockchain"; Danezis & Meiklejohn (2015) also use Bitcoin Nakamoto (2008) as a consensus protocol for transaction validation.

Although not empirically validated, four Solution Proposal papers (Fig. 2.9), provide a novel protocol and PoC (Wu et al. (2019)) and two models (Borgonovo et al. (2018), Brunnermeier & Niepelt (2019)). Those contributions are communicated as purely academic articles and *via* grey literature and the first was available in 2018 following the other three in 2019. There are 19 Philosophical papers contributing 13 novel frameworks and nine taxonomies, as some of those contributed both. The majority of those are communicated from industry through grey literature and the availability of these was steadily growing each year, peaking in 2019 at seven publications and two papers for half of 2020. Opinion research is also heavily dominated by grey literature (12 out of 17 papers) and its availability grew constantly, with 2019 bucking the trend with eight papers and one in 2020. Experience papers only briefly appeared in 2017 (one paper), 2018 (two papers) and 2019 (one paper) and all are provided as grey literature. This data indicates that industry is also heavily involved in the theoretical discussion about CBDC.

Wu et al. (2019) (Solution proposal) suggest using PoC, a Bitcoin blockchain (Nakamoto (2008)) based electronic currency protocol to support anonymous payments. The protocol provides full access of transaction history to supervisors and auditors. The authors use blind signature technology, public key signatures and Proof-of-Work (PoW) consensus. In another Solution Proposal paper, Borgonovo et al. (2018) provide a "primer model" to analyse demand for CBDC by identifying drivers of the political consensus in favour or against it. The research uses a "financial portfolio approach" and assumes that the prospect of issuance of CBDC would influence individual portfolio choices. Brunnermeier & Niepelt (2019) provide a "generic model of money and liquidity which identified sources of seigniorage rents and liquidity bubbles" and apply that model in the context of CBDC introduction for the use by general public. Their results imply that: "CBDC, coupled with central bank pass-through funding, need not imply a credit crunch nor undermine financial stability".

Philosophical type is utilised in 19 papers to communicate research approach and contributions to knowledge. Amongst those, nine contribute taxonomies (Lipton (2018), Bech & Garratt (2017), Berentsen & Schar (2018), Shirai (2019), Griffoli et al. (2018), Cœuré & Loh (2018), Khiaonarong & Humphrey (2019), Auer & Böhme (2020), Didenko et al. (2020)) and 13 add new frameworks (Arthur et al. (2018), Engert & Fung (2017), Danezis & Meiklejohn (2015), Griffoli et al. (2018), Fung & Halaburda (2016), Cœuré & Loh (2018), Kahn & Wong (2019), Koumbarakis & Dobrauz-Saldapenna (2019), Pfister (2019), Brunnermeier & Niepelt (2019), Han et al. (2019), Agur et al. (2019), Didenko et al. (2020)), three of which (Griffoli et al. (2018), Cœuré & Loh (2018), Didenko et al. (2020)) contribute both. Out of nine novel taxonomies, two papers provide taxonomies of potential benefits and cost for a central bank from issuing CBDC (Khiaonarong & Humphrey (2019), Bech & Garratt (2017)); the other two propose taxonomies of existing forms of money in relation to CBDC (Berentsen & Schar (2018), Cœuré & Loh (2018)); a further four offer taxonomies of CBDC projects and ongoing technical design efforts in other countries' by central banks (Khiaonarong & Humphrey (2019), Shirai (2019), Auer & Böhme (2020), Didenko et al. (2020)); Auer & Böhme (2020) sets out an additional taxonomy in the same paper for the underlying design trade-offs that maps consumer needs hierarchy for designing a retail CBDC; Lipton (2018) suggests a general taxonomy of potential blockchain applications to money and banking. Out of 13 novel frameworks, eight papers offer new conceptual frameworks to characterise various design features of potential CBDC: Didenko et al. (2020) consider design parameters for CBDC such as: users, scope, architecture and technology, within which they envisage three alternative CBDC architectural approaches: 1) central bank accounts with general access, 2) central bank accounts with intermediated access, and 3) new digital forms of fiat currency. By doing so, they analyse the impact of DLT and blockchain onto monetary and payment systems (Didenko et al. (2020)); Engert & Fung (2017) set out a framework of the features for a benchmark CBDC that are similar to cash; Agur et al. (2019) build a theoretical framework tailored at analysing the relationship between CBDC design, welfare analysis, the demand for money types and financial intermediation; Han et al. (2019) provide a theoretical guidance for a three layered blockchain-based CBDC framework that includes supervisory, network and user layers, incorporating account-based and wallet based mainstream models; Cœuré & Loh (2018) and Pfister (2019) propose conceptual frameworks for understanding the difference between a retail or general purpose CBDC and a wholesale CBDC; Kahn & Wong (2019) provide a theoretical framework for account-based, token-based and delegated (i.e., as custodians and intermediaries) CBDC schemas; Koumbarakis & Dobrauz-Saldapenna (2019) set out a framework and formulate broad design principles for CBDC in line with the central bank's function as a Lender Of Last Resort (LOLR). Furthermore, Griffoli et al. (2018) offer a conceptual framework to compare different forms of money and another framework that provides an understanding about the roles of CBDC from a user perspective. Fung & Halaburda (2016) propose a framework for central banks for accessing why and how they should consider issuing CBDC. The same framework can be used by the general public to make payments and could be implemented to improve the efficiency of retail payment system. Brunnermeier & Niepelt (2019) provide a general framework for the analysis of monetary economics in the context of introduction of CBDC. Their framework: "Augments the standard asset pricing formula with a liquidity kernel". Danezis & Meiklejohn (2015) present the first cryptocurrency framework "RSCoin" that provides control over issuance of CBDC and the monetary policy to a central bank. The remaining papers – Opinion (Seretakis (2019), Raskin & Yermack (2018), Geva (2018), Engert & Fung (2017), Furche & Sojli (2018), Berentsen & Schar (2018), Khiaonarong & Humphrey (2019), Nabilou (2019), Bordo & Levin (2017), Murray (2019), Shirai (2019), Kahn & Wong (2019), Koumbarakis & Dobrauz-Saldapenna (2019), Nabilou & Prum (2019), Didenko et al. (2020)) and Experience (Engert & Fung (2017), Cœuré & Loh (2018), Kahn & Wong (2019), Pfister (2019)) papers provide a discussion on design characteristics of CBDC (Bordo & Levin (2017), Murray (2019), Shirai (2019), Geva (2018), Berentsen & Schar (2018), Koumbarakis & Dobrauz-Saldapenna (2019), Pfister (2019), Didenko et al. (2020)), why and how a central bank should issue CBDC (Kahn & Wong (2019), Seretakis (2019), Engert & Fung (2017), Cœuré & Loh (2018), Furche & Sojli (2018)) and potential hazards from CBDC issuance (Raskin & Yermack (2018), Engert & Fung (2017), Geva (2018)).

Payment Clearing and Settlement Systems Operated by Central Banks

The Payment Clearing and Settlement (PCS) systems of a central bank are characterised by processes, such as payments (i.e., order management, including trade validation (Benos et al. (2017))), post-trade securities clearing (i.e., the calculation of counterparties' obligations (Benos et al. (2017))), and post-trade settlement (i.e., the final transfer of assets (Benos et al. (2017))). Those systems also involve several different types of financial intermediaries (Mills et al. (2016), Didenko et al. (2020)) and infrastructures invoked from the time a trade in a financial security is agreed to the time when it is finally settled (Benos et al. (2017)). "Central banks have traditionally played an important catalyst role in payments and settlements" (BIS (2017)). PCS processing systems of today are cumbersome and involve lengthy reconciliation tasks (Benos et al. (2017)). Finally, operational, settlement, legal and financial risks are inherent in the conduct of PCS system activities (Mills et al. (2016), Didenko et al. (2020)).

Fig. 2.10 represents a blockchain underpinned PCS use-case in the included literature. There are 27 papers (Fig. 2.8a) providing various contributions (Fig. 2.10a). Empirical Research is only presented in three Evaluation papers (Chiu & Koeppl (2019), Hileman & Rauchs (2017), Mills et al. (2016)) – 9% of all papers on blockchain applications for PCS and those papers were available

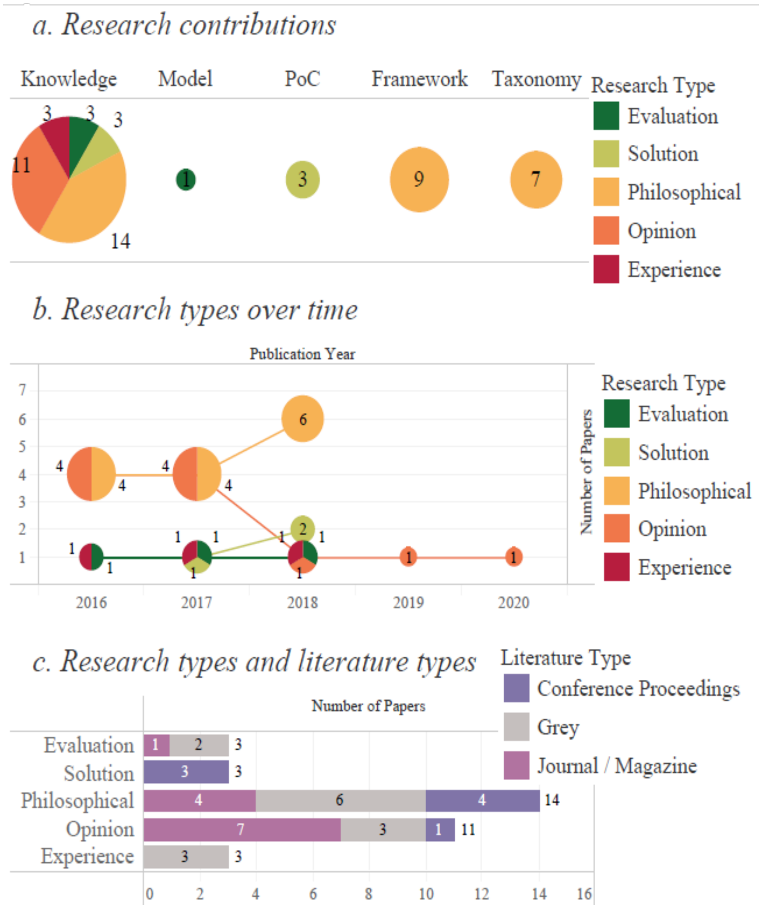


Figure 2.10: Payment Clearing and Settlement Systems Operated by Central Banks (PCS).

between 2016 - 2018; two of them are grey literature. There is no Validation research available in the included publications. One model is added as a technological contribution by Chiu & Koeppl (2019). The remainder of all research only make theoretical contributions. The data indicates that, although comparatively small, all empirical research has industry input or drivers, because the Evaluation research approach involves industry participation (Petersen et al. (2008)).

As a step towards understanding the implications of DLT deployment to PCS systems and to identify the opportunities and the challenges facing its long-term implementation and adoption, a research team of the Federal Reserve Bank (FED) (Mills et al. (2016)) conducted interviews with focus groups interested in participating in, or otherwise contributing to, the evolution of DLT (Mills et al. (2016)). In their report, the team summarised the approaches taken by industry to investigate the potential of blockchain (Mills et al. (2016). Hileman & Rauchs (2017)), also based on the results of surveys and focus groups report that overall: “55% of central banks are exploring DLT-based payment systems for remittance transfers, inter-bank payments and other uses” (Hileman & Rauchs (2017)).The only model contributed by Chiu & Koeppl (2019) investigates the extent of potential financial gains or losses, if financial securities were to be settled on blockchain. The distinctive technological features of blockchain are explicitly modelled for asset settlement. They investigated, both qualitatively and quantitatively, using mathematical analysis: its feasibility, optimal block size and time. The authors chose to consider a permissionless blockchain, which ensures delivery-vs-payment (DvP) by linking transfers of assets with payments and where updating of records is based on a proof of work (PoW) protocol.

Three papers utilise Solution Proposals for research communication of PoCs (Tsai, Deng, Ding & Li (2018), Wu & Liang (2017), Wang et al. (2018)), all of which are proceedings of conferences, one in 2017 and two in 2018 (Fig. 2.10). 42.42% (or 14 publications) are Philosophical papers which provide nine novel frameworks and seven taxonomies, with two adding both (BIS (2017), Cœuré & Loh (2018)). Over time, the addition of those papers to research was steady, with four for each of 2016 and 2017, increasing to six in 2018; the variety of literature types is relatively balanced, with grey literature slightly leading that trend. A third of publications (or 11) are Opinion Papers rising in availability in 2017 and being the only paper published for 2019 and for 2020. Those papers are principally shared as pure academic articles. Three experience papers (Mills et al. (2016), Cœuré & Loh (2018), Chapman et al. (2017)) are all shared as grey literature, one for each of 2016 - 2018. This data indicates that the theoretical elaboration on the topic of utilisation of DLT for PCS is consistent and well-balanced between academics and industry. However, there are potential early signs of reduction in interest due to lack of availability of new research and the creation of a gap in the state of knowledge; as for the majority of 2019, there were noticeably few new research engagements on this topic. Out of three Solution proposals, two (Wu & Liang (2017), Wang et al. (2018)) explore application of blockchain for "inter-bank payment systems (IBPS)" and one explores use of blockchain as a "clearinghouse" (Tsai, Deng, Ding & Li (2018)). For their project, Tsai, Deng, Ding & Li (2018) adopt a permissioned DPT (Double-chain Parallel-processing Technology) developed at Tiande to facilitate a "multi-blockchain clearinghouse" experiment and demonstrate its feasibility via PoC. (Wu & Liang (2017)) utilise a Bitcoin blockchain (Nakamoto (2008)) to build a distributed ledger prototype system for credit matching of trading system for X-Swap and Wang et al. (2018) use Hyperledger Fabric (*Hyperledger Fabric – Hyperledger* (n.d.)) to develop an end-to-end IBPS prototype to design a fund transfer functionality enabling gross settlement for Real Time Gross Settlement (RTGS) systems. A total of 14 papers use Philosophical methods to communicate new knowledge. Amongst those, nine add new frameworks (Tsai, Deng, Ding & Li (2018), Shah & Jani (2018), Milne (2018), Wang et al. (2018), BIS (2017), Tsai et al. (2016), Zhai & Zhang (2018), Fung & Halaburda (2016), Cœuré & Loh (2018)) and seven contribute taxonomies (Lipton (2018), Pinna & Ruttenberg (2016), Mills et al. (2016), Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017), Benos et al. (2017), BIS (2017), Cœuré & Loh (2018)), with two of them (BIS (2017), Cœuré & Loh (2018)) contributing both. Out of nine frameworks, two papers offer multi-blockchain frameworks for integrating DLT into PCS processes (Tsai, Deng, Ding & Li (2018), Tsai et al. (2016)). Three papers consider blockchain-based IBPS frameworks (Shah & Jani (2018), Wang et al. (2018), Zhai & Zhang (2018)). The remaining papers, such as 11 Opinion (Chiu (2017), Guo & Liang (2016), Yoo (2017), Seretakis (2019), Pinna & Ruttenberg (2016), Priem (2020), Ben Dhaou & Rohman (2018), Ducas & Wilner (2017), Benos et al. (2017), Caytas (2016), Didenko et al. (2020)) and three Experience papers (Mills et al. (2016), Chapman et al. (2017), Cœuré & Loh (2018)) provide a discussion on the potential impact of DLT on PCS processes.

Asset Transfer and Ownership

Any financial instrument, such as a monetary instrument, security, commodity or derivative is an asset (Mills et al. (2016)). "PCS systems are typically organised around a specialised third-party called Central Securities Depository (CSD), which are responsible for transfers of legal ownerships of securities/assets against payments" (Chiu & Koeppl (2019)). Additionally, a variety of financial intermediaries, on behalf of their clients, can hold or trade those assets or securities (Mills et al. (2016)). In today's markets, it is a common occurrence for investors that are not the direct owners of the traded assets, to hold them indirectly through chains of financial intermediaries that operate between asset issuers and those investors (Benos et al. (2017)). "This is partly a legacy from the time where securities were issued as paper certificates and had to be immobilised to facilitate their trading through book-entry transfers" (Benos et al. (2017)).

Fig. 2.11 shows a representation of the assets use-cases for blockchain in the included literature. Out of 12 papers (Fig. 2.8a), describing assets transfer and ownership, one adds a model (Fig. 2.11a). There are only two Evaluation papers published (Chiu & Koeppl (2019), Hileman & Rauchs (2017)) as an article and as grey literature, one in 2017 and another in 2018; one contributes

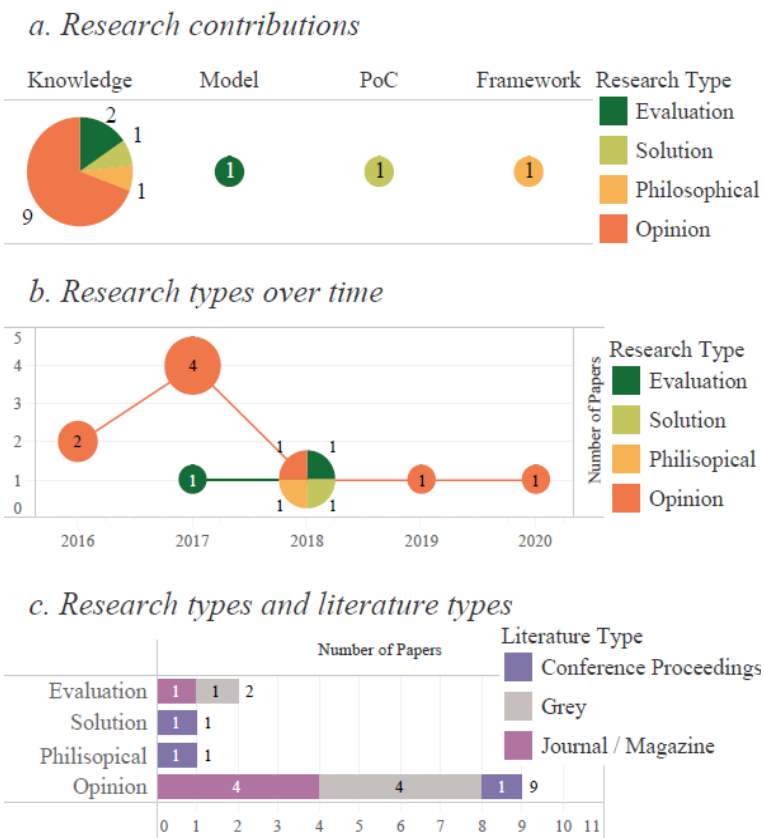


Figure 2.11: Assets Transfer and Ownership (Assets).

a model (Chiu & Koeppl (2019)). There are no Validation papers on this use-case. Overall, the data on empirical papers does not provide a particular pattern, apart from that its availability is low and all available research has industry involvement. This might indicate a potential knowledge gap for empirical research. Out of two Evaluation papers, (Hileman & Rauchs (2017)) state that only 23% of central banks were investigating the ownership record management capabilities of blockchain. Chiu & Koeppl (2019), whilst explicitly modelling feasibility of blockchain for assets trading, establish that the key innovation from blockchain to their model is that it provides a shared database of security ownerships that can be updated without relying on multiple, specialised intermediaries or a third-party infrastructure.

One solution proposal (Chen et al. (2018)) provides a PoC in 2018 *via* a conference (Fig. 2.11). The same paper also adds a framework. The majority - 69.23% or nine - are Opinion papers, four of which were published in 2017, with 2018-2020 supplying one additional paper for each year. One of those papers came from a conference, with industry and academics providing an additional four each. There is no Experience research available for this use-case. This data indicates that theoretical discussion on this topic is mainly hypothetical as there is no practical experience available upon which to draw justifiable conclusions. Chen et al. (2018) utilise two research types for their paper to communicate two contributions, a Solution Proposal for PoC where they propose a "financial product management platform" that provides capabilities for multi-function financial data inquiries, routine maintenance of financial products and multi-institution traceability. Their platform is based on Hyperledger (*Hyperledger Fabric – Hyperledger* (n.d.)). The researchers also construct a "financial product management framework" for deployment of transactional logic for blockchain. Opinion papers provide a high-level discussion of both hypothetical benefits and limitations from application of blockchain to asset transfers and ownerships (Seretakis (2019), Priem (2020), Mills et al. (2016), Ducas & Wilner (2017), Benos et al. (2017), BIS (2017), Geva (2018), Tsai et al. (2016)).

Audit Trail

The BoE states that as part of a central bank's accounting reporting procedures, it: "Has a responsibility for reviewing the findings of internal and external auditors and monitoring outstanding actions. It receives and reviews reports on the risk profile of a central bank and inter-bank market participants" (Wohlin (2014)). A large number of auditing processes believed to be simplified or even eliminated by automation of the audit trail on blockchain (Manda & SS (2018)). Fig. 2.12 provides a representation of the audit trail use-cases for blockchain in the included literature. Out of 14 papers (Fig. 2.8a), exploring the influence of blockchain on auditing performance of a central bank (Fig. 2.12a), one adds a protocol. There was only one Evaluation paper available in 2017 which only contributes to discussion *via* an academic journal (Hileman & Rauchs (2017)). There are no Validation papers for this use-case. The data indicates that empirical research is comparatively low (signalling a potential knowledge gap), purely theoretical and, again, only with industry cooperation. The only available Empirical paper by Hileman & Rauchs (2017) established that only a comparatively small proportion of central banks (18%) had specifically mentioned that audit trails, e.g., tracking of payments, are under investigation.

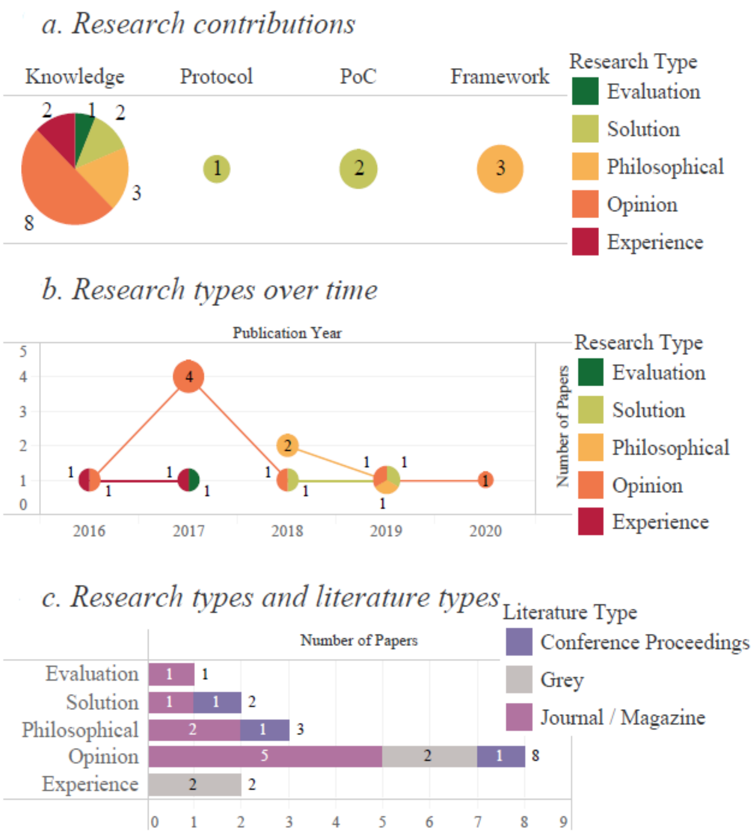


Figure 2.12: Audit Trail (Audit).

Two papers provide Solution proposals (Fig. 2.12), *via* PoC (Wang et al. (2018), Chen et al. (2018)) through a journal and a conference in 2018 and 2019, and one of them provides a protocol (Chen et al. (2018)). Three Philosophical papers all add frameworks in 2018 and 2019, two in journals and one through a conference. 50% of all papers are Opinion papers, peaking in 2017 at four publications; one more was added for 2018 - 2020. Interestingly, there are two Experience papers published as grey literature at the beginning of the period in 2016 and 2017. The data indicates that, although comparatively low, theoretical discussion on this topic has been underpinned by some practical experience from industry practitioners, although academic journal articles are now leading the conversation. Both Solution proposals utilise Hyperledger (*Hyperledger Fabric – Hyperledger* (n.d.)) for their underlying architecture. Chen et al. (2018) propose a "financial product management platform" that provides a multi-institution/multi-function data audit capability. Wang et al. (2018) introduce an "end-to-end IBPS protocol" - that provides provenance tracking functionality for auditors. By leveraging the immutability of blockchain ledger, their protocol equips auditors with the ability to trace back the history of records and conduct reconciliation. There are three papers that contribute *via* frameworks. Chen et al. (2018) and Wang et al. (2018) construct frameworks for auditors to track financial product data provenance on a blockchain. Kavassalis et al. (2018) provide a framework for financial transactions as well as financial risk reporting; they report a transactional audit trail to the qualified authorities about all significant circumstances under which a transaction took place. Eight Opinion papers (Seretakis (2019), Pinna & Ruttenberg (2016), Priem (2020), Grody (2018), Ben Dhaou & Rohman (2018), Ducas & Wilner (2017), Benos et al. (2017), BIS (2017)) and two Experience papers (Mills et al. (2016), Chapman et al. (2017)) provide a high-level discussion of how implementation of DLT in central banks could affect their auditing capabilities.

Regulatory Compliance

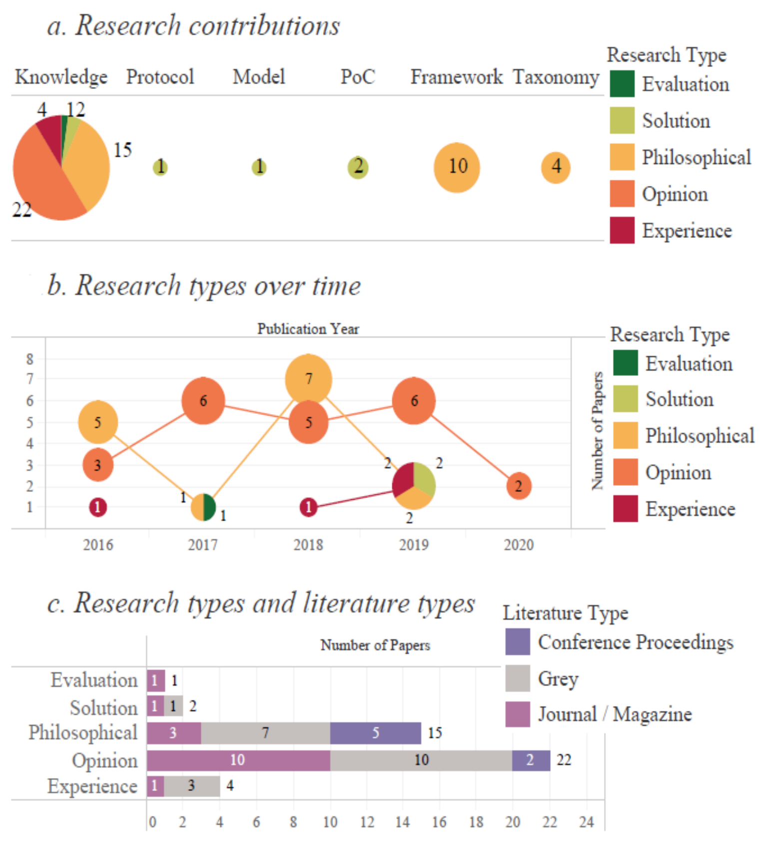


Figure 2.13: Regulatory Compliance (Regulators).

Hayes (2016) states that the most visible function of a central bank is that of a monetary authority. A generalised legal consideration for a central bank acting as a financial regulator consists of a legal framework which includes general laws, regulations, rules, procedures and contracts (BIS (2017)). Fig. 2.13 represents the regulatory compliance use-cases for blockchain in the included literature. In total, 37 papers (Fig. 2.8a) examine the impact from blockchain on the functionality of a central bank as a financial regulator (Fig. 2.13 a). Only one publication is empirical - an Evaluation paper (Hileman & Rauchs (2017)) in 2017 in a journal. The only Empirical paper of Hileman & Rauchs (2017) reports a response from surveys and focus groups that: "36% of central banks have been investigating DLT for regulatory compliance, such as automatically enforce market regulation" (Hileman & Rauchs (2017)). This data reveals that empirical research is comparatively low and only with industry cooperation, signalling a potential knowledge gap.

Two Solution Proposal papers (Wu et al. (2019), Auer (2019)) contribute two PoCs and a model in 2019 as a pure academic article and as grey literature (Fig. 2.13). Over a third of all papers on this topic (15 publications) are Philosophical papers adding nine novel frameworks and four taxonomies through diverse literature cohorts. Most of those papers were published in 2016 and 2018. Over half of all regulation use-case papers are Opinion papers (22 in total), with almost half of those available as academic articles, presenting a somewhat steady trend in popularity over the years. There are also a total of four Experience papers added in 2016, 2018 and 2019, with three of those represented by practitioners *via* grey literature and one as a purely academic article. This data indicates that the theoretical discussion on this use-case is ongoing, diverse and potentially underpinned by practical experience from industry practitioners and academics. Using Poc, Auer (2019) models a blockchain based automated "embedded supervision" functionality for novel distributed markets. The model provides for economic finality in a permissioned market with decentralised verification. A CBDC protocol proposed by Wu et al. (2019), based on a Bitcoin blockchain (Nakamoto (2008)), provides supervisors with the ability to oversee unanimous payments *via* unrestricted access to the blockchain ledger. 15 Philosophical papers mostly contribute frameworks - 10 papers (Tsai, Deng, Ding & Li (2018), Shah & Jani (2018), Hayes (2016), Arthur et al. (2018), Tsai et al. (2016), Kavassalis et al. (2018), Fung & Halaburda (2016), Kahn & Wong (2019), Auer (2019), Han et al. (2019)), vs. four taxonomies (Nguyen (2016), Mills et al. (2016), Benos et al. (2017), Cœuré & Loh (2018)). Four of those papers (Arthur et al. (2018), Fung & Halaburda (2016), Kahn & Wong (2019), Han et al. (2019)) propose frameworks that utilised CBDC as a transparent transactional ledger visible to regulators, e.g., "custodians and intermediaries CBDC schemas" of Kahn & Wong (2019) or the three-layered CBDC framework of Han et al. (2019) that includes a supervisory layer. Two other papers utilise a blockchain-based PCS architecture as: "a promoter of regulatory informant" (Tsai et al. (2016)) or as a participating regulatory node in DLT-based PCS (Tsai, Deng, Ding & Li (2018)). Two further papers offer frameworks for central banks and regulators to assess legal risks from blockchain, such as risks to legal settlement finality, issues with a management and protection of data, connectivity with legacy systems, standards development (BIS (2017)) and suitability for KYC compliance (Shah & Jani (2018)). Hayes (2016) provide a conceptual framework for a workable decentralised central bank (DAO bank) to perform functionality of a "technocratic, rules-following monetary authority". Kavassalis et al. (2018) propose a novel framework for a "regular technology (RegTech) approach for financial transactions, as well as financial risk reporting based on distributed computing, decentralised data management technologies such as blockchain, distributed storage, algorithmic financial contract standards, automated legal text and document engineering methods and techniques". The researchers provide a proposal of: "How to develop a new layer of algorithmic regulation functionality, that enhances a supervisor's capacity to monitor the evolution of risk in the system" (Kavassalis et al. (2018)). Auer (2019) makes a "case for embedded supervision, i.e., a regulatory framework that provides compliance in tokenized markets to be automatically monitored by reading the market's ledger, thus reducing the need for firms to actively collect, verify and deliver data".

Out of four taxonomies, Nguyen (2016) classifies overall legal and policy challenges about potential blockchain applications for banking. In relation to potential implications from the regulatory point of view onto blockchain-underpinned PCS, Benos et al. (2017) provide a taxonomy of potential regulatory improvement, whereas Mills et al. (2016) offer a set legal challenges. Cœuré & Loh (2018) categorise Monetary Policy aspects for CBDC issuance.

Out of 22 Opinion Papers, seven discuss various ways of how DLT could be approached from a regulatory perspective (Seretakis (2019), Guo & Liang (2016), Grody (2018), Ducas & Wilner (2017), Priem (2020), Berentsen & Schar (2018), Micheler & Whaley (2019)), four examine impact of blockchain on Monetary Policy and Monetary Reforms (Ducas & Wilner (2017), Priem (2020), Ben Dhaou & Rohman (2018), Dow (2019)), seven deliberate on regulatory motivation for CBDC issuance and its effects on Monetary Policy transmission (Cœuré & Loh (2018), Bordo & Levin

(2017), Nabilou (2019), Murray (2019), Kahn & Wong (2019), Nabilou & Prum (2019), Didenko et al. (2020)) and four reflect on the role of regulators for DLT-based PCS (Chiu (2017), Benos et al. (2017), BIS (2017), Caytas (2016)). Also, four Experience papers (Ozili (2019), Mills et al. (2016), Cœuré & Loh (2018), Pfister (2019)), discuss questions that need to be considered by the regulators when assessing adoption of DLT for financial markets (Guo & Liang (2016)), legal considerations for PCS and blockchain (Mills et al. (2016)) and Monetary Policy implications of CBDC (Cœuré & Loh (2018), Pfister (2019)).

2.6.10 Discussion

Threats to Validity

For any empirically-based research, we need to consider the threats to the validity of the work *et al.*Petersen et al. (2015). The following types of validity have been considered, enabling awareness of the potential limitations to the classification schema: theoretical validity, descriptive and interpretive validity and possibility of missing relevant articles.

Theoretical validity: there is potential for *researcher bias* in the selection of the studies and reporting of the results as the majority of work for this SMS was conducted by an individual researcher. To reduce this threat and gain confidence in the results, study identification was additionally evaluated through forward snowball sampling, where only 13 new studies were identified, indicating no measurable change to the search results. Additionally, one should keep in mind potential for the *publication bias*, as new controversial negative views are less likely be published Petersen et al. (2015). To minimise this bias, only well-known scientific databases, in combination with rigorously designed search protocol were used to collect as many as possible available papers. However, as the research topic has proven to be a rather young research area, it is conceivable that further research has been administered by the industry and potentially either published as the "white papers" or kept confidential. SMS research on this topic using focus on grey literature as its source, could be an area for additional future research direction.

Descriptive and interpretive validity: there is a potential threat to accuracy of data extraction, recording and description, since in this qualitative study those processes are partially underpinned by the researcher's knowledge and understanding of the domain. To increase the descriptive validity of the study and following the guidelines of Petersen *et al.*Petersen et al. (2015), a data collection form was designed and implemented. This allowed us to make the data extraction process objective and, if necessary, amendable.

Possibility of missing relevant articles: The decision to limit this mapping study to the literature published since January 2008 does mean that there is a possibility of missing some relevant publications from before this time. However, given that the results show that there was no literature available even before 2016 on this topic, it is highly unlikely that even if there were potential unidentified papers available before 2008, that they would significantly impact final conclusions. Furthermore, creation of a search phrase was a challenging task, in particular the differences in functionality and sophistication between the different mainstream search engines, because each search engine required a different search expression syntax. To mitigate the challenges of the search phase the search for relevant literature was conducted as thorough as possible, by including an automated database search, followed by manual search, followed by forward snowball citation checking. Despite this thoroughness, there is always a possibility that some relevant articles were missed.

Research Maturity

Although the hype about the capabilities of the blockchain started between 2008 - 2009, when its novel implementation through Bitcoin cryptocurrency reached worldwide news channels, it is evident from the data that the attention of research community to this topic is very recent, where first publications were first available from 2016 (Fig. 2.5b). This falls in line with other researchers' opinions, that the application of blockchain to the business of central banks is at a very early stage (Barrdear & Kumhof (2016), Berentsen & Schar (2018), Hayes (2016), Pinna & Ruttenberg (2016), Chiu (2017), Mills et al. (2016), Ben Dhaou & Rohman (2018), Hileman & Rauchs (2017), Bordo & Levin (2017), Benos et al. (2017), Kumhof & Noone (2018), Meaning et al. (2018), Griffoli et al. (2018), Cœuré & Loh (2018), Murray (2019), Arthur et al. (2018), Engert & Fung (2017)). Industry is still providing large proportions of empirical technological and theoretical contributions to the field, with participation of academia predominantly on the non-empirical side of the research. Furthermore, the data implies that the overall trend of the engagement from the research community is growing, although it is difficult to judge with confidence about the trend for 2020 since our database search was done during the beginning of the second quarter, where a proportion of papers are still unpublished, but this does not invalidate the results we have presented.

As the topic of this study is a comparatively new area, there is also a distinct lack of validated research or data to support hypotheses. As described in Section 2.6.7, Empirical Research was only used 13% of the times and the majority of that research was Evaluation Research, involving participation of industry experts. The study has identified a clear need for more quantitative/empirical work in the area to evaluate aspects of blockchain. A common criticism of many areas of software engineering is that academic studies fail to appreciate the demands and pressures exerted on industry. As a result, there is almost a chasm between what academic studies do and what industry wants. The trend seems to be being repeated in this relatively new area. Empirical studies should involve industry and academia, address pressing issues in industry and focus on industrial impact. The results in this paper show a mixed picture thus far.

Use-Cases

Section 2.6.8 showed that uses-cases for application of blockchain for central banks belonged largely to CBDC, Regulation or PCS. The largest proportion of empirical research and novel technological contributions were applicable to CBDC use-cases, where again, we can see a heavy presence of grey literature. The regulatory compliance use-case for blockchain closely follows CBDC by the amount of interest, although the majority of that research is done utilising non-empirical methods to generate large ongoing discussion from a diverse cohort of researchers. Interestingly, although a very popular use-case from the onset of the research availability, DLT-based PCS systems exhibit a sudden knowledge gap between 2019 and 2020. Further evaluation of the reasons for this lack of interest from the research community could reveal some hidden insights. In relation to asset transfer and audit trail use-cases, both present somewhat similar trends, showing comparatively low engagement from researchers, providing non-empirically validated, theoretical views in the main.

Discussion of each of the separate use-cases in Section 2.6.8 indicates that, although there are numerous advantages from application of DLT to the business of a central bank, potential limitations and issues constitute a comparatively large proportion of the debate:

CBDC models receive attention from the research community and central banks. Researchers are focusing on design characteristics of CBDC such as account versus token based CBDC or those designed for retail or wholesale money customers. If CBDC could pay interest on its holdings, researchers argue that it could remedy competition problems in the banking sector and promote financial inclusion. However, the questions of the role for central banks, disruption of commercial banks' business models, risks to smooth operation of payment systems, conduct of monetary policy and numerous legal challenges still remain unanswered.

In relation to hypothetical blockchain underpinned *PCS operated by central banks*, it is argued that for inter-bank, large-value wholesale payments blockchain could provide faster, close-to-real time 24x7x365 processing, reducing the need for centrally maintained back-up systems and reducing the number of intermediaries. By streamlining and speeding up post-trade value chain, PCS systems on DLT could free up collateralised liquidity quicker, thus improving availability of assets and resolving shortcomings of fractional reserve banking. On the other hand, faster processing will abolish the net benefits for liquidity provided by the (T+3) days settlement cycle. Furthermore, the probabilistic nature of blockchain-based settlements is a serious issue. Other limitations of current blockchains are its operational capacity, performance-based scalability, limitation of block size and issues with self-executing code. Immutability of DLT is also a problem, since PCS systems require a capability for error management, maintenance and management of technological failures or misuse.

Transfer and ownership of the assets through central bank-maintained systems has also been argued as a hypothetical beneficiary from blockchain adaptation. The tamper-resistant nature of blockchain could reduce legal, operational and overall systemic risks. Business sensitive information could be protected through encryption, while improving regulatory supervision and increasing transparency of asset ownership. On the other hand, issues with proprietary rights and obligations of assets on DLT and enforceability of the rights of the transacting parties in single or multiple jurisdictions are not assured by the current financial regulators and supervisors. There is not even a standardised definition of what constitutes a digital blockchain-based asset.

Small amounts of research are devoted to the enhancements to the *audit trail* for regulatory purposes from blockchain application. The immutable, tamper resistant nature of DLT promises to ensure traceability and transparency of audit for any history of funds and securities. However, blockchain-based auditing still requires regulators to accommodate record keeping by providing authoritativeness and reliability checks for those records.

Blockchain innovation for *regulatory compliance* is also extensively covered by the research. Development of blockchain-based technical interoperability standards, as a base connectivity layer promises to lower technological integration cost, provide access to more granular standardised data, thus bringing network scale efficiencies. Moreover, establishment of regulatory sandbox models should ease regulation in testing, development and delivery of blockchain solutions for central banking. Nevertheless, if blockchain application were to create risks in one area of central banking through interconnection of existing financial markets and interdependence of legacy payment infrastructures, these risks will be transmitted to the whole financial system. Furthermore, interoperability between blockchains and legacy financial systems or even between different niche DLT architectures is still in its infancy, leading to additional complexity, incompatibility and operational risks.

Influence of *CBDC regime onto Monetary Policy* operations is also discussed by the research as another aspect of financial regulation. On the positive side, CBDC is seen as an appropriate policy response to the payment innovation, where it could be utilised as an additional monetary policy tool used e.g., as a policy transmission channel, simplifying systematic and transparent conduct of it, or a type of QE. A Central Bank can also commit to an algorithmic rate of money creation. On the negative side, the highly discussed issues are the immaturity of current blockchain architectures for CBDC adaptation and lack of empirical research on the impact of such CBDC regimes onto monetary policy performance. This leads to the conclusion that the move towards CBDC adaptation would be premature.

Lastly, the discussion on how *PCS application on blockchain can improve regulation* concludes that there is an opportunity for central banks to enhance their regulatory auditing functions, utilising data visibility offered by blockchain, hypothetically improving resolution management capabilities. On the other hand, issues arising from such novel systems attracts more attention from researchers as current legislation is not adopted to cover DLT-based PCS. The other issue is the importance of legal settlement finality for PCS activities as a key element of risk management. Blockchain's ability to sustain settlement is not clear, as current consensus protocols are probabilistic, further imperilled by the existence of forks. Furthermore, the ability to host those PCS systems in multiple jurisdictions opens them up to the risk of regulatory arbitrage, complications with compliance with BSA, KYC, ALM, CFT, GDPR etc. As these novel blockchain technologies are not fail-safe, operational risks, recovery and litigation expenses could be greater than the promised potential rewards from DLT-based PCS systems. In fact, the blockchain implementation for the central banking industry is one where practical application and theory *both* have integral roles to play in moving forward. The theory can be supported well by research in best practice and accompanied by sound and rigorous empirical studies that evaluate and compare different strategies. We are at a timely stage in blockchain's evolution for these to be now mandated.

One other criticism of some academic studies is that they are often not trialled in the field and are conducted in the rarefied and some would say artificial atmosphere of the student classroom. While there is no disadvantage to using non-industrial subjects *per se*, the industry knowledge transfer this creates is limited. If there is one over-riding lesson that this mapping study shows, it is that a coordinated and collaborative approach should be adopted between industry and academics to avoid the pitfalls of the past and to generate knowledge that progresses blockchain application, rather than widening the chasm that often emerges between the two.

2.6.11 Conclusions

The purpose of this mapping study was to examine existing peer-reviewed publications concerning the influence of blockchain technology on the business of central banks. The particular emphasis was on identifying what type of use-cases were considered for blockchain adaptation, what the research trends were and who provided that research. Discussion about why those use-cases were considered and potential benefits, risks and issues arising from blockchain adaptation to those use-cases were summarised using relevant literature.

The Systematic Mapping Study identified a spectrum of existing blockchain-based use cases for central banks covered by academic research and presented a detailed statistical and thematic analysis of those use-cases and of the overall topic. Narrative summaries of contents of the research for each of the identified use-cases was also provided. In respect of the topic of this study, overall research maturity was established by presenting frequency of publications over time with papers categorised by research channels; research depth and breadth is demonstrated via research types, research contribution and cohorts of researchers.

A critical discussion point in this review is the understanding of which exact areas and functionality of the central banking business is under the academic lens of interest. However, as the goal of the SMS was to provide an overview and to be a guiding input for SLR, a trade-of between effort and reliability of the outcome has to be made (Petersen et al. (2015)). For more informed decisions and to provide a deeper understanding of each of the areas, performing a more focused review of each separate central bank uses-case for blockchain category is needed.

This mapping is a reflection of the state-of-knowledge in blockchain for central banks at present. One valuable activity would be to update our mapping study with new publications as they arise. The concept of a *living review* (i.e., one that evolves over time and is current at all times) is one that we feel would be useful to follow.

Chapter 3

Research Methodology

Chapter 3 of the thesis introduces the research methodology underpinning the Blockchain Financial Statements project, focusing on the Design Science Research Methodology (DSRM). This approach is central for merging design creativity with scientific investigation, aiming to deliver a technology solution that not only embodies innovative capabilities but also addresses practical needs within its application environment. The chapter outlines the comprehensive steps taken from the BFS's conceptualisation to its development and eventual validation, emphasising the methodological rigor and iterative process inherent in DSRM that ensures the artefact's relevance and functionality.

Diving deeper, the chapter elaborates on the application of DSRM through various facets of the research process. It discusses the BFS artefact, detailing its functions and the problems it aims to solve, and examines the context within which the BFS operates, including the technological and regulatory frameworks that influence its design and utility. Further, it explores stakeholder interactions with the BFS, the conceptual framework guiding its development, and the design activities involved in its creation. Lastly, it identifies the key stakeholders and outlines the functional requirements of the BFS, ensuring that the artefact meets the expectations and needs of its intended users. Through a narrative that bridges theoretical underpinnings with practical application, Chapter 3 sets the stage for a transparent and replicable research journey, contributing significantly to the discourse on applying DSRM in technology-driven projects.

3.1 Definition of Design Science Research Methodology

This chapter describes the Design Science Methodology (DSRM) approach implemented in this research project and is guided by fusion of the implementation guidelines communicated by prominent academics such as Wieringa (2014), Peffers et al. (2007) and Geerts (2011), where:

Wieringa (2014) focuses on providing guidelines of how information systems and software engineering research can be performed by iterating through activities of designing the artefact that improves something for stakeholders and empirically investigates the performance of the artefact in a context.

Peffers et al. (2007) communicates the methodology for conducting design science (DS) research in information systems (IS) providing a nominal process model for research execution and the mental model for such research presentation and evaluation.

Geerts (2011) illustrates the application and integration of DSRM to accounting information systems (AIS) research through retroactive analysis.

This Thesis' exploratory nature is encapsulated in the *Single Case Mechanism* approach outlined by Wieringa (2014), focusing on a detailed, context-rich examination of the BFS artefact within a narrowly defined scenario. This approach ensures depth and specificity in investigating the artefact's design, functionality and potential impact. The reason for adopting a combined research approach in this Thesis stems from the multidisciplinary nature of the study, which integrates software architecture, blockchain technology, accounting, central banking and business orchestration. The ambition of the BFS project is to unify distinct blockchain frameworks into a novel dual-blockchain architecture, together with adaptation of selected components and processes of these separate blockchains into the BFS framework. This innovative configuration is further integrated with traditional accounting to facilitate transactional and accounting processes within the BFS ecosystem.

The rationale for selecting DSRM over other methodologies is multi-faceted and is driven by its problem solving investigative and exploratory nature involving structured and creative process of designing, building and evaluating artefacts while ensuring a practical and iterative development approach (Wieringa (2014)):

Facilitating Interdisciplinary Integration: The BFS is inherently multidisciplinary, involving concepts from blockchain technology, software engineering, financial accounting and central banking. DSRM is well-suited for such interdisciplinary research, offering a flexible, yet structured framework that can accommodate and integrate diverse concepts and effectively merge these diverse fields, methodologies and techniques necessary to tackle complex problems that span multiple domains. This is essential for the successful integration of blockchain features with accounting cycles, data and ledgers in BFS, ensuring that the technological advancements in BFS are relevant and applicable to financial accounting.

Problem-Centric Approach: DSRM is inherently problem-oriented, making it an ideal choice for this research and which begins with a clearly defined problem - the need for a more efficient, accurate and transparent method of validating financial claims post-accounting period. This methodology aligns with the research's focus on the risks of fraudulent financial reporting and the need for universally accepted authenticity controls for economic data integrity; this makes it an ideal choice for research that aims to create and assess a practical, innovative solution like BFS, where the BFS is developed as a response to the challenge of establishing a trusted liquidity provisioning channel between economic entities and public money providers.

Incorporation of Single Case Mechanism: In DSRM, a single case mechanism allows us to employ focused: "simulation of a sociotechnical system"(Wieringa (2014)) (the adaptation of a single real-life use case), to: "learn which phenomena can be produced by which mechanism" (Wieringa (2014)), of such a system, though testing the utility of novel technical system (the BFS artefact), or of its model (the novel architecture of the BFS and the interaction processes of the entities of the single case ecosystem) (Wieringa (2014)).This approach allows a detailed analysis of practical application for an artefact such as BFS. A single case study offers focused and relevant insights into the requirements for BFS's functionality to facilitate credit provisioning challenges of real-life complex economic conditions.

Alignment with Research Objectives: DSRM is inherently suited for research involving the design, creation and assessment of a new IT or software artefact (Wieringa (2014)). This aspect also aligns with the project's goal of designing and developing the BFS, but also to evaluate, through demonstration its practical viability of using blockchain for authentication of financial claim validity; fraud limitation *via* real-time or on-demand auditability and by offering a model that can be replicated or adapted in different financial contexts.

Practical and Theoretical Contributions: DSRM bridges the gap between theoretical research and practical application. DSRM supports this blend of theory and practice, ensuring that the research stays grounded in practical applicability while being informed by theoretical underpinnings. Such an approach ensures that the BFS contributes theoretically by examining blockchain's potential in BEING applied to financial reporting.

Artefact Creation and Iterative Development Emphasis: DSRM is designed to be leveraged for the development of such artefacts, from its conceptualisation through design to an implementation. The iterative process of this methodology connects problem identification to its demonstration and validation (Wieringa (2014)). The creation of the BFS architecture and the demonstration of its functionality are the central aspect of the research described in this Thesis. Emphasising the process of iterative development allows for continuous refinement of the BFS prototype, based on an ongoing validation process (Wieringa (2014)). Given the innovative integration of complex elements such as blockchain technology with traditional accounting processes, such an iterative approach ensures rationality of the outcome and alignment with the requirements of accounting practices and real-world applicability.

Importantly, this methodology is subdivided into two critical and interlinked cycles: the *Design Cycle* and the *Empirical Cycle*, each addressing distinct aspects of research.The design cycle is guided by the*design problem*, such as development of a blockchain-based accounting system that can address risks of information manipulation in financial reporting and the *empirical cycle*, aimed at answering *knowledge questions*, such as the feasibility of such blockchain technology in creating a reliable audit mechanism.

Communication: The final stage of DSRM is reporting, ensuring that the findings and contributions of the BFS are effectively communicated to both academic and industry stakeholders, facilitating broader impact and application.

3.1.1 Activities of the Research Methodology

This section of the Thesis outlines foundational elements for designing, developing and validation of the BFS's architecture and its implementation, employing combined principles and guidelines for DSRM of (Wieringa (2014), Peffers et al. (2007), Geerts (2011)). Tab. 3.1 displays the format of DSRM to guide the research process. *The first column* in Tab. 3.1 lists 6 *DSRM activities* as a nominal sequence,that make up the adopted methodology of Peffers et al. (2007) and Geerts (2011). *Column two* describes each of these activities, starting from *"what to do?"*, followed by the narrative details. Lastly, the *third column* links generalised *knowledge base* with the different activities to elaborate on how these are expected to be executed. This provides raw material to guide accomplishment of design science research process (Geerts (2011)). It accounts for identification of the most appropriate and effective knowledge tools and best practices that can be applied to enable a rigorous demonstration of the utility, quality and effectuate of the designed artefact (Geerts (2011)). These are comprised of: foundational theories, instruments, frameworks, constructs, models, methods, instantiations and more (Peffers et al. (2007), Geerts (2011)).

3.2 Application of Design Science Research Methodology

This section outlines foundational elements for implementation of this iteration of DSRM, following consolidated principles and guidelines of Wieringa (2014), Peffers et al. (2007) and Geerts (2011). To successfully utilise DSRM for the design of the BFS architecture, the development of its artefact and the demonstration of integration of blockchain technology into the accounting domain, this projects starts from clear definition and understanding of the major concepts of such methodology, namely:

1. ***the object of its study***, consisting of the artefact itself, the contexts, the interaction and the problem, and
2. the two primary activities - the ***design*** of the artefact to improve the problem context and ***investigation*** of answers to knowledge questions about this artefact in its context (Wieringa (2014)).

The ultimate goal of the research in this Thesis is to explain and predict the overall behaviour of the BFS and its ecosystem from its architectural knowledge, to enhance the efficiency and transparency of financial reporting in the digital age.

3.2.1 Object of the Study in Research Methodology

This section outlines the concept of the object of this study. The "Object of the Study" forms the core around which this research is centred. According to (Wieringa (2014)), it is essential to start by clearly defining this object, to guide a clear understanding of the scope, depth and the direction of the research work. The object often becomes the tangible output of the research process (Wieringa

DSRM Activity	Activity Description	Knowledge Base
Problem Identification and Motivation	*What is the problem?* (Geerts (2011)). Define the core research problem and justify the value of the solution (Peffers et al. (2007), Geerts (2011))	State of knowledge of the problem and understanding of the importance of the solution (Peffers et al. (2007)); Understand relevance of the problem, its current solutions, weaknesses of these solutions; Justification of the solution motivates the pursuit of such solution development and illustrates the reasoning behind the problem (Peffers et al. (2007), Geerts (2011))
Define Objectives of the Solution	*How should the problem be solved?*(Geerts (2011)). Define the objectives of the solution	State of knowledge of the problem; knowledge of state of art if any (knowledge of solutions) if any, and their efficacy (Peffers et al. (2007)). Rationally infer the objectives from the problem specification (Peffers et al. (2007), Geerts (2011))
Design and Development	*Create an artefact that solves the problem* (Geerts (2011)). Create the treatment, the artefact, its design specification and its implementation in which research contribution is embedded in the design(Geerts (2011), Wieringa (2014)). Determine the artefact's desired functionality and its architecture and then create the actual artefact (Peffers et al. (2007))	Elaborate on the "Design and Development" activity in terms of iterative process; stages of design cycle.
Validation and Communication	*Demonstrate the use of the artefact* (Geerts (2011)) Validate by demonstration that he artefact works and solves one or more problems. This could involve simulation, case study, proof, etc. Communicate the problem, its importance, the artefact, its utility and novelty, the rigour of the design and effectiveness of the solution to the relevant audience (Peffers et al. (2007), Geerts (2011))	Knowledge of how to use artefact to solve the problem (Geerts (2011)); Knowledge of the disciplinary culture (Peffers et al. (2007), Geerts (2011))

Table 3.1: Activities of the DSRM

(2014)). Definition of the object encapsulates several key components: the *Artefact* (the Blockchain Financial Statements (BFS)), the *Context* (financial reporting and liquidity management), the *Interactions* (transactional interchange between various economic entities and with the BFS); and the *Problem*(fraudulent financial reporting and lack of or inadequate on-demand auditability).

3.2.2 The Artefact

In DSRM, an artefact refers to a construct developed to address a specific problem and embodies the practical application of theoretical concepts (Wieringa (2014)). Within the setting of this research, the object of the study is the exploration and development of the Blockchain Financial Statements (BFS), as the innovative artefact that integrates blockchain technology with financial accounting processes. This artefact represents a novel combination of two patterns' different blockchain architectures and integration of such technology into a financial accounting system, with the aim of addressing the issues of authenticity and integrity of post accounting monetary claims. The primary role and the re-purpose of the BFS's functionality is to provide a solution to the problems of fraudulent financial reporting, born from inadequate auditability processes. It is designed to facilitate an on-demand authenticity mechanism for validation of these post-reporting period monetary claims. The guarantees provided by the BFS are supported by the maintenance of a tamper-evident blockchain ledger that contains financial statements of an entity.

3.2.3 The Context

In DSRM, the context for the artefact centred around of critical definition of its operational environment, the settings or the conditions in which this artefact is applied and holds relevance to, or a specific domain it impacts (Wieringa (2014)). The research in this Thesis is contextualised within the domain of financial reporting and liquidity management.This domain is characterised by the risks of intentional fraudulent reporting in financial statements, necessitating a reliable, verifiable and tamper-evident mechanism for auditability. In response to these challenges, the innovation of the BFS is to provide secure and transparent framework for reporting of the outcomes of business transactions. This role is increasingly vital, given the evolving landscape and increasing digitisation of financial interactions, where traditional mechanisms fall short in addressing new complexities. The necessity for development of such a solution targets the enhancement of financial reporting mechanisms, especially focusing on liquidity management among a range of economic entities. It ensures that the BFS is relevant and highly applicable to the real-world environment and that it improves business processes in the economy. This application is demonstrated practically though adaptation of the real-world scenario, born from the Global Financial Crisis and COVID disruptions, demonstrating BFS's capability to manage real-time financial complexities.

3.2.4 The Interactions

According to Wieringa (2014): *"the design science researcher designs not just an artefact, but also designs a desired interaction between the artefact and the problem context"*. Exploring such interactions between the BFS and its contextual ecosystem of distinct economic entities is important, as these combine diverse and dynamic relationships and activities that could transpire between the BFS and its users. In this research, these interactions are illustrated through transactional exchanges between economic entities and coordination of responsibilities between them. These are the stakeholders (see Section 3.2.7), which interact with the BFS in various capacities, such as incorporating a business, conducting transactions, performing reconciliation and generating reports. The exploration of interactions is descriptive and also serves several important functions for the research:

Demonstrating Practical Applicability: The interactions provide evidence of how the BFS can be implemented and utilised in real-world financial settings, thus bridging the gap between theoretical development and future practical deployment.

Assessing System's Efficacy: Through these interactions, the research assesses the artefact's effectiveness in facilitating transparent and secure financial transactions, a key concern highlighted throughout initial chapters.

Understanding Impact: By studying these interactions, the research gains insights into the system's impact on routine business operations and its role in enhancing overall financial reporting quality.

Identifying Opportunities for Improvement: These interactions also serve as a feedback mechanism, highlighting areas where the BFS System can be further refined to better meet the needs of its potential users.

Adopting a single case-based experiment technique Wieringa (2014), the structure of the BFS architecture is explored through the lens of an individual real-life use-case study. This approach involves studying the system's architecture, identifying mechanisms that produce system-level phenomena (e.g., transactional interactions among heterogeneous economic entities) and generalising the functional requirements for the final BFS prototype. This in-depth examination of interactions thus plays a significant role in validating the BFS as a viable and innovative solution in the field of financial technology. They provide insights into the practical application and transformative potential of the BFS by illustrating how it is operated as an innovative channel for the distribution of liquidity. Lastly, they demonstrate the artefact's functionality and its capability to facilitate trust through secure financial transactions and transparent reporting.

3.2.5 BFS Conceptual Framework

In this section, the *Conceptual Framework* for the BFS artefact and its context is presented. It is constructed leveraging the DSRM of Wieringa (2014) and outlines a structure that guides the development, implementation and validation of the BFS artefact. The development and illustration of the methodological framework for the BFS is a crucial part of the interpretation of resulting artefact. It is represented by a unique combination of its two primary components, blending design of technology with its ecosystem. It encompasses:

1. *Technology Integration:* The BFS encompasses a hybrid blockchain technology that fuses the traditional Bitcoin-style blockchain data structure with smart contract functionality, along with components and network communication models of the Corda architecture and its Flow Framework. This innovative combination forms the technological backbone of the BFS, enabling a robust and versatile accounting platform for automation of financial recording and reporting processes.
2. *System/Ecosystem Design:* The design of the BFS system and the overall ecosystem extends beyond simple technological integration. At the system level, it involves identification and implementation of accounting and business processes with this novel blockchain architecture. The goal is to outline an ecosystem where these diverse elements integrate seamlessly into the interaction between economic entities to identifying mechanisms that produce and drive system-level phenomena, such as transactional interactions among heterogeneous economic entities.

To design and investigate the BFS in these contexts, the work in this Thesis builds on the a conceptual framework of Wieringa (2014) that defines the structures within the BFS and its ecosystem interactions. This framework is based on an architectural conceptual structure that views "the world as a hierarchy of interacting systems" (Wieringa (2014)). In such a structure, each of these systems is: "an entity comprising components that collectively influence overall system behaviour" (Wieringa (2014)). These components can be further deconstructed into lower-level elements or can be consolidated into some other composite system (Wieringa (2014)). The conceptual constructs that represent these components of the BFS framework are outlined in Table 3.2.

Leveraging on the above phases of this project implementation and the components of the framework, the BFS conceptual framework, as illustrated in Fig. 3.1 demonstrates how the application of knowledge within the context of identified problems can lead to the creation of a BFS solution using a Design Science approach. This framework emphasises rigour and relevance, illustrating the system's contribution to both knowledge and practice in the field. Utilizing design, building and validation stages of this methodology, the final BFS can be rigorously examined and refined, contributing to both the academic knowledge and practical applications in financial reporting and liquidity management.

3.2.6 Design Activity

The intention of this work is to predict and explain the overall system's behaviour, based on the knowledge of its architecture - the BFS. To achieve this, the work focuses on a single case exper-

Concept Name	Concept Description
Populations	This refers to the population of economic entities involved in the BFS ecosystem, encompassing a diverse range of actors in the financial sector; technological components such as blockchain.
Entities	These are the actors and components within the BFS system. They include businesses, regulatory bodies and other stakeholders who interact with the BFS System.
Processes	These are the interactions or sequential steps in the use-case, detailing how the BFS is employed in real-world scenarios for implementation of financial transactions, transformation of transactional data into accounting format, reporting of accounting results and overall accounting process.
Events	These are pivotal moments that trigger specific actions or processes of the BFS system. Key events within the BFS framework include business transaction completions and significant periods like the end of accounting periods which are critical markers in financial reporting.
Taxonomic relationships	This aspect covers the business transaction registers, accounting ledgers and wallet types within the BFS implementation, outlining how different financial elements are categorised and interconnected.
Cardinality relationships	These define the relationships between accounts and ledgers in the BFS system, including one-to-one, one-to-many and many-to-many relationships.
Procedure specifications	These include accounting procedures, transactional agreements, transactional execution protocols and cryptographic security verification protocols. They are essential in ensuring that the BFS system operates securely and in accordance with established and expected accounting standards.
Variables	These are the properties of the system and the interactions of its components, which include both technical and operational aspects of the BFS system and crucial for understanding system behaviour and functionality.

Table 3.2: BFS contracts that define its conceptual structure

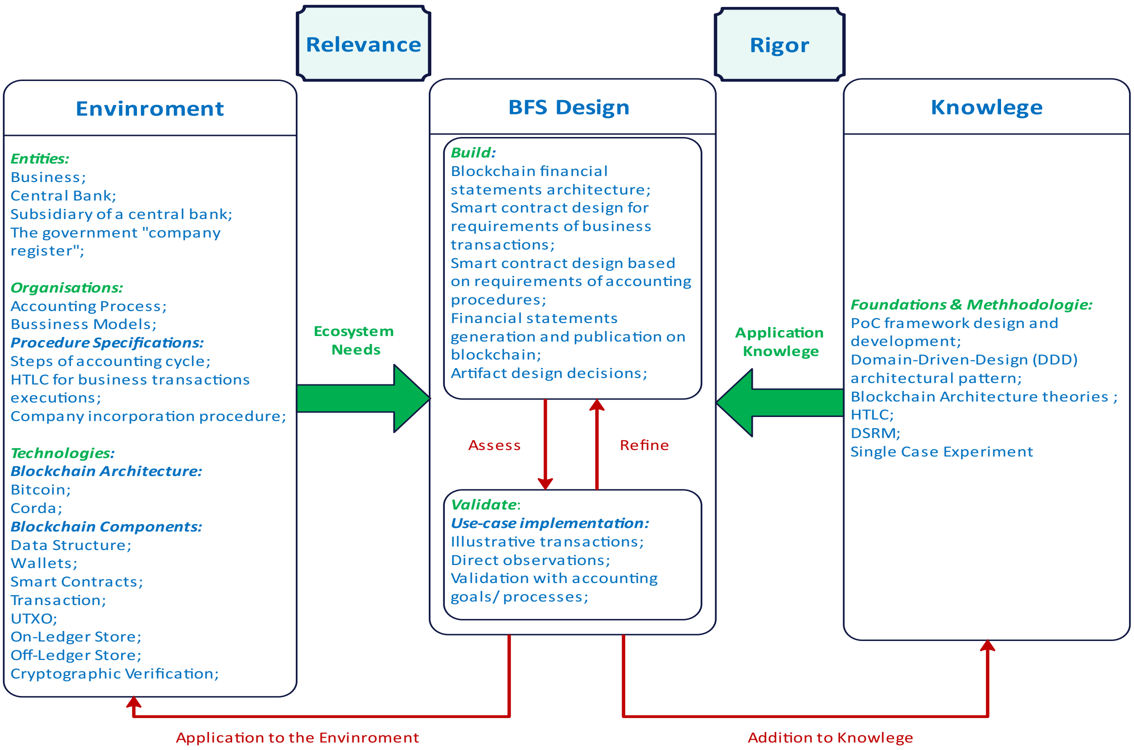


Figure 3.1: BFS Conceptual Framework.

BFS conceptual framework addresses research through the *building* and *validation* of BFS artefact (central top rectangle), designed to meet the identified business need of its *environment* (rectangular most left box). The *knowledge base* (rectangular most right box) provides raw materials from and through which DSRM based BFS research is accomplished. The knowledge base is composed of *foundations and methodologies*. *Methodologies* provide guidelines used in the *validate* phase (central bottom rectangle of the design box). *Rigour* is achieved by appropriately applying existing foundations and methodologies. The contributions of DSRM based BFS research are assessed as they are applied to the business need in an appropriate environment and as they add to the content of the knowledge base for further research and practice (bottom red arrows).

iment technique (Wieringa (2014)) where the structure and the functional requirements for the BFS architecture are explored through the lens of an individual real-life use-case study. Such decomposition of the use-case dictates the identification of key mechanisms responsible for creating a system-level phenomena and its components (Wieringa (2014)). Analysing how the various components of the BFS ecosystem interact within the network of heterogeneous economic entities, such as businesses, to facilitate transactional interactions involves testing the BFS, making inferences, drawing conclusions and then applying new tests to further refine or enhance the resulting artefact, or leaving further optimisation of the BFS to future researchers. As outlined in Tab. 3.1, the work flow is outlined next.

The first step is ***Problem Identification and Motivation***. This step provides the foundation for the entire research by clearly identifying the problem that the BFS artefact intends to solve and establishing the motivation for it. To understand and elaborate on the problem, the *research activity* is to include: in-depth examination of the current financial and accounting systems, with a particular focus on the issues of accounting fraud, lack of real-time auditability and the inefficiencies in the liquidity support distribution mechanisms. The motivation behind investigating these problems stems from the potential of blockchain technology to improve accounting conduct by introducing transparency, security and efficiency.

The second step in Tab. 3.1 involves ***Defining Objectives of the Solution***. This step specifies what the BFS must achieve to address the identified problem. This step bridges the gap between the problem space and the solution space, guiding the design process of the BFS artefact. It will include *stakeholder identification and their goal analysis* to guide specifications for functional requirement of the BFS. Each stakeholder group has distinct goals, such as enhancing financial reporting accuracy, ensuring real-time auditability and streamlining liquidity support mechanisms. The functional requirements are aimed at leveraging blockchain technology to improve financial and accounting processes. The design goals for the BFS are directly informed by the functional requirements, focusing on creating a blockchain-based system that meets the identified needs of the stakeholders and integrates into existing accounting infrastructures.

The third Step of Tab. 3.1 is ***Design and Development***. This step focuses on creating the BFS itself and involves conceptualising, constructing and iterating the design, based on the previously defined objectives. Initially, the conceptual architectural framework of the BFS is developed. This includes defining how the dual-blockchain system can interact with the accounting and transactional processes, ensuring that stakeholder requirements are met. Next, a detailed description of design and data structures for BFS components is presented, such as smart contracts for automating accounting and transactional processes, the integration framework for Bitcoin and Corda blockchains and the user wallets for stakeholder interaction. Each component is designed to fulfil specific functionalities outlined in the solution objectives. The BFS undergoes several iterative design and development cycles. Each iteration involves building a prototype, testing its functionality and compliance with stakeholder requirements, analysing compliance with these requirements and refining the design. This iterative process ensures that the final BFS artefact effectively addresses identified challenges.

Theoretical and practical knowledge of Domain-Driven Design (DDD) is essential for modelling the BFS to ensures its functionality is directly relevant to the needs of its users. Familiarity with iterative design and development methodologies, versioning control to ensure that the BFS can be developed and refined in a responsive and adaptive manner. Understanding of DSRM principles to guide the BFS's development process ensures that each design iteration contributes to the artefact's ability to meet its objectives and support stakeholder goals.

The fourth and final step of Tab. 3.1 is ***Validation and Communication***. The final phase involves validating and demonstrating the BFS artefact's capabilities *via* its implementation and communicating its utility and impact to the relevant audience. Validation of the BFS serves to bridge the gap between theoretical design and practical application. Guided by the knowledge base, this step ensures the BFS is functional and also impactful in addressing the challenges of financial reporting and accounting in the blockchain era. This involves executing the BFS code to showcase its innovative integration of blockchain technology with traditional accounting processes. It includes demonstrating the execution of business and accounting transactions through the system, showcasing the automated generation and publication of financial statements, and illustrating the artefact's blockchain-based record-keeping and data integrity features. Furthermore, it is crucial to validate that these functions effectively address the problems identified in earlier steps, such as improving financial transparency, enhancing data integrity and reducing fraud in accounting practices. Scenario-based testing, derived from the use-case is used to simulate the BFS's deployment. This testing method assesses how the artefact interacts within its environment and processes data, ensuring it behaves as expected and is reflective of its operational context. This includes transaction execution, data transformation and the generation of financial statements, ensuring the artefact operates as expected.

Communication activity involves articulation of the importance of the artefact, its utility and novelty, the rigour of the design and effectiveness of the solution to the relevant audience. This involves highlighting the BFS's unique features such as its dual-blockchain architecture and its utility in providing a secure, transparent and efficient platform for financial management and accounting. This involves presenting the effectiveness of the BFS in achieving its intended goals and the benefits it offers to its stakeholders.

The *knowledge base* supporting this step includes, for the validation process, an understanding of the application of blockchain technology in financial and accounting contexts is essential to validate the artefact's blockchain-related functionalities, such as smart contract execution and ledger integrity. Furthermore, familiarity with accounting principles and financial reporting requirements is necessary to validate that the financial statements generated by the BFS are accurate, compliant and meaningful to stakeholders. Knowledge of DSRM guidelines will inform this validation process, ensuring it is systematic, thorough and aligned with the project's research objectives. This includes employing appropriate validation techniques like simulation or laboratory manual testing where applicable. For the communication activity, necessary skills can be utilised for communication of both, the technical aspects of the BFS and its broader implications for financial and accounting practices to a diverse audience.

3.2.7 Knowledge Base for Design and Development Activity

The knowledge base of the BFS is interdisciplinary. The BFS is informed by existing theories of computer science and software design, including those specifically related to blockchain, theories of accounting and auditing, central banking and business. This section provides an overview of the various elements that constitute the knowledge context for the BFS project, underlining the significance of integrating insights from different domains to inform the design and implementation of the BFS (Wieringa (2014)).

Theoretical Foundations (Wieringa (2014)). These theories provide a foundational understanding of the principles guiding the development of blockchain applications, requirements of accounting and the architectural frameworks that ensure the system's functionality.

- ❖ Blockchain architectures, such as Bitcoin and Corda, includes a range of disciplines including cryptography, computer science, economics and distributed systems theory. Although they apply theoretical principles differently to suit their unique goals and use cases. Bitcoin focuses on creating a decentralised digital currency, while Corda aims to facilitate private, efficient and legally-binding transactions between entities in a business network.
- ❖ Software architecture and design theories such as Domain-Driven Design (DDD) architectures.

- ❖ Accounting - principles and frameworks necessary for systematic process for recording, analysis, interpretation of business transactions and for recording, classifying, summarizing and interpreting financial data to produce financial statements. Among others are double entry bookkeeping and regulatory reporting frameworks.
- ❖ Central Banking theoretical foundations involve around its role and functions, such as a lender of last resort to stimulate the economy during economic dysfunction, provision of money supply and overseeing payment systems. (QE involves the purchase of long-term securities from the open market to increase the money supply and encourage economic activity.)

Empirical and Practical Knowledge from Existing Products (Wieringa (2014)). Exploring useful facts about currently available products and established standards and practices reveals practical insights into how data models are structured and how common practices can be embedded within a BFS. These include understanding the stages of accounting and specifications derived from existing accounting products such as "Sage 50" accounting software, accounting documentation and reporting requirements, such as those obtained from the Companies House UK and generalised blockchain knowledge of Bitcoin and Corda:

From Bitcoin:

- ❖ Cryptographic Hash Functions: Bitcoin relies on cryptographic hash functions like SHA-256 for various purposes, including the creation of bitcoin addresses, the mining process (proof-of-work) and the integrity of transaction data in blocks.
- ❖ Public-Key Cryptography: Bitcoin uses public-key (asymmetric) cryptography to ensure secure transactions between parties. Each user has a pair of keys: a public key, which is shared and a private key, which is kept secret. This allows for secure digital signatures, essential for the ownership and transfer of bitcoins.
- ❖ Bitcoin Data Structure: The blockchain is a distributed ledger technology that underpins Bitcoin. It is essentially a chain of blocks that contain transaction data, secured by cryptographic principles. The concept of a blockchain combines elements from data structures (like linked lists and trees) and distributed systems to achieve decentralisation, integrity and transparency.

From Corda:

- ❖ Flows: Corda introduces the concept of "Flows" for automating transactions. Flows enable the coordination of complex multi-step protocols between nodes, ensuring all necessary steps are completed successfully for a transaction to be finalised. This concept leverages programming models and distributed systems theories focused on process orchestration and transactional consistency.
- ❖ Pluggable Consensus: Corda allows different consensus mechanisms to be plugged into the network accommodating various requirements for notarisation and validation. This flexibility is built on theoretical work in distributed consensus and Byzantine fault tolerance, allowing Corda to cater to a wide range of use cases with differing needs for security, finality and privacy.
- ❖ Smart Contracts: Corda uniquely integrates legal prose with its smart contracts, ensuring that each transaction is directly tied to legal terms and conditions. This approach is grounded in contract law and aims to bridge the gap between traditional legal processes and digital transactions.

Key Stakeholders

The BFS project, situated within the social context of a design science framework, brings together diverse array of stakeholders, each with distinct roles, expectations and potential impacts from the implementation of the system (Wieringa (2014)). In this context, the identification or hypothesizing about stakeholders is important for understanding the social dynamics and for defining functional requirements for the design and implementation of the BFS and to achieve the **5th Objective** of this research - to offer actionable insights and recommendations for these identified stakeholders.

Given the exploratory nature of the BFS project, it is possible that not all potential stakeholders are immediately aware of their stake in the project's outcomes. This section describes stakeholders of the BFS, exploring their potential roles, interests and the implications of these roles within the project's ecosystem. Nonetheless, hypothesizing about potential stakeholders and their interests is a useful exercise for anticipating the project's broader impacts (Wieringa (2014)).

Key "benefiting" Stakeholders:

1. *Economic Entities.* These include businesses and organisations that engage in economic activities. These entities can utilise BFS for transparent, secure and efficient transaction processing and financial management. They stand to benefit from the enhanced auditability and reduced risk of fraud.
2. *Company Members and Customers.* This group includes individuals or entities directly involved with the companies using the BFS, such as employees, shareholders and customers. They are interested in the reliability, security and efficiency of the transactions and financial reporting facilitated by the BFS.
3. *FinTechs and Accountants.* Accountants play a central role in the operation and maintenance of the BFS and the implementation of accounting processes. As the primary users of the system's accounting and auditing functionalities, they require a system that enables automation to streamline existing procedures, enhance accuracy and provide real-time auditability. Furthermore, a new professional niche, driven by FinTechs may emerge at the intersection of current accounting practices and blockchain technological adaptation. This could lead to the creation of novel roles for accountants who specialize in integrating blockchain into financial reporting and management, requiring innovative designs and methodologies.
4. *Government (Sponsor).* The BFS project aligns with governmental goals of improving societal practices and enhancing economic well-being. Agencies like the Engineering and Physical Sciences Research Council (EPSRC) in the UK, which invest in research projects to benefit society and the economy are key sponsors. The success of the BFS could facilitate more effective distribution of financial support and improve regulatory oversight, thereby fostering economic stability and growth.

Key "worse-off" Stakeholders:

1. *Manual Accountants.* As the BFS integrates blockchain into financial operations, traditional accountants might find themselves needing additional training to adapt to the new technology. There is a risk of job displacement for those unable to transition, highlighting the need for educational programs and support structures.
2. *Fraudsters.* By its very design, the BFS aims to reduce the potential for financial fraud through its tamper-evident and transparent transaction ledger. This stakeholder group therefore stands to lose from the successful implementation of the BFS, as it would significantly hamper their malicious activities.

These stakeholders in the BFS project bring diverse perspectives and requirements to the table. By identifying these stakeholders and understanding their goals and potential conflicts, the BFS project can navigate the social context effectively, ensuring that the system is technologically sound and socially relevant/beneficial. Addressing stakeholder needs and resolving conflicts among their goals are vital steps toward the successful implementation and adoption of the BFS in the broader economic ecosystem.

Functional Requirements

Wieringa (2014) defines functional requirements as: "the results of the design choices that we make jointly with or on behalf of the stakeholders" (Wieringa (2014)). Wieringa (2014) further states that, in order to justify these requirements, the contribution for each must be articulated. The important point is to keep in mind that this contribution is *predictive* (Wieringa (2014)): "because it argues that when the artefact would be inserted in the problem context, it would interact with it in a way that contributes to stakeholder goals. This argument is fallible, because it does not provide deductively certain support to its conclusion" (Wieringa (2014)).

To achieve stakeholder goals described in the previous section, it is predicted that the BFS has to fulfil following functional requirements:

- Tamper-Evident Record-Keeping: Implement blockchain technology to ensure that all transactional data and financial statements are immutable and verifiable. This requirement is important to ensure the integrity and immutability of financial records to prevent fraud and unauthorised alterations.
- Real-Time Auditability: Enable real-time or on-demand verification of financial transactions and statements to enhance transparency and trust. This requirement is necessary to enable stakeholders to verify the authenticity and accuracy of financial data on demand, enhancing transparency and trust.
- Privacy vs. Accountability: Balance transactional privacy with regulatory compliance and accountability needs, using technologies like the Corda Flow framework for controlled data sharing. This is to facilitate confidentiality of sensitive financial data while ensuring that traceability and transparency requirements are met.
- Automated Transactional and Accounting Processes: Utilise smart contracts to automate business and accounting transactions, reducing manual errors and improving efficiency.
- Digital Identity: Provide digital wallets, implemented as secure repositories for cryptographic identities ensuring that all participating entities can be uniquely identified and verified when engaging in transactions.
- Role-Based Access and Transactional Capability: Implement business-role based wallets to enable precise control over access to sensitive financial and ledger data, combined with business role defined authority and transactional capabilities, to enable entities to execute and track transactions.
- Secure Multi-Blockchain Transactions: Support secure and efficient cross-blockchain transactions to enable interoperability among different economic entities and their blockchain systems.

Chapter 4

The BFS Architectural Approach

Back in the late 1960s, Melvin Conway made an observation that has become known as *Conway's law* Bird (2010):

> *Organisations which design systems ... are constrained to produce designs which are copies of the communication structures of these organisations.*

Chapter 4 of the Thesis describes the architectural strategy employed for the BFS project, emphasising the critical role of Domain-Driven Design (DDD) in harmonizing the advanced capabilities of blockchain technology with the specific requirements of accounting and business reporting. The DDD approach, known for its efficacy in handling complex domain logic, provides a structured methodology for the BFS's architecture. At the core of this approach is the categorisation of the BFS into three principal sub-domains; financial accounting, blockchain, and business entity. Each of these represents a distinct aspect of the BFS's operational domain and playing its part in achieving the overarching objectives of the BFS project.

Further exploration of the BFS architecture reveals its implementation through bounded contexts, a strategy that segments the BFS into more manageable components, each dedicated to a specific sub-domain. This segmentation facilitates a detailed focus on the individual aspects of Financial Accounting, Blockchain technology, and Business Entity, allowing for specialised development and optimisation within each context. It provides a clear delineation of domain boundaries, promoting an efficient allocation of resources and focused attention on domain-specific challenges and solutions.

On the technological front, Chapter 4 introduces a technological stack for BFS, and introduces innovative dual blockchain framework as a cornerstone of the BFS's implementation. This integration forms the foundation of the BFS's proof-of-concept (PoC) implementation, demonstrating the system's capacity to operate within a complex multi-ledger ecosystem. By leveraging distinct blockchain platforms, the BFS achieves a balance between security, transparency, and operational efficiency. This dual-framework approach introduces the project's technological strategy, aiming to harness blockchain's full potential to revolutionise financial reporting and management practices. The architectural and technological pathways charted in this chapter lay the groundwork for a BFS that not only addresses the immediate challenges of blockchain application in accounting but also sets a precedent for future innovations in the domain.

4.1 Architectural Approach

BFS architecture is developed and refined through a blend of design and empirical methodologies (Wieringa (2014), Peffers et al. (2007), and Geerts (2011)). To enable implementation of the iteration stages of these, the architectural pattern or approach that emphasises alignment of software architecture with complexities of business domain and its processes must be selected. As already outlined in the previous chapters, the goal of the design and implementation for the BFS is to address existing challenges in financial reporting and liquidity management. These can be summarised as follows:

- ❖ The need to establish tamper-evident and secure connections between diverse economic entities and public money providers is identified as critical. BFS aims to automate and verify accountability and compliance with some arbitrary rules defined by these public money providers for liquidity distribution - a complex task that requires deep understanding of domain-specific nuances.
- ❖ Another key aspect of BFS is the establishment of mechanisms for post-reporting period validation of monetary claims, ensuring financial data integrity and chronological validity that defined these monetary claims. This requires a complex design alignment between diverse accounting processes and blockchain technological specifications.
- ❖ BFS also seeks to establish reliable connections between the financial statements of public money providers and heterogeneous businesses, enabling the assessment and transmission of liquidity through these connections. This direct connectivity is necessary to facilitate authenticatable and timely financial information flow, essential for informed decision-making across diverse financial landscapes and to provide a robust, efficient and transparent financial reporting system.

To facilitate such a complex integration of blockchain-enabled framework and its components with accounting standards and business reporting requirements, the current project utilises the **Domain-Driven Design (DDD)** architectural pattern, as a logical process for design of the BFS architecture (Richards & Ford (2020)).

The following sections will describe how DDD was utilised for this project with significant domain complexity, to enable timely and secure liquidity transmission channel between economic entities, with blockchain supported verifiable authenticity, guarantee of integrity, chronology and cryptographic security of the reported transactional data.

4.2 Domain-Driven-Design (DDD) Architecture

Domain-Driven-Design as an architectural approach emphasises deep understanding of complex application domain logic, focusing on complexities of business domains, intricacies of its logic while offering patterns to model any software design. Core components in the DDD utilised for the design and implementation of the BFS artefact are: domain, domain model, bounded contexts and ubiquitous language.

Domain in DDD is defined as the subject area around which the software solution, such as the BFS is built.

Domain model is a conceptual object model representing different parts of the domain that will be used in a software solution, such as BFS. It includes both behaviour and data. The purpose of the domain model is to show what information is needed and the structure of that information. It can be applied to design of technical diagrams defining data schemas.

Bounded contexts are the context into which a model can be applied to. A complex project often has multiple domains, each with their own bounded context.

Lastly, the ***ubiquitous language*** is the common domain language used for descriptions and understanding of entities, elements, methodologies and all relevant components within domain models and bounded contexts.

4.2.1 Top-Level Business Domain

According to Evans (2004), in DDD, a *business domain* refers to a specific area of business expertise or knowledge, processes and rules that the software system is designed to address or support. It ensures that the business rules and behaviours are encapsulated within that domain's entities and its services promote clear understanding and representation of the implementation. It prioritizes creating a software model that mirrors complex business entities, reflecting their relationships and business rules to enable their interactions. The application of DDD in BFS is demonstrated by a single use-case scenario, showcasing how a BFS underpinned ecosystem can handle financial transactions and reporting within a blockchain-enabled framework. This scenario demonstrates applicability and effectiveness of DDD in creating a solution which is both technically sound and closely aligned with this domain. The DDD pattern enables decomposition of BFS into its composite domains, facilitating creation of comprehensive logical models that encapsulate various blockchain elements such as smart contracts, its data ledgers and cryptographic protocols, alongside accounting principles and procedures. These models serve as a blueprint for the BFS system's design and its implementation.

Another central concept of the DDD is the development of a *ubiquitous language* which bridges the gap between technical experts and domain practitioners and resonates across all aspects of the application. This language enables all parties to have a mutual understanding of the key concepts and processes within the application, including software developers, blockchain experts, financial accountants, auditors and business managers. This common communication mechanism encompasses terms to describe the blockchain technology together with financial accounting, maintains consistency in terminology and conceptual understanding. This uniformity is crucial for ensuring that system requirements, design models and implementation are all aligned with the project's goal as well as being understood by stakeholders. As importantly, the process of development of such a language is often collaborative and non-static, as it evolves as the system develops and as new insights about the domain are gained. It enables development of documentation, fostering communication and consistency and ensuring that the language accurately reflects both the technical aspects of blockchain and the nuances of financial reporting for the BFS. Through this language, BFS can effectively integrate the intricacies of blockchain technology with the domain-specific requirements of financial reporting, leading to a system that is both technologically robust and domain-aligned.

At the heart of BFS is its primary top-level domain model described as: *"an economic entity, that leverages blockchain technology, customised to serve as this entity's financial accounting system"*. This technology continuously records and periodically reports on economic activity of such an entity and provides verifiable, on-demand validation of the financial health of such an entity based on its past transactional interactions.

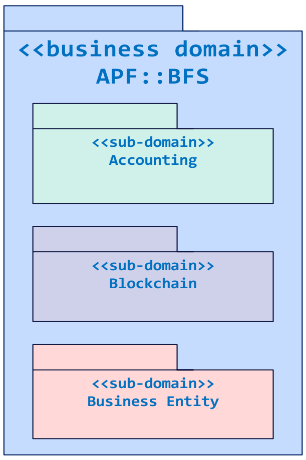


Figure 4.1: Top level business domain model for the BFS artefact.
The top level business domain of the BFS is associated with the light blue colour. The accounting subdomain is associated with the turquoise colour. The blockchain sub-domain is associated with the lighter blue colour. The business entity sub-domain is associated with the salmon colour.

In Fig. 4.1 the core domain model for the BFS is illustrated. It is sub-divided into three key *sub-domains:*

1. **Financial Accounting**;
2. **Blockchain**;
3. **Business Entity**.

4.2.2 Bounded Contexts

Bounded contexts are a fundamental concept of DDD, responsible for defining clear boundaries within the application's system and encompassing a specific subset of the domain model. Together these contexts ensure that a software application, such as BFS is modular, with each module having a distinct and well-defined role. By dividing the design of the architecture of the BFS and its implementation into distinct bounded contexts illustrated in Fig. 4.1, the overall project achieves clarity and focus. This focus enables addressing of separate and specific aspects of business management and its overall existence. Contextualizing this process helps in clarifying specifications for integration of accounting and financial reporting within blockchain framework (thereby reducing complexities) and ease adaptability to changing requirement or emerging needs of the application.

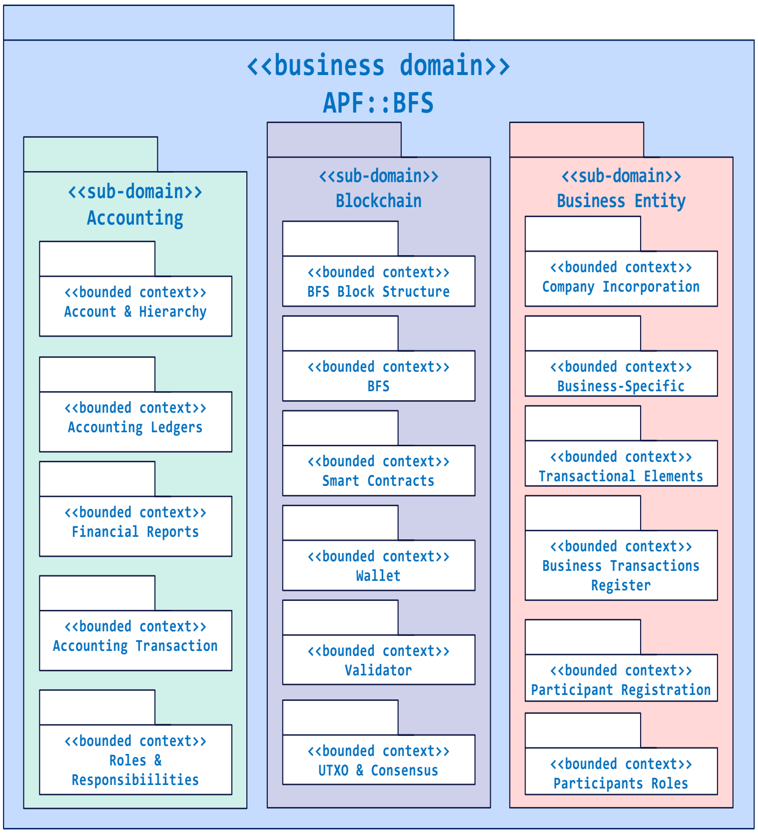


Figure 4.2: Top level business domain model for the BFS with its bounded contexts. The top level business domain of the BFS is associated with the light blue colour. The accounting subdomain is associated with the turquoise colour. The blockchain sub-domain is associated with the lighter blue colour. The business entity sub-domain is associated with the salmon colour. Each of these subdomain contains that subdomain relevant contexts.

Each of the sub-domains of the BFS, identified in the previous Section 4.2.1 and illustrated in Fig. 4.1 consists of a number of bounded contexts relevant for BFS artefact, as illustrated in Fig. 4.2. Each of these contexts has its own ubiquitous language, granular sub-domain models, contexts, data repositories, services and events, optimised for its specific area of responsibility within BFS. This structure ensures that the BFS comprehensively addresses complexities of business existence and its transactional interactions, necessary accounting processes, blockchain protocols and compliance rules.

Each of these contexts, with its specific focus and responsibilities, contributes to the overall coherence of the BFS architecture, ensuring that it meets technical specifications of blockchain technology and aligns with the complicated needs of financial accounting in the digital age.

Financial Accounting Sub-Domain

Financial Accounting - represented by traditional accounting roles, processes and procedures, adapted and integrated into the BFS. It is structured around several bounded contexts, each addressing a distinct aspect of these financial accounting processes and their further integration within blockchain technology. The application of DDD principles has led to the segmentation between such bounded contexts, to ensure that the BFS architecture is modular and clear, focusing on specific aspects of financial management and reporting and so on. These bounded contexts must meet with technical requirements of blockchain technology and must align with the nuanced needs of financial accounting, ensuring that the BFS is a comprehensive and user-centric solution.

1. *Account and its Hierarchy Context* encompasses a fundamental concept of accounting variable - the *financial account* - representing a monetary denomination owned by or owed to the business. This includes the definition of account ID, account types such as cash, equity and debt securities, among others. The classification and hierarchical relationships between accounts within this context are tailored to reflect the economic activities and financial structure of the business entity and are specific to the use-case - the APF. Elements of this context provide a structured framework for categorising and tracking aspects of business activity within the BFS, facilitating precise financial analysis and reporting.
2. *Accounting Journals and Ledgers Context* is dedicated to the data structure of the accounting ledgers and the management of financial accounts within these ledgers, emphasising organisation and governance of these financial accounts. The ledgers that are relevant to the use-case of BFS project are: Chart of Accounts (COA); General Journal (GJ), General Ledger (GL), and the Running Trial Balance (RTB). Governance of these ledgers ensures adherence to sequential rules for data transmission between traditional accounting ledgers and the maintenance of the balance of the primary accounting equation: *Assets = Liabilities + Equity* through the *double-entry accounting method* (Stolowy & Ding (2019)). This diligent management of ledgers is fundamental for preparing data for final *end of period* financial reporting (Stolowy & Ding (2019)).
3. *Financial Reporting Context* encompasses the preparation and presentation (via publication) of final financial statements. This context utilises data from the previous ledger context to generate these key financial documents. Traditionally, these include the *balance sheet*, *income statement* and *cash flow statement*, which collectively provide insights into the financial health and performance of a business (Stolowy & Ding (2019)). This context is essential for transparent and accurate external reporting and decision-making, aligning with the overall goals of the BFS.
4. *Accounting Transactions Context* is focused on *accounting rules* for the retrieval of economic data governed by *accounting cycle* process. It starts with the recording and management of a company's economic interactions. This context ensures that every completed financial event is accurately captured, reflecting the company's economic activity in real-time and facilitating subsequent accounting processes. This involves anonymization and transformation of such data into accounting variables, such as accounts and mapping these variables in accordance with accounting principles. It adapts traditional steps of the accounting cycle to the needs of the BFS. This adaptation involves streamlining relevant processes to include only those necessary for the BFS demonstration, ensuring the system's efficiency while accurately reporting the entity's financial health. This context underscores the sequential steps, starting from business transaction recording and leading all the way to the preparation of financial statements, adapted for blockchain integration. It also encompasses the double-entry accounting system principles, ensuring that each transaction is accurately represented

in accounting ledgers, maintaining the integrity accounting equation that governs all financial records (Stolowy & Ding (2019)).

5. *Accountant's Role and Responsibilities Context* highlights the role and responsibilities of hypothetical accountants within the BFS ecosystem. It includes this accountant's critical function in collecting, managing, interpreting and reporting financial data, ensuring compliance with accounting standards and contributing to the strategic decision-making process of any business. This context underscores the human element in the BFS architecture, bridging the gap between technology and traditional accounting practices. Furthermore, management of a repository for accounting transactions (accounting transaction register) within this context is crucial for verifiable maintenance of chronological records for referencing of all financial activities. Lastly, the role and responsibility of an accountant includes initiation and implementation of all accounting transactions.

Blockchain Sub-Domain

Blockchain focuses on the adaptation of blockchain technology for the single economic entity's financial statements. A blockchain utilisation context defines how blockchain technology is adapted by the BFS to record and report the financial activities of a single economic entity. It includes customisation of blockchain architectures to suit the entity's financial statement needs and to enable transaction implementation process.

1. *BFS Block Structure Context.* Defines variations in the type of data stored on the BFS filing history ledger. This context defines not transactions, but aggregated accounting data with some abstracted identifiers from underlying business and accounting transactions allowing for combination of confidentiality with transparency;
2. *The BFS and Ledger Context.* Defined as a chronological repository of state of business at a particular point in time.
3. *Smart Contracts Context.* Defines the rules generated to enable automation of business and accounting processes which can be uniquely tailored to each economic entity's needs.
4. *Wallets Context.* In the BFS, this handles wallets for members and for counterparties, defining access permissions to internal company data and the type of business activity performed by such wallet owners. It also manages cryptographic elements like hashes and electronic signatures. Components of this context enable standardised approach to management of distinct role-specific operations for each member within economic entity, embodying unique digital identity and operational authority of an individual user.
5. *Validator Context.* Manages cryptographic elements such as hashes, electronic signatures and generation and verification of cryptographic secrets, particularly for transaction implementation. This context contains processes for verifying various aspects of transactions without storing any data, aligning with the principles of statelessness and immutability inherent in blockchain technology. This class contains the default implementation for the methods to verify the integrity of transaction data. These methods are critical for ensuring that each transaction adheres to predefined rules and conditions before being processed further by the BFS.
6. *UTXO and Consensus Context.* Manages a UTXO generation mechanisms and component based on the internal accounting process within the BFS. In this Thesis, the UTXO is not a data model, but a funds verification process designed as a novel process relevant to accounting. A consensus mechanism is also tailored to the needs of the BFS artefact and is used in this project for funds availability verification between transacting counterparties.

Business Entity Sub-Domain

Business Entity. Encapsulates the entity itself, its internal operations, together with entity's transactional interactions with other BFS ecosystem participants. This bounded context is specific the use-case for the BFS artefact and it is centred around:

1. *Company Incorporation Context.* This deals with the foundational aspects of a business existence, including incorporation processes together with its application form and its data, submission of this documentation to the relevant authorities, attainment of an incorporation documentation package, which includes, amongst other things, the "Certificate of Incorporation", company registration number and a copy of the Genesis Block for BFS instantiation.
2. *Business-Specific Context.* Encompasses various types of business specific activity and procedures other than business transaction orchestration. Specific to the use-case for the BFS project, this context includes operationalisation of the lending facility - as a smart contract within BFS. This will enable the process of arranging a "Lending Agreement" between the APF and the BoE, followed by the drawdown requests to draw funds on this agreement for facilitation of necessary funds for QE implementation.
3. *Business Transaction Elements.* Contains all use-case specific and BFS generalised components necessary for generation and execution of business transactions by the APF::BFS. This context encompasses various types of business transactions the economic entity engages in, ensuring that they are recorded and processed in alignment with the entity's structure and objectives. It handles various business transactions all specific to the use-case of this project, such as:
   - ❖ Share issuance and sale at incorporation of the company (Delivery vs Payment (DvP) business transaction);
   - ❖ Financial loan inception with the lender (the BoE) and drawdown on that loan by the APF. The lent asset is Treasury Bills (One way Payment (OwP) business transaction);
   - ❖ Purchase of Commercial Paper financed by the Treasury Bills lent to the APF (Payment vs. Payment (DvD) business transaction).
4. *Business Transactions Register Context.* Encompass application and wallet local register to track ongoing transactional interactions of the economic entity, such as APF::BFS. It ensures that business-sensitive information is only accessible to relevant parties, maintaining confidentiality while also allowing for transaction traceability. Such an approach preserves privacy and supports compliance and transparency.
5. *Participants Registration Context.* Manages and facilitates relationships between the company, such as APF::BFS, and its internal members, related parties and external counterparties through a well-defined wallet registration and management system. Focus here is on members and counterparties relation to the APF::BFS, through recording and recognition of individuals interacting within the BFS domain.
6. *Participants Roles Context.* This includes appointment of BFS members and identification and registration of counterparties to this BFS. Here, the focus is on the appointment and management of members within the business entity, such as directors, shareholders, lenders, employees and registration of transactional counterparties.

4.3 Technological Approach

This section elaborates on the elements for the technological stack for the design and implementation of the BFS. In the earlier sections, the BFS ecosystem was characterised as a network of distinct economic entities, each of which maintained individual financial statements (BFS filing history), encapsulating summaries of that entity's unique economic engagements. Standardised management of such individual financial statements by the businesses ensures a decentralised yet coherent financial reporting structure. These entities (the BFS artefact) are interconnected with their counterparties (other BFSs) within this network *via* fundamental units of such engagements: the business transactions (smart contracts). These business transactions drive the overall economic dynamics of such an ecosystem by generating economic data to be consolidated into individual financial statements of relevant economic entities, utilising accounting processes automated by accounting transactions (smart contracts).

To encapsulate this complex relationship and interactions of such an ecosystem into the design and implementation of the BFS artefact, we innovatively integrate two distinct blockchain frameworks, each contributing uniquely to the artefact's functionality. This dual blockchain framework is central to understanding how the BFS operates as a proof of concept (PoC) Java prototype, functioning within a complex multi-ledger ecosystem of diverse business entities. Furthermore, this technological approach to the design and implementation of the BFS serves as a treatment to the issue of verifiable but privacy preserving techniques for secure sharing of sensitive financial information.

4.3.1 Dual Blockchain Framework Integration

First, the traditional "Bitcoin" (Nakamoto (2008)) blockchain framework is employed to design and implement the data structure utilised to store financial statements of an entity - BFS "Filing History" (see Fig. 4.3). This decision is motivated by the guarantee of such a data structure to maintain a tamper evident and cryptographically linked chronology of events. The BFS system is designed to handle the transformation of raw economic data into a format suitable for financial reporting:

- ❖ Economic data generated from completed business transactions is transformed into accounting data, anonymised, classified, aggregated and then packaged into financial statement reports.
- ❖ These reports are stored in a traditional chain of cryptographically-linked blocks within a permissioned blockchain data structure.
- ❖ This data structure forms the filing history of the entity, owned and distributed among members of such entity such as business owners, shareholders, employees and so on.
- ❖ Additionally, access permissions to this ledger can be modified and extended to qualified external parties such as other controlling entities or government bodies for regulatory or tax purposes.

Serving as a filing history, this blockchain ledger is utilised as a tamper evident financial reporting tool for documenting financial health and trajectory of the business. However, in the traditional Bitcoin blockchain architecture, every peer participating in the network receives and processes every transaction ever conducted. Despite Bitcoin's architecture being known for ensuring participant anonymity, its broadcast communication method raises ongoing concerns regarding the privacy of business-sensitive transactional data.

To address this critical challenge of ensuring transactional privacy and to provide tractability for the cases of compliance validation and transparency, the research in this Thesis incorporates components of the "Corda" platform architecture (R3 (2023)). These components are:

- ❖ Corda *Flow Framework:* (R3 (2023)). BFS implies point-to-point messaging (see Fig. 4.4), governed by the Corda network (R3 (2023)). This approach ensures that sensitive transactional data is only transmitted between the parties relevant to the transaction, thereby remaining confidential between them and safeguarding business-sensitive information.
- ❖ Corda *Transaction Volts:* (R3 (2023)). One important component of the BFS application is adaptation of the Transaction Volt architecture of Corda to implement a Business Transaction Register; an off-ledger repository that stores all business transactions of an entity (see Fig. 4.4). These transactions reposted in their "raw state" - untransformed for accounting. That way it is possible to preserve transactional privacy and each transaction maintains a traceable link (transaction hash, together with all public keys of the counterparties) to support verification or investigative needs.

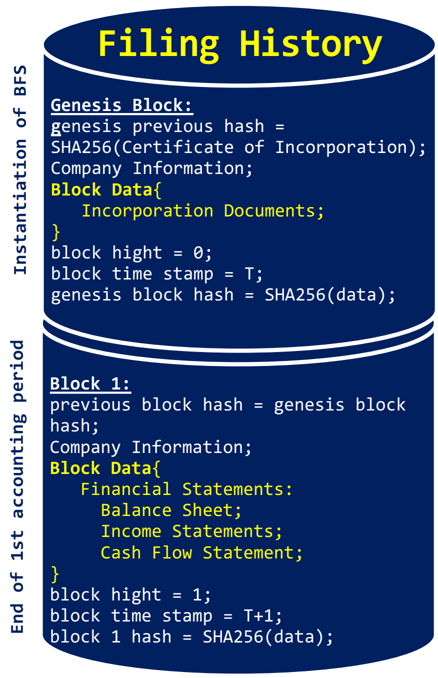


Figure 4.3: BFS "Filing History" indicative data structure.

In the BFS block structure, illustrated in Fig. 4.3, following terms are essential for understanding how the it operates and how data is securely stored and managed within BFS system: block, genesis block, genesis previous hash, company information data instance, block data, block height, time stamp, the final hash, and the first hash of the next block:

- A *block* is a container that holds a batch of transactions or records. In the BFS system, a block contains accounting-related data, such as financial statements. Each block is cryptographically linked to the previous block in the chain, called *BFS filing history*, ensuring overall data integrity and security. Blocks are organised chronologically, forming a continuous ledger of all accounting history of an entity.
- The *genesis block* is the first block in the blockchain. It serves as the foundation of the entire blockchain ledger, from which all subsequent blocks are linked. In the BFS architecture, the genesis block contains company incorporation documentation, such as the company's incorporation certificate. Unlike other blocks, the genesis block does not reference a previous block since it is the first one.
- The *company information data instance* refers to a specific set of unique data attributes, representing a business entity's identification details, such as its registration information, address, company incorporation number and date and so on.
- *Block data* refers to the actual informational content stored within a block. In the BFS, this data includes financial statements, incorporation documentation, or other relevant financial information, as per regulatory reporting requirements.
- *Block height* represents position of a block in the blockchain, starting from the genesis block, which has a height of 0. Each subsequent block added to the chain increases the block height by 1.
- The *block timestamp* is a record of when a block was added to the blockchain. This ensures that all chronological data sequencing, stored within BFS filing history is accurately time-stamped, providing an audit trail.
- The *block hash* is a cryptographic signature generated by running the block's data through a hashing algorithm (such as SHA-256). This hash acts as a unique identifier for the block and is used to link it to the next block in the chain. Any modification to the block's data would result in a completely different hash, making it easy to detect tampering.
- The *next block* refers to the block that follows a particular block in the blockchain. Each block (except the last block in the chain) points to its previous block by including the *previous block's hash*. This linkage creates the chain of blocks that makes up the blockchain.

4.3.2 PoC Implementation Tool

The implementation of the technological PoC prototype using Java serves as a tangible representation of the BFS architecture and its operational capability, showcasing the system's functionality

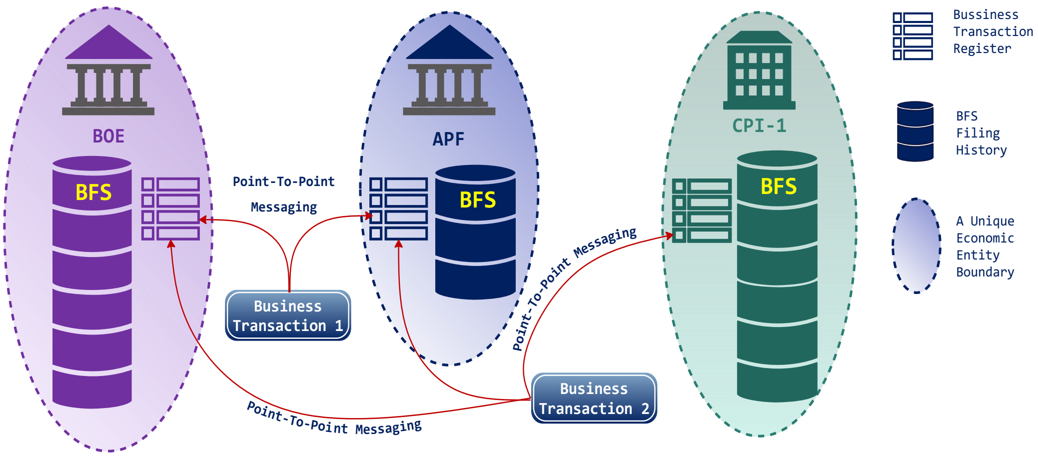


Figure 4.4: BFS ecosystem participants and their transactional interactions using point-to-point communication and transactional volts.
The overall BFS ecosystem consist of 3 distinct economic entities contained in dashed ovals: the BoE - as the purple coloured entity, the APF as the blue coloured entity and the CPI - 1 as the green coloured entity. Each entity owns distinct BFS filing history ledger that contains and records accounting activity of that entity, together with an off-ledger application local repository - Business Transaction Register - that records all ongoing business transactions of its owning entity. The interactions between these entities is implemented via business transactions, depicted as blue rectangles, with the red arrows indicating to in which Business Transaction Registers these copies of these business transactions are persisted.

in a controlled environment. This Java BFS blends the secure and immutable nature of blockchain with the flexibility and privacy requirements of financial transactions. The approach offers a significant advancement in the field of financial technology and paves the way for further development and refinement of the BFS system.

Chapter 5

The Architecture of BFS

Moving from the broad conceptual overview in Chapter 4, this chapter presents a comprehensive taxonomy of key BFS architectural components. This journey into the architecture is essential for understanding how blockchain technology can enhance financial accounting processes. By detailing the architectural pieces, their organisation, and interactions, this chapter prepares the groundwork for Chapters 6 and 7, where the exploration and demonstration of the BFS's implementation and application is presented.

The previous Chapter 4 outlined the BFS architecture's scope and structure. The design logic is guided by by combination of principles of DDD with empirical methods, creating a framework that facilitates design of an architecture capable of addressing challenges of misconduct in financial reporting, and provides solutions for liquidity management. The DDD architectural pattern, known for its focus on deep understanding the solution's domain, is critical in guiding the design of BFS though modular and iterative evolution of its components. This approach is essential in constructing a system capable of managing complex business transactions and accounting processes with assured authenticity and integrity.

Following this, current Chapter 5 presents the architecture of the BFS, focusing on the sub-domains of financial accounting, blockchain technology, and business entity interactions. Each of these sub-domain is analysed within its context, highlighting the architecture's nuances, including key actors, elements, and their interactions. This presentation aims to showcase the BFS's design alignment with the operational needs of integrating blockchain into financial accounting within requirements of the APF use-case. As this chapter concludes, BFS architecture is linked with communication of its practical implementation within the context of the APF use-case.

The narrative then leads into Chapters 6 and 7, where the theoretical and architectural foundations laid here will be demonstrated through practical application. A proof-of-concept will showcase the BFS system's functionality and describe its effectiveness for real-world setting, providing insights into the potential of BFS. This progression from theoretical foundation to practical application highlights the importance of the DDD methodology in creating a domain-focused, coherent, and effective architecture. This architecture promises to advance financial accounting and liquidity management into a new era of efficiency and transparency.

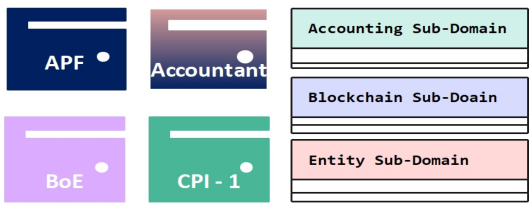


Figure 5.1: Illustrative colours for key BFS roles and of Java implementation of class and hierarchy diagrams.

For illustration purposes, All BFS project primary roles, that are utilised to execute demonstrated functionality, transactions and processes described in this report, when illustrated in the diagrams are colour-coded as illustrated in wallets on the left of Fig. 5.1. Additionally, all BFS class and hierarchy diagrams in this Thesis are colour-coded to align with the specific BFS sub-domains to which they belong (see Fig 5.1). Furthermore, in this report, class structures in the BFS are represented using class UML diagrams and mind maps. Class UML diagrams provide a clear visualisation of the relationships, attributes within the classes, and methods within interfaces, illustrating their structures and interactions. Mind maps are employed to demonstrate the values and some of the hierarchical arrangements of the classes, offering an intuitive representation of the connections and dependencies within the system. Together, these diagram types effectively capture the architectural and conceptual organisation of the BFS framework.

Lastly, the primary focus of the BFS framework was not on the performance optimisation of specific data structures but rather on demonstrating the feasibility and functionality of the framework within the context of its conceptual design and use-case requirements. The choice of data structures was guided by their ability to fulfil functional requirements and maintain logical coherence within the design, rather than performance considerations. As the primary aim of this research was to explore the integration of blockchain technology within financial reporting and accounting practices, performance benchmarking was outside the scope of this study. Future iterations or further research could investigate the performance aspects of these data structures, but at this stage, it was not a consideration. This decision aligns with the purpose of validating the conceptual framework rather than optimising implementation specifics.

5.1 Introduction

Wieringa (2014) describes a system as an assembly of elements that engage with each other to constitute a whole. It is further noted that these components themselves might be systems composed of lower-level components or, alternatively, can be integrated into more complex systems.

Current research goal is to predict and explain the overall system behaviour from the knowledge of components of its architecture (Wieringa (2014)), which is understood through these components and the interactions between them. Accomplishment of this goal starts from identification of system's behaviour though narrative description of the step-by-step process of the APF use-case, utilised for BFS project, illustrated in Fig. 2.3. This supporting narrative step-by-step description enables definition of primary actors (as different players in the project's scenario), top-level set of fundamental elements, together with use-case related actions (or behaviours) that together define and guide composition of the architectural structure of the BFS artefact.

The organisation of these BFS components and orchestration of their behaviours are vital for overall ecosystem to perform related processes and respond to events. The inter-dependencies and interactions among such components are not random but are strategically guided by the business logic embedded within the bounded contexts of the BFS ecosystem. This logic ensures that the interactions align with the system's objectives, operational principles, and business needs of an economic entity.

As it was outlined earlier (4.2.2), mapping the definition of the architectural components within the boundaries of identified sub-domains and their constituent bounded contexts ensures that the interactions and dependencies between these components are guided by the mechanisms of the business logic of these bounded contexts, so enabling accurate reflection of the mechanisms necessary to reflect, in turn, the intricacies of financial accounting and respond to business needs of its users.

In Figure 5.2, an illustration of the key BFS components such as actors and fundamental elements are presented, together with key relationships between them, illustrated as red arrows. These are categorised by their taxonomic relationships, which help in classifying and understanding the roles and functions within the BFS system. This help in classifying and understanding the

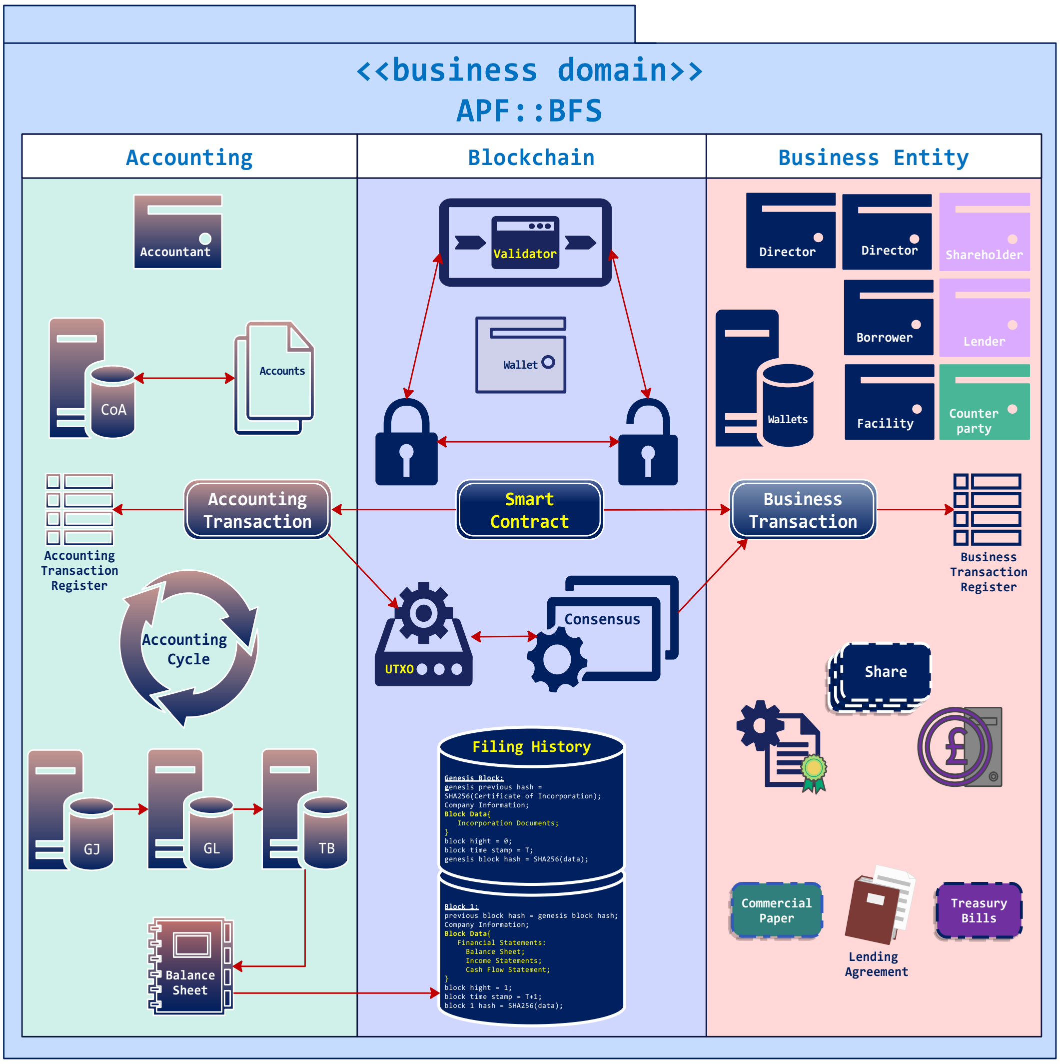


Figure 5.2: APF::BFS architectural components and key relationships.

In the BFS system, a *business entity* refers to a legally recognised economic unit created by one or more individuals to carry out the functions of a business. It is an organisation, such as a company or financial institution, that participates in the BFS ecosystem. Each business entity has its own unique set of business processes, rules, and requirements, which are encapsulated within the *business entity sub-domain*. BFS entities interact with one another through transactional smart contracts in the *blockchain sub-domain* and are responsible for maintaining their own financial records and reporting their financial activities through the *accounting sub-domain* in the BFS system. This modular relationship flow within the BFS architecture allows for efficient treatment of each business as an individual accounting unit, with financial records maintained utilising a blockchain data structure - the BFS filing history. Red arrows represent key relationships that connect primary flow of data between relevant components. More detailed definitions of each of the component and their corresponding relationships is described in the following sections of this thesis.

roles and functions of each component within the system, and through cardinality associations, which elaborate on how components are linked, detailing the extent and nature of relationships between different components (Wieringa (2014)). These associations are important for interpreting the dynamics of the BFS ecosystem and understanding how changes in one component might affect others. Such associations are important in providing interpretations of the dynamics of the BFS ecosystem and in understanding how changes in one component might affect others. These analytical frameworks help us comprehend the dynamic nature of the BFS architecture, allowing us to see how changes in one component may influence others.

In the following sections of this Chapter, the description of the BFS components, such as actors, fundamental elements and the mechanisms are presented solely from the perspective of the Asset Purchase Facility (APF), which functions as the primary business entity engaging in transactions with its counterparties, namely the Bank of England (BoE) and the hypothetical economic entity Commercial Paper Issuer - 1 (CPI-1).

5.2 Actors

Before describing these actors, the overarching definition for the business entity within BFS ecosystem is necessary. In the BFS as a business entity refers to a legally recognised entity or an economic unit created by one or more individuals to carry out the functions of a business. It is treated as a distinct accounting entity for which financial records are maintained utilising blockchain data structure. This entity can be further decomposed into acting members such as directors, employees shareholders and so on, each of which can have relationships and responsibility with this business entity dictated by their roles. In this project, based on the scenario of the us-case, the ecosystem consists of three distinct business entities interacting with one another through business transactions (see Fig. 4.4).

Within the architecture of the BFS ecosystem, there are several key use-case based participants that play distinct roles, interacting with and through the system to achieve specific accounting and transactional outcomes. These actors, central to this exploration, are mapped meticulously across various sub-domains and bounded contexts, revealing the intricate design and functionality of the BFS architecture.

The first and the primary one is the **Asset Purchase Facility (APF)** (the whole top-level business domain of the (represented as green wallet in Fig. 5.2)). It stands at the core of this research project. It is the main entity through which the architecture of the BFS artefact is explored. All transactional and accounting processes within the BFS are designed, implemented, and demonstrated with the APF as the focal point. This central role of the APF ties it directly to several sub-domains and bounded contexts within the BFS architecture, including:

❖ *Business Entity Sub-Domain* (see most right section of Figure 5.2), within this sub-domain, the APF is integrated into every context, starting from the *company incorporation* (marking the beginning of its BFS journey), its *business specific activities*, and so on. Together, these contexts encapsulate the operational dynamics and transactional interactions of the APF, underscoring its central role in the BFS ecosystem.

❖ *Financial Accounting Sub-Domain* (see most left section of Figure 5.2), where all APF's activities are recorded and processed within this sub-domain, particularly in contexts like *accounting transactions*, for detailing the flow and recording of financial activities, and *financial reporting* for the generation and publication of financial statements. These areas reflect the APF's economic interactions and their subsequent representation in financial statements.

❖ *Blockchain Sub-Domain* (see central section of Figure 5.2), within this sub-domain, the APF is also integrated into every context, where the APF's involvement extends deeply into the BFS as the blockchain framework, showcasing how traditional financial entities can leverage blockchain technology for enhanced operational efficiency and transparency. Smart Contracts Context: The APF utilises smart contracts to automate and secure its financial transactions and interactions with counterparts such as the BoE and CPI-1. These contracts codify the terms of transactions, ensuring that they are executed precisely as agreed upon, thereby reducing the risk of disputes and enhancing transactional integrity. Within *wallets context* the APF manages digital wallets for its members and various stakeholders, including directors, shareholders, and counterparties, such as market participant CPI-1. These wallets are essential for the secure storage and transfer of digital assets, serving as a critical component of the blockchain's infrastructure within the BFS. Within *validator Context* the APF plays a role in validating transactions ensuring that all adhere to the predefined rules and conditions, maintaining the integrity and reliability of the resulting financial records. Also, within *BFS block structure context*, the APF acts in defining the structure of blocks within its BFS ledger. This involves approving the type of data stored on the blockchain, how it is organised, and ensuring that the blockchain ledger accurately reflects the APF's financial transactions and accounting records.

Next is the **Bank of England (BoE)** (represented as purple wallets in Fig. 5.2), acting as the public money provider and the counterparty to the APF in two main capacities. It plays dual roles: as a shareholder acquiring shares during first business transactions, and as a lender, providing necessary funding for QE implementations carried out by the APF. This multifaceted interaction showcases the BFS's capacity to handle complex financial relationships. Such dual role places the BoE within critical intersections of the BFS architecture and is mapped to several bounded contexts:

❖ *Blockchain Sub-Domain*, in the smart contracts context within the blockchain sub-domain, the automation of contractual obligations of the "lending agreement", established between the APF and the BoE, dictates the terms of the second business transaction, which includes Treasury Bills lending the BoE to the APF.

❖ *Business Entity Sub-Domain*, specifically as a transacting counterparty to APF, and in relevance to the various roles it plays as this counterparty, captured within *participants roles contexts* - the shareholder and the lender. This BoE's interactions with the APF are detailed further in this report, highlighting its influence on liquidity provision and financial support.

❖ *Financial Accounting Sub-Domain*, within *account and its hierarchy context*, the BoE's role as a shareholder is captured within a specific financial accounts of the APF, highlighting its investment in the APF and its impact on financial structures.

The third actor within BFS is **Commercial Paper Issuer - 1 (CPI - 1)** (represented as green wallet in Fig. 5.2) — a hypothetical market participant that seeks financial support from the APF, acting as another vital counterparty. Its inclusion in the BFS architecture is illustrative, showing how the system can support real-world financial interactions. The role of the CPI-1 is mapped to the following contexts:

❖ *Business Entity Sub-Domain*, Within the *business transactions* context, CPI-1's engagement in commercial paper transactions with the APF demonstrates the BFS's ability to facilitate financial support and liquidity management.

❖ *Blockchain Sub-Domain*, The interaction between CPI-1 and the APF, mediated through *smart contracts*, involving the use of blockchain component for secure, transparent, and efficient transactions. Also, within *wallets context* the digital identities and transactions of CPI-1 are managed, showcasing the technological prowess of the BFS in handling diverse market participants.

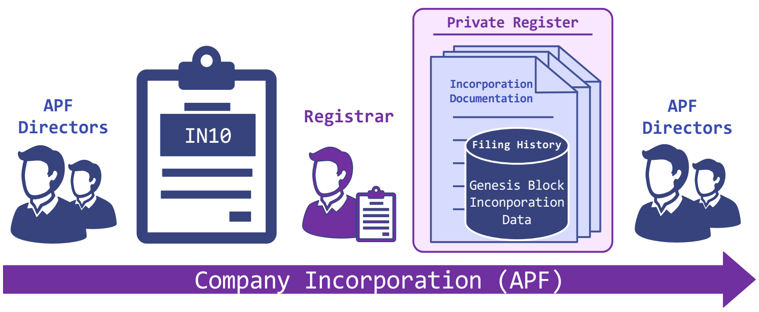


Figure 5.3: Company Incorporation Process Flow.
The arrow direction of the figure indicates the sequential direction of the company incorporation steps implemented in the BFS research. Registrar related steps and elements are illustrated as purple, and APF is depicted as blue.

The fourth hypothetical actor is the **Registrar**, as the company incorporation officer within "Companies House" UK (Companies House (2024*c*), Companies House (2024*a*)). The Registrar is responsible for formal incorporation of companies like the APF and, in the BFS architecture, for generating the Genesis Block. Company directors submit documentation required for the incorporation (form IN10 (*Form IN10. Register a private or public company. Application to register a company* (2024))) to this officer. The copy of this Genesis Block is securely stored in the private database of the Registrar (e.g., in "Companies House" (Companies House (2024*d*))) and is securely communicated to the directors of this business, together with the documentation (see Fig. 5.3). While the Registrar's function is critical at the inception phase of the APF, for the purpose of this project, its operations are assumed rather than detailed, focusing instead on the accounting processes. This actor is associated with:

❖ *Business Entity Sub-Domain*, specifically within the Company Incorporation context, where the Registrar's actions enable the formal inception of the APF, marking the beginning of its operational life of the APF within the BFS ecosystem, and laying the foundation for subsequent financial and transactional activities.

Lastly, incorporating the role of an **Accountant** into the BFS architecture provides that critical bridge between the traditional financial accounting practices and the innovative application of blockchain technology within the APF::BFS. The Accountant plays an important role in overseeing and managing the integrity of financial transactions and reporting, ensuring that the BFS adheres to established accounting principles and practices. The Accountant is responsible for the accurate recording, verification, and reporting of financial transactions within the APF::BFS. This role encompasses application of accounting principles to transactional data, and the preparation of financial statements that accurately reflect the APF's financial status. By mapping the Accountant across relevant contexts, the BFS architecture ensures that the system not only leverages the benefits of blockchain technology but also adheres to the rigorous standards of financial reporting and accountability. This role can be mapped to the contexts of the following sub-domains:

❖ *Financial Accounting Sub-Domain*, within it, the Accountant is integrated into every context, including: *account and hierarchy context* where the Accountant ensures that financial accounts are accurately created, classified and maintained within the BFS, reflecting the APF's economic activities and financial structure correctly. Net, the *accounting journals and ledgers context*, where the Accountant is responsible for the management of accounting ledgers, ensures organisation and governance of financial accounts, adhering to the principles of double-entry accounting. This links next to the *financial reporting context*, where the Accountant compiles and presents financial statements based on data from accounting ledgers, facilitating transparent and accurate external reporting. Also. within *accounting transactions context*, the Accountant Oversees accounting cycle process, captures and records each economic event, ensuring that it is accurately reflected within the APF's financial records.

❖ Within *Blockchain Sub-Domain*, in the *smart contracts context* Accountant plays a role in defining the rules and conditions of accounting smart contracts to ensure that they accurately represent the financial transactions and obligations of the APF.

❖ Lastly, within *Business Entity Sub-Domain*, the *business transaction elements context* encapsulates Accountants role in verification of the economic substance of business transactions, ensuring they are recorded in accordance with legal and regulatory requirements. Lastly, *participants roles context* includes Accountant's involvement as the role, played within economic entity.

These actors, through their defined roles and interactions, shape the architectural and operational landscape of the BFS artefact. Their activities, grounded in the system's sub-domains and bounded contexts, not only demonstrate the practical application of the BFS but also underscore the system's potential to facilitate and improve financial accounting and reporting processes in a blockchain environment.

5.3 Fundamental Elements

Next, this subchapter will focus to the fundamental elements that constitute the backbone of the BFS architecture. As a complex and innovative system, is comprised of numerous elements, each serving a distinct function within the overarching framework. These elements are the building blocks that facilitate the BFS's ability to streamline and secure financial transactions and accounting processes through blockchain technology.

Given the breadth and depth of the BFS architecture illustrated in Fig. 5.2, and considering the substantial number of fundamental elements involved, exploration of these elements will be organised within the context of relevant sub-domains. This approach allows for a more structured and comprehensible presentation, ensuring that each element is examined within its appropriate operational and functional context. By aligning the discussion of fundamental elements with specific sub-domains, this Thesis can better illustrate how these components interrelate and contribute to the BFS's objectives.

In Fig. 5.2 the BFS architecture is dissected into several key sub-domains, each encapsulating a specific aspect of the system's functionality. These sub-domains include Financial Accounting, Blockchain, and Business Entity, among others. Within each sub-domain, this research finds a unique set of fundamental elements that play crucial roles in fulfilling the BFS's operational requirements and achieving its strategic goals. This structured presentation not only highlights the diversity of elements involved but also showcases how they collectively form a cohesive and robust system capable of revolutionizing financial accounting and reporting practices.

5.3.1 Elements of Financial Accounting Sub-Domain

Here, elements related to the traditional accounting cycle, financial reporting, and compliance with accounting standards are paramount. Elements such as accounting ledgers, journals, and financial statements are adapted to the BFS framework, ensuring accuracy and transparency in financial reporting.

❖ *Account and its Hierarchy Context*

Account ID. It is a hierarchical account number (see Fig. 5.4) which enables identification of a unique account in the accounting ledgers, such as Chart of Accounts (COA), General Journal (J/L), General Ledger (G/L), Trial Balance (T/B) and in the final Financial Statements. In addition to the account name, each Account is assigned and is identified primarily by the this Account ID number and these numbers are typically organised into logical groups, e.g.:

- 10000 - 19999: asset `Account` numbers range in G/L;
- 20000 - 29999: liability `Account` numbers range in G/L;
- 30000 - 39999: equity `Account` numbers range in G/L;
- 40000 - 49999: income `Account` numbers range in G/L;
- 50000 - 59999: expenses `Account` numbers range in G/L.

Account Category. Financial accounting reflects the economic activity of a business. Its users must be able to compare the financial statements of any entity in order to identify trends in financial position and performance Stolowy & Ding (2019). To enable such comparability, financial accounting employs standardisation of representation of financial information, by structuring consolidation of accounting information into key classes of accounts, such as Stolowy & Ding (2019).:

1. assets (e.g., cash, accounts receivables, fixed assets);
2. liabilities (e.g., accounts payable, long term liabilities);
3. equity (e.g., retained earnings, capital);
4. income and expenses.

These classes of accounts can be further subdivided into still standardised, but more business specific categories of accounts Stolowy & Ding (2019). To build on these principles, the architecture of the BFS reflects this hierarchical consolidation rules for the accounts. These rules will govern classification of various accounts balances on their categories and final consolidation of the ending balances into financial reports.

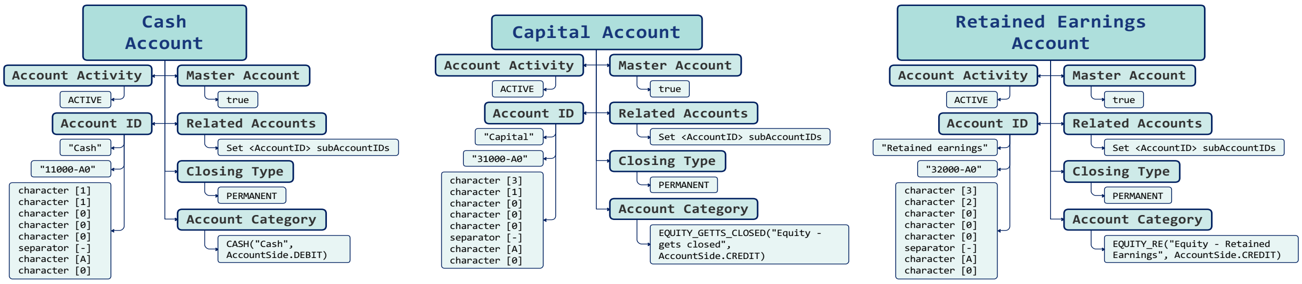


Figure 5.4: Examples of default Accounts for BFS use case

This figure demonstrates three initial Accounts instantiated at the APF::BFS instantiation. For each of the Accounts, their instants variables are assigned use-case relevant values illustrated in the faded blue rectangles mapped to its light blue rectangle types. The turquoise colour represents relationship to the accounting sub-domain.

Account. A financial record's account is a fundamental concept of the financial accounting framework, and is employed to systematically record and report on the evolution of the economic variables, primarily expressed in monetary units Stolowy & Ding (2019). It serves as a structured approach to classify and summarize financial information resulting from business transactions. Each account signifies uniform changes in liquidity associated with a specific category or element of a business's operations. This methodical approach to accounting is pivotal in ensuring the precision, uniformity and transparency of financial reporting. In accordance with the double-entry accounting framework, every completed business transaction is assigned to at least two `Accounts` - with a debit and a credit entry.

Building on these principles, in the BFS project (see Fig. 5.4) as an illustrative example of the hypothetical accounts implemented in the BFS artefact), the Account is designed as a generalised identifier for homogeneous units of monetary value. The moderate values extracted from the completed business transactions, could be aggregated either into unique and desecrate categories (a child `Account`), or accumulated into a master-category (or a class) of homogeneous `Accounts` (a master `Account`), to communicate consolidated 'ending balance' of its constituencies.

Account Activity. Components that activate or deactivate accounts that are relevant and used in the current reporting period (see Fig. 5.4).

Related Accounts. Concept of 'related accounts' (see Fig. 5.4) can be best understood through the lens of the hierarchical structure of relationships between parent (primary or master) accounts and child (sub or subsidiary) accounts in the financial accounting. This structure is used to organise and categorise financial information in a systematic way to ensure consistency of aggregation of multilevel relationships in the financial statements. This approach to consolidation of financial variables enables both: the overview of the high-level financial position, and the detailed breakdown of specific categories.

❖ *Accounting Journals and Ledgers Context* Accounting Ledgers. Local data repositories for inter-accounting period recording of accounting data in chronological order Stolowy & Ding (2019).

Chart of Accounts (COA) - an application local repository for pre-established, logically organised list of all recognised and authorised accounts, used in recording of all transactions the business can engage in Stolowy & Ding (2019). Accounts in COA are assigned their unique "account ID", and are hierarchically organised based on that ID within the COA data structure (see simplified example of COA in Table 5.1 below).

#	Account ID	Account Name	Account Class	Account Activity
1	110000-A0	Cash	Assets	active
2	310000-00	Capital	Equity	active
3	320000-00	Retained earnings	Equity	active
...	...	...	...	...

Table 5.1: Simplified example for chart of accounts (COA).

General Journal (GJ). A data repository for a day-to-day chronological registration of accounting information of all ongoing allowable transactions Stolowy & Ding (2019). It is a first point of entry for the data from the completed BTr, transformed though ATr.

Tiem Stamp	Transaction ID	Accounts	Debit	Credit
30/01/2009	Hash(transaction)	Cash (110000-A0)	100.00	0
30/01/2009	Hash(transaction)	Capital (310000-00)	0	100.00
...	...	...	...	...
...	...	...	...	...

Table 5.2: Simplified example for General Journal.

General Ledger (GL). A data repository for a set of homogeneously defined accounts; a classification category which contains cryptographically-linked chronological details of the relevant side of the Journal entries and provides an ending (aggregated) account balance Stolowy & Ding (2019).

Cash (110000-A0)			
Time Stamp	Debit	Credit	Current Balance
	Beginning Balance		0
30/01/2009	100.00	0	100.00
...	...	...	...
	Ending Balance		100.00

Table 5.3: Simplified example for General Ledger Account for "Cash".

Capital (310000-00)			
Time Stamp	Debit	Credit	Current Balance
	Beginning Balance		0
30/01/2009	0	100.00	100.00
...	...	...	...
	Ending Balance		100.00

Table 5.4: Simplified example for General Ledger Account for "Capital".

Trial Balance (T/B). A list of the debit and credit entries and footings for each ending account balance from the GL.

❖ *Financial Reporting Context*

Financial Statements. These are outputs of the accounting process forming a reporting package through a Balance Sheet (BS), Income Statement (I/S) etc Stolowy & Ding (2019). These contain anonymised and aggregated data on the business transactions over a selected accounting period published in the block of the chronological filing history of

Retained earnings (320000-00)			
Time Stamp	Debit	Credit	Current Balance
	Beginning Balance		0
30/01/2009	0	0	0
...	...	...	...
	Ending Balance		0

Table 5.5: Simplified example for General Ledger Account for “Retained Earnings”.

Account ID	Account Name	Beginning Trial Balance	Trial Balance Entry	Ending Trial Balance	...	Trial Balance Entry	Ending Trial Balance
110000-A0	Cash	0	100.00	...	...	...	...
310000-00	Capital	0	(100.00)	...	...	...	...
320000-00	Retained earnings	0	0	...	...	...	...
...	...	...	...	...	...	...	....

Table 5.6: Simplified example for Running Trial Balance.

BFS. In this example, I generate only BS built according to the double-entry accounting practices of Stolowy & Ding (2019) and represents the APF’s financial position after initial investment by its shareholder; this is through the issue of shares by the APF and the purchase of these by the BOE at incorporation.

❖ *Accounting Transactions Context*

Accounting Transaction. A set of pre-defined Smart Contract rules that govern transformation of transactional data into accounting data, classification of this accounting data into corresponding accounts and aggregation of the values in this account into corresponding accounting journals (ledgers). Each individual accounting transaction has two sides (debit and credit) which are always balanced Stolowy & Ding (2019).

5.3.2 Elements of Blockchain Sub-Domain

This sub-domain focuses on elements that leverage blockchain technology for the BFS. It includes smart contracts, digital wallets, blockchain ledgers, and consensus mechanisms, which collectively ensure the security, immutability, and efficiency of the BFS’s transactional processes.

❖ *BFS Block Structure Context.*

Block. A container for the final publishable accounting data, such as Financial Statements. In this architecture, a block can contain a number of data types, replicating the filing requirements of the APF documentation published by Companies House Companies House (2023*a*). This example illustrates two data types stored in separate blocks (see Fig. 4.3).

Block Data Types. Unlike the standard Bitcoin-style blockchain, where each block stores homogeneous data types such as transactions, the blocks in the BFS filing history chain are designed to store a variety of standardised data types. These data types are determined by the kinds of documentation that economic entities are required to submit to Companies House. In designing BFS artefact, the simplified representation of the distinct data types, published in the blocks of the chain are (see Fig. 4.3):

- incorporation data block type,
- accounting data block type.

❖ *The BFS and Ledger Context.*

Blockchain “Filing History”. A cryptographically linked chronological chain of blocks. This example presents a blockchain consisting of two blocks (incorporation block (height 0) and accounting block (height 1). This is illustrated in Fig. 4.3.

❖ *Smart Contracts Context.*

Smart Contracts. As another fundamental element of the blockchain sub-domain, smart contracts are bespoke, digitally programmed protocols are designed to facilitate, verify, and enforce contractual or transactional agreements automatically when predefined conditions are met, eliminating the need for intermediaries. Within BFS framework these smart contracts are key drivers of automation, ensuring the seamless execution of business and accounting processes while maintaining compliance with predefined rules and standards. The BFS ecosystem incorporates smart contracts to automate the execution of the following operations:

- Accounting transactions - where the BFS's accounting transaction smart contract transforms economic events into standardised accounting entries, applying principles of double-entry bookkeeping. These contracts ensure that all economic outcomes from business transactions are accurately reflected in accounting ledgers and in the final financial statements. Thesis provides detailed Java implementations of accounting transaction smart contract in the Section 6.1.4, where it is demonstrated through combination of Java classes `AllowableAccountingTransaction` and `AccountingTransaction`.
- Business transactions - where BFS's business transaction smart contracts are designed to autonomously handle complex economic interactions, to facilitate use-case specific business transactions such as asset transfers and payments. They ensure that terms and conditions are met before execution and final settlement of value and exchange of assets, enhancing the reliability and efficiency of business continuity. Thesis provides detailed Java implementations of business transaction smart contract in the Section 6.3.3, where one example of such is represented by the Java class `InitialShareSettlementBusinessTransaction_DvP`. This class implements business-specific functionality of the APF use-case, where it extends its generalised BFS superclass `AbstractBusinessTransactionData` and the interface `BusinessTransactionDataInterface`. It additionally enables automation of Delivery versus Payment (DvP) principles.
- Lending agreement - where a use-case specific smart contracts is designed and demonstrated to facilitate loan agreement, such as one between the APF and the BoE, automating drawdown requests and transfer of the Treasury Bills into the ownership of APF. Thesis provides detailed Java implementations of lending agreement smart contract in the Section 6.3.2, by presenting and describing `LoanAgreement` class. This smart contract automates loan terms, including funding requests and repayments, facilitating efficient and transparent financial management.
- Company incorporation and BFS instantiation - where a BFS smart contract is designed and described, to enable automation of the registration and establishment of economic entity, such as APF. This includes generating incorporation documentation, creating the Genesis Block for the BFS, and setting up initial share issuance. This ensures a secure and structured approach to entity creation. Thesis provides descriptive Java implementations of company incorporation smart contract in the Section 6.3.1.

Architectural relevance of the BFS smart contracts embodies principles of modularity and domain-driven design. They serve as reusable components tailored to address specific operational requirements while maintaining compatibility with the BFS framework. By automating essential processes such as financial reporting, transaction execution, and organisational setup, smart contracts demonstrate feasibility of integrating BFS to support real-world applications, showcasing how blockchain technology can redefine financial accounting and business operations.

❖ *Wallets Context.*

Wallet - is a superclass to all wallet types. This wallet stores keys and records of all transactions that send and receive value from those keys. Using these keys, the wallet is able to create new transactions of accept transactions on behalf of the economic entity it is member of. Wallets that extend this superclass are differentiated by the additional functionality, designed to accommodate complex and diverse business needs of BFS use-case.

❖ *Validator Context.*

Cryptography. Adaptation of cryptography within BFS is utilised for securing digital information, communications and transactions. It involves various cryptographic techniques to ensure the integrity, confidentiality and authenticity of data. In the BFS artefact cryptography is utilised to:

- Generation and verification of hashes, where SHA256 hash function is used to convert input data of any size into a fixed-size string of characters. Each unique piece of data produces a unique hash and any alteration in the data results in a completely different hash. This property is crucial for maintaining the integrity of data in blocks in a blockchain, as each block's hash is dependent on the previous block's hash, creating a secure chain.
- Validation of data integrity in transit, where the SHA256 cryptographic protocol ensures that the data being transmitted remains secure and unaltered. This includes mechanisms to protect against unauthorised data modification, interception and other forms of cyber attacks.
- Signature generation and verification, where digital signatures are used to authenticate the identity of transaction participants and ensure transactional data integrity. A private key generates a unique signature for each transaction, which can be verified by others in the network using the corresponding public key. This process ensures that transactions are secure and have not been tampered with.
- Tokenization process involves converting sensitive data into non-sensitive data tokens that can be shared within the blockchain ecosystem. Tokenization helps to maintain data privacy while allowing transactions to be processed and verified on the network. The BFS project uses tokenisation to undertake transfer of ownership of financial assets, relevant to the use-case. The assets that are tokenized in this project are: *Shares*, *Share Certificates*, *Commercial Paper*, *Treasury Bills.*

❖ *UTXO and Consensus Context.*

Consensus facilitates funds availability verification incorporating accounting in the concussions protocol.

UTXO - in BFS UTXO is designed as a process, not a data model. It is utilised by the funds verification consensus designed as a novel process relevant to accounting.

5.3.3 Elements of Business Entity Sub-Domain

Elements within this sub-domain address the operational and structural aspects of business entities engaging with the BFS. This includes incorporation documents, business transaction records, and participant roles, which are essential for establishing and maintaining the legal and operational framework of entities like the APF.

❖ *Company Incorporation Context.*

Form IN10 . An application form to register a private or public company in the UK *Form IN10. Register a private or public company. Application to register a company* (2024). In BFS architecture, the Registrar utilises this form (see Fig. 5.3) to automate generation of the incorporation documentation, Genesis Block for the BFS instantiation and further sets the criteria for the Director Wallets generation, based on the information provided in this form.

Private Register a data record item in the hypothetical entity called "Companies House" Companies House (2024*d*) - the government managed database. Form IN10 becomes part of the Private Register (see Fig. 5.3). This data record is generated when the Registrar approves the incorporation of the said company. The copy of such a record is assumed to be permanently and securely stored in the hypothetical "Companies House".

This record contains all company incorporation documentation (such as Certificate of Incorporation) together with a copy of BFS Genesis Block for instantiation of the BFS as a blockchain-based economic entity. Upon its generation, its copy is shared with the entities directors to enable them to instantiate BFS as the blockchain underpinned entity.

❖ *Business-Specific Context.*

Lending Agreement. This is a document designed as a complex 'smart contract' which automates hypothetical legal contractual agreement between its parties in the application through an executable code. In this project, the lending agreement data (variables) and its rules (functionalities) are specified through adaptation and simplification for illustrative purposes of the use-case specific requirements. These requirements are drawn from the public data souses. This agreement is between the APF and the BoE. The high-level requirements are as follow:

- Inception and agreement of the 'lending agreement' between the BoE and the APF;
- This is to enable of purchases of private sector assets by the 'Facility'. These purchases to be financed solely by the "deposit from the Bank" (Bank of England (2009*b*)).
- Initial purchases of Commercial Paper (the asset) must be financed by the issuance of Treasury Bills (Bank of England (2009*b*)).
- These Treasury Bills are lent by the BOE to the APF, which in turn "uses these as SourceOfFunds" to purchase Commercial Paper from its issuers of such assets. This means that the APF swaps illiquid Commercial Paper issued by private sector for highly liquid Treasury Bills borrowed for the Bank.
- Upon receipt of notification from the APF, of APF's intention to draw down under this loan, the Bank is required to make the advance (Bank of England (2010*b*)).

Draw Down Request. This is a document designed as a complex 'smart contract' that delineates and automates the legal contractual agreement between its parties in the application through an executable code. In this project the drawdown request data (variables) and its rules (functionalities) are specified through adaptation and simplification for illustrative purposes of the use-case specific requirements. These requirements are drawn from the public data souses. This drawdown request is sent from the APF to the BOE. The high-level requirements are as follows:

- This drawdown request is to cover "near-term cash flow requirement" of the APF to finance its assets purchases from the wider market participants;
- This drawdown request is to receive a requested amount of Treasury Bills;
- Treasury Bills for this purpose are issued with a maturity of three months and price matched to the requested nominal value by the BOE, which then makes them available immaterially to be borrowed by the APF upon the request.

❖ *Business Transaction Elements.*

Business Transaction. This is an event of economic nature with monetary implications for the business. According to Stolowy & Ding (2019), transactions that are not explicitly completed at the end of the accounting period are considered to be opened Stolowy & Ding (2019). In this architecture, only completed transactions are pushed though accounting cycle.

Shares. This is an asset issued by the APF. For this asset to be utilised in the digital infrastructure of the BFS ecosystem, it undergoes processes of Tokenisation. Then, as a digital Token, it is issued into the ownership of the APF. Shares are issued and tokenized during incorporation of APF and at instantiation of the BFS. The data for the share particularities is provided by the APF directors in the Form IN10 to the Registrar. The specification for this Share Token is as follow:

- 100 shares issued;
- Issued at par value of £1.00 per share;

- APF is the Issuer;
- other issuer relevant data;
- Tokenisation relevant procedures implementation.

These Shares themselves are not transferred during business translation; the Share Certificate (outlined next) that pertains to all relevant sold shares, such as date and shareholder are generated, tokenized and transferred to that shareholder.

Share Certificate. The tokenized document that demonstrates the purchase of Shares from the Share issuer (the APF in out use-case) by the shareholder (the BOE in BFS use-case). This tokenized certificate pertains to all relevant details relating to the shares that have been taken on a certain date by a particular shareholder. During a business transaction, this Certificate is generated and is transferred to the shareholder's ownership.

Share Stock. The issued Shares are stored in Share Stock (off-ledger repository of the BFS application) in the possession of the APF. Newly issued Shares are deposited as authorised shares and when these are sold to the shareholders, these Shares become outstanding shares, still stored as the part of the Share Stock repository of the APF.

- Authorised Shares refer to the maximum number of shares that a corporation is legally permitted to issue; it includes already-issued stock, along with shares that have the management's approval but have not yet been released onto the trading market, including "Initial Authorised Share Capital" that is the share capital issued at incorporation of an entity.
- Outstanding Shares - Outstanding shares include those held by shareholders and company insiders.

Commercial Paper. This is a debt asset issued by the CPI-1. For this asset to be utilised in the digital infrastructure of the BFS ecosystem, it undergoes a process of Tokenisation. Then, as the digital Token, it is issued into the ownership of the CPI-1 and then transferred as part of the business transaction into the ownership of APF. The specification for this Commercial Paper Token is as follow:

- Issued at discount to its nominal value;
- CP is issued on the agreed trade date (agreed with APF);
- Three month to maturity;
- other issuer relevant data;
- Tokenisation relevant procedures implementation.

Treasury Bill. This is a debt asset issued by the BOE. (In this project, for illustrative purposes of the BFS, this Treasury Bills will be issued not by the DMO, but directly by the BOE.) For this asset to be utilised in the digital infrastructure of the BFS ecosystem, it undergoes a process of Tokenisation. Then, as the digital Token, it is issued into the ownership of the BoE and transferred as part of the business transaction into the ownership of APF, who subsequently will use it to fund the purchase of Commercial Paper from the CPI-1. The specification for this Treasury Bills Token is as follow:

- No coupon - vanilla bond;
- The issuance of TB is matched with the issuance of CP and both securities are issued on the same day trade with the same day maturity.
- Three month to maturity;
- Nominal value matched to the requested funds value from the APF;
- Tokenisation relevant procedures implementation.

❖ *Participants Registration Context.*

Member Wallet. Individuals or entities that have ownership, interests in this entity, or are employed by such entity in any capacity. These wallet types are registered with its BFS entity and are awarded role-relevant functionality and access rights to the internal BFS data, e.g., directors, employees and others. Fig. 5.5 illustrates an example of two APF Member Wallets owned by the directors of the APF. These specific wallets were instantiated at incorporation of APF Companies House (2023*b*) concurrently with

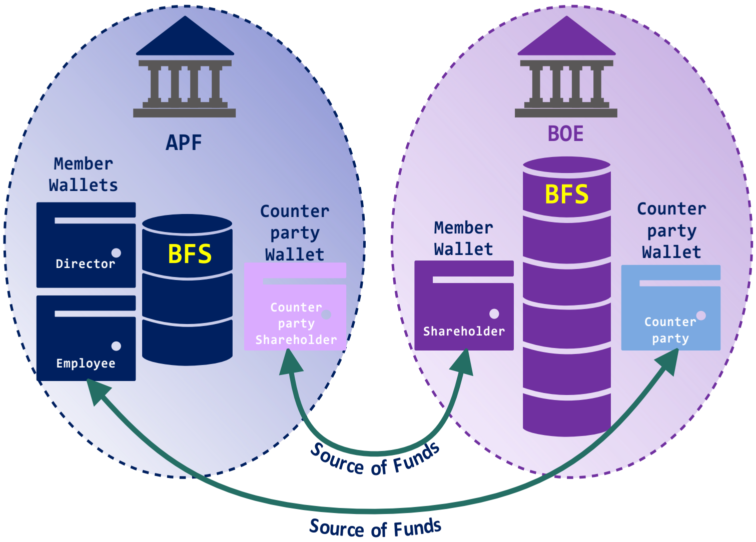


Figure 5.5: Wallet Types: Member and Counterparty Wallets.
In the BFS ecosystem, each business entity is composed of various members who perform specific roles critical to business operations, such as members and counterparties. These are materialised as Wallets and are allocated different roles that define the responsibilities and relationships each individual has within their respective entity. Wallets of the APF Directors illustrated as dark blue on the left to the APF entity - as the member of the APF, and in lighter blue colour instantiated with the BoE entity on the most right - as the counterparty wallet. The green arrow indicates connection that is responsible for provision of funds for the counterparty wallet from its primary entity, to cover transactional obligations. The single Member Wallet of the BoE (as the Shareholder) is illustrated as darker purple and is a part of the BoE entity. In order to transact with APF, a counterparty wallet, illustrated in th lighter purple colour is instantiated within APF, and it is connected to its primary entity via source of funds relationships.

instantiation of its BFS. In Fig. 5.5 these are depicted as dark blue on the left to the APF entity. The single Member Wallet of the BoE is illustrated as darker purple in Fig. 5.5 and is a part of the BoE personnel.

Counter-party Wallet. This concept for a wallet is based on a ‘customer account’ registered with an in e-commerce website. In BFS example, for the BoE to transact with the APF, the BoE has to register a counter-party type wallet in the APF. This counter-party wallet (light purple in the Fig. 5.5) is part of the APF BFS infrastructure, although it is owned by the BoE. This wallet’s access to the internal business data of the APF is prohibited, but it does have access to the APF transactions that this wallet is a counterparty to. Furthermore, this wallet supports a direct communication channel with its owning BFS (the BoE in this example). This data communication channel is called “Source of Funds” and is utilised to move/ transfer of funds necessary to cover relevant transactions.

❖ *Business Transactions Register Context.*

Business Transactions Register. Application local repository for registering of member and counterparty wallets for this business entity.

❖ *Participants Roles Context.*

Director. This role and its responsibilities within APF::BFS artefact are modelled on a hypothetical company director with specific focus on the use-case implementation, such as instantiation af APF::BFS, and implementation of initial share issuance and sale business transaction demonstrated in Section 7.2.

Shareholder. The use-case specific role for the implementation for BFS wallet is the counterparty role encapsulated by the shareholder of the APF - the BoE (Companies House (2023*a*), Companies House (2023*b*)). In BFS PoC implementation, this wallet acts as a counterparty to the initial share purchase transaction outlined in Section 7.2.

Borrower. This role serves firstly as a digital embodiment of a hypothetical employee of the APF, operating within the BFS ecosystem, facilitating the APF's business objective of obtaining funding, required for QE implementation. Use-case based arrangement for QE implementation is supported by the illustrative adaptation of the scenario for simulation of issuance and subsequent lending of tokenized Treasury Bills to the APF by the BoE. In BFS PoC implementation, this wallet acts as acquirer of funding for QE implementation, through Drawdown Request on Loan Agreement with HTLC transaction described in Section 7.4, and implemented in Section 7.5.

Lender. In alignment with the use-case role is modelled to operationalize implementation of the QE process on the side of the BoE - as the lender. This function is realized through a conceptual lending agreement between the APF and the BoE, demonstrated through the use-case specific issuance and lending of tokenized Treasury Bills to the APF from the BoE, presented in Section 7.4), and implemented in Section 7.5.

Facility. As "Commercial Paper Facility (CPF)", this role for a hypothetical employee of the APF, whose wallet is modelled on the use-case for this research project. It acts as *Commercial Paper Facility* schema of the APF, where its functions are o "undertake assets purchases, with initial focus on investment grade sterling Commercial Paper (CP) issued by the UK corporates" (Bank of England (2009*c*)). Following adaptation of sequential historical events of the use-case, first purchases of the CP by CPF to be financed by the issue of Treasury Bills (TB) (Bank of England (2009*a*)). In this project, for illustrative purposes of the BFS, Treasury Bills will be issued not by the DMO, but directly by the BOE. In BFS PoC implementation, this wallet acts as QE provisioning entity in the Atomic Three-Party QE with HTLC transaction outlined in Section 7.5.

Counterparty. This role is the counterparty to APF, specifically to the Facility role above. Counterparty embodies the role of **Commercial Paper Issuer - 1 (CPI-1)** acting as a market counterparty in QE business transactions (see Section 7.5). This role's functionality is governed by the specifics of the use-case and it aim to obtain financial support from APF, by issuing and sale CP to the APF::BFS in exchange for TBs. In BFS PoC implementation, this wallet acts as QE requesting entity in the Atomic Three-Party QE with HTLC transaction outlined in Section 7.5.

5.4 Mechanisms

This section provides a examination of how the BFS architecture facilitates financial operations between distinct entities within the BFS ecosystem. Central to this discussion is the interplay between APF and it use-case relevant counterparties: the BOE and the hypothetical market entity referred as CPI-1.

5.4.1 Mechanisms of Financial Accounting Sub-Domain

Financial Accounting within architecture of BFS plays critical role in ensuring the system's functionality aligns with the established traditional principles of financial accounting. This sub-domain, within its bounded contexts, introduces a set of mechanisms that are fundamental drivers for the operational mechanisms of the BFS architecture. These mechanisms facilitate accurate processing of business transactions that affect entity's finances, gathering these transactions, and analysing them for financial implications. Economic variables from these transactions are logged in accounting journals, ledgers and reports, ensuring chronological documentation of financial activities.

❖ *Account and its Hierarchy Context*

Account generation and Classification - are the mechanisms for constructing and categorising financial accounts into types (cash, equity, debt securities) based on their function and nature, reflecting the entity's financial structure. This mechanism employs *Account ID* together *Account* and *Account Side* elements, described in Section 6.1.1 of this report.

Account Hierarchy Management - organises accounts in a hierarchical structure to simplify reporting and analysis, supporting the efficient aggregation and disaggregation of financial data. Implementation of this mechanism is focused on *Account Class*, *Account Category* and *Related Accounts* components, described in Section 6.1.1 of this report. Demonstration of the hierarchical consolidation of accounts' data flow is provided in the Section 6.1.1

- *Accounting Journals and Ledgers Context*

  Chart of Accounts Management - mechanisms for taxonomy and identification financial accounts within business entity. Implementations of these are described in Section 6.1.2.

  Recording of Accounting Data - captures accounting data within accounting journals and ledgers, providing a chronological record and initial analysis of each economic event. Implementations of these are described in Section 6.1.2.

- *Financial Reporting Context*

  Financial Statement Generation - utilises end of period entries of a trial balances(adjusted) to produce key financial statements, such as balance sheet, offering insights into the entity's financial health (see Section 6.1.3).

- *Accounting Transactions Context*

  Economic Event Analysis - identifies and analyses economic events within business transactions, for their financial implications mapping to entity specific accounting processes, ensuring accurate financial representation (see Section 6.1.4 for implementation).

  Double-Entry System Application integration and enforcement of double-entry accounting principle, maintaining the integrity of the accounting equation (Assets = Liabilities + Equity) across all accounting transactions (see Section 6.1.4 for implementation).

  Accounting Cycle Management - mechanisms for implementing logical binding of procedural data flow for generation of accounting rectors, through adaptation of traditional accounting cycle steps to the BFS environment (see Section 6.1.4 for implementation and Fig. 2.1 illustrating these procedural steps).

- *Accountant's Role and Responsibilities Context*

  Data Collection and Management - mechanism enables Accountant to manage and perform all necessary functionalities (see Section 6.1.5) to collect, record, and communicate of economic data, aligning with traditional accounting principles.

  Accounting Repositories Management - maintains a verifiable, chronological record of financial activities, supporting the transparency and auditability of the BFS (see Section 6.1.5).

The taxonomy of mechanisms within the Financial Accounting Sub-Domain of the BFS architecture demonstrates integration of domain-driven design principles with the practical requirements of financial accounting. By describing specific mechanisms within each bounded context, the BFS architecture achieves modular and clear structure that not only meets the technical demands of blockchain technology but also addresses the nuanced needs of financial accounting. This structured approach ensures that the BFS is a comprehensive, user-centric solution capable of enhancing the efficiency, accuracy, and transparency of financial management and reporting in the digital age.

5.4.2 Mechanisms of Blockchain Sub-Domain

In the BFS architecture, the Blockchain Sub-Domain stands out for its innovative adaptation of blockchain technology to enhance financial reporting and transaction processing for economic entities. This sub-domain is designed around several bounded contexts, each contributing to the functionality and efficiency of the BFS by leveraging blockchain's inherent strengths. Below, several key mechanisms are presented within each of these bounded context, highlighting their roles in the

BFS architecture.

❖ *BFS Block Structure Context.*

Data Storage Customisation - as a mechanism that involves defining the types of data stored within each block of the BFS's filing history, focusing on the aggregation of accounting data rather than individual transactions. It allows for a balance between maintaining confidentiality and ensuring transparency of financial activities. Variations and functional management of the BFS's unique Block data types and their roles, including traditionally expected block header and block body, and their significance are described in the following Section 6.2.1.

❖ *The BFS and Ledger Context.*

Ledger Design - tailors the BFS ledger to meet the specific needs of financial statement reporting, ensuring that its structure accommodates aggregated accounting information with appropriate identifiers for underlying economic activity.

Chronological Data Repository - establishes the BFS ledger as a cryptographically linked time-sequenced record of the business's state, capturing the evolution of financial data over time. This mechanism is central to providing verifiable historical account of economic activities, accessible for audit and analysis.

The instantiation and presentation of BFS mechanisms is demonstrated in Section 6.2.2.

❖ *Smart Contracts Context.*

Process Automation use-case specific mechanisms utilised for design and implementation of diverse smart contracts to automate business and accounting processes, automation of lending. By reducing manual intervention, smart contracts provide reliance on the contractually defined execution of events within BFS - as and entity, and between BFS ecosystem participants. These contracts are customised to fit the unique requirements of each event and functional requirements of the APF use-case.

Rule Definition - enable requirement specific design and implementation of arbitrary rules embedded within smart contracts of the BFS, providing governance of transaction processing, ensuring that all financial activities are executed in accordance with established protocols, agreements and standards.

Broader definition, together with BFS implementation of smart contract mechanisms are outlined in Section 6.2.3.

❖ *Wallets Context.*

Identity Definition and Access Management - mechanism that provides ability to uniquely define and identify individuals within digitised realm of BFS ecosystem. These mechanisms are designed and implemented to manage digital BFS wallets for both members and counterparties, defining the permissions for accessing internal company data and executing specific types of business activities. This mechanism safeguards sensitive financial information while facilitating authorised transactions.

Cryptographic Security - oversees the generation and verification of cryptographic elements, such as hashes and electronic signatures within these digital BFS wallets, enhancing the security and authenticity of transactions processed through the BFS.

These mechanisms are outlined in the Section 6.2.4.

❖ *Validator and UTXO and Consensus Contexts.*

Transaction Integrity Verification - implements processes for verifying the integrity of received business transactions, maintaining the statelessness and immutability principles of blockchain technology. This mechanism ensures that each transaction strictly adheres to the BFS's predefined rules and conditions.

Funds Verification Process - implements consensus mechanisms tailored to suit the BFS artefact, facilitating the verification of funds availability between transacting counterparties. This mechanism plays a critical role in maintaining the integrity and reliability of financial transactions within the BFS. Facilitates UTXO processes that is adapted to serve as a novel mechanism for verifying funds availability within the accounting process.

This approach is specifically designed to align with the BFS's accounting needs. These mechanisms are outlined in Sections 6.2.5, and 6.2.6.

The mechanisms within the Blockchain Sub-Domain are fundamental to the BFS architecture, enabling the precise and secure recording, reporting, and verification of financial activities through blockchain technology. These mechanisms not only enhance the efficiency and integrity of financial management within the BFS but also demonstrate innovative approaches to financial accounting and reporting in the digital era. By leveraging blockchain's capabilities, the BFS architecture offers a forward-looking solution that aligns with the evolving landscape of financial transactions and compliance requirements.

5.4.3 Mechanisms of Business Entity Sub-Domain

The Business Entity Sub-Domain within BFS architecture encapsulates the operations, structures, and transactional dynamics of business entities engaged in the BFS ecosystem. This sub-domain, defined by its bounded contexts, outlines a series of mechanisms crucial for the orchestration and management of business entities and their interactions within the BFS framework. These mechanisms ensure that entities, such as APF, can efficiently and securely conduct their business within the blockchain-enabled environment of the BFS.

- *Company Incorporation Context.*
  Incorporation Process Management - mechanisms here facilitate the formal establishment of a business entity, guiding them through the submission of digitised incorporation documents to relevant authorities (such as Registrar) and the subsequent receipt of BFS essential documents like the Certificate of Incorporation and the Genesis Block for BFS instantiation.
- *Business-Specific Context.*
  Operational Facility Management - includes mechanisms for setting up and managing specific business operations, such as lending facility example for the APF use-case. Generalised concept for this mechanism allows entities to formalise lending agreements and manage drawdown requests. In the use-case specific implementation, this mechanism enables APF to acquire necessary funding for QE operations.
- *Business Transaction Elements.*
  Transactional Execution Framework - contains mechanisms for the generation, management, and storage of elemental components necessary for execution of business transactions, aligning them with the entity's structure and objectives. This framework supports use-case specific elements to orchestrate various transaction types, outlined in the Section 6.3.3.
- *Business Transactions Register Context.*
  Transactional Interaction Tracking - mechanisms here ensure the maintenance of an application local repository to track the entity's transactional interactions. This supports confidentiality, compliance, and transparency by making business-sensitive information accessible only to relevant parties.
- *Participants Registration Context.*
  Relationship Management - mechanism for a systematic wallet registration and management within BFS entity. This mechanism ensures that each participant's role and identity are accurately recorded and recognized, facilitating smooth interactions within the BFS domain.
- *Participants Roles Context.*
  Role Assignment and Management these mechanisms focus on appointment and management of entity members, such as directors, shareholders, lenders, and employees. It includes the registration of transactional counterparty wallets, ensuring each participant's role and responsibilities are clearly defined and managed within the BFS.

The Business Entity Sub-Domain's mechanisms enable functionality of the BFS architecture. By establishing a structured approach to company incorporation, operational management, transactional execution, and participant interaction, the BFS design ensures that business entities can navigate and build on complexities of blockchain integration. These mechanisms not only facilitate the secure and efficient operation of entities within the BFS but also enhance the overall integrity and reliability of the financial reporting and transactional ecosystem. Through these structured processes and interactions, the BFS architecture demonstrates its capability to support and enhance the operational efficiency and transparency of business entities in the evolving landscape of financial technology.

5.5 Design Prioritisation in the BFS Architecture

Design components of the BFS architecture, which includes actors, mechanisms, and foundational elements, can be classified into two distinct groups based on their criticality: "must-have" elements (foundational to the system's functionality) and "value-judgment" elements (designed to enhance flexibility, scalability, or usability). This categorisation reflects a deliberate approach to balancing core functionality with flexibility and adaptability, making sure that the BFS architecture meets its objectives, while remaining versatile for future iterations and diverse use-case scenarios.

This layered classification philosophy enables the BFS to function effectively in its current implementation and provides flexibility for future iterations and broader adoption. Must-have elements form the foundational backbone of the functionality, ensuring that BFS meets its essential objectives of accuracy, transparency, and compliance. Value-judgment elements, on the other hand, enhance the system's adaptability to varied use cases and future requirements, ensuring the BFS remains relevant and practical across different organisational contexts.

5.5.1 Must-Have Design Elements

These elements are indispensable for the BFS architecture to meet its functional, operational, and technical goals. Non-negotiable inclusion od these ensures that the BFS aligns with its core objectives, adheres to established accounting principles, and satisfies the requirements of blockchain integration. These elements include:

Core Accounting Framework - integration of the double-entry accounting system ensures compliance with established accounting standards and maintains the integrity of financial data by upholding the accounting equation (Assets = Liabilities + Equity). Integrated into the Accounting Transactions Context and mapped to ledgers (e.g., General Ledger, Trial Balance). This framework forms the backbone of financial data processing and reporting within the BFS.

Blockchain Filing History - a secure, immutable ledger that records aggregated accounting and financial data chronologically. This element ensures transparency of financial activities, auditability, and resistance to tampering. Custom block structures store aggregated accounting data instead of raw transactional data, offering confidentiality with transparency.

Aggregation of Accounting Data in Blockchain Block Structures - BFS ledger aggregates data at the account level instead of storing raw transactional details. This design protects business sensitive information while maintaining auditability, balancing privacy with compliance requirements.

Ledger and Financial Reporting Mechanisms - essential ledgers (e.g., General Ledger, Trial Balance) and reporting mechanisms (e.g., balance sheets, income statements) provide accurate and transparent financial insights. These components are critical for standardised and reliable financial reporting and for auditability.

Smart Contracts -automate essential processes such as accounting transactions, business agreements (e.g lending agreement), together with BFS incorporation and instantiation processes.

They enforce predefined rules enhancing operational efficiency and facilitate reliability of transactional executions.

UTXO and Funds Verification Cansensus - a novel approach to adapting the UTXO (Unspent Transaction Output) model for accounting purposes. It ensures secure fund verification, enhances transaction integrity by aligning funds availability checks with accounting practices.

Role-Based Access through Wallets - digital wallets represent actors (e.g., directors, shareholders, and counterparties) and enforce access permissions to ensure secure and authorised operations. Includes member wallets for internal operations and counterparty wallets for external interactions. Includes member wallets for internal operations and counterparty wallets for external interactions.

5.5.2 Value-Judgment Design Elements

These elements, while not strictly essential, add significant value by enhancing the BFS's usability, adaptability, and scalability. Their inclusion reflects subjective decisions based on stakeholder priorities, specific use-case requirements, or anticipated benefits, and are designed with the vision of alignment with emerging technologies or anticipated benefits. Examples include:

Customisable Financial Reporting Templates - while the BFS generates standard financial statements (demonstrated through a balance sheet), additional templates for income statements and specialised reporting (e.g., regulatory compliance reports) that reflect stakeholder preferences can be customised and adopted to aligns with diverse reporting requirements.

Customisation of Wallet Features - extend standard wallet functionality to align with specific roles (e.g., director wallets for governance, counterparty wallets for restricted access).

Smart Contract Rules for Custom Processes - while essential contracts (e.g., business and accounting transactions, lending agreements) are must-haves, additional contracts (e.g., for specific business workflows) can be customised and value-driven to support diverse business scenarios.

Tokenisation of Financial Instruments - converts assets such as shares, commercial paper, and treasury bills into digital tokens for efficient management and transfer. While processes and assets, demonstrated in this project are essential to satisfy current use-case, variety of tokenisation processes and underlying financial elements can can be implemented and adapted to the specific operational needs of an organisation.

Registrar Integration for Genesis Block Creation - the Registrar's role in automating company incorporation and generating the Genesis Block is significant but not essential for all BFS implementations. Streamlines onboarding and establishes a clear starting point for blockchain-enabled entities.

Data Repositories for Business-Specific Contexts - all additional repositories (e.g., for share certificates, drawdown requests) enhance data organisation and retrieval but could be streamlined or omitted based on operational needs. These add granularity and facilitates complex workflows.

This Chapter 5 presented design of the BFS architecture, underpinned by the Domain-Driven Design (DDD) methodology. This approach facilitated development of a domain model encompassing three core sub-domains—Financial Accounting, Blockchain, and Business Entity—each divided into bounded contexts that address specific operational requirements (see Fig. 5.2).

The actors, mechanisms, and components within the BFS architecture were categorised and described, illustrating how their interactions enable the BFS ecosystem to address challenges in financial reporting, liquidity management, and transactional transparency. From the foundational mechanisms of account generation and financial reporting to the innovative use of smart contracts, tokenisation, and blockchain-based ledgers, the architecture demonstrates its capacity to bridge traditional financial practices with cutting-edge technology.

By embedding transparency, security, and efficiency into its structural framework, the BFS sets a precedent for blockchain-based financial systems. These design principles provide a base-line platform for the BFS to operate effectively while remaining adaptable to evolving technological and regulatory landscapes.

Moreover, this Chapter elaborated on the must-have foundational elements necessary for the BFS to function, and on the value-judgment elements that provide flexibility and scalability for future iterations

Transition to Chapter 6, building on the theoretical foundation established in this Chapter, the focus is shifted to the practical application of the BFS architecture. Next Chapter will go through the implementation and proof-of-concept, demonstrating capabilities of the BFS's in a controlled laboratory setting. By translating the architectural blueprints into functional systems, the Chapter 6 provides tangible evidence of the BFS's effectiveness in managing transactional data, automating accounting processes, and enhancing financial transparency.

Chapter 6

BFS Implementation and Proof of Concept

The Domain-Driven Design methodology (DDD) serves as a guiding principle for the BFS architecture, facilitating the formulation of a domain models that encapsulate the essence of business entities, blockchain technology and financial accounting, (Fig. 4.1). Three core sub-domains are dissected into bounded contexts (see Fig. 4.2), each embodying distinct aspects of the BFS's operational domain. Such a stratified architectural setup ensures that the BFS architecture aligns with the technical nuances of blockchain and resonates with the nuanced demands of financial accounting in a digitised ecosystem. In this chapter, through a proof-of-concept demonstration of the data structures and processes within these bounded contexts, the Thesis seeks to illustrate the functionality and effectiveness of BFS in a controlled environment, thereby contributing valuable insights into the ongoing discourse on the future of financial technologies.

Chapter 6 ventures into the practical application and verification of the BFS implementation, unravelling the functionality and potential of BFS within controlled laboratory setting. This Chapter aims to transition from the theoretical discussions and architectural blueprints previously laid out to demonstrating the tangible application and effectiveness of the BFS in embodying the principles of blockchain, accounting and business orchestration. Through a detailed proof-of-concept demonstration, this section will illustrate how BFS implemented and how it operates.

The proof-of-concept approach used in this Chapter focuses on translating theoretical designs into practical application by implementing specific use-case scenarios to demonstrate the BFS's capability in managing transactional data, automating accounting processes. This demonstration not only validates the BFS architecture, but also demonstrates the system's adaptability and potential for real-world deployment.

The demonstration begins with a deep dive into the implementation of the Financial Accounting sub-domain. This segment presents adaptation of the conventional accounting processes, detailing integration of the accounting cycle into the BFS framework. It explores the establishment of accounts and their hierarchies, the structuring of ledgers, the compilation of financial reports, and the dynamics of accounting transactions within the BFS environment. Furthermore, it sheds light on the roles and responsibilities that accountants hold in this blockchain-integrated system, redefining their functions and interactions within the digital accounting domain.

Following financial accounting implementation, the description shifts to the Blockchain sub-domain, unveiling how the incorporation of blockchain architecture not only reinforces the integrity and security of financial data, but also reshapes the transparency and efficiency of transactions and record-keeping. Detailed discussions encompass the BFS block structure, ledger context, and the critical role of smart contracts, wallets, and validators in providing the system's functionality. Special emphasis is placed on the UTXO and consensus mechanisms, highlighting their innovative application, providing verification, and ensuring consistency of financial records.

The Chapter then transitions to the Entity Sub-Domain, illustrating how BFS accommodates the incorporation of companies, manages business-specific components, and facilitates the registration and role definition of participants. This section encapsulates the operationalisation of business transactions within the BFS, presenting a comprehensive consolidation with blockchain technology to be leveraged to enhance accuracy, accountability, and accessibility of financial statements, setting a new benchmark for financial reporting in the blockchain era.

Additionally, the Chapter describes BFS implementation using Java as the programming language. Java's use in the BFS examples ensures a stable, scalable, and secure environment for implementing blockchain technologies, aligning with the system's need for efficient execution of smart contracts, data management, and future adaptability.

6.1 Implementation of Sub-Domain for Financial Accounting.

This section describes aspects of the implementation of the accounting sub-domain within the overall Blockchain Financial Statements framework. It illustrates how traditional accounting processes, components and roles can be integrated within the architecture of the BFS. However, these accounting aspects are not merely replicated. They are adapted to showcase integration of conventional accounting into practical and operational components within the innovative blockchain-based environment of the BFS. This integration is illustrated though a range of bounded contexts, including configuration for accounts and their hierarchical structures, accounting ledgers, the accounting cycle, financial statements and associated accounting roles, along with their corresponding components necessary for BFS. This sub-domain is tasked with gathering and organising such components and methodologies, transforming them to align with the innovative BFS system, thus contributing to the advancement of financial reporting in the digital age.

6.1.1 Account and its Hierarchy Context

The context of Account encapsulates its configuration together with overall hierarchical organisation between such accounts.

Account ID

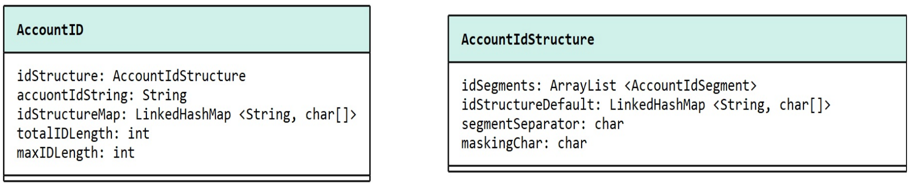
The concept of "Account ID" is a fundamental element in accounting with a multitude of uses. It is a unique identifier for each account within an accounting system which streamlines tracking and reconciliation of individual transactions. It allows for referencing and cross-checking, ensuring that each transaction is accurately accounted for and aligned with the correct account. Traditionally, each account is uniquely identified by either or both:

1. its name (or description), and
2. an account ID - featuring hierarchical numbering, so as to enhance comparability and consolidation (Stolowy & Ding (2019)). It can be made up of numbers, letters and the combination of both.

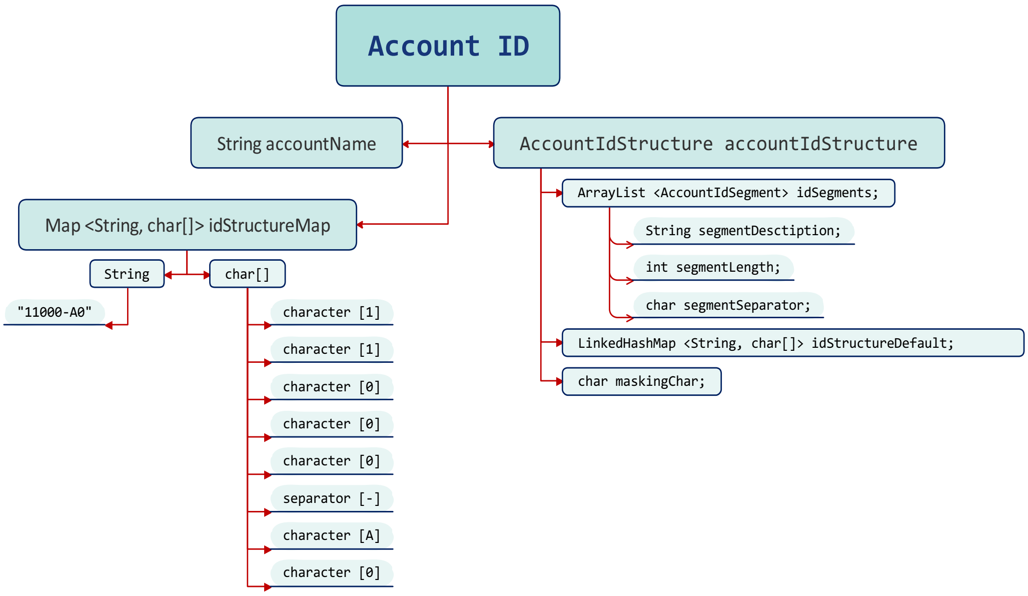
The internal logic of a hierarchical numbering of `Accounts` contributes to a deeper understanding of the nature and the role of each `Account` within the financial accounting framework (Stolowy & Ding (2019)) which, in turn, is designed to reflect business activity of an economic entity. This Thesis adapts this concept of the account ID (or account number) as an alphanumeric code utilised to uniquely and hierarchically identify each `Account`. Each `AccountID` is set up for each individual `Account`, so as to allow for the tracing of the many transactions that will impact the final balance of such `Account` in the normal course of the business within an accounting period. This `AccountID` is also used by 'accounting smart contracts to match completed business transactions with the relevant `Accounts`, to record economic activity in accounting journals and report economic variables in financial statements and managerial accounts. More specifically, the `AccountID` is used to identify and manage the same `Account` in relation to:

❖ Accounting Transactions;

- ❖ COA - a complete collection of all Accounts of a business entity;
- ❖ GL - as well as inter-accounting period transactional values and the account balances for a specific period of time;
- ❖ Running Trial Balance;
- ❖ Reports, such as Financial Statements;
- ❖ the BFS.



(a) UML class diagrams for AccountID and AccountIdStructure classes.



(b) Account ID class instance value mapping.
This figure represents visually logical assignment of values to the instances of the Account ID

Figure 6.1: Account ID class.

his study now illustrates some of the primary variables of the `AccountID` class, as illustrated in Fig. 6.1:

1. `String accountName` (e.g.“Cash”) - the human readable name assigned to this `Account`, that will be displayed in all accounting reports.
2. `String accuontIdString` (e.g.,“11000-A0”) - a human readable representation of the alphanumeric account number;
3. `Map <String, char[]> idStructureMap` - actual initiated map of the `AccountID`. For example, one could set up an `Account` with the `AccountID` containing four segments formatted as `Strings`: “1” “1000” “-” “A0”:
   (a) First `String` is a one character length numeric `AccountClass` (see 6.1.1) identifier such as: 1 = assets, 2 = liabilities, 3 = equity, 4 = income, 5 = expenses. in the hypothetical example of this project this first character is:
      - character [1] - Assets

(b) The second String is a four character length hierarchical identifier for a specific Account within that AccountClass (see 6.1.1):
- character [1] - categorised as "Cash"
- character [0] - no further accounting categorisation
- character [0] - no further accounting categorisation
- character [0] - no further accounting categorisation

(c) The third String is a separator character used for human readable division between the GL filtering of Accounts and a separate department within an organisation (if it exists):
- character [-]

(d) The last String orders hierarchically internal details for such departments and indicates department identifier:
- character [A] - Cash operations department
- character [0] - no further categorisation

4. AccountIdStructure accountIdStructure - the structure of the AccountID is an important concept, as it is crucial in the hierarchy of relationships between the Accounts within the entity's accounting framework. It allows for filtering of the information on the financial statements by pre-defined segments that are decided on (and set up) at the point of instantiation of BFS (see Section 6.2.2). This segmentation of the AccountID enables 'Account ID Masking' protocol (see 6.1.1), which employs automation to the process of filtering of internal financial data by, e.g.,: AccountCategory (see 5.3.1) and/or AccountClass (see 6.1.1) and RelatedAccounts (see 6.1.1) or any other type of business unit such as: department, location and so on. The AccountIdStructure class is designed to contain:

   (a) List <AccountIdSegment> idSegments - where each AccountIdSegment class is initiated at instantiation of BFS (see Section 6.2.2) to serve a pre-defined logical role the internal accounting framework. These account ID segments are critical to the concept of Account ID Masking (see 6.1.1) and are applied to chronological ordering of internal accounts. These segments are utilised for on demand reports filtering on a particular segment, e.g., by class of accounts, a particular department and so on. For example, one could set up an Account with "11000-A0" as AccountID, which will contain three distinct segments: [1][1000-][A0]. Each of these AccountIdSegments is a class that encapsulates metadata for:

      i. String segmentDesctiption - this description could be a class of the accounts or a department, division or any other designator for that segment of an AccountID;

      ii. int segmentLength - the length of that segment, e.g., length from one to more characters;

      iii. char segmentSeparator - a separator between this segment and the segment following it (if any). It is a pre-defined character utilised to divide segments to ease readability, e.g., '-' (dash);

   (b) Map <String, char[]> idStructureDefault - a pre-agreed, at the point of instantiation of BFS (see Section 6.2.2), default structure of all AccountIDs employed in the accounting framework of this economic entity. This 'default ID structure' includes both, segments and their separators (if any). This structure is copied into every AccountID class, where it is utilised as the repository for the actual ID segments (see point 3 below).

   (c) char maskingChar - a pre-defined character, e.g., '*' (star), that can be utilised to mask (exclude) Accounts that are not required for a specific report request. E.g., if you only want to filter information for all (and only) assets, you would utilise masking characters like so: 1****-**. The account masking character (often a placeholder or wildcard, such as an asterisk * or another specific character) is used to represent part of an account number that is intentionally hidden. For instance, if only the first digit of an account number is shown, this indicates that requested for analytics account is, e.g. "Total Assets", while the remaining digits are masked to filter out more granular details.

Account ID Masking

`AccountIdMasking` class is utilised for the on demand filtering of the reports, based on the account ID hierarchical order within the accounting framework of an economic entity. These reports are filtered using a pre-defined masking character, e.g., '*' (star), and can be based on a specific segment or group of segments to produce financial summaries. E.g., if you only want to produce an inter-accounting period report on all financial resources controlled by an entity (assets), you would utilise masking character like so: 1****-**. The only fields of this class are:

1. `AccountIdStructure idStructure` (see 4a);
2. `String accountMask` - a string of the masked `AccountID` that the data records must be filtered for. E.g.,if the masking `char - '*'`, then for `String accountMask` = `''1*****-**''` will filter for all asset accounts requested.

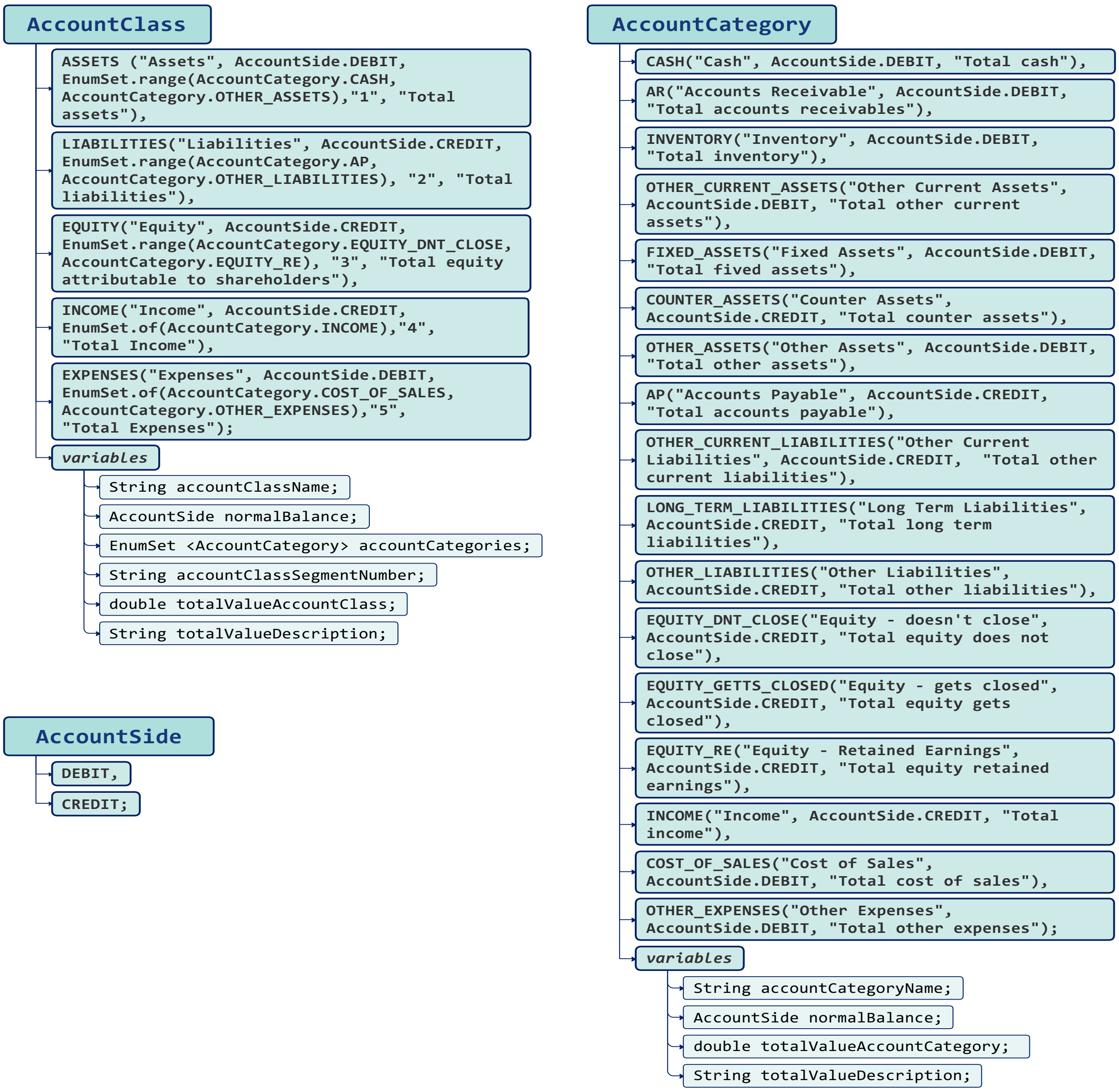


Figure 6.2: Account Class, Account Category classes, and Account Side Enumerators.

Account Side

For the implementation of the `enum AccountSide`, the notation usage of `enum` as a Java class type here and further in the implementation of the BFS is necessary. This implementation choice is motivated by the specific aspects of the `enum`. It is ideal when dealing with a fixed set of constants, which is exactly the case with `AccountSide` (representing either debit or credit), `AccountClasse`s (representing specific categories like assets, liabilities, equity, etc.), and further in the implementation of the BFS. Additionally, using enum ensures type safety, and guarantees that only valid, predefined values for `AccountSide` and other traditionally required accounting categories can be used throughout the BFS system. This means that developers or users cannot mistakenly assign an invalid value, which significantly reduces errors and improves reliability.

At the bottom of Fig. 6.2 another important concept of accounting is illustrated. This is `enum AccountSide` which contains `DEBIT` and `CREDIT`. In a double-entry accounting system, every financial transaction affects at least two accounts. If one account is debited, another must be credited by an equal amount, maintaining the balance in the accounting equation (Assets = Liabilities + Equity). Each `Account` in double-entry financial accounting has a so-called "normal-side" as its account side, winch is deflated to a `DEBIT` and `CREDIT`. For assets and expenses accounts, debit is their normal (default) account side, whereas, for equity and the revenue - credit is the normal side. In addition to accounts, an accounting entry (a change in its monetary value) can be either debit or credit. In this case, such marking indicates in which direction (incremented or decremented) the monetary value of that account must be changed in relation to its normal account side. In other words, a debit entry is an accounting entry that either increases an asset or expense account or decreases a liability or equity account. A credit entry - is an accounting entry that either decreases an asset or expense account or increases a liability or equity account.

ACCOUNT CLASS	INCREASES BALANCE	(DECREASES BALANCE)
Assets	DEBIT	(CREDIT)
Liabilities	CREDIT	(DEBIT)
Equity	CREDIT	(DEBIT)
Income	CREDIT	(DEBIT)
Expenses	DEBIT	(CREDIT)

Table 6.1: Debits and Credits Balances.

Account Class

Financial accounting reflects economic activity of a business. Its users must be able to compare the financial statements of any entity to identify trends in financial position and performance Stolowy & Ding (2019). To enable such comparability, financial accounting employs standardisation of representation of financial information, by structuring consolidation of accounting information into key categories and classes of accounts. These classes are (Stolowy & Ding (2019)).:

1. assets - things of value such as cash, accounts receivable, bank accounts, computers and furniture;
2. liabilities - things you owe including accounts payable notes payable and bank loans;
3. equity - represents your ownership or financial interest in the business;
4. income - monies received for products or services during the course of doing business;
5. expenses - the cost of doing business including office supplies, insurance, rent, payroll expenses and postage etc.

These classes of accounts can be further subdivided into still standardised, but more business specific categories of accounts Stolowy & Ding (2019) that will be outlined in the next part 6.1.1.

In this Thesis, a for`enum AccountClass` was designed to represent and contain the following (see left of Fig. 6.2):

1. `ASSETS` - set-up as:

   (a) `"Assets"` - class name;
   (b) `AccountSide.DEBIT,` - normal side;
   (c) `EnumSet.range(AccountCategory.CASH, AccountCategory.OTHER_ASSETS)` - constituent account categories;
   (d) `"1"` - where `AccountID` starts from 1 (masking: 1*****-**);
   (e) `"Total assets")` - name of its totalling account.
2. `LIABILITIES` - set-up as:
   (a) `"Liabilities"` - class name;
   (b) `AccountSide.CREDIT,` - normal side;
   (c) `EnumSet.range(AccountCategory.AP, AccountCategory.OTHER_LIABILITIES)` - constituent account categories;
   (d) `"2"` - where `AccountID` starts from 2 (masking: 2*****-**);
   (e) `"Total liabilities")` - name of its totalling account.
3. `EQUITY` - set-up as:
   (a) `"Equity"` - class name;
   (b) `AccountSide.CREDIT,` - normal side;
   (c) `EnumSet.range(AccountCategory.EQUITY_DNT_CLOSE, AccountCategory.EQUITY_RE)` - constituent account categories;
   (d) `"3"` - where `AccountID` starts from 3 (masking: 3*****-**);
   (e) `"Total equity attributable to shareholders")` - name of its totalling account.
4. `INCOME` - set-up as:
   (a) `"Income"` - class name;
   (b) `AccountSide.CREDIT,` - normal side;
   (c) `EnumSet.of(AccountCategory.INCOME)` - constituent account categories;
   (d) `"4"` - where `AccountID` starts from 4 (masking: 4*****-**);
   (e) `"Total income")` - name of its totalling account.
5. `EXPENSES` - set-up as:
   (a) `"Expenses"` - class name;
   (b) `AccountSide.DEBIT,` - normal side;
   (c) `EnumSet.of(AccountCategory.COST_OF_SALES, AccountCategory.OTHER_EXPENSES)` - constituent account categories;
   (d) `"5"` - where `AccountID` starts from 5 (masking: 5*****-**);
   (e) `"Total expenses"` - name of its totalling account.

Account Category.

To further adapt traditional accounting principles to the BFS implementation, the implementation of further categorisation for classes of accounts is necessary to account for expected granularity of hierarchical structuring in accounting Stolowy & Ding (2019). To accomplish this, the BFS prototype is designed so that each `Account` is linked to a use-case specific, but standardised `AccountCategory`. This category, in turn, is consolidated further into a standardised `AccountClass`es, such as `Assets`, `Liabilities`, `Equity`, `Income` or `Expenses`, outlined in the previous part of this section. Such categorisation is required to serve as a structure for representation of financial information for the purposes of reporting, analysis and decision-making. Furthermore, the presented structure and ordering of these categories for accounts will construct a hierarchical order for all `Accounts` in the Trial Balance and the financial statements reports. The ordering of `AccountCategory`is designed as a `enum` class (see right of Fig. 6.2) and it contains:

1. `CASH("Cash", AccountSide.DEBIT,"Total cash")`
2. `AR("Accounts Receivable", AccountSide.DEBIT ,`
   `"Total accounts receivables")`

3. `INVENTORY("Inventory", AccountSide.DEBIT, "Total inventory")`
4. `OTHER_CURRENT_ASSETS("Other Current Assets", AccountSide.DEBIT, "Total other current assets")`
5. `FIXED_ASSETS("Fixed Assets", AccountSide.DEBIT, "Total fixed assets")`
6. `COUNTER_ASSETS("Counter Assets", AccountSide.CREDIT, "Total counter assets")`
7. `OTHER_ASSETS("Other Assets", AccountSide.DEBIT, "Total other assets")`
8. `AP("Accounts Payable", AccountSide.CREDIT, "Total accounts payable")`
9. `OTHER_CURRENT_LIABILITIES("Other Current Liabilities", AccountSide.CREDIT, "Total other current liabilities")`
10. `LONG_TERM_LIABILITIES("Long Term Liabilities",  AccountSide.CREDIT, "Total long term liabilities")`
11. `OTHER_LIABILITIES("Other Liabilities", AccountSide.CREDIT,, "Total other liabilities")`
12. `EQUITY_DNT_CLOSE("Equity - doesn't close", AccountSide.CREDIT, "Total equity does not close")`
13. `EQUITY_GETTS_CLOSED("Equity - gets closed", AccountSide.CREDIT,, "Total equity gets closed")`
14. `EQUITY_RE("Equity - Retained Earnings", AccountSide.CREDIT, "Total equity retained earnings")`
15. `INCOME("Income", AccountSide.CREDIT, "Total income")`
16. `COST_OF_SALES("Cost of Sales", AccountSide.DEBIT, "Total cost of sales")`
17. `OTHER_EXPENSES("Other Expenses", AccountSide.DEBIT, "Total other expenses")`

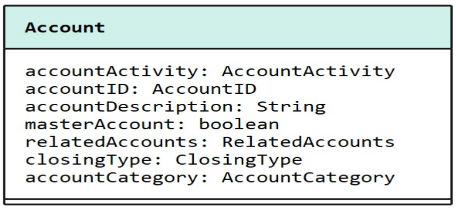
In the research contained in this Thesis, the ordering of `AccountCategory` is designed as a `enum` class (see right of Fig. 6.2) and is self-explanatory. The categories of accounts from the right of Fig. 6.2, are aggregated into standardised classes of financial accounts that will be further aggregated into financial reports (financial statements) described in the following part (see 6.1.3). Each newly created Account will contain a local field to represent its `AccountCategory`, with the description of its utilisation illustrated in Fig. 6.3, Fig. 6.4 and Fig. 6.10.

Account

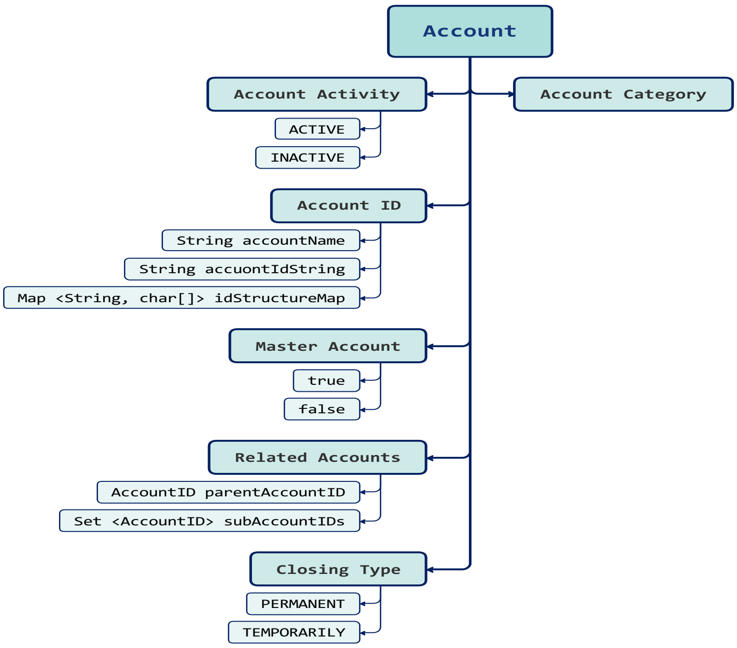
A Financial record's account is a fundamental concept of the financial accounting framework, where it is employed to systematically record and report on the evolution of the economic variables, primarily expressed in monetary units (Stolowy & Ding (2019)). It serves as a structured approach to classify and summarize financial information resulting from business transactions. Each account signifies uniform changes in liquidity associated with a specific category or element of a business's operations. This methodical approach to accounting is pivotal in ensuring the precision, uniformity and transparency of financial reporting. Building on these principles, in the BFS, the `Account` is designed as an identifier for homogeneous units of monetary value. These could be aggregated either into unique and desecrate categories (a child `Account`), or accumulated into a master-category (or a class) of homogeneous `Accounts` (a master `Account`), to communicate consolidated 'ending balance' of its constituencies. (see Section 5.3.1).

In accordance with the double-entry accounting framework, every completed business transaction is assigned to at least two `Accounts` - with a debit and a credit entry (see part 6.1.1). In the BFS, the `Account` class contains (see Fig. 6.3):

1. `AccountActivity accountActivity` - a `enum` class categorising state of the `Account`, i.e., if this `Account` is utilised during the latest accounting period's recording and reporting process.
   (a) `ACTIVE` - is utilised within current accounting cycle;
   (b) `INACTIVE` - is not utilised within current accounting cycle;
2. `AccountID accountID` is a hierarchical account number for COA, General Journal, General



(a) UML class diagram for Account class.



(b) Account class instance value mapping.
This figure represents visually logical assignment of values to the instances of the Account class.

Figure 6.3: Account class.

Ledger, Trial Balance and in final Financial Statements (see 5.3.1). Each `Account` is identified primarily by the `AccountID` number and these numbers are typically organised into logical groups (see 5.3.1);

3. `String accountDescription` - a human readable description of each `Account`, e.g., "Cash", "Accounts Receivables", "Other Liabilities" and so on. Accounts are usually named to reflect the transactions that are assigned to them.
4. `boolean masterAccount` - if set to `true`, is a top account within its hierarchical category, i.e., there is no parent account to this account. For example, in this project, a liability account such as `AccountID` ("2","1000", "-", "00"), "Loans and other borrowings" - is a 'master account as it is the top account within its hierarchical order. Whereas another liability account, such as AccountID("2","1100", "-", "00"), "Loan from Bank of England" - is not a 'master account, as it is a sub-account to the "Loans and other borrowings" account. This means that the closing balance of the "Loan from Bank of England" account forms a part of the closing balance of the "Loans and other borrowings" account at the end of the accounting period. In other words, it is a flag to indicate if this `Account`'s value is a base value (`masterAccount = false`) or its value is calculated by hierarchical accumulation of balances from its child-accounts (`masterAccount = true`).

5. `RelatedAccounts relatedAccounts` (see 5.3.1 and 6.1.1);
6. `ClosingType closingType` - is an `enum` class that is an indicator of how the final/closing balance on a particular `Account` is treated at the end of the accounting period. i.e.,:
   (a) `PERMANENT` - are `Accounts` that are not closed at the end of each accounting period. At the start of the new accounting period, the closing balance from the prior accounting period is brought forward and becomes the new opening balance on the same account;
   (b) `TEMPORARILY` - are `Accounts` that are only last for an accounting period. At the end of the accounting period, the balance is transferred to its master account and the account is closed with a zero balance. At the start of a new accounting period, an account will be opened only if there are transactions relating to that account.
7. `AccountCategory accountCategory` -see 5.3.1.

The `Account` class does not store any monetary balances. The monetary changes to its balance that are born from business transactions and the iterative uprating of the final balances for each of the `Accounts` are stored in the financial journals and reports, outlined in the Section entitled 6.1.2.

Hierarchical Consolidation of Accounts

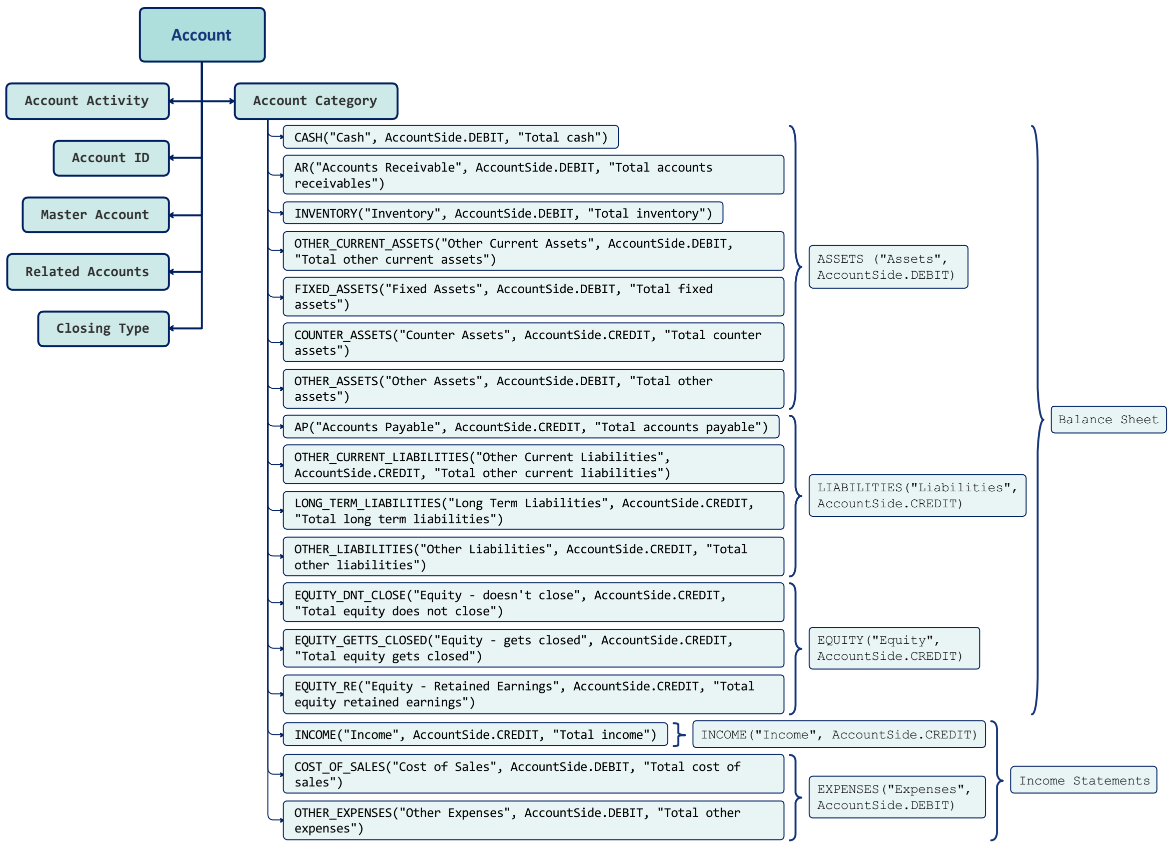


Figure 6.4: Consolidation of an Account into a Report.

Using the `AccountCategory` variable within each of the `Account` of the BFS, enables automation for the grouping of `Accounts`, based on the accounting principles outlined in an earlier section. Such implementation enables consolidation paths for the economic values into relevant financial reports. Fig. 6.4 illustrates such a grouping of `AccountCategory` for the `Account` into an element of `AccountClass` and further into the final `BalanceSheet` or `IncomeStatement`, which will described later in Section 6.1.3. Fig. 6.4 of this report visually presents the accounting logic for accounting data consolidation process (see the right side of Fig. 6.4). As one can see, `enum AccountClass` together with `enum AccountCategory` classes are used to consolidate relevant `Accounts`. Each of these `Accounts` is extracted from the `COA` repository based on their `AccountID`.

Related Accounts

The Concept of "related accounts" can be best understood through the lens of the hierarchical structure of relationships between parent (primary or master) accounts and child (sub or subsidiary) accounts in financial accounting. This structure is used to organise and categorise financial information in a systematic way to ensure consistency of aggregation of multilevel relationships in financial statements. This approach to consolidation of financial variables enables both the overview of the high-level financial position and the detailed breakdown of specific categories. In BFS prototype this concept is implemented by defining:

❖ parent `Account` - is a primary account under which one or more subsidiary (child) accounts may exist. It provides a consolidated summary view of the financial category it represents - the total of ending balances of all its child accounts.

❖ child `Account` - is a sub-account to its parent account. It provides more granular detail about the specific type of financial activity or activities within its parent account category.

In a hierarchy of relationships between `Accounts`, each child `Account` can have only one parent `Account`, whereas any parent `Account` can have one or more child `Accounts`. Furthermore, a child `Account` can act as a parent to its sub-accounts, creating a multi-layered structure. This project defines the `RelatedAccounts` class as a repository for:

1. `AccountID parentAccountID` - the `AccountID` of its direct parent `Account`, if any, otherwise it is null.
2. `Set <AccountID> subAccountIDs` - a hierarchical set of `AccountIDs` of all its child `Accounts`, if any; otherwise it is null.

6.1.2 Accounting Ledger Context

The concept of accounting ledgers represents local BFS repositories to store financial `Accounts`, outlined in the previous concept (Chart of Accounts) and to record chronological changes in the monetary values for these `Accounts` (General Ledger). Furthermore, economic data from each completed business transaction is deposited in the General Journal with corresponding debited and/or a credited entry. And lastly, a ledger for running trial balance is necessary to prepare end of period accounting data for financial reports.

Chart of Accounts (COA)

Chart of Accounts (COA) is a logically organised list of all recognised and authorised `Accounts` used in recording all transactions that the economic entity can engage in (see simplified example in table 5.1). It is a repository, local to the application, fully managed by an individual instance of an `Accountant` or an accounting department (Stolowy & Ding (2019)). This repository is a class variable of the instance of the `Accountant` class, outlined further in this report (see Part 6.1.5)

In this implementation of the BFS, the `COA` class is contained within the `Accountant` class (see below 6.1.5), i.e., it is its class variable. All `Accounts` in COA are unique and are organised hierarchically by the order of the corresponding `AccountID`, i.e., each `Account` is identified and retrieved from the COA by either: the number of `AccountID` and/or its `accountName` (see 5.3.1).

Implementation of the BFS prototype presents a design for the `COA` class to contain (see Fig. 6.5):

1. `AccountIdStructure accountIdStructure` - default structure of the `AccountID` that has been a priori decided upon at instantiation of BFS. Each newly created `Account` implements this ID structure to initiate its `AccountID`;

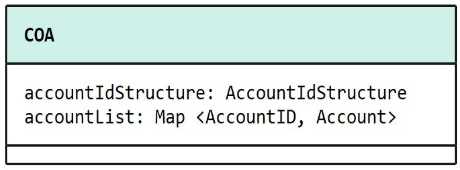


Figure 6.5: Chart of Accounts (COA) class.

2. TreeMap <AccountID, Account> accountList - the COA repository itself is an application local store for all Accounts of this economic entity. This accountList is organised hierarchically by AccountID.

In traditional accounting, the COA and the general ledger (G/L) both commonly contain the same set of Accounts. The difference between the two repositories is the fact that ledger accounts reflect monetary balances, while the chart of accounts does not. Lastly, in the implementation of the BFS prototype, when the role of the accountant is assigned and the instance of one is created (see part 6.1.5), this Accountant's constructor instantiates an empty repository for COA. When creating a new Account, the constructor of such Account automatically adds this newly created Account into the COA::Map <AccountID, Account> accountList repository.

Journals and Ledgers

In traditional accounting, journals and ledgers are day-to day chronological registries of accounting information found in the completed business transactions Stolowy & Ding (2019). This part of the section will outline implementations for the General Journal (GJ) and its entries, General Ledger (GL), its account type and the entries of G/L., the running trial balance, its account type and the entries to one.

1. **General Journal (GJ).**

   Figure 6.6 illustrates the data structures for the GeneralJournal and its JournalEntry classes. As was described in Section 6.1.2, GJ is the first point of entry for the economic data from all the completed business transactions of an entity, often within the current accounting period (Stolowy & Ding (2019)). The high-level data flow for the journaling process is as follows:

   ❖ When business transaction, reposted in the BusinessTransactionRegister marked as TransactionComplitionState.COMPLETED (see Section 7.3), corresponding "accounting smart contract" (see 6.1.4) is activated.
   ❖ Based on the pre-defined accounting rules for the said contract, the economic variables (variables of monetary value) are extracted from such transaction.
   ❖ These monetary values are mapped to a predefined Account as a debited or a credited entry.

   The detailed data flow for this process will be outlined further in Section 7.3. For the purpose of description of the implementation of accounting journaling, the BFS is designed to: first - instantiate a JournalEntry from the data, extracted from completed business transaction. This instantiation process is designed to encapsulate both traditional accounting methods and the advanced traceability features offered by blockchain integration.

   Initially, a JournalEntry is instantiated from data extracted from a completed business transaction. Traditionally, a journal entry records the debit and credit aspects of a transaction, capturing the financial impact on the accounting ledger. However, in the BFS application, this traditional data structure is significantly enhanced. Building on the capabilities of Corda's Transaction Vault , which facilitates asset traceability and ownership history, the BFS innovatively incorporates blockchain's data provenance features into each journal entry.

In Corda, each transaction and the changes to its states are continuously tracked, allowing for the lineage of an asset to be traced from its creation through all subsequent transactions. Each asset (State) in the Corda framework contains a hash of the transaction that created it, thereby maintaining a secure and verifiable link in the transactional chain.

In the BFS, each instance of the journal entry contains the unique hash of its source business transaction as well as the accounting transaction hash - the hash of accounting smart contract that has been activated to create this journal entry Such enhancement of the data structure for the journal entry captures the essential accounting data and adds an additional layer of blockchain-based traceability and verification. The inclusion of these hashes ensures that the provenance of each record is securely and persistently stored within the General Journal and is passed further into relevant records.

Such enhancement to the data structure of the journal entry within the BFS also facilitates future auditing and verification processes. If additional checks or validation of a financial record are required, the embedded transaction hashes provide a clear and unalterable trail back to the original business transaction used for the reported economic data, together with the accounting transaction governing the classification of this reported economic data. This integration of traditional accounting practices with blockchain's data provenance capabilities demonstrates the innovative approach to the design choices for the components of the BFS, ensuring enhanced transparency, security and reliability in financial reporting.

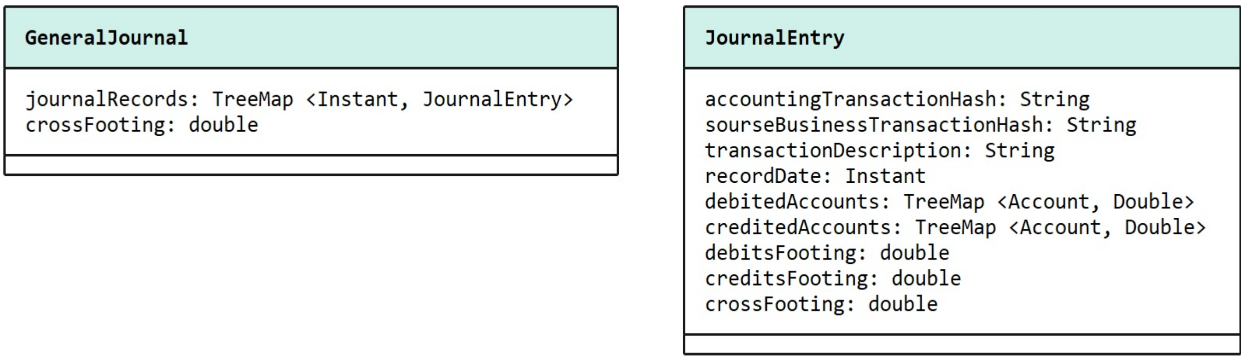


Figure 6.6: General Journal and General Entry classes.

Figure 6.6 illustrates data structure of both: the General Journal and its Journal Entry classes. The description for this data is as follows:

(a) `JournalEntry` - an elemental data structure that contains a monetary unit of value, extracted from business transaction;

  i. `String accountingTransactionHash` - a unique hash of the accounting smart contract that governed the classification of the economic data extracted from the completed business transaction;

  ii. `String sourseBusinessTransactionHash` - a unique hash of the source business transaction;

  iii. `String transactionDescription` - a human readable description extracted from the source business transaction;

  iv. `Instant recordDate` - the time stamp of this entry;

  v. TreeMap <Account, Double> debitedAccounts - the affected accounts, whose values will be debited, irrespective of the normal account side for these accounts (see 6.1.1);

  vi. TreeMap <Account, Double> creditedAccounts - the affected accounts, whose values will be credited, irrespective of the normal account side for these accounts (see 6.1.1);;

vii. `double debitsFooting` - the cross-footing value for all debited accounts;
viii. `double creditsFooting` - the cross-footing value for all credited accounts;
ix. `double crossFooting` - the checksum of both cross-footing values, which must be equal to 0 (zero), otherwise there is an error made in this record.

(b) `GeneralJournal` - the G/J recording only recognises the existence of the elements of the business transaction. This repository is a class variable of the instance of the `Accountant` class, outlined further in this report (see Part 6.1.5). The data structure for this variable is as follows (see Fig. 6.6):

i. `TreeMap <Instant, JournalEntry> journalRecords` - all `JournalEntry` records for current accounting period;
ii. `double crossFooting` - the check sum of all cross-footing values for each `JournalEntry` in the `TreeMap` of `journalRecords` above, which must be equal to 0 (zero), otherwise there is an error made in this record;

The process of transferring entries from the G/J to the G/L is called posting Stolowy & Ding (2019). Each account debited or credited in the journal is posted as debit or credit of the relevant account to the G/L Stolowy & Ding (2019). The subsequent step is therefore is to post the relevant data from the `JournalEntry` into G/L. Traditionally, in manual accounting systems, these entries are posted periodically, with the frequency varied based on the size of the business, the volume of the transactions and the common accounting policies of an organisation.To the contrary, in the automated accounting systems these entries are posted instantly, often in real-time. Following this practice, the process of posting to G/L from the G/J is set to be instantaneous and is described next.

2. **General Ledger (GL).**

Fig. 6.7 illustrates the data structure for the `GeneralLedger` repository, together with its components: the `GeneralLedgerAccount` that is stored in this repository, and the `GL_Entry` - that is a data structure that is a container for relevant data items to represent the change in monetary value for the `GeneralLedgerAccount`. The BFS is designed and implemented in such way that each new `JournalEntry` is transformed into `GL_Entry` and posted into relevant `GeneralLedgerAccount` within `GeneralLedger` repository. Traditionally, only debit or credit value from the `JournalEntry` is posted to relevant `GeneralLedgerAccount`. However, in the BFS application, the existing traditional data structure is significantly enhanced by integrating with innovative concepts of blockchain:

❖ Firstly, utilizing "Corda's Transaction Vault" (Brown (2018)) framework, the BFS embeds a layer of blockchain traceability within each `GL_Entry`. Each of these entries contains a list of unique hashes from all source business transactions relevant to its creation (`List <String> sourceAccountingTransactionHashes`), together with all hashes of accounting smart contracts (`List <String> sourceBusinessTransactionHashes`) that governed the accounting rules for the debit or credit value posted in this `GL_Entry`. This novel feature ensures that the provenance of each financial record is securely and persistently stored, enhancing the integrity and verifiability of BFS accounting records.

❖ Second, variables from both the `GL_Entry` and `GeneralLedgerAccount` are utilised by the `Validator` of the BFS, a key actor within each BFS application, responsible for verifying fund availability by applying "enough funds verification consensus" within the BFS ecosystem (see 6.2.5). This role leverages the variables illustrated in Fig. 6.7, such as: `GL_Entry`::`AccountID glAccountID`, `GL_Entry`:: `Instant entryInstant`, and `GeneralLedgerAccount`::`double currentBalance` to generate `TransactionInputValue` (see 6.2.6), internal to the BFS. The concept for `TransactionInputValue` is based on the UTXO (Unspent Transaction Output) data model of traditional Bitcoin blockchain (Rohrer et al. (2017)) (see 6.2.6), facilitating secure and verified business transactions between counterparties. This process is essential to provide pre-settlement verification of monetary claims, made by the counterparties of the business transactions, ensuring that the offered funds are available.

Figure 6.7 illustrates the data structure of all three components: General Ledger, General Ledger Account and General Ledger Entry classes. The description for this data is as follows:

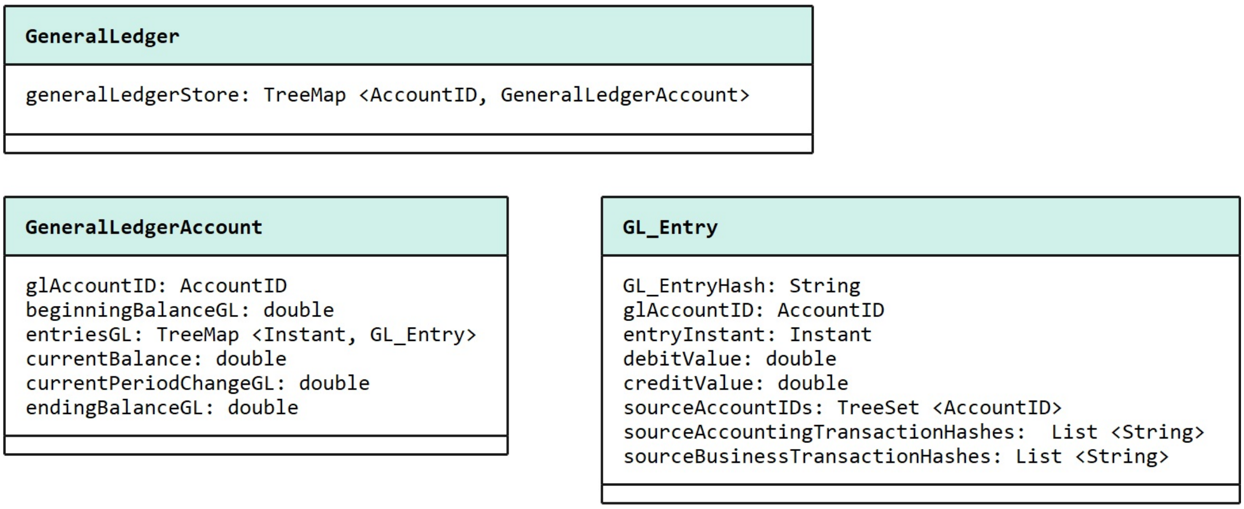


Figure 6.7: General Ledger, General Ledger Account and General Ledger Entry classes.

(a) GL_Entry class - is the value change that posted from the transformed GJ entry. The data structure for this variable is as follows (see Fig. 6.7):

i. String GL_Entryhash - hash of the this entry, providing a unique identifier that ensures the integrity and non-reputability of this value. This variable is also utilised in the TransactionInputValue by the Validator in the implementation of consensus protocol (see 6.2.6).

ii. AccountID glAccountID - the AccountID of the Account, whose value will be impacted by this entry. This variable is also utilised in the TransactionInputValue by the Validator in the implementation of consensus protocol (see Section 6.2.6).

iii. TreeSet <AccountID> sourceAccountIDs - the AccountIDs of the child Accounts, if these exist (see 6.1.1). This is a collection of AccountIDs from child accounts showcasing the hierarchical nature of accounting records. A TreeSet repository type is chosen because it is a type of set in Java that automatically orders its elements in ascending order, which is particularly useful when dealing with Account IDs, while also keeping a unique set of these Account ID, thus preventing duplication;

iv. Instant entryInstant - the timestamp of this entry, which marks the chronological placement of entry within the financial record. This variable is also utilised in the TransactionInputValue by the Validator in the implementation of consensus protocol (see 6.2.6);

v. double debitValue - the debit entry value that was posted from JournalEntry::debitedAccounts 1a, or is assigned to 0 (zero), reflecting the financial impact of the transaction (see second columns of Tab. 5.3 - Tab. 5.5 for visual reference);

vi. double creditValue - the credit entry value that was posted from JournalEntry::debitedAccounts 1a, or is assigned to 0 (zero), reflecting the financial impact of the transaction (see third columns of Tab. 5.3 - Tab. 5.5 for visual reference);

vii. List <String> sourceAccountingTransactionHashes - list of all accounting smart contract hashes, used to generate this entry, including for child accounts if these exist, providing a comprehensive audit trail (see 6.1.1). Repository type `List` is selected, because in the context of accounting transaction hashes, there is no requirement to keep the elements in a specific order or to categorise them. This Java type allows the system to simply store the hashes in the order they are added, without the overhead of maintaining any sort order or categorisation, which is unnecessary for this data;

viii. `List <String> sourceBusinessTransactionHashes` - list of all source business transaction hashes used to generate this entry, including for child accounts if these exist, providing a comprehensive audit trail (see 6.1.1). This variable is also utilised in the `TransactionInputValue` by the `Validator` in the implementation of consensus protocol (see 6.2.6). Repository type `List` is selected, because in the context of business transaction hashes, there is no requirement to keep the elements in a specific order or to categorise them. This Java type allows the system to simply store the hashes in the order they are added, without the overhead of maintaining any sort order or categorisation, which is unnecessary for this data.

(b) `GeneralLedgerAccount` - The account within G/L concept, which corresponds to the `Account` in `COA`, but which contains all relevant monetary data for a particular aspect of a business transaction, such as cash aspect, capital aspect etc Stolowy & Ding (2019). The data structure for this variable is as follows (see Fig. 6.7):

i. `AccountID glAccountID` - the `AccountID` for the `Account` form `COA` repository. The difference in the `Accounts` stored in `COA` and the `GeneralLedgerAccounts` is that the latter do contain monetary values;

ii. `double beginningBalanceGL` - opening (beginning) balance on the `GeneralLedgerAccount` at the beginning of the accounting period;. Depending on the closing type of the account (see part 6 and Fig. 6.3), PERMANENT vs TEMPORARY, this balance is either “Brought Down” from the previous accounting period `endingBalanceGL` for the former, or it is 0 (zero) for the latter;

iii. `TreeMap <Instant, GL_Entry> entriesGL` - the chronologically ordered store of all `GL_Entry`s for this reporting period;

iv. `double currentBalance` - this is so-called running balance of this `GeneralLedgerAccount` (see last columns in Tab. 5.3 - Tab. 5.5 for visual reference). In the automated accounting systems, this balance is also posted to the `RunningTrialBalance` as the “Trial Balance Entry” (see Tab. 5.6). Furthermore, this variable is also utilised in the `TransactionInputValue` by the `Validator` in the implementation of consensus protocol (see 6.2.6) as the value for the transaction input based on the UTXO model of blockchain.

v. `double currentPeriodChangeGL` - this is an optional value that can be utilised by the optional analytics reports. e.g., if one needs a statement for the last month, or the last quarter and so on. This is the monetary change for the selected account from the beginning of pre-specified period, e.g., change from the beginning of the month.

vi. `double endingBalanceGL` - last closing (or ending) balance at the end of current accounting/reporting period. This balance value is posted as the final “Ending Trial Balance” value for each relevant `Account` (see Tab. 5.6).

(c) `GeneralLedger` - is the repository for all `GeneralLedgerAccount`, each of with is a grouping for a particular aspect of a business transaction, such as cash aspect, capital aspect etc. Stolowy & Ding (2019). It is an intermediate database in the traditional accounting process that can provide internal analytics capabilities by the categories of transactions Stolowy & Ding (2019) (see Fig. 6.7). This repository is a class variable of the instance of the `Accountant` class, outlined further in this report (see Part 6.1.5). The description for this data is as follows:

i. `TreeMap <AccountID, GeneralLedgerAccount> generalLedgerStore` -

the application-local store for all `GeneralLedgerAccount`s.

Typically, the `endingBalanceGL` from `GeneralLedgerAccount` are posted to the trial balance at the end of an accounting period in traditional manual accounting. This period can be monthly, quarterly or annually, depending on the business's accounting practices. Posting usually occurs just before the preparation of financial statements. This ensures that the trial balance reflects all the activities of the period and aids in the accuracy of the financial statements.

In automated accounting systems, the posting of ending balances from general ledger accounts can be performed instantaneously directly into "running trial balance", as"Ending Trial Balance" entry for each relevant `Account` (see Tab. 5.6), thus maintaining a continuously accurate trial balance. This is done by automation of calculation of the said ending balance in the general ledger account by automatically summing all debits and credits. This process is also embedded in the design and implementation of the BFS.

3. **Running Trial Balance.**

As a general principle, the trial balance is the list of the debit and credit entries and cross-footings for each account in general ledger (Stolowy & Ding (2019)). The aim for this exercise is to check that the sum of the debit entries (or debit balances) is equal to the sum of all the credit entries (or credit balances), i.e., the verification that the "total debits = total credits" (Stolowy & Ding (2019)). Accounts in the trial balance ledger are nearly always listed in the order of the account ID. The objectives for this ledger are:

- to audit the accounting entries so to reveal potential errors and anomalies by highlighting out-of-balance accounts (Stolowy & Ding (2019));
- to provide a simple determination fro the net profit/loss without construction of balance sheet or income statement (Stolowy & Ding (2019));
- to be used in preparation for creating both adjusting entries and closing entries, as well as other financial statements (Stolowy & Ding (2019)).

In this work, the posting of G/L balances into the running trial balance is an automated step of the accounting process and this step is the preceding step before the generation of the final financial statements and publication of these into the `Block` of the `FilingHistory` blockchain.

This automation is implemented by the following means: when a new `GL_Entry` is added to the `GeneralLedgerAccount` of the `GeneralLedger` repository, as part of the same process flow, a new `TbAccountEntry` is created from the relevant data extracted from `GL_Entry` (see steps 4-5 of the generalised accounting process illustrated in Fig. 2.1). This `TbAccountEntry` (see Fig. 6.8 - the most right class) is added to the chronological map of the `TbAccount::tbAccountEntries` (see Fig. 6.8 - the middle class), thus updating the "Ending Trial Balance" value illustrated in Table 5.6.

Furthermore, in this implementation of the BFS, while a trial balance ledger communicates the "end-balance" value of each financial account, in contrast to`GL_Entry`, the `TbAccountEntry` does not incorporate any cryptographic details of any accounting and business transactions. As was outlined earlier, the goal for the construction of the trial balance ledger is to verify if the double-entry accounting practices were followed correctly and all accounts are balanced. Lastly, although adjustments (adjusting entries) are needed to be made prior to closing the accounting period and publication of the financial statements, this implementation of the BFS omits this step of traditional accounting. The reason for such a decision is two-fold: first, as the use-case based implementation only considers 2 business transactions, the accounting data from these transactions does not include any data relevant to generate "adjusting entries" of the final Trial Balance; second, such a detailed implementation is not consequential for the illustrative nature of the demonstration of the BFS's utility.

Figure 6.8 illustrates the data structure of all three components relevant to the trial balance: Running Trial Balance, Trial Balance Account and Trial Balance Entry classes. The description for this data is as follows:

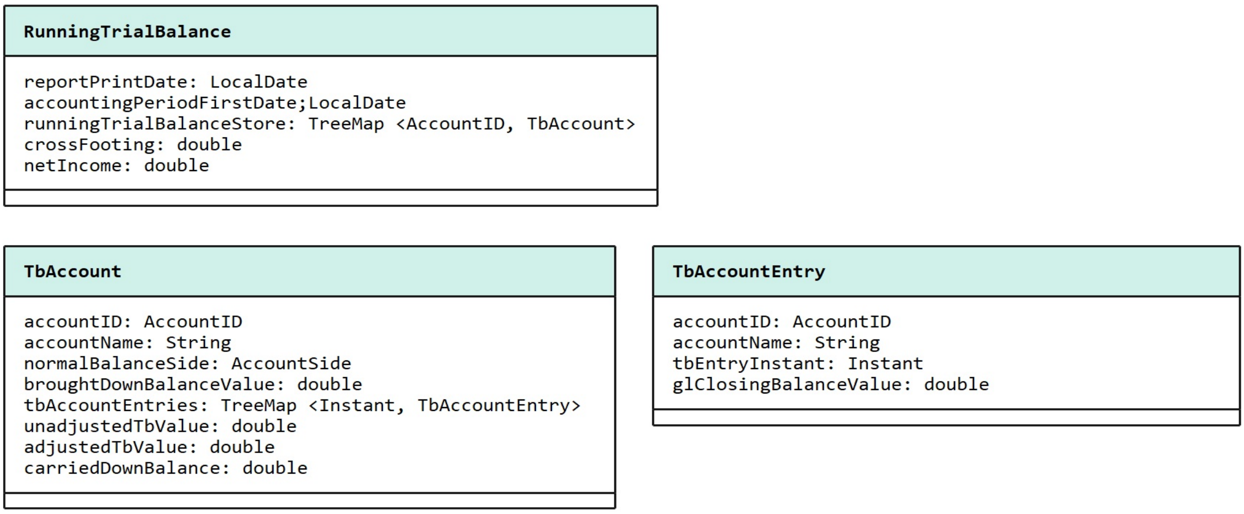


Figure 6.8: Running Trial Balance, Trial Balance Account and Trial Balance Entry classes.

(a) TbAccountEntry - captures the consolidated financial impact derived from the relevant GL_Entry. The data structure for this variable is as follows (see Fig. 6.8 - the right class):
   i. AccountID accountID - the AccountID for the corresponding Account form COA repository;
   ii. String accountName - the name of the Account for the corresponding AccountID form COA repository;
   iii. Instant tbEntryInstant - the time stamp of this TbAccountEntry;
   iv. double glClosingBalanceValue - the endingBalanceGL extracted from GeneralLedgerAccount.

(b) TbAccount - the account within running trial balance concept that corresponds to the Account in COA, but which does contain relevant monetary data for a particular aspect of a business transaction such as cash aspect, capital aspect etc. Stolowy & Ding (2019). The data structure for this variable is as follow (see Fig. 6.8 - centre class):
   i. AccountID accountID - the AccountID for the Account form COA repository.
   ii. String accountName - the name of the Account for this AccountID form COA repository;
   iii. AccountSide normalBalanceSide - the representation of the default accounting setting for this entry (debit or credit (see Section 6.1.1));
   iv. double broughtDownBalanceValue - the opening (starting) value for this TbAccount. This is the opening balance of a ledger account brought into the books from a previous accounting period and is made at the beginning of the current accounting period;
   v. TreeMap <Instant, TbAccountEntry> tbAccountEntries - the chronologically ordered repository of all TbAccountEntrys for current reporting period;
   vi. double unadjustedTbValue - the updated value of for this TbAccount, after each new TbAccountEntry is added to the tbAccountEntries. This value is used in the generation and the publication of the final financial statements accounts at the end of the reporting period.
   vii. double carriedDownBalance - the closing value for this TbAccount. This is the post -reporting closing balance of a ledger account carried forward to the next

accounting period and is the last posting (balancing) to the trial balance and general ledger done at the the end of the current accounting period. This value will become the new `broughtDownBalanceValue` for the same `TbAccount` at the beginning of the next accounting period.

(c) `RunningTrialBalance` - this is the repository for all `TbAccounts`, each of which is a grouping for a particular aspect of financial accounts. It is a final database of the steps for the traditional accounting process (Stolowy & Ding (2019)) (see steps 5 of the generalised accounting process illustrated in Fig. 2.1).This repository is a class variable of the instance of the `Accountant` class, outlined further in this report (see Part 6.1.5). The data structure for this variable is as follows (see Fig. 6.8):

  i. `LocalDate reportPrintDate` - the date of the generation of this report, outlining all economic activity since the beginning of the accounting period;

  ii. `LocalDate accountingPeriodFirstDate` - a set date for the beginning of the current accounting period;

  iii. `TreeMap <AccountID, TbAccount> runningTrialBalanceStore` - the application-local repository for all `TbAccounts`;

  iv. `double crossFooting` - the check sum of all cross-footing values for each `TbAccountEntry` for each of the `TbAccount` in the `TreeMap` of `runningTrialBalanceStore` above, which must be equal to 0 (zero), otherwise there is an error made and intervention is required to track and correct such reporting error;

  v. `double netIncome` - the simple representation for the net profit/loss that can be communicated to the relevant stakeholders, prior and without construction of balance sheet or income statement. This value can be tracked instantly to the ongoing business activity of an economic entity, providing on-demand analytics.

6.1.3 Financial Reports Context

In the landscape of financial reporting, the BFS presents a novel solution that integrates blockchain technology with traditional accounting process. Traditionally, at the end of the accounting period and as the culmination of the accounting cycle, financial statements are generated. This is the primary focus of this context, where the implementation of the data structures and hierarchical relationships govern the processes for generation of financial statements - such as a Balance Sheet. This selective implementation, driven by the illustrative nature of specific business transactions, underscores the BFS's capability for instantaneous verification of financial data. The accounting period, for illustrative purposes, spans the interval between successive business transactions, showcasing the BFS's innovative approach to real-time financial reporting.

By implementing domain-specific complexities, data relationships and representation requirements, this part of the design for the BFS illustrates its ability to provide accurate, real-time financial reporting, crucial for informed decision-making within the agility of the modern economy. This section describes the data structures that underpin and govern the Balance Sheet's generation within the BFS, highlighting the interconnectedness of the Account Hierarchy Accounting Process and Blockchain contexts. Through an exploration of Java class implementations and the broader data flow model, this part illustrates architectural connectivity of BFS elements, designed to ensure clarity, modularity and adaptability in financial reporting for the digital age.

Financial Statements

The establishment of financial statements is the final stage of the accounting process (Stolowy & Ding (2019)) traditionally known as "end-of-period entries" (Stolowy & Ding (2019)). These statements are reports that consist of the:

- ❖ Statements of Financial Positions as at the end of the period (Stolowy & Ding (2019)) - or a Balance Sheet;
- ❖ Statement of Profit or Loss and other Comprehensive Income for the period (Stolowy & Ding (2019)) - or the Income Statement;
- ❖ Statement of Changes in Equity for the period (Stolowy & Ding (2019));
- ❖ Statement of Cash Flow for the period (Stolowy & Ding (2019));
- ❖ Notes, comparing summary of significant accounting policies and other explanatory information (Stolowy & Ding (2019)), others that are specific to the business entity.

For the purposes of this Thesis, adaptation is necessary for such specifications for financial accounting reports. It is done by providing design and implementation of the data structure and the related processes for the contraction and publication of the financial statements. As the BFS focuses on the specific use-case to demonstrate the implementation of the designed BFS, the decision is made to only demonstrate the implementation of the Balance Sheet, leaving the other financial statements implementation for the future research effort. This decision is further underpinned by the illustrative nature of only two business transactions.

Furthermore, the accounting period in this Thesis, for illustrative purposes, is defined as the period between the previous and the current business transaction. Such implementation for the BFS represent real-time accounting process that demonstrates required instantaneous auditing and financial data verification capability - the goal for this project. To illustrate the implementation of construction of the Balance Sheet that reports of financial position of an economic entity, at the end of the accounting period the decision is made to run a full accounting cycle after each business transaction. This way, two separate balance sheet reports are generated, enabling comparison between the accounting periods and evaluation of the financial "health" of the business entity after each economic interaction with its counterparties.

Figure 6.9: Financial Statements, Report, Report Entry classes, and Report Type enum.

Fig. 6.9 illustrates primary elements of the implementation, instrumental in preparation and presentation of financial statements. These are: `ReportEntry`, `Report`, `enum ReportType` and final `FinancialSatements`. Generation of the financial statements is step 6 of the accounting cycle illustrated in Fig. 2.1. The high-level process is as follows: at the end of the reporting period, after all necessary validations of the trial balance and subsequent end-of-period adjustments have been made (see Section 6.1.2), the final `endingBalanceGL` form G/L is used to instantiate new `ReportEntry` for each of the relevant `AccountID`.

In the BFS, the enum type is utilised to categorise various types of reports and financial statements, ensuring that each report entry aligns with its specific report type. This is particularly evident in how the relationship between `ReportEntry`, `Report`, `enum ReportType` and `FinancialSatements` is structured. Each of these classes plays a role in consolidating financial data from discrete `Accounts` into the final financial statements, and the enum type ensures correct categorisation and flow of information. Each enum value corresponds to a specific type of financial statement that is produced from the aggregation of account data. For example, `ReportType.BALANCE_SHEET` indicates that the associated `ReportEntry` or Report is related to a balance sheet report. The specific example of the data flow, that is implemented for the consolidation and grouping of discreet `Accounts` onto the final financial statements is described further in the next Section 6.1.3.

It is important to highlight here the unique and novel data feature of the `ReportEntry` within the BFS. In contrast to traditional accounting practices, where the only final balances are published in the accounts within financial reports, the design and implementation of such statements is integrated with blockchain's data provenance capabilities. This is another instance of adaptation of Corda's Transaction Vault framework (Brown (2018)) that allows for the continuous tracking of each transaction, facilitating additional layer of traceability and verification of each record published.

In the BFS, each instance of the `ReportEntry` for the financial statements contains a full and final map of all `transactionsAndParticipants` that were instrumental in the evolution of the value for the `endingBalanceGL` of relevant financial account from the begging of the accounting period. Each element of the `transactionsAndParticipants` map within the `ReportEntry` contains a `sourseBusinessTransactionHash` together with `PublicKeys` of all counterparties to that source business transaction. By embedding these unique cryptographic transaction hashes with public keys of counterparties within report entries, the BFS ensures that the origin and authenticity of every financial account can be securely traced back to its source, significantly enhancing the transparency, security and reliability of published reports. This innovative feature makes accounting falsification challenging, providing regulators with an additional data authenticity tool within the domain of financial accounting and potentially setting a new standard for auditability and verifiability in the digital age. The detailed description of primary field included in the reporting context of the financial statements classes of the BFS is as follows:

1. `enum ReportType` - an initial categorisation mechanism for generation of financial statement reports. This enumerator (see bottom element of Fig. 6.9) serves as a final consolidation tool for hierarchical grouping of accounts illustrated in the right of Fig. 6.4 of Section 6.1.1. Each element of this enumerator also marks the type of the financial report added to the final reporting package of the `TreeMap <ReportType, Report> financialStatements` (see top left element of Fig. 6.9). The data structure of `enum ReportType` is as follows:
   (a) `BALANCE_SHEET` - a balance sheet report type, which is set-up as:
      i. `("Statement of financial position",` - the published name of the Report;
      ii. `EnumSet.of` - set of `(AccountClass`s consolidated in this report:
         A. `(AccountClass.ASSETS,`
         B. `AccountClass.LIABILITIES,`
         C. `AccountClass.EQUITY))`
   (b) `INCOME_STATEMENT` - an income statement report type, as a place holder for future implementation (out of scope of current project). It is that is set-up as:
      i. `("Statement of comprehensive income",`- the published name of the Report;
      ii. `EnumSet.of` - set of `(AccountClass`s consolidated in this report:
         A. `(AccountClass.INCOME,`
         B. `AccountClass.EXPENSES))`
   (c) `CASH_FLOW ("Cash Flow") -` the cash flow report type, as a place holder for future

implementation (out of scope of current project);
   (d) `CHANGES_IN_EQUITY("Changes in Equity")` - the changes in equity report type, as a place holder for future implementation (out of scope of current project);
   (e) `NOTES("Notes")` - the notes report type, as a place holder for future implementation (out of scope of current project);
   (f) `RETAINED_EARNINGS("Retained Earnngs")` - the retained earnings report type, as a place holder for future implementation (out of scope of current project);

   Each of these elements, amongst other fields, contains a `EnumSet <AccountClass> accountClass` that provides an interface between the context for "Account and its Hierarchy" (see Section 6.1.1) and the context of "Financial Reports" (see 6.1.3). The data flow illustrating these is communicated further in Section 6.1.3 and Fig. 6.10.
2. `ReportEntry` - an elemental data structure that represents the final `endingBalanceGL` extracted form G/L (see top right element of Fig. 6.9);
   (a) `AccountID accountID` - the ID of the corresponding account published in the final reports;
   (b) `double currentPeriodClosingValue` - the final `endingBalanceGL` extracted form G/L
   (c) `Instant entryTimeStamp` - the time stamp of this entry;
   (d) `HashMap<String, List<PublicKey>> transactionsAndParticipants` - is a map of cryptographically secured elements, each containing a `sourseBusinessTransactionHash` together with `PublicKeys` of all counterparties to that source business transaction.
3. `Report` - an integration of the traditionally defined financial report, such as a "Balance Sheet" within the BFS. The data elements for this class are illustrated in the top middle element of Fig. 6.9. The data structure for this class is as follows:
   (a) `String companyName` - an official name of the economic entity whose economic activity is represented in his report;
   (b) `LocalDate reportDate` - the date of publication of the report;
   (c) `Instant reportTimeStamp` - the time stamp of the generation of the report;
   (d) `ReportType reportType` - the type of this report, e.g. as a `BALANCE_SHEET`;
   (e) `String reportName` - the name extracted from the `reportType`;
   (f) `EnumMap <AccountClass, TreeMap <AccountID, ReportEntry>> reportParts` - the grouped consolidation of the accounts published in this report, e.g., for the `AccountClass.ASSETS` grouping, the hierarchical map of all `ReportEntry`s will consist of all relevant records, including a `sourceBusinessTransactionHash` together with `PublicKeys` of all counterparties to that source business transaction.
4. `FinancialSatements` - the final packaging for all reports for this accounting period. This object is the one that will be subsequently added as the data element to he `Block` of the `FIlinghistory` blockchain, illustrated in step 7 of the accounting cycle (see Fig. 2.1). The data structure for this class is as follows:
   (a) `LocalDate financialSatementsDate` - the date of publication of the final statements;
   (b) `Instant financialSatementsTimeStamp` - the time stamp of the generation of the final statements;
   (c) `TreeMap <ReportType, Report> financialStatements` - the map of all financial reports included in this publication;
   (d) `String statementHash` - the cryptographic hash of the finalised package. This hash is generated to provide verification of integrity of the data within this reporting package. This is another novel integration added to complement traditional accounting practices. By including this hash, an extra layer of validation is introduced, ensuring that the data in the financial statements remains unchanged after being finalised and published on the BFS blockchain.

The next important step of this report is to illustrate how the data communication is designed between the contexts of the "Journals and Ledgers", "Account and its Hierarchy", "Financial Reporting" and "Blockchain" (see 4.2.2). Such data flow is illustrated in the next Section of this report, and will explore the hierarchical data structure designed for accounting information consolidation within the BFS, emphasising the importance of this data flow in ensuring the accuracy and integrity of financial reporting.

Account to Financial Statements Data Flow

Fig. 6.10 illustrates the process of data communication across several contexts within the BFS architecture, namely “Journals and Ledgers”, “Account and its Hierarchy”, “Financial Reporting” and “Blockchain”. This data flow is essential for understanding of how the BFS follows traditional accounting practices. It demonstrates the logical path of data that underpins the generation of a financial statement such as “Balance Sheet”.

The data flow model demonstrates how the hierarchical data structure for the consolidation of accounting information is designed within the BFS and further reinforces how the data path is designed for the BFS accounting flow. By incorporating traditional accounting practices within the design and implementation of the BFS, the `COA` repository houses `Accounts` that do not contain monetary units. This data flow is further focused on showcasing how this fundamental financial `Accounts`, that do not contain monetary units are processed to derive such values from the ending balance of the `GeneralLedgerAccount` of the G/L for corresponding `AccountIDs`, together with cryptographic transaction hashes and public keys of counterparties, ensuring the origin and authenticity of every financial account.

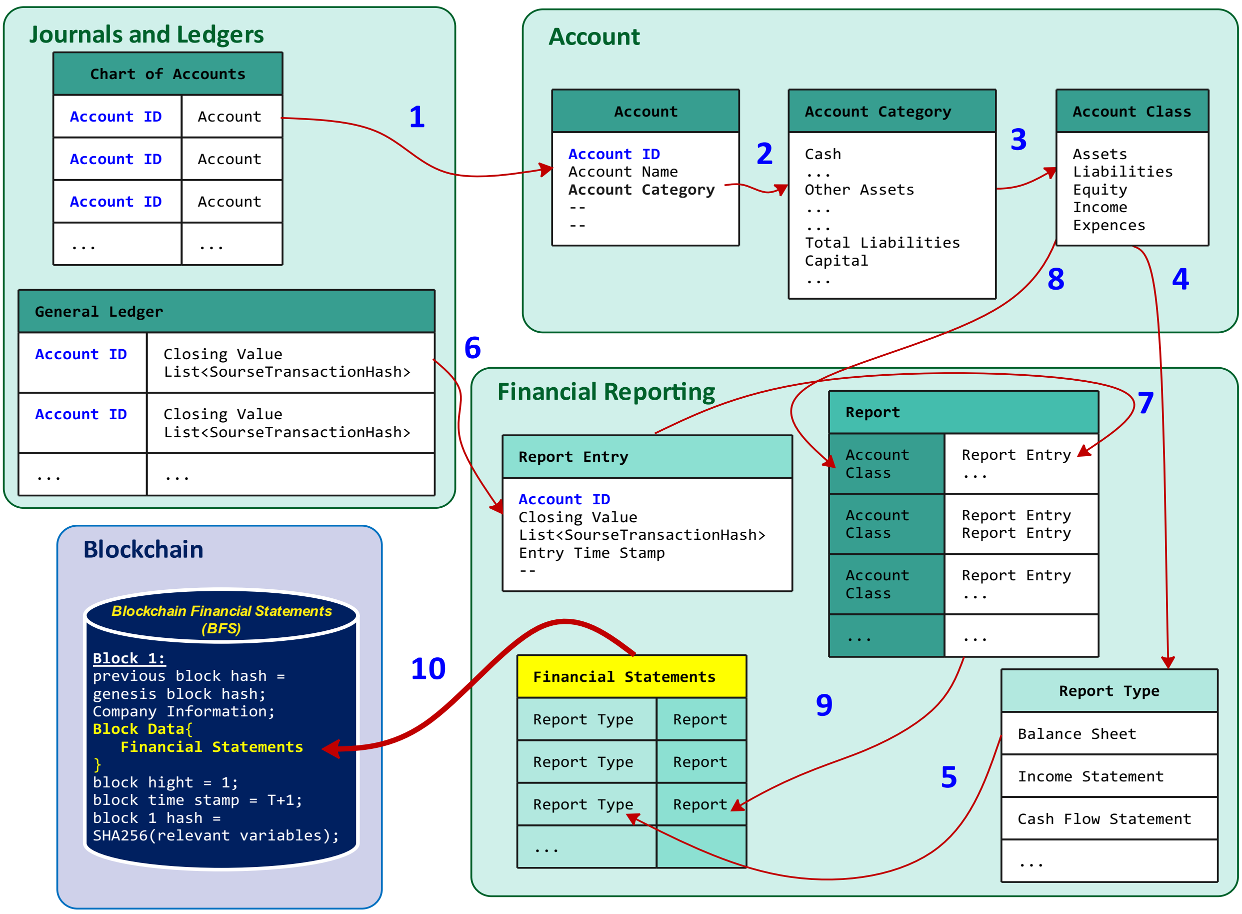


Figure 6.10: Account to Financial Statements Data Flow.
Different colours of the rectangles represent corresponding sub-domains of the BFS architecture, where: the accounting subdomain is associated with the turquoise colour. The blockchain sub-domain is associated with the lighter blue colour. Yellow heading of the Financial Statement class maps in the step 10 with its location within BFS block.

The data flow illustrated in Fig. 6.10 is initiated, e.g., at the end of the reporting period, on-demand, or instantaneously, as part of the process for generation of the financial reports. The steps of the data flow path that as illustrated in Fig. 6.10 are as follows:

1. Utilising `AccountID`, the `Account` is retrieved from the `COA` repository;
2. Every `Account` (see 6.1.1) is assigned a field for `AccountCategory` at the instantiation of such `Account` (see 6.1.1). This field for `AccountCategory` is mapped to a particular accounting category, such as "Cash" (see 6.1.1);
3. This field for `AccountCategory` is then consolidated within the `AccountClass` (see 6.1.1) into a group, e.g.. "Assets";
4. This group is then further consolidated by the `ReportType` into a grouping for the financial report such as balance sheet;
5. This grouping of the `ReportType` with govern mapping hierarchy within the final financial statements for each `Report` when generated;
6. Concurrently, the final closing value, together with relevant provenance data from the G/L is instantiated into new `ReportEnrty` for each `AccountID` published in the financial statements;
7. These`ReportEnrty`s are collected within the relevant `Report`s, where these entries are hierarchically arranged based on their `AccountID`, providing structural integrity of the traditional financial statement;
8. Next, in the final `Report`, the `AccountClass` grouping is employed once more to logically consolidate relevant `ReportEnrty`s.
9. As the final step in the generation of `FinancialStatements`, the mapping between each `ReportType` and a corresponding `Report` is completed.
10. Lastly, agreed and hashed `FinancialStatements` are published as par of the data structure within the `Block` of the BFS.

6.1.4 Accounting Transaction

Integrating the context of *Accounting Transactions* (see 4.2.2) with the context *Smart Contract* of blockchain (see 4.2.2) within the BFS represents a novel approach to automation of financial transactions and record-keeping, leveraging blockchain-enabled framework to enhance accuracy, transparency and efficiency in financial reporting. This project designs and implements *Accounting Smart Contracts* that govern such automation of traditional and established accounting rules to manage financial data drawn out from economic activities, such as completed business deals. This allows for the subsequent valuation of these financial data within the accounting framework (Stolowy & Ding (2019)).

In the context of *Accounting Smart Contracts*, the generalised structure for the implementation of classes and their relationships, illustrated in Fig. 6.11 serves as a blueprint for these smart contracts. This blueprint can become a foundational model for guiding the future development and implementation of a wide variety of specifications for business model dependant accounting rules, ensuring that the system can be adapted and expanded to meet diverse accounting needs. Application of the framework of *Accounting Smart Contracts* will ensure that financial transactions are processed and recorded with enhanced reliability and clarity, building the way for verifiable and accurate financial management within blockchain platforms.

In a generalised context of accounting and business, the type of economic entity often dictates the types of transactions (or types economic events) that the business is allowed to undertake (Stolowy & Ding (2019)). In this Thesis, the primary component designed to govern the overall control and sequencing of the necessary processing for the `AccountingTransaction` class described later in this chapter. The generalised structure for the `AllowableAccountingTransaction` class is illustrated in Fig. 6.11 and it defines transactions through economic events and their impacts on accounts. This class is uniquely defined and instantiated by the `Accountant`, to incorporate a specified type of economic event, the `EconomicEvent` enumerator that will impact the monetary values of a set of specified financial accounts, the `AffectedAccounts` class.

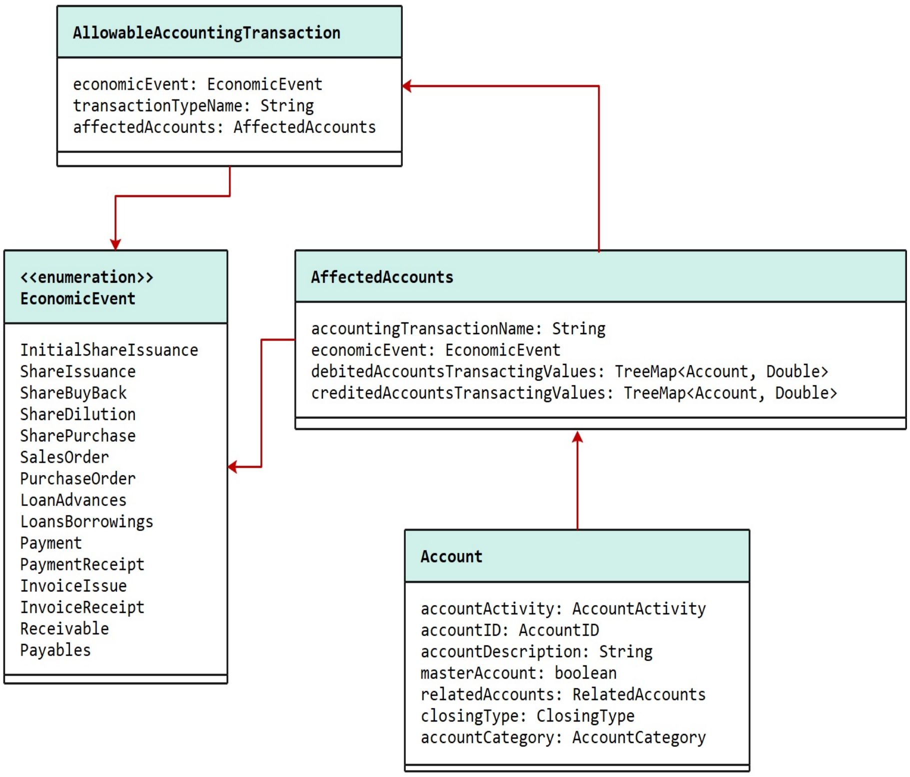


Figure 6.11: Allowable Accounting Transaction Framework

In the use-case of this Thesis, the business type of the primary entity and the 2 overall business transactions implemented define specifications for the design and implementation of accounting smart contracts. In this use-case:

- the APF is an economic entity, whose business activity is "central banking";
- APF's economic activity is illustrated by 2 business transactions;
- these two business transactions will specify a unique list of economic events captured within the `EconomicEvent` enumerator.
- each specific value of the `EconomicEvent` is associated with the aspects of the APF's business and corresponds with accounting.

Fig. 6.11 illustrates primary components of the implementation for "Accounting Smart Contract". This structure illustrates how transactions are tied to specific economic events, affect various accounts and how those accounts are detailed and categorised within the system. The implementation of the data structure and their relationships for the primary components in this figure are described in more detail next:

1. `AllowableAccountingTransaction` is linked to both `EconomicEvent` and `AffectedAccounts`, indicating that an accounting transaction is defined by an economic event and affects multiple accounts;
   (a) `economicEvent` refers to the specific economic event associated with the transaction, such as a share issuance, payment or invoice.
   (b) `transactionTypeName` is a string attribute that provides additional information about the transaction.
   (c) `affectedAccounts` represents the accounts affected by the transaction.
2. `EconomicEvent` is detailed with its enumeration values, including types of events such as `InitialShareIssuance`, `ShareBuyBack` and so on.
   Its values are used by both `AllowableAccountingTransaction`, and `AffectedAccounts` to

specify the type of event.

3. `AffectedAccounts` manages the details of debit and credit transactions involving multiple `Account` entities (complying with double-entry accounting), and tying these back to an `EconomicEvent`.
   (a) `accountingTransactionName` is a string attribute that describes the accounting transaction.
   (b) `economicEvent` specifies the economic event associated with the transaction defined in the `EconomicEvent` enumeration.
   (c) `debitedAccountsTransactingValues` and `creditedAccountsTransactingValues` are `TreeMap` data structures that track the debited and credited accounts and their corresponding transaction values.
4. `Account` encapsulates the details of individual accounts, including their activity, category and relation to other accounts.
   (a) `accountActivity` describes the activity or purpose of the account.
   (b) `accountID` is a unique identifier for the account.
   (c) `accountDescription` provides a textual description of the account.
   (d) `accountCategory` categorises the account (e.g., assets, liabilities, equity).
   (e) `masterAccount` is a boolean attribute that indicates whether the account is a master account.
   (f) `relatedAccounts` represents relationships between accounts.
   (g) `closingType` specifies the type of account closure if applicable.

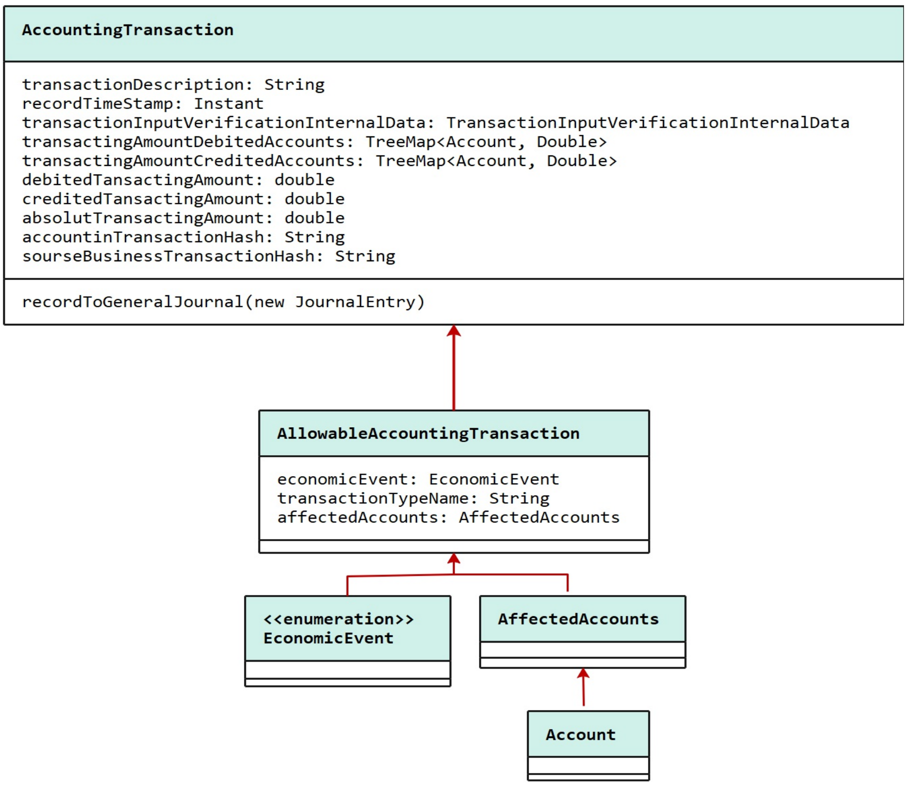


Figure 6.12: Accounting Transaction class

This report now describes how the implementation of the `AllowableAccountingTransaction` and its components utilised for governance and execution of the `AccountingTransaction` class. The `AccountingTransaction` - is the key component of the accounting smart contract that encapsulates the main data and processing logic for the overall process for generalised Accounting

Smart Contract. To create an instance of the AccountingTransaction class, the Accountant passes preset AllowableAccountingTransaction as a parameter to its constructor, together with sourseBusinessTransactionHash, transactionDescription, transactionCompletionTimeStamp, absolutTransactingAmount obtained from a completed BusinessTrtansaction.

AllowableAccountingTransaction class plays a crucial role in defining how AccountingTransaction class is created and processed. It defines the allowable parameters and rules for the specific type of financial transaction, including details of the economic event, transaction type, affected accounts and transaction amounts. These relationships are defined by and associated with the EconomicEvent parameter of the AllowableAccountingTransaction, together with AffectedAccounts component, which represents accounts affected by this AccountingTransaction.

Subsequently, when instantiated, the AccountingTransaction will trigger a chain of various actions, such as updating account balances within journals and ledgers, generating financial reports and maintaining transaction history. Description of the parameters of the AccountingTransaction class (see Fig. 6.12) initiated to perform additional tasks like validation and recording to a general journal and ledger:

1. String transactionDescription - stores a description of the business transaction, providing a human-readable explanation of the transaction's purpose or nature;
2. Instant recordTimeStamp - represents the precise timestamp when the accounting transaction is recorded.
3. TransactionInputVerificationInternalData transactionInputVerificationInternalData - holds data related to the verification of transaction input values, including details about the economic event associated with the transaction and verification results. This class is the part of the funds verification process. The high-level process for this class is as follows:
   - the latest balance for each of such Accounts is extracted from the currentBalanceof the relevant GeneralLedgerAccount (see 2), otherwise it is set to 0 (zero). See last columns in Tab. 5.3 - Tab. 5.5 for visual reference;
4. TreeMap<Account, Double> transactingAmountDebitedAccounts - represents the debit side of the transaction which associates the relevant Account with the corresponding transaction amounts (as Doubles). The Account for this map defined by the AffectedAccounts :: debitedAccountsTransactingValues component;
5. TreeMap<Account, Double> transactingAmountCreditedAccounts - similar to the debited accounts, this attribute represents the credit side of the transaction using AffectedAccounts :: creditedAccountsTransactingValues component.
6. double debitedTansactingAmount -stores the total debited amount for the transaction;
7. double creditedTansactingAmount - holds the total credited amount for the transaction;
8. double absolutTransactingAmount - stores the absolute value of the transaction amount, which is used when the same value is set to he debited an the credited account.
9. String accountinTransactionHash - a hash value of this AccountingTransaction, utilised for data integrity and security purposes;
10. String sourceBusinessTransactionHash - the hash value that references the source business transaction associated with this accounting transaction.

For the demonstration of the use-case specific implementation of this step, the Java example of the instantiation of the AllowableAccountingTransaction is illustrated in Fig. 6.13. The EconomicEvent.InitialShareIssuance is assigned to this AllowableAccountingTransaction, together with other relevant Accounts, for the AffectedAccounts class. This structure is then stored in the AccountingTransactionRegister (see Section 6.1.5), managed by an Accountant. The instantiation of the AllowableAccountingTransaction initialInvestment class is an example of definition of the governance rules for propagation of accounting data that will be extracted from the business transaction.

When initialised AllowableAccountingTransaction initialInvestment becomes a param-

```
AllowableAccountingTransaction initialInvestment = new AllowableAccountingTransaction(
        EconomicEvent.InitialShareIssuance,
        "Initial Investment by the Initial Members",
        new AffectedAccounts("Initial Investment by the Initial Members",
                        COA.getAccountByName("Cash"),
                        COA.getAccountByName("Capital")));

AccountingTransactionRegister.addNewAllowableAccountingTransaction(initialInvestment);
```

Figure 6.13: Allowable Accounting Transaction class instantiation.

eter of the `AccountingTransaction` class (see Fig. 6.12), the final construct created is a representation of the *accounting transaction smart contract* example.

The generalised processes flow for each accounting smart contract is as follows:

1. Each `AllowableAccountingTransaction` is set up by the `Accountant` prior its utilisation, where it is assigned specific `EconomicEvent`, together with relevant set of `Accounts` within the `AffectedAccounts` class that will govern which `Accounts` to be debited and/or credited during execution;
2. Next, when a `BusinessTransaction` is flagged as as completed (instantaneously), or at the end of the accounting cycle, the monetary units/values are extracted from the `BusinessTransationRegister` repository;
3. In the `AffectedAccounts` class these values are assigned to the relevant `Accounts` on the debit and credit side, i.e., posted into the `TreeMap<Account, Double>` for `debitedAccountsTransactingValues` and `creditedAccountsTransactingValues` respectively;
4. Next, at the high-level, the latest balance, as input for this accounting transaction, for each of such `Accounts` is extracted from the `currentBalance`of the relevant `GeneralLedgerAccount` (see 2), otherwise it is set to 0 (zero).
   This balance is used in `TransactionInputVerificationInternalData`
5. New `JournalEntry` (1a) is instantiated for each of the `Accounts` within the `TreeMap <Account, Double>` for `debitedAccountsTransactingValues` and `creditedAccountsTransactingValues`;
6. Each new `JournalEntry` is posted to the new `GeneralJournal` by calling `recordToGeneralJournal(new JournalEntry)` method, continuing further steps of accounting cycle, described earlier in the Section 6.1.2 of this report.

6.1.5 Accountant's Roles and Responsibilities

The overall role of the Accountant and the set of responsibilities assigned to it within the BFS are central to the overall implementation and operation of this innovative blockchain based financial reporting system. In the implementation of the BFS, the Accountant class embodies and coordinates a set of traditional accounting responsibilities. These incorporate the Accountant's involvement in various stages of the accounting cycle, from the initial transaction registration to the final stages of financial reporting. The Accountant is further responsible for maintenance of the BFS's application local accounting repositories, such as the Chart of Accounts (COA), the General Journal, managing of the General Ledger and overseeing the Running Trial Balance.

All these traditional activities are underpinned by the innovative integration with the blockchain architectural data structures for publication of the financial statements; the conceptualisation of the Accountant's role within an economic entity with the context of a blockchain wallet; enrichment of the traditional accounting data with components of blockchain cryptography and so on.

Furthermore, this illustrative implementation of the Accountant class, with its logical relationships and methodologies for orchestration of financial data management and reporting, presented in the report, can serve as an additional conceptual framework for future incorporation of the traditional accounting roles and automation of responsibilities with the novel technological models, such as blockchain.

Accountant Role

In the overall domain of the BFS, the integration of traditional accounting roles with innovative technological frameworks of DLT presents both challenges and opportunities. The accountant, traditionally seen as an individual, department or external entity responsible for managing an organisation's financial records. This role undergoes a transformation within the BFS, adding dimensions through incorporation of blockchain technology.

In the implementation of BFS, the Accountant is re-imagined as a digital entity that facilitates traditional accounting responsibilities through enhanced functionality of BFS environment. This conceptualisation aligns with the project's aim to integrate and adapt conventional financial operations to the blockchain framework, ensuring integrity, transparency and efficiency. In the context of the BFS, the `Accountant` is defined as a hypothetical individual, who is the member (i.e., is employed) by the APF organisation. As a persona of the FBS framework, at its core, the `Accountant` is designed as a `Wallet`, a digital representation that embodies the financial identity and capabilities of the Accountant on the blockchain. As the `Wallet`, the `Accountant` serves as the digital custodian and the manager of the company's financial `Account`s within the BFS. This design choice is pivotal for several reasons:

1. *Digital Identity*: Encapsulating the Accountant as a `Wallet`, the BFS ensures that all interactions are securely linked to a verified digital identity (as a set of Public and Private keys, generated by and linked with this BFS entity), leveraging blockchain's inherent features;
2. *Member Wallet*: association explicitly links such digital identity with the APF as its member, `RelationToCompany.CompanyMember`, outlining its role within the company's operational structure. This association is not just administrative but functional, granting the `Accountant` necessary access rights to transactional and accounting data. This access is crucial for maintaining the accuracy, compliance and integrity of financial records.
3. *Controlled Access and Permissions*: is defined through `WalletControlType.Accountant`, within the BFS. This is to manage the scope and extent of the functional access the Accountant has over financial data and transactions and the scope of activity assigned to a traditional accountant within an organisation. It signifies a controlled, yet comprehensive access level, tailored to fulfil both traditional accounting responsibilities within BFS implementation.

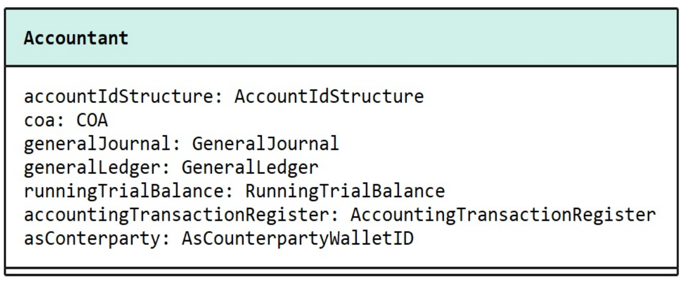


Figure 6.14: Accountant class.

As part of the implementation, the `Accountant` as the class extends a `Wallet` functionality defined in Section 6.2.4 and is assigned a role of the member of the organisation it is acting for. As the `MemberWallet` and as an `Accountant` (the traditional industry based role), it is assigned full access rights to the internal transactional and accounting data of the company. Fig. 6.14 illustrates the main data elements that falls into a primary "ownership" and responsibility of the `Accountant`. These are:

- ❖ AccountIdStructure accountIdStructure - a pre-defined standardised structure for the logical and hierarchical identification of Accounts within ledgers and reports;
- ❖ COA coa - see Section 6.1.2 for a detailed description of its design and implementation;
- ❖ GeneralJournal generalJournal - see Section 6.1.2 for a detailed description of its design and implementation;
- ❖ GeneralLedger generalLedger - see Section 6.1.2 for a detailed description of its design and implementation;
- ❖ RunningTrialBalance runningTrialBalance - see Section 6.1.2 for a detailed description of its design and implementation;
- ❖ accountingTransactionRegister accountingTransactionRegister - is the repository for accounting smart contract specifically designed and implemented for each economic entity to govern economic data transformation into accounting data. The detailed description of the structure of this class is presented in Section 6.1.5 of this report.

Accountant Responsibilities

Key Responsibilities:

- ❖ Accounting Transaction Implementation: where journalInitialShareSettlementBusinessTransactions_DvP() method of the Accountant class is executed as a the first *accounting transaction smart contract* within the BFS artefact(demonstrated in Section 7.3). It is designed to automate generation of accounting entries that are specific to the share settlement transaction (the use-case of this study), enamelling accuracy and transparency of financial recording.
- ❖ Business Transaction Register Management: The accountant plays a critical role in recording and verifying the details of business transactions. This responsibility includes ensuring that each transaction is accurately reflected in the BFS ecosystem and categorised according to its nature and impact on the organisation's financial position.
- ❖ Accounting Transaction Register Oversight: Beyond recording business transactions, the accountant is responsible for the oversight of accounting transactions. This involves translating business activities into accounting entries that reflect the financial implications of these activities, ensuring compliance with relevant accounting standards and principles.

AccountingTransactionRegister
beggeningPeriodDate: LocalDate endingPeriodDate: LocalDate allowableAccountingTransactions: HashMap <String, AllowableAccountingTransaction> accountingTransactionsRegister: HashMap<String, AccountingTransaction>

Figure 6.15: Accounting Transaction Register class

Fig. 6.15 illustrates the data elements within the AccountingTransactionRegister repository. These elements and their corresponding roles are as follows:

1. LocalDate for the beggeningPeriodDate and the endingPeriodDate - represent the beginning and end of the accounting cycle;
2. HashMap <String, AllowableAccountingTransaction> allowableAccountingTransactions - the repository of all AllowableAccountingTransaction designed by the Accountant to govern the rules for data transformation within AccountingTransactions. See detailed description of this element in Section 6.1.4

Fig. 6.11.

3. `HashMap<String, AccountingTransaction> accountingTransactionsRegister` - the implemented `AccountingTransaction`s of this accounting cycle. These are necessary to provide for tractability, error management and data verification necessary in traditional accounting processes. See a detailed description of this element in Section 6.1.4.

- ❖ Navigating the Accounting Cycle: The accountant oversees the entire accounting cycle within the BFS system, from initial transaction recording to the preparation of final financial statements. This includes managing journals and ledgers, adjusting entries and closing accounts at the end of the accounting period.
- ❖ Journals and Ledgers Maintenance: A key aspect of the accountant's role involves maintaining the Chart of Accounts (COA), General Journal, General Ledger and Running Trial Balance. The accountant ensures that these foundational components are up-to-date, accurate and reflective of the organisation's financial activities and status.
- ❖ Financial Reporting: One of the most critical responsibilities of the accountant within the BFS is the generation and analysis of financial reports. This includes the Balance Sheet, Income Statement and other reports that provide insights into the organisation's financial health. The accountant ensures that these reports are prepared in accordance with accounting standards and are available for review by stakeholders.

6.2 Implementation of Blockchain Sub-Domain

The integration of the blockchain architecture with traditional accounting processes and data provides an innovative approach to the accounting data validity, integrity, transparency and security of transactions and record-keeping. The blockchain sub-domain serves as the foundational framework upon which the BFS is built. The mechanics and bounded contexts within this tub-domain are critical for the BFS's ability to offer decentralisation of accountability by building on the traditional Bitcoin ledger data structure to store data and to provide tamper evident verification mechanism for such data, anchored in cryptographically linked sequential validity of events recorded within such a blockchain. Such exploration is crucial for understanding of how the BFS design extends components of blockchain technology to offer a novel approach to accounting operations that is secure, transparent and immutable.

6.2.1 BFS Block Structure Context

A Block is a fundamental element in any blockchain system, serving as a container for a specific type of data (Antonopoulos (2014)). Typically, it bundles together data such as transactions (Antonopoulos (2014)) or states (Brown (2018)) and includes metadata that secures its position in the blockchain's sequence (Antonopoulos (2014)). In essence, each block is securely connected to its predecessor through cryptographic means, establishing a backward link that forms the chain.

The BFS utilises blocks as the core structure for securely recording financial statements, together with other relevant metadata. Each block is uniquely identified by its hash, produced by applying the SHA256 cryptographic hash algorithm to the block's header (Antonopoulos (2014)). This hash acts as a unique identifier, a digital fingerprint, which secures the integrity of each block against alteration and falsification. Therefore, a block in the BFS stores data and is also integral to maintaining the system's security, ensuring the accuracy and unchangeability of historical records. By embedding these cryptographic links within each block, the BFS ensures a continuous and tamper-evident financial statements ledger. This mechanism is crucial for building trust and reliability in the system, as it guarantees that once data is entered into the blockchain, it cannot be altered retroactively without detection. This foundational aspect of blockchain technology is what makes the BFS a robust and secure platform for financial record-keeping and verifiable transactions.

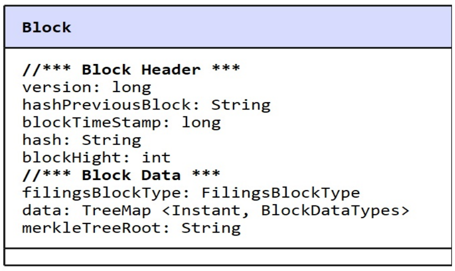


Figure 6.16: Block Data Structure

The block structure is illustrated in Fig. 6.16 and it can be broadly segmented into two main parts: the block header and the block body.

Block Header

In the BFS filing history, the block header functions as a key component for gathering metadata, providing crucial details about each block. This metadata acts as a unique identifier and enhances the security of the blockchain by establishing cryptographic connections to previous blocks, ensuring the system's unalterable nature and time sequencing. Consequently, blocks are not merely containers for data; they are essential for preserving the blockchain's consistency and unbroken sequence, reinforcing its overall trustworthiness and reliability. In the BFS implementation, the block header consist of the traditional Bitcoin blockchain style metadata (Antonopoulos (2014)), incorporating several key fields defined as part of the protocol format (see Fig. 6.16).

1. `long version` - represents the current software version of the BFS, allowing for the tracking of software or protocol upgrades over time. It ensures that nodes in the network can identify and operate under a unified set of rules.
2. `String hashPreviousBlock` - reference to the hash of the previous (parent) block in the chain; ensures the continuity of the blockchain by linking each block to its predecessor. It is a fundamental component that enables the construction of the blockchain's immutable chain.
3. `long blockTimeStamp` - UNIX time (seconds from the Unix Epoch), this timestamp marks the moment the block was added to the BFS. The timestamp's significance lies in its ability to anchor the block in a specific chronological context, making the timing of financial filings and transactions both transparent and verifiable.Adding a new block to BFS filing history sets this Block's time stamp, simultaneously calculating the hash of the block.
4. `String hash` - a cryptographic hash generated by applying the SHA256 algorithm to the block header. This digital fingerprint uniquely identifies the block and secures the blockchain against tampering. It's important to note that the block hash is computed by each node upon receiving the block and is not stored within the block's data structure itself.
5. `int blockHight` - indicates the block's position within the blockchain, with the genesis block assigned a height of 0 (zero). Each subsequent block appended to the chain increases the height by one, providing a measure of the blockchain's length and the relative position of each block within it (Antonopoulos (2014)).

In traditional blockchain implementations, the `nonce` and `difficulty` fields are expected as these play crucial roles in the mining process, regulating the creation of new blocks and the security of the blockchain (Antonopoulos (2014)). However, in the context of the BFS blockchain, these fields are omitted. This decision aligns with the project's focus on demonstrating the feasibility and functionality of a blockchain-based financial statements without the computational overhead associated with mining.

Block Data

In addition to the block header, the block's structure contains its data (see Fig. 6.16). The block data is a crucial part of the block's architecture within the BFS as it houses the actual content of the filing history blockchain - the financial statements or other mandatory reporting documents. This section explores the components of block data, focusing on their roles and significance within the BFS implementation.

Traditionally, blockchain blocks aggregate homogeneous data types, creating an organised and searchable ledger. The BFS enhances this concept by designing a set of varied data types that are stored within the blocks of the filing history blockchain. This decision was motivated by the integration of the regulatory reporting requirements of Companies House within the architecture of the BFS. Companies House is the UK's registrar of companies that is responsible for the record keeping of the entity's necessary filing history such as accounts, capital, incorporation documents etc. These categories are mandatory, because they represent key aspects of a company's structure, activities and compliance requirements (UK Government (2023)). This is to ensure that companies operate within the legal framework set out by the UK government, promoting a transparent, fair and stable business environment.

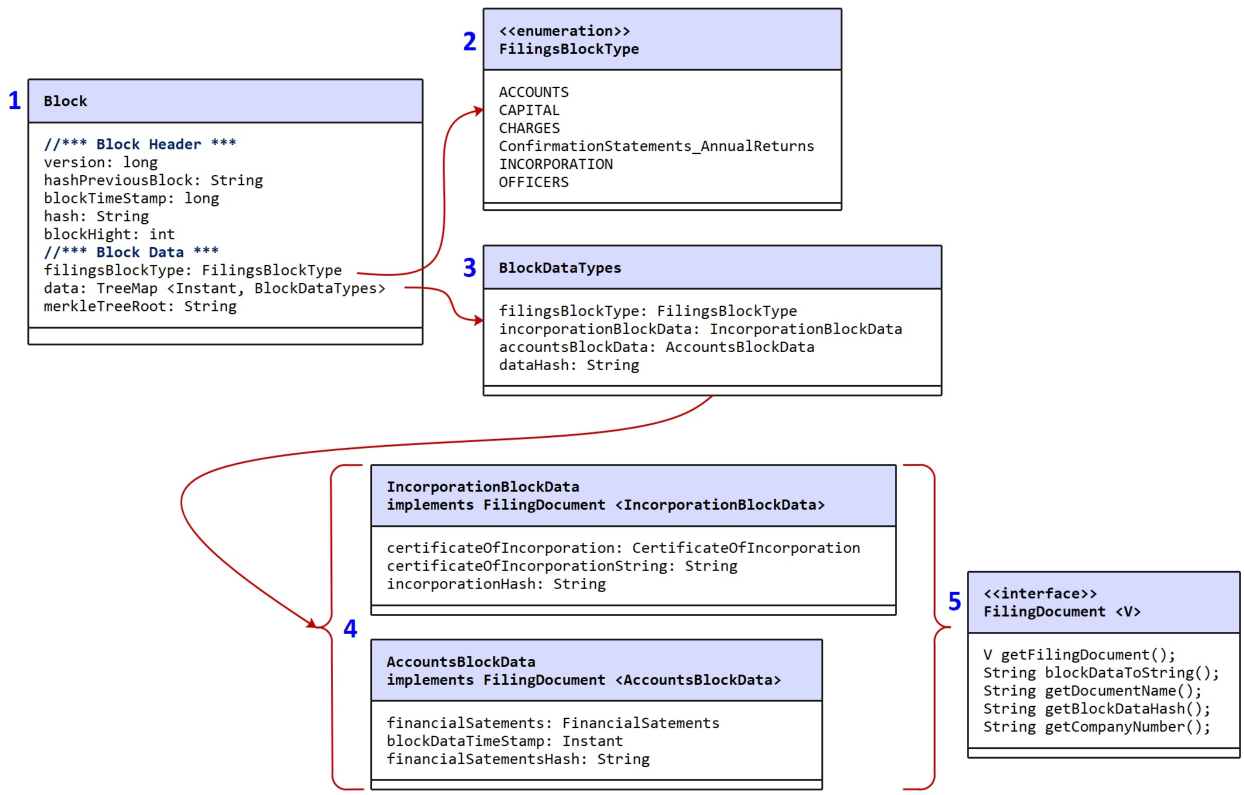


Figure 6.17: Block Data

The overview of all building components for the creation of the data attributes stored within the block of the BFS filing history are illustrated in Fig. 6.17. To enable compliance with the UK reporting regulation (Companies House (2024*b*)), the block data within the BFS filing history was designed to hold multiple pre-defined data types, to reflect the reporting structure of Companies House. In the implementation of the BFS, the traditional heterogeneous data structure of the block is implemented in an innovative way, integrating the multiple data types required in accounting and reporting. The data components within the `Block` of the filing history blockchain are as follows:

1. `FilingsBlockType filingsBlockType` - an enumeration that categorises the block by its filing type for accounting and reporting purposes;

2. `TreeMap <Instant, BlockDataTypes> data` - a chronological repository that enables filing of multiple `BlockDataTypes` within the same reporting cycle of economic entity. It utilises the novel `BlockDataTypes` class designed as the wrapper and a container for all necessary types required by the regulation in the UK. For illustrative purposes in this Thesis, only 2 use-case specific data types were designed, such as:
   (a) `IncorporationBlockData` - represents data specific to the incorporation filing documents, such as Certificate of Incorporation and so on;
   (b) `AccountsBlockData` - represents financial statement data associated with a block;
3. `String merkleTreeRoot` - the root of the Merkel Tree built by a sequential hashing of the hashes specific to each `BlockDataTypes` stored within `TreeMap <Instant, BlockDataTypes> data` of this block.

In Fig. 6.17, starting from marked blue number 1, the generalised structure of the data fields of the BFS Block are illustrated. Leading to the marked number 2 from the `filingsBlockType`, the association with the `enum FilingsBlockType` is illustrated. Next, this report describes the categories for each data type captured in the `FilingsBlockType`. The components of this enumerator `FilingsBlockType` embody the mandatory reporting requirements for companies in the United Kingdom, which are (Companies House (2024*b*)):

1. `ACCOUNTS` - includes financial statements, such as balance sheets, profit and loss statements and cash flow statements that provide insights into the financial health and performance of the company;
2. `CAPITAL` - relates to the company's share capital and any changes to its ownership structure;
3. `CHARGES` - refer to any security interests or liabilities that a company has granted over its assets, such as mortgages or debentures;
4. `ConfirmationStatements_AnnualReturns` - a snapshot of general company information, including registered office address, directors, shareholders and share capital;
5. `INCORPORATION` - encompasses documents related to the formation of the company, such as the memorandum and articles of association, certificate of incorporation and other incorporation-related filings. These documents establish the legal framework and governance structure of the company;
6. `OFFICERS` - individuals appointed to key roles within the company, such as directors, secretaries and other officers authorised to act on behalf of the company.

`FilingsBlockType` are also used to filter `Blocks` by its filing type from `FilingHistory` for accounting purposes. For example, establishing your business means that the directors are required to file certain documents every year such as annual accounts (`ACCOUNTS`) and a confirmation statement (`ConfirmationStatements_AnnualReturns`). They must also inform Companies House about any changes, such as the appointment or resignation of directors (`OFFICERS`). For illustrative purposes in this Thesis, only two use-case specific components of the enumerator `FilingsBlockType` were utilised: `INCORPORATION` and `ACCOUNTS`.

Next, in Fig. 6.17, starting from the generalised structure of the data fields of the BFS Block leading to the marked number 3, the `BlockDataTypes` are illustrated. This class is the wrapper and a container class for different Block Data types. Although it is possible to store multiple data types within this wrapper class, for illustrative purposes only one block data type is stored into the `Block :: TreeMap <Instant, BlockDataTypes> data`. The overall structure of the `BlockDataTypes` class is as follows:

1. `final FilingsBlockType filingsBlockType` - an association of the class that indicates what type of data is stored within this wrapper class;
2. `IncorporationBlockData incorporationBlockData` - contains information about the certificate of incorporation
   (a) `CertificateOfIncorporation certificateOfIncorporation` - the certificate of incorporation class - a digital version generated by the Registrar during company incorporation process;
   (b) `String certificateOfIncorporationString` - the string representation of the

certificateOfIncorporation class. This string is used in Genesis Block initial hash generation, described in the next Section 6.2.1;

(c) `String incorporationHash` - the hash of this class to provide for integrity of its data and used in the Merkel Tree construction of its `Block`;

3. `AccountsBlockData accountsBlockData` - holds financial statements relevant to the block;
   (a) `FinancialSatements financialSatements` - the final publishable financial statements
   (b) `Instant blockDataTimeStamp` - the time stamp of the addition of this class to the `Block`;
   (c) `String financialSatementsHash` - the hash of this class to provide for integrity of its data and used in the Merkel Tree contraction of its `Block`;
4. `String dataHash` - the overall hash of this wrapper to guarantee integrity of data within this class.

Both classes, `IncorporationBlockData` and `AccountsBlockData` implement `FilingDocument <V>` interface illustrated with the blue mark 5 in Fig. 6.17. The `interface FilingDocument <V>` is the marker interface that enables necessary common functionality for all types of data added to the `Block`s of the filing history blockchain of the BFS.

Genesis Block Implementation

This section of the report will describe how the data structure for the `Block` is utilised to generate the initial block of the BFS. The first block in any blockchain is traditionally called - the Genesis Block. It is a common ancestor for all subsequent blocks in any chain and this block cannot be altered (Antonopoulos (2014)). Every participant/node in a blockchain "knows" the genesis block and its hash, together with all the fixed data attributes within it (Antonopoulos (2014)). It can be thought of as a "secure starting point (root), from which to build a trusted blockchain" (Antonopoulos (2014)).

```
Constants
---------------------------------------------------------
bfsConstants: Constants
GENESIS_PREV_HASH: String
textForGenesisHash: String
---------------------------------------------------------
Constants(String textForGenesisHash) {
  Constants.GENESIS_PREV_HASH = Sha256Hash.generateHash(textForGenesisHash);
  Constants.textForGenesisHash = textForGenesisHash;
}
---------------------------------------------------------
Constants getInstance(String textForGenesisHash) {
  if (Constants.bfsConstants == null) Constants.bfsConstants = new Constants(textForGenesisHash);
  return Constants.bfsConstants;
}
...
```

Figure 6.18: Constants for BFS.

Before definition of the genesis block, it is necessary to describe the `Constants` class (see Fig. 6.18). This class is an important component of a blockchain workflow, as it is within this class that the immutable variables for creation of the genesis block are encapsulated. This definition is necessary to ensure that every participating member of the BFS entity starts with an identical copy of the blockchain, necessary for the maintenance of the ledger's integrity and trust.

The `Constants` class is illustrated in Fig. 6.18 and it encapsulates critical parameters that are used during the instantiation of the genesis block, ensuring consistency, security and integrity across the blockchain network. This class is critical for the instantiation of the genesis block for several reasons:

- This definition will ensure that for immutable reference, all variables of this class are `public static final`. These class variables are immutable after their initial assignment. By using a `final` class and `static` variables, these constants are locked down at compile time and cannot be modified, which is crucial for maintaining the consistency and integrity of the blockchain.

- By using a `private` constructor and a Java singleton pattern, the `Constants` class ensures that only one instance of such class and its class variables can ever exist. This single instance can be accessed globally within the application, providing a single source of truth for the genesis block's parameters.
- The `GENESIS_PREV_HASH` is a fixed value that is used by all the participants. This value is predetermined and it is critical because it signifies the start of the blockchain and its uniqueness helps in distinguishing the genesis block from other blocks of its blockchain or any other blockchain.

The class variables of the `Constants` class, illustrated in Fig. 6.18, encapsulate:

1. `static Constants bfsConstants` - the singleton instance of itself;
2. `static String GENESIS_PREV_HASH` - the very initial hash that will be used in the Genesis Block. This is the hash of the next class variable - `textForGenesisHash`;
3. `static String textForGenesisHash` - the `String certificateOfIncorporationString` from the `IncorporationBlockData` outlined in Section 2b.

Next is the description of the implementation of the "Genesis Block" instantiation, designed in reference to the traditional practices, but adapted to the use-case specific requirements of the incorporation process of the APF BFS and as per public documentation available through Companies House (Companies House (2023a,b)).

During the implementation of the incorporation process of the APF by the `Registrar` of the hypothetical Companies House illustrated in Fig.5.3 (Company Incorporation Process Flow) the `IN10` form is provided by the perspective directors of the APF to the said `Registrar`.

```java
30 public final class Registrar {
31     private PrivateKey privateKey;
32     private PublicKey publicKey;
33     private static final HashMap <String, PrivateRegister> privateRegister =  new HashMap <String, PrivateRegister>();
34 
35     private Registrar() {
36         KeyPair keyPair = Cryptography.ellipticCurveCryptoKeyPair();
37         this.privateKey = keyPair.getPrivate();
38         this.publicKey = keyPair.getPublic();
39     }
40 
41     public static PrivateRegister incorporateCompany(IN10 in10) {
42         CertificateOfIncorporation certificateOfIncorporation = new CertificateOfIncorporation(in10);
43         IN10 in10Updated = in10;
44         in10Updated.getPart1_CompanyDetails().setCompanyRegistrationNumber_A1(certificateOfIncorporation.getCompanyRegistrationNumber());
45         in10Updated.setDateOfIncorporation(certificateOfIncorporation.getDateOfIncorporation());
46         PrivateRegister pr = new PrivateRegister(in10Updated, certificateOfIncorporation);
47         for (Map.Entry <String, PrivateRegister> form: privateRegister.entrySet()) {
48             if(!privateRegister.isEmpty()
49                     &&(form.getKey()==in10Updated.getPart1_CompanyDetails().getCompanyRegistrationNumberString()
50                         && form.getValue() == pr)) {
51                 System.out.println("The company"+in10.getPart1_CompanyDetails().getProposedCompanyNameInFull_A1()
52                         +" has been already regestered.");
53                 return null;
54             }
55         }
56         Registrar.privateRegister.put(certificateOfIncorporation.getCompanyRegistrationNumberString(), pr);
57         return pr;
58     }
59 
60 
61 
```

Figure 6.19: Registrar class as a hypothetical role / actor of the "Companies House" entity within the BFS ecosystem.

The partial Java code implementation for the `Registrar` class is demonstrated in Fig. 6.19. One can see that this entity possesses Public and the Private set of keys (see Fig. 6.19, lines 31-32), enabling it to interact with other ecosystem participants. Furthermore, the `privateRegister` map (line 33 of the Fig. 6.19) - is a store (a private repository) for all existing incorporation documentation of all economic entities within overall BFS ecosystem, registered/incorporated by this `Registrar`. The key for each of such records within the the `privateRegister` map is the unique company registration number generated for each of these entities (analogues to the current practice).

During the incorporation process of a business, the process for generation of the `PrivateRegister` instance is implemented by calling a `incorporateCompany(IN10 in10)` method of the `Registrar` class, illustrated in line 41 of Fig. 6.19. Here, the finalised `IN10` form is utilised as the parameter of the `incorporateCompany(IN10 in10)` method that returns new instance of the `PrivateRegister`. This new `PrivateRegister` instance is then used as the value entry of the `privateRegister` map, in line 33 of Fig. 6.19, for each new corresponding registered company.

The instantiation process of this new `PrivateRegister` instance - is where the initial Genesis Block is implemented during company incorporation. As is evident from Fig. 6.19 line 46, the copy of form `IN10` and the newly instantiated `CertificateOfIncorporation` (Fig. 6.19 line 42) both become parameters of the `PrivateRegister` constructor (see Fig. 6.19 line 46, and Fig. 6.20 line 30).

```
17 public class PrivateRegister {
18     private char [] companyRegistrationNumber = new char[8];
19     private String companyFullName;
20     private final LocalDate dateOfIncorporation;
21     private LocalDate accountingReferenceDate;
22     private final CertificateOfIncorporation certificateOfIncorporation;
23     private MemorandumOfAssociation memorandumOfAssociation;
24     private ArticlessOfAssociation articlesOfAssociation;
25     private HashMap <String, ShareCertificate> issuedShareCertificatesActive;
26     private final IN10 in10;
27     private final String certificateOfIncorporationString;
28     private final Block genesisBlock;
29
30     public PrivateRegister(IN10 in10, CertificateOfIncorporation certificateOfIncorporation) {
31         this.companyRegistrationNumber = certificateOfIncorporation.getCompanyRegistrationNumber();
32         this.companyFullName = certificateOfIncorporation.getCompanyNameInFull();
33         this.in10 = in10;
34         this.certificateOfIncorporation = certificateOfIncorporation;
35         this.memorandumOfAssociation = in10.getMemorandumOfAssociation();
36         this.articlesOfAssociation = in10.getPart1_CompanyDetails().getArticlesOfAssociation_A8();
37         this.dateOfIncorporation = in10.getDateOfIncorporation();
38         setAccountingReferenceDate(this.dateOfIncorporation);
39         this.certificateOfIncorporationString = certificateOfIncorporation.getCertificateOfIncorporationString();
40
41         String GENESIS_PREV_HASH = Constants.getInstance(this.certificateOfIncorporationString).get_GENESIS_PREV_HASH();
42         this.genesisBlock = new Block(GENESIS_PREV_HASH, FilingsBlockType.INCORPORATION, 1);
43         this.genesisBlock.addData(this.dateOfIncorporation,
44                 new BlockDataTypes(new IncorporationBlockData(certificateOfIncorporation)));
45     }
46
```

Figure 6.20: Private Register class.

The partial Java code implementation for the `PrivateRegister` class is illustrated in Fig. 6.20 (instance variables and constructor). One can see on the line 28 of Fig. 6.20, the permanent copy of the genesis block for the APF entity, to whom this `PrivateRegister` was created is stored. This block is generated in the constructor of `PrivateRegister`, the lines between 41 - 44 of Fig. 6.20 and this process is as follows:

- ❖ in line 39, the `String` representation of the `CertificateOfIncorporation` is retried and stored within fields of the `PrivateRegister` (line 27);
- ❖ `this.certificateOfIncorporationString` is then passed to the singleton `Constants` class, where `String GENESIS_PREV_HASH` is generated, permanently stored and returned for the `Block genesisBlock` instantiation;
- ❖ next, on line 42 of Fig. 6.20 a new instance of a `Block` is initialised, taking `String GENESIS_PREV_HASH`, `FilingsBlockType.INCORPORATION` and `1` (as the software version) in its constructor;
- ❖ In Fig. 6.21, the partial Java code implementation for such `Block` class is illustrated. The constructor of this class (see lines 25-31 of Fig. 6.21) takes `String hashPreviousBlock`, `FilingsBlockType filingsBlockType`, and `long version` as parameters (the data relationships and associations are illustrated in Fig. 6.17 of the previous Section);
- ❖ next, in line with Fig. 6.20 line 43, the

```java
15 public class Block {
16     private long version;
17     private String hashPreviousBlock;
18     private long blockTimeStamp;
19     private String hash;
20     private int blockHight;
21     private FilingsBlockType filingsBlockType;
22     private TreeMap <Instant, BlockDataTypes> data;
23     private String merkleTreeRoot;
24 
25     public Block (String hashPreviousBlock, FilingsBlockType filingsBlockType , long version) {
26         this.filingsBlockType = filingsBlockType;
27         this.version = version;
28         this.hashPreviousBlock = hashPreviousBlock;
29         this.data = new TreeMap <Instant, BlockDataTypes>();
30         System.out.println("Block Constructor was called.");
31     }
32 
33     public void addData(LocalDate documetFilingDate, BlockDataTypes document) {
34         if(document != null) {
35             Instant docDate = documetFilingDate.atStartOfDay(ZoneId.systemDefault()).toInstant();
36             this.data.put(docDate, document);
37             this.blockTimeStamp = docDate.toEpochMilli();
38             this.merkleTreeRoot = null;
39             this.hash = null;
40             this.merkleTreeRoot = calculateMtrHash();
41         }
42     }
43 
```

Figure 6.21: Block class.

`addData(LocalDate documetFilingDate, BlockDataTypes document)` method of this new `Block` is called. The implementation of this method is illustrated in Fig. 6.21 lines 33-41.

- ❖ in line 43 of Fig. 6.20, the `this.dateOfIncorporation`, retrieved from the `IN10` form and set on the line 37 is passed to the `addData(...)` method;
- ❖ in line 44 of Fig. 6.20 of the same `addData(...)` method, the new `BlockDataTypes` instance is created, utilising `certificateOfIncorporation` and the relationships of the data described and illustrated earlier in Fig. 6.17.
- ❖ through this process the new and `final` instance of the `Block genesisBlock` is established and stored within this `PrivateRegister`.

At the end of this process, the newly created `PrivateRegister` instance (see Fig. 6.19 lines 46-56) is added to as the governmental record of the instantiation of this company to the `privateRegister` map (line 33) of the `Registrar` entity (see Fig. 6.19). Furthermore, the copy of that `PrivateRegister`, together with the copy of the `Block genesisBlock` is shared with the company directors (as illustrated in Fig.5.3 - Company Incorporation Process Flow), who is then able to instantiate their business as the `BFS` entity and commence planned business activity.

6.2.2 The BFS and Ledger Context

The prospective directors of the APF, upon receiving the `PrivateRegister` instance, are able now to instantiate `APF::BFS` as a "private company limited by shares" (Companies House (2023*a*)). In the context of the BFS, the instantiation of the BFS as a private company limited by shares marks the beginning of the APF's economic activity. At the high-level, the `BFS` class is a primary repository that hosts:

- ❖ all the company related data generated during its incorporation;
- ❖ the blockchain itself - i.e., the reporting filing history of the economic entity;
- ❖ the primary connections to the wallets relevant to this economic entity;
- ❖ and the digitised financial holdings of the economic entity, such as shares or the debt assets belonging to it.

To ensure that the implementation of the BFS aligns with the real-world reporting and information standards of the Companies House in the UK, the existing user interface for the company

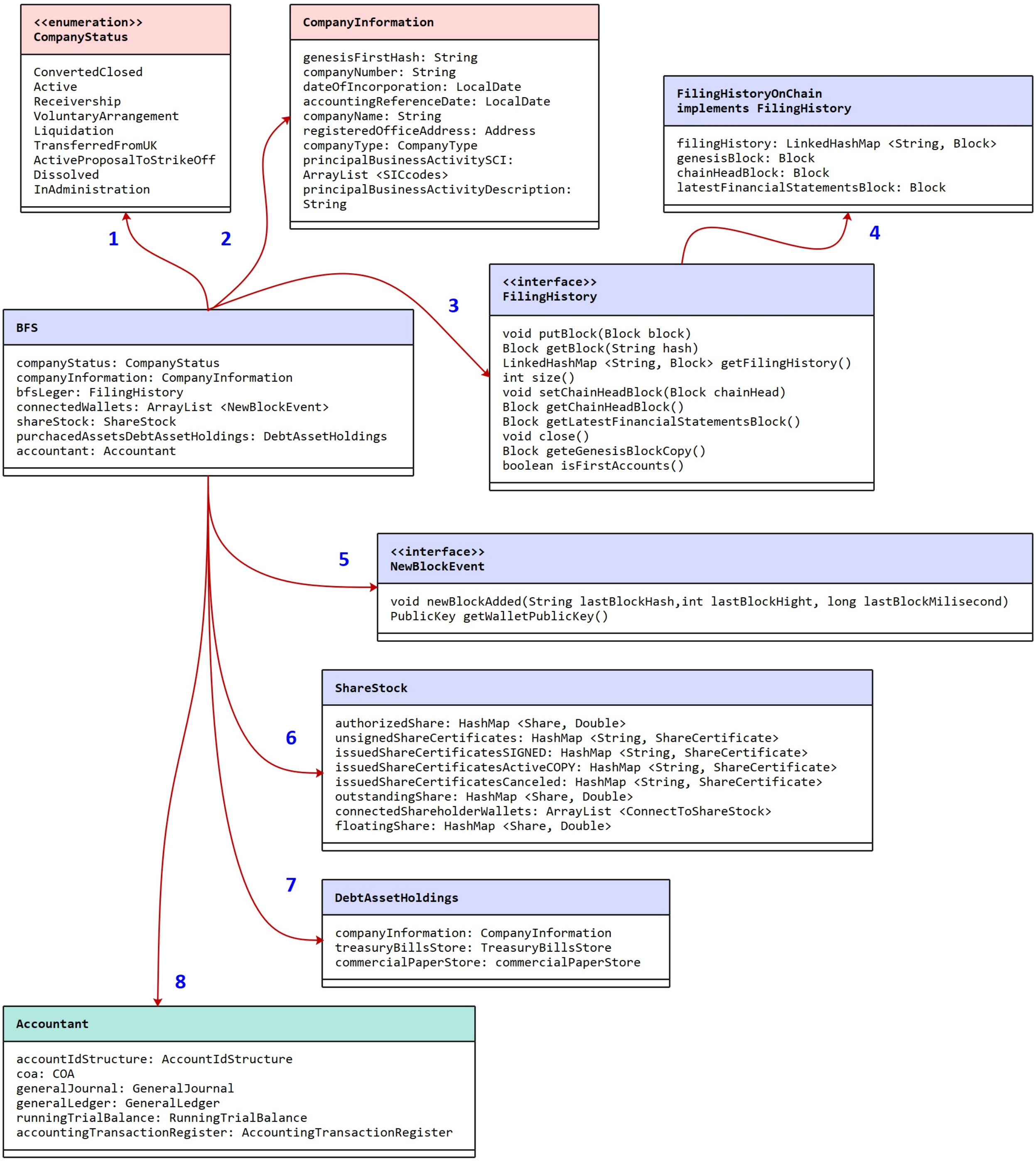


Figure 6.22: Blockchain Financial Statements Data Structure.

overview of the Companies House website (Companies House (2023*a*,*b*)) served as the inspiration for structuring the entity-related data within the BFS class, ensuring conformity with real-world reporting norms. The data elements that are reflected as the entity global data variables are illustrated in Fig. 6.22 and are as follows:

- current status of the company, options for winch are encapsulated within the `CompanyStatus` enumerator;
- overview of the company, winch is implemented in the `CompanyInformation` class;
- a tab for the “filing history” tab within companies house UI is implemented by the `FilingHistory` - the data store for all mandatory reporting of an economic entity;

- the “people” tab within companies house UI is implemented by the `ArrayList <NewBlockEvent> connectedWallets` - which connects every new created `Wallet` relevant to this economic entity;

The overall data structure and the most relevant building blocks for the `BFS` class are illustrated in Fig. 6.22. The description of roles for its elements is as follows:

1. `CompanyStatus companyStatus` - is an enumerator that encapsulates some regulatory expected states that an economic entity can be in, such as active, dissolved an so on;
2. `CompanyInformation companyInformation` - this class generalises the global entity specify identification data created during company incorporation or amended during entities going concern;
3. `FilingHistory bfsLeger` - an interface that ensures that it the blockchain implementation adheres to the methods and standards required for filing history within the blockchain system. The `FilingHistory` interface in the BFS serves a critical role, managing the storage and retrieval of blocks within the blockchain, symbolizing the company’s evolving financial narrative. As the BFS ecosystem grows, storage challenges may emerge, similar to those faced by the Bitcoin blockchain, prompting future innovative solutions for a filing history blockchain. Such future implementation could allow for a “lightweight” blockchain variant that could maintain only essential recent records on user devices, offloading older data to more permanent cold storage solutions. This strategic approach ensures scalable and compliant record-keeping within the BFS framework.
4. `FilingHistoryOnChain` that implements `FilingHistory` - class within the BFS serves as a blockchain-based accounting recording and reporting system, crucial for the storage and management of financial and non-financial blocks of the BFS. Within its `AccountingBlock` type, it captures snapshots of account balances for distinct accounting periods, maintaining a comprehensive history of all entries of the financial ledger. Through such mechanisms, the BFS ensures efficient and compliant record-keeping, underpinning the APF’s financial transparency and accountability.
5. `ArrayList <NewBlockEvent> connectedWallets` is the primary registration and blockchain related data update point for all wallets, relevant to this economic entity.
   The `NewBusinessTransactionEvent` interface is implemented by all `Wallets` (see Section 6.2.4), utilises `PublicKey getWalletPublicKey()` method to notify connected parties within the BFS ecosystem about the addition of new `Block` to the `FilingHistory`.
6. `ShareStock shareStock` - this is a business type specific repository for a private or public company limited by shares. This repository records all share issuance and sale related activity of the entity;
7. `DebtAssetHoldings purchacedAssetsDebtAssetHoldings` - this is the use-case specific repository for the tokenized debt assets, such as “Treasury Bills” and “Commercial Paper”. This repository records all debt asset issuance and liquidation related activity of the entity;
8. `Accountant accountant` - the default hypothetical accountant role created at the first installation of the BFS entity. During this process in the BFS constructor, the default hypothetical entity is generated as a `Accountant` of the `APF::BFS`, who is set up as the company member, i.e., it is employed by the APF::BFS. The `Accountant` class is a repository of all necessary accounting journal and ledgers for its employer - the APF::BFS. Furthermore, the default `Accountant` is enabled with the functionality to generate an initial set of `Accounts`, to facilitate the initial booking activity and record initial investment of the initial members of the APF. This is known as equity/capital or net worth and is the amount owners have invested in a business (Sage50 (2023)). In accordance with traditional accounting practices (Sage50 (2023)) and the use-case of this research, the initial set of default `Accounts` is: cash, retained earnings and capital accounts (see “Account class” Fig. 6.3). The data structure for each `Account` is illustrated in Fig. 5.4, each of which is set up in accordance with “Financial Accounting Sub-Domain” outlined in 5.3.1 and the “Account and its Hierarchy Context”.

6.2.3 Smart Contracts Context

The concept of smart contracts stands central in the blockchain domain, providing an flexible mechanism for automating the execution of agreements without the need for intermediaries. Within the BFS artefact, smart contracts are critical components that drive the functionality of business and accounting transactions (see Sections 6.3.3 and 6.1.4), as well as the facilitation of use-case specific lending agreement (see Section 6.3.2).

As a general concept, smart contracts are self-executing contracts, e.g. with the terms for such agreement between buyer and seller directly written into lines of code. The code and the agreements exist across a distributed, decentralised blockchain network. The code controls execution, and transactions are trackable and irreversible, providing a trustworthy environment for parties to transact. The BFS artefact builds this technology for the following:

Business Transaction Smart Contracts: Designed to autonomously perform complex business transactions between entities. These smart contracts record the terms, verify conditions, and automatically execute the agreed-upon details.

A designated `BusinessTransactionRegister` repository is implemented to record all economic interactions, including the exchange of assets, payment processes, and the confirmation of agreement conditions.

Accounting Transaction Smart Contracts: Transform economic data from business transactions into standardised accounting entries. They apply principles of double-entry bookkeeping and ensure that all transactions are reflected accurately in financial statements. The `AccountingTransactionRegister` repository is dedicated to store these rules governing the automated transformation of business transaction data into accounting records. These smart contracts systematically process economic data, applying relevant accounting standards to generate ledger entries and eventually financial statements, showcasing real-time financial health.

Loan Agreement Smart Contracts: Facilitate the process of drawdown requests, issuance, and settlement of funds between entities like the APF and the BoE. These contracts automate the terms of the loan, including interest calculations, repayment schedules, and trigger necessary actions based on predefined criteria.

In the BFS artefact, smart contracts are uniquely designed to fulfil specific roles within the ecosystem. They are the core mechanism upon which the BFS artefact operates, ensuring accuracy, efficiency, and integrity in financial transactions and reporting.

The Java implementation of these smart contracts is provided in Sections 6.3.3, 6.1.4, and 6.3.2, and it illustrates the feasibility of integrating blockchain technology into traditional financial and accounting systems. Through the BFS artefact's proof of concept, the research demonstrates how blockchain technology can be harnessed to significantly enhance financial management and reporting processes.

In essence, the application of smart contracts within the BFS artefact is a demonstration to the potential for blockchain technology to reinvent how financial interactions are conducted, providing a secure, transparent, and efficient system for all parties involved in the economic ecosystem.

6.2.4 Wallet

The Wallet class is one of the most important classes in the BFS. It is an abstract class that must be extended by any use-case specific implementation of a BFS Wallet, such as `DeirectorWallet`, `ShareholderWallet` and so on (see Section 6.3.6). This design decision enables future extensibility and flexibility to cater diverse requirements of the heterogeneous economic practices and participants. The design and implementation of this class within a BFS was built on a combination of existing blockchain practices of a traditional wallet, where it starts from a simple public/private key store to uniquely identify an entity participating in the blockchain network. The wallet in the BFS serves as the repository for cryptographic identity and as a comprehensive transactional

Figure 6.23: BFS Wallet class.

history manager, a controller of access permissions and a secure vault for cryptographic secrets. Furthermore, as each wallet relates first to the business entity it is designed to transact for, it contains the relevant standardised `CompanyInformation` for that business entity.

In Fig. 6.23 the data structure and the functional capability of the interfaces that this class implements are described. These interfaces are identified by the blue arrow connectors leading from the Wallet class in Fig. 6.23. The extended data of the `Wallet` class is identified by the red arrows of the same Fig. 6.23. The detailed description below outlines the key features and functionalities of such an advanced blockchain wallet.

1. interfaces implemented by the Wallet class:
   - `NewBlockEvent` - a functional interface to be extended by Wallet that allows the host BFS to communicate with it native `Wallets`. When the filing history of the BFS receives a new `BLock`, and appends it to the `LinkedHashMap <String, Block> filingHistory`, it activates relevant BFS functionality that notifies all connected Wallets about this new Block.
     - in the BFS: Block is inspected for Transactions and Public Keys are extracted;
     - in the BFS: `Wallets` whose public keys are found, are sent relevant transactions for inspection and local in `Wallet` updates;
     - in the `Wallet`: local in `Wallet` updates, e.g., to mark relevant transactions as closed (completed) or discontinued;
     - in the `Wallet`: the metadata from this new added `Block`, such as last block hash, last block height and last block time stamp (see Fig. 6.23 - 1) is copied to this `Wallet` for future fast referencing.
   - `NewBusinessTransactionEvent` - Used by an element of the `BusinessTransactionRegister`.
     The `ArrayList <NewBusinessTransactionEvent> connectedParties`:
     - registration with the `connectedParties` happens when Wallet is added to the BFS
     - called by the `BusinessTransactionRegister` but implemented by the `Wallet`
     - called when the registered Wallet's `PublicKey` is a `TransactionParticipant` within added business transaction and sends a copy of such transaction into the`Wallet`'s local transaction repository, such as
       `Map <String, BusinessTransactionDataInterface <?>>`
       `businessTransactionsALL`.
2. data elements of the `Wallet` class:
   - `WalletControlType walletControlType` differentiation of the level of control the Wallet has in relation to the data internal to the entity. This distinction is necessary because company member, counterparty and related party are two distinct terms used in accounting that refer to different types of relationships to the economic entity. This enumerator class marks the level of control over a company's activity and the access to the internal business sensitive data within an organisation. The high-level distinction here is:
     (a) Members of the company - such as shareholders or directors, typically have a higher level of control over the company's activities than related parties or counterparties. They may have voting rights and decision-making power that can influence the direction and operation of the company.
     (b) Related parties - such as parent companies, subsidiaries, or key management personnel, may have some level of influence over a company's activities. However, their level of control is typically limited to their specific relationship with the company and may not extend to broader decision-making or strategic planning.
     (c) Counterparties - such as customers, suppliers or lenders, typically have no direct control over a company's activities. Their relationship with the company is based on a transactional exchange of goods, services or financing and they do not have voting rights or decision-making powers.

- `PrivateKey privateKey` and `PublicKey publicKey` - enable secure, cryptographic transactions within the BFS ecosystem. These keys are part of a cryptographic algorithm used for encrypting and decrypting information and to uniquely identify wallet owners during transactional interactions to enable the secure, anonymous and autonomous operation;
- `int version` - used to track future changes in the Wallet format;
- `CompanyInformation companyInformation` - a BFS designed standardised way of capturing relevant and unique publicly available information to identify each economic entity within the ecosystem;
- `String walletOwnerName` - a distinct human readable name assigned to an individual or an entity
- `SecretStore secretStore` - a wallet local repository for a cryptographic secrets. It is an implementation of a "Hash Time Lock Contract" (HTLC), a key element that enables cross-blockchain atomic swaps, or atomic transactions (), whilst minimising the risk to transacting counterparties.HTLC consists of the cryptographic hash function and a time lock (or time out) element. Below is the high-level the HTLC process flow is presented:
  - entity A agrees to purchase an asset from the entity B for an offered amount of digital funds;
  - an entity A first blocks offered amount of digital funds, agreed for transaction, on its ledger for a time period T;
  - the wallet of the entity A generates a `Secret String` X_{APF}, and its `SecretHash` that contains $Y_{APF} = \mathrm{H}(X_{APF})$;
  - the wallet of the entity A communicates the `SecretHash`, containing the Y_{APF} with the entity B and stores the `Secret String` X_{APF} and its `SecretHash` locally in the `Wallet::SecretStore ::HashMap <String, SecretHash> mySecretHashStore`.
  - This `Secret String` X_{APF} will be utilised by A to claim the asset from B for the payment offered.
  - during the execution of atomic cross-ledger transaction, if the time interval is still T, the entity A shares the `Secret String` X_{AOE} with entity B;
  - entity B verifies if `Secret String` X_{AOE} satisfies `SecretHash` $Y_{AOE} = \mathrm{H}(X_{AOE})$ and, if so, the entity B releases the asses to the entity A and retrieves the digital funds from the entity A to cover the agreed price.
- Next, the `Map <String, BusinessTransactionDataInterface <?>>` for `businessTransactionsALL`, `businessTransactionsOpened`, `businessTransactionsClosed`, and `businessTransactionsDiscontinued` - represent wallet local repositories for the business transaction specific only to this wallet owner. It includes initiated, ongoing, completed and discontinued transactions, offering a holistic view of the wallet's activity over time. These various pools for transactions give quick access to wallet-relevant transactions by the state they're in;
- `String lastBlockHash` - the hash of the last block in the filing history blockchain for the quick referencing and updates;
- `int lastBlockHight` - the height of the last block in the filing history blockchain for quick referencing and updates;
- `long lastBlockMilisecond` - the time stamp of the last block in the filing history blockchain for quick referencing and updates.

6.2.5 Validator Context

In a traditional blockchain architecture, the validator is responsible for verifying the legitimacy of transactions before they are added to the blockchain (Nakamoto (2008)). Validators check transaction details against the network's consensus rules, ensuring that each transaction is valid, not duplicated and that participants have the necessary funds for the transaction. Once a transaction is verified, validators are involved in the process of creating new blocks, which are then added to the blockchain after a consensus is reached among the nodes, e.g., a proof-of-work (PoW) consensus in Bitcoin(Nakamoto (2008)). This process ensures that only valid and agreed-upon transactions are recorded, preserving the blockchain's trustworthiness and reliability.

The design and implementation of the BFS introduces a novel approach to the role of validators within the blockchain network specifically tailored to enhance the transparency, efficiency and integrity of financial reporting between central banks and businesses. Unlike traditional blockchain models where validators add blocks to a blockchain in the BFS design the responsibility for adding blocks is reserved for an `Accountant` and can involve explicit approval of the company directors. This design decision was motivated by aligning the functionality of the BFS with traditional accounting reporting practices. The validator's role is redefined to focus solely on:

- verifying the *integrity of transactional data*, which includes checks for the completeness, accuracy and consistency of transactional data;
- the *validity of signatures*, to confirm that transactions have been initiated by legitimate parties and have not been tampered with during transmission;
- importantly, the Validator *confirming input values* for business transactions without revealing the data. This approach aligns with the principles of zero-knowledge proofs (ZKP), a cryptographic method that allows one party (the prover) to prove to another party (the verifier) the truth of a statement without conveying any information apart from the fact that the statement is indeed true (Ben-Sasson et al. (2014)). This is another innovative aspect of the BFS, where the Validator adapts a principle of ZKP, verifying the sufficiency of funds for business transactions and confirming funds availability claim made that is based on the internal accounting record, without disclosing the actual balances or business sensitive details of the internal accounting ledgers. In this way, the Validator implements the "funds verification consensus", described in Section 6.2.6 of this report.

This specialisation of the design and implementation of the role and responsibilities for the Validator ensures that it concentrates on the accuracy and authenticity of transactions without being involved in block creation, streamlining the validation process and enhancing system security. By separating the roles of validators and accountants, the BFS design enhances the focus on transactional integrity and privacy. In Fig. 6.24, the implementation of the components of validator package for the BFS is illustrated. This package is designed to host a series of specialised classes, each tailored to perform specific functions within the validation process, thus facilitating integration of blockchain technology with financial reporting processes.

- `Validator` class itself is designed to verify various aspects of transactions without storing any data, aligning with the principles of statelessness and immutability inherent in blockchain technology. This class contains the default implementation for the methods to verify the integrity of transaction data. These methods are critical for ensuring that each transaction adheres to predefined rules and conditions before being processed further in the BFS system.
  - `boolean verifyTransactionDataIntegrutyTradeINPUT (TradeOrderFinancialSecurity tradeInput)` - verifies the integrity of trade order transactions involving financial securities, ensuring that the trade inputs are valid and tamper-proof;
  - `boolean verifyTransactionDataIntegruty (LoanUtilisationRequest loanUtilisationRequest)` - checks the integrity of loan utilisation requests, validating the request's completeness and accuracy;
  - `boolean verifyTransactionDataIntegruty (FundsTransferConfirmationByLender fundsTransferConfirmationByLender)` - verifies the integrity of funds transfer confirmations by lenders, ensuring the transfer details are correct and authorised.
  - `boolean verifyTransactionDataIntegruty`

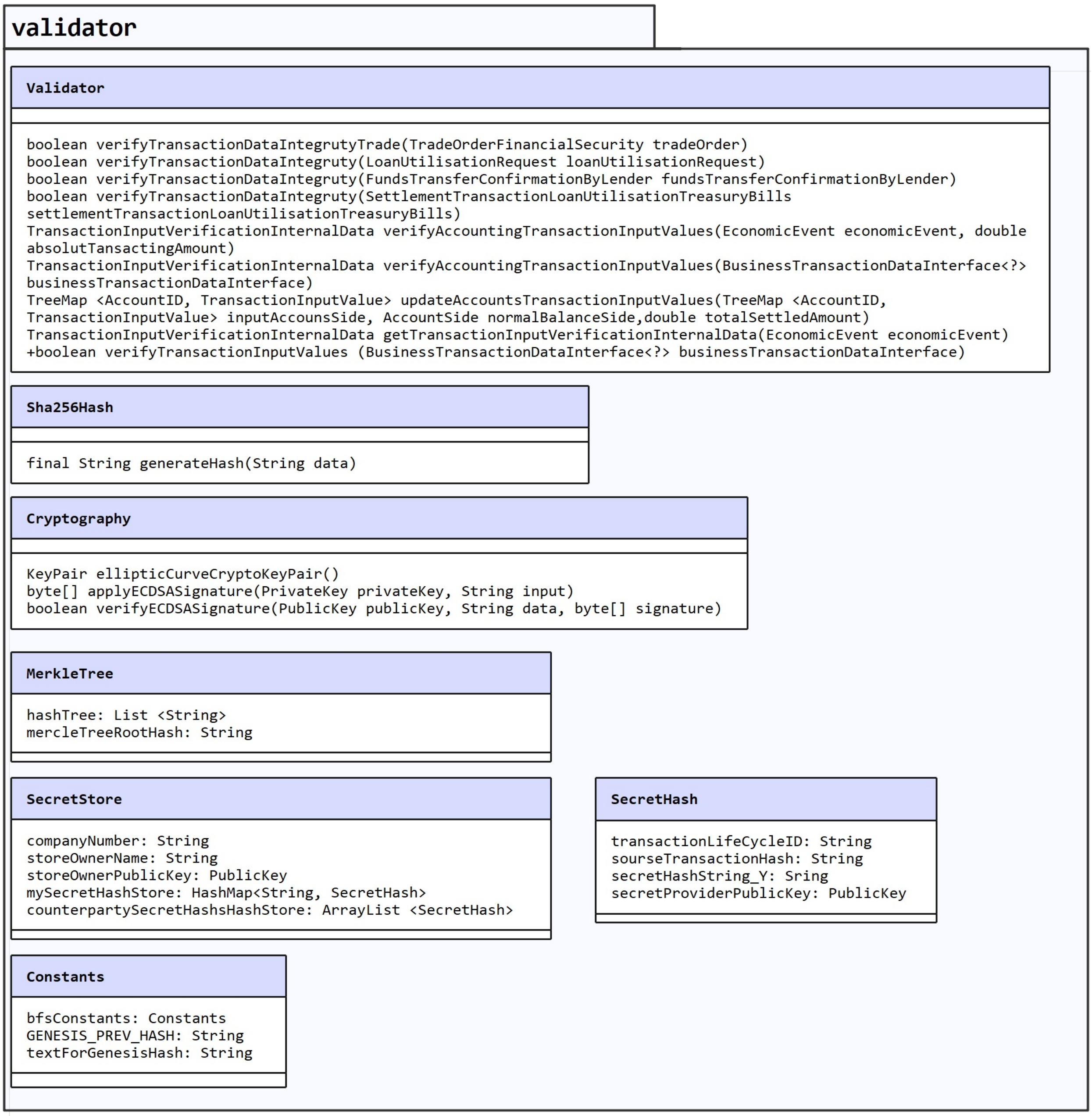


Figure 6.24: Package "validator".

`(SettlementTransactionLoanUtilisationTreasuryBills settlementTransactionLoanUtilisationTreasuryBills)` - verifies the integrity of settlement transactions for loan utilisation involving treasury bills, validating the transaction's compliance with regulatory and system rules;

- `TransactionInputVerificationInternalData verifyAccountingTransactionInputValues(EconomicEvent economicEvent, double absolutTansactingAmount)` - verifies the input values of accounting transactions against the economic events and transacting amount, ensuring the transactions meet the necessary conditions without revealing sensitive data;
- `TransactionInputVerificationInternalData verifyAccountingTransactionInputValues (BusinessTransactionDataInterface<?> businessTransactionDataInterface)` - verifies the input values of accounting transactions for the business transaction themselves,

utilising `BusinessTransactionDataInterface<?>` of these business transactions, ensuring the transactions meet the necessary conditions without revealing sensitive data;
  - `TreeMap <AccountID, TransactionInputValue> updateAccountsTransactionInputValues (TreeMap <AccountID, TransactionInputValue> inputAccounsSide, AccountSide normalBalanceSide, double totalSettledAmount)` - updates the transaction input values for accounts, maintaining the integrity and balance across the BFS ecosystem;
  - `TransactionInputVerificationInternalData getTransactionInputVerificationInternalData(EconomicEvent economicEvent)` - retrieves internal data necessary for verifying transaction input, facilitating a secure and efficient verification process.
  - `boolean verifyTransactionInputValues (BusinessTransactionDataInterface<?> businessTransactionDataInterface)` - takes a settlement ready (pre-settlement) business transaction and verifies against internal accounting records if the value, stated in the business transaction bu the economic entity is available for spend.

- `Sha256Hash` - leverages the SHA-256 hashing algorithm to generate secure, irreversible hash values for transaction data, ensuring that once a transaction is recorded, its integrity is immutable.
- `Cryptography` - provides essential functionalities such as digital signature generation, verification and generation of the public and private keys for the wallets of the participants. Its role is critical in maintaining the confidentiality and authenticity of transactions and protection of the identities of the wallet owners;
- `MerkleTree` adapts the concept "Merkel Tree" algorithm and `Sha256Hash` hashing. In the implementation of the BFS, the leaf nodes of the tree contain hashes of the serialized contents of the financial statements data stored in the accounting blocks of the BFS filing history blockchain. This structure provides a highly efficient mechanism for verifying the integrity and completeness of reported accounts within the BFS system.
- `SecretStore` together with `SecretHash` - manages the secure storage and retrieval of cryptographic secrets, utilised in the HTLC implementation by the transaction wallets of the BFS ecosystem. It ensures that sensitive information is kept secure and is accessible only to authorised entities, thereby bolstering the BFS system's security architecture. The use-case specific element of this class are as follows:
  1. `Secret String` - X - is a `String` that is generated by the `Wallet`, (that has generated and shared with counterparty its transaction proposal) and is stored in the wallet's `SecretStore::HashMap<Secret String, SecretHash> mySecretHashsHashStore`. A one way SHA256 hash function is applied in this string to generate Y (`String secretHashString_Y`) which is the variable of the `SecretHash` class.
  2. `SecretHash` - does not contain `Secret String`. Its variables are:
     (a) `String transactionLifeCycleID` - Transaction Life-Cycle ID;
     (b) `String sourceTransactionHash` - Source Transaction Hash;
     (c) `String secretHashString_Y;` - Secret Hash (Y);
     (d) `PublicKey secretProviderPublicKey` - Secret Provider Public Key;
  3. The Secret String generation use-case based example will be demonstrated in Chapter 7 of this report.
- `Constants` - the structure, implementation and the utilisation of this class was described in Section 6.2.1 and illustrated in Fig. 6.18 of this report.

The described design and implementation of the validator package reflects integration of both blockchain technology and financial accounting processes, marking a significant step forward in the evolution of financial systems, so to enhanced transparency, efficiency and integrity in financial reporting.

6.2.6 UTXO and Consensus Context

This section describes the unique application of the concept of the UTXO and the consensus. It will underline the significance of these concepts within the BFS, particularly emphasising their integration with accounting practices and ledgers. In existing applications of blockchain technology, the concepts of UTXO (Unspent Transaction Output) and consensus mechanisms stand as critical architectural elements that underpin the functionality and security of diverse blockchain ecosystems. These concepts are instrumental in defining the operational dynamics and integrity of blockchain networks, facilitating the decentralised verification and recording of transactions. Traditionally, UTXO models offer a unique method for tracking transaction outputs that are yet to be spent, serving as a fundamental component of the transaction verification process in several blockchain architectures (Rohrer et al. (2017)). Similarly, consensus mechanisms ensure that all participants in the network agree on a single version of the truth, thereby achieving distributed trust and preventing double-spending without the need for a central authority (Garay et al. (2015)).

However, in the context of the BFS, both UTXO and consensus mechanisms are re-imagined and adapted to address the specific needs of integrating blockchain technology with established accounting practices, ledgers and are designed to address validity of monetary claims made by transacting counterparties within the BFS ecosystem. This adaptation provides a novel mechanism for financial reporting and liquidity management by leveraging the inherent transparency, security and efficiency of blockchain technology.

Within the BFS, the UTXO is not a data model, but a funds verification process designed as a novel mechanism and component of the internal accounting process. This mechanism is then utilised by the `Validator` during "funds verification consensus" implementation (see Section 6.2.5) to verify to a transacting counterparty that the monetary claim, made by the member wallet of host BFS is correct, without revealing the internal business sensitive data. Through this specialised consensus approach, the BFS system facilitates real-time auditability, enhancing trust among participants.

At a high-level the traditional process for construction of the UTXO data structure is as follows: the initiation of a transaction involves selecting of transaction inputs, which are essentially cryptographic references to the latest outputs from preceding transactions. These references indicate the available balance of a hypothetical cryptocurrency that can be utilised for new transactions. The value of these inputs, when aggregated with the transaction's intended amount, culminates in the final transaction output. This process ensures that any residual balance - the difference between the input value and the transaction's required amount - is earmarked for future expenditures, effectively creating a new UTXO.

One of the data elements of the `AccountingTransaction` class is the field for transactionInputVerificationInternalData, named `transactionInputVerificationInternalData`.

The `TransactionInputVerificationInternalData` class holds data related to verification of transaction input values, including: details about the economic event associated with this transaction and the verification results for input values for the debit and credit side of the accounting equation (Stolowy & Ding (2019)). This class represents the adaptation of the UTXO process by the BFS. It is also a central opponent of the BFS "funds verification consensus" mechanism, implemented by the `Validator` during the business transaction settlement stage. Furthermore, during execution of business transactions conducted between reporting periods, as part of the "funds verification consensus", the call to the `Validator`'s `verifyTransactionInputValues (BusinessTransactionDataInterface<?> businessTransactionDataInterface)` method is initiated, which utilises the `TransactionInputVerificationInternalData` class to verify if enough funds are available to cover transactional obligations of transacting counterparties.

Transaction Input

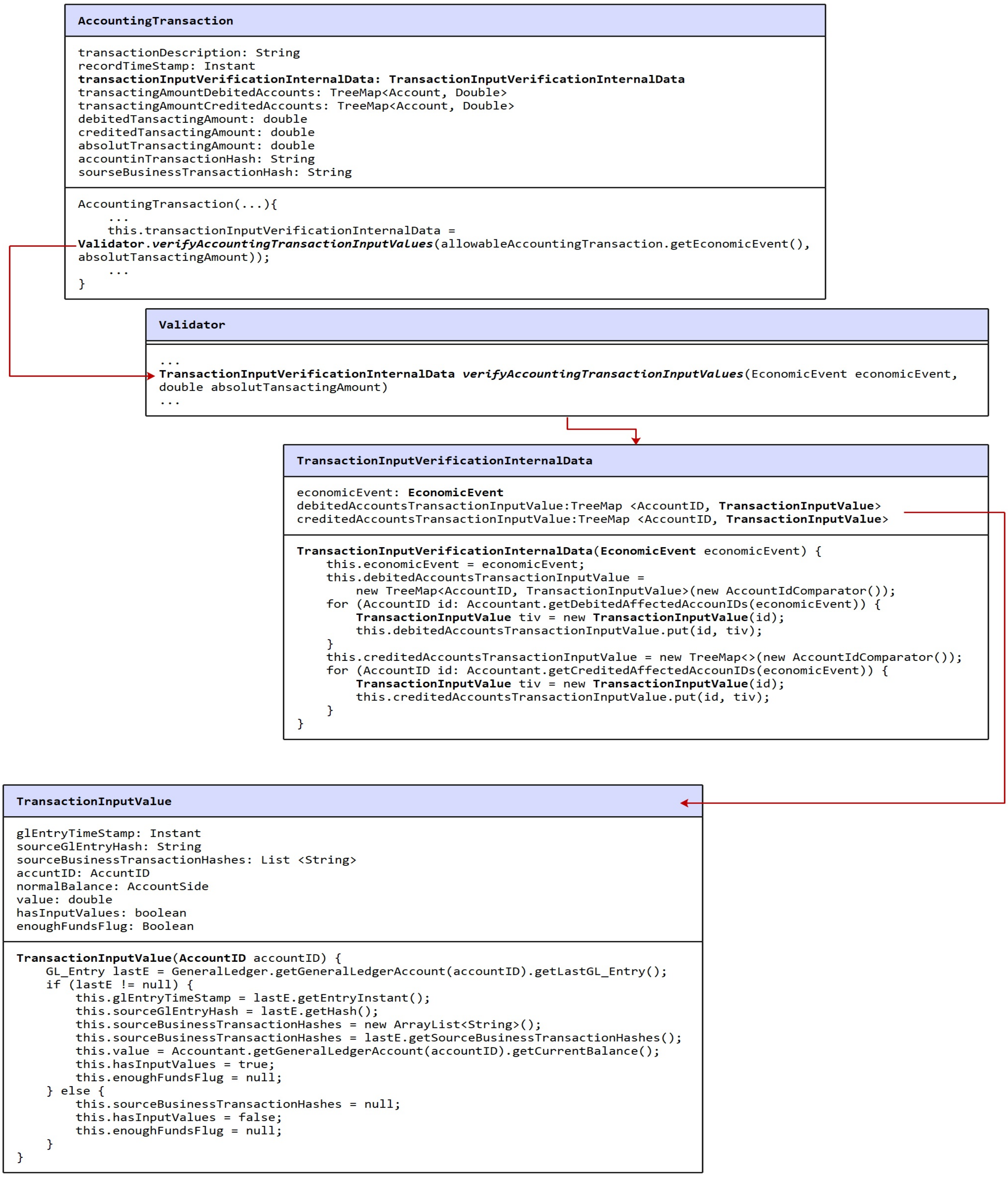


Figure 6.25: Transaction Input.

In the BFS, the implementation of the UTXO process is integrated within accounting. Fig. 6.25 illustrates the data structure and the implementation flow of the transaction input generation within the BFS.

This process starts within `AccountingTransaction` class, where the final generated transaction input value, represented by the `TransactionInputVerificationInternalData` class in the BFS design is stored for future accounting referencing and validations, amongst other uses (see Fig. 6.25 variables of `AccountingTransaction` class). In the constructor of this class, amongst other things, the call to a method of the `Validator` class is made (see Section 6.2.5 describing the `Validator`). As illustrated in the simplified version of the `Validator` class in Fig.6.25, the return Object of this method is a newly created `TransactionInputVerificationInternalData` class. This class acts as a container for the transaction input values, adapted to represent input values for both debited and credited accounts within the BFS. The data structure for this class is as follows:

- ❖ `EconomicEvent economicEvent` - is the economic event of the Accounting Transaction that has triggered the generation of this transaction input values. This enumerator is passed as a parameter of the `TransactionInputVerificationInternalData` constructor (see Fig. 6.25). This enumerator is utilised in the call to `Accountant. getDebitedAffectedAccounIDs(economicEvent)` method to retrieve from the `AccountingTransactionRegister` :: `HashMap <String, AllowableAccountingTransaction> allowableAccountingTransactions` a set of `AccountIDs` to be used for `debitedAccountsTransactionInputValue` and `creditedAccountsTransactionInputValue` described next;
- ❖ `TreeMap <AccountID, TransactionInputValue>` for `debitedAccountsTransactionInputValue` and for `creditedAccountsTransactionInputValue` represent respective maps that associate `AccountIDs`, obtained in the previous step, with their respective `TransactionInputValues`. These values are the distinct transaction input values derived from debited and the credited general ledger accounts respectively, in accordance with double entry accounting.

  The `TransactionInputValue` class encapsulates the essence of an individual transaction input, consolidating data necessary for the BFS's UTXO model. It is designed to incorporate both blockchain-related data and traditional accounting information, thus serving as a bridge between these two domains. The `TransactionInputValue` class is designed to extract and utilise the most up-to-date and relevant values for the transaction input data directly from the last or latest entry within a general ledger account (see Section 6.1.2). This approach ensures that each transaction input reflects the current financial state and historical transactional context of the account, thereby maintaining the integrity and accuracy of the BFS ecosystem's financial records.

  The innovative functionality of the `TransactionInputValue` class is underpinned by its ability to access and interpret the most recent `GL_Entry` associated with a specific general ledger account, combined with `currentBalance`of the this `GeneralLedgerAccount`. This is crucial for several reasons:

  1. *Current Financial State*: The latest `GL_Entry` contains the most current balance of the account, which is essential for verifying the sufficiency of funds for upcoming transactions. By basing transaction inputs on this entry, the `TransactionInputValue` ensures that validations and subsequent transactions are grounded in the most recent financial data.
  2. *Historical Transactional Context*: Beyond just the current balance, the latest `L_Entry` encapsulates the historical context of the account's transactions. This includes information on source transaction hashes and the timestamp of the last entry. Such data is critical for constructing a comprehensive view of the transaction history, aiding in auditability and traceability within the BFS.

  The `TransactionInputValue` class structure includes several key attributes (see Fig. 6.25):

  – `Instant glEntryTimeStamp` - records the timestamp of the most up-to-date `GL_Entry` entry, providing a chronological context for the transaction input.
  – `String sourceGlEntryHash` - a cryptographic hash of the same `GL_Entry`. This hash serves as a unique identifier, ensuring the traceability and integrity of the transaction input back to the data original source.
  – `List <String> sourceBusinessTransactionHashes` - that contains hashes of the source business transactions. This list links the transaction input to its originating business

transactions, facilitating an audit trail.

- AccountID accuntID - object that identifies the account associated with the transaction input. This identification is crucial for attributing the transaction input to the correct ledger account.
- AccountSide normalBalance - enumeration that indicates the normal balance side (debit or credit) of the account, aligning with existing accounting principles.
- double value - represents the value (in monetary units) of the transaction input. This value is critical for the financial calculations and validations performed by the BFS. The high-level process to obtain this value is as follows:
  * the latest balance for each of such AccountIDs is extracted from the currentBalance of the relevant GeneralLedgerAccount, referencing the most recent update from GL_Entry (see 2), otherwise it is set to 0 (zero). (See last columns in Tab. 5.3 - Tab. 5.5 for visual reference);
- boolean hasInputValues- this flag indicates if general ledger account related to this accuntID already had transactional entries (true), or if it is a newly created Account (false);
- Boolean enoughFundsFlug - a wrapper object used for boolean flag to indicate whether the account has sufficient funds for the transaction. This flag is essential for the "enough funds verification consensus" within the BFS ecosystem. The wrapper class is used to add a null value to the default true/false of the boolean.

The implementation of the TransactionInputValue class involves initialising the the attributes outlined based on data from the most recent GL_Entry and its GeneralLedgerAccount contained in the general ledger. The constructor of this class (see Fig. 6.25) is designed to facilitate this initialisation process in two contexts: when GL_Entry is available, i.e., there were already some entries made in the general ledger account within this accounting period; and when this data is absent, for whatever reason.

This constructor demonstrates how the TransactionInputValue class integrates GL_Entry and currentBalance of the relevant GeneralLedgerAccount data into the BFS's transaction validation process. By capturing the GL_Entry timestamp, source GL_Entry hash and business transaction hashes, along with the account's current balance and normal balance side, the class effectively consolidates transaction input data in accordance with both blockchain and accounting standards. And by utilizing the latest GL_Entry with currentBalance of the relevant GeneralLedgerAccount for extracting transaction input data, the BFS ensures that all financial claimed, made during business transactions are validated against the most current and comprehensive financial information available, thereby safeguarding the integrity of the BFS ecosystem's financial records.

Transaction Output

In traditional blockchain systems, the generalised concept of a transaction output plays an important role in the construction of the Unspent Transaction Outputs (UTXO) data structure. At its core, a transaction output represents the mathematical resulting outcome of a transaction process, where the initiation of a transaction involves the selection of inputs. These inputs are cryptographic references to the latest outputs from preceding transactions, indicating the available balance that can be used for new transactions. When these input values are aggregated with the transaction's intended amount, the resulting value is the final transaction output. This output signifies the completion of a transaction and generates future residual balance — which is the difference between the total input value and the transaction's required amount - earmarked for future expenditures. Thus, each transaction output finalises a current transaction and serves as an input for future transactions, thereby creating a new UTXO.

Within the BFS, the design for the transaction output follows similar generalised model, but it is implemented in a novel way, to accommodate traditional accounting practices within it. In this innovative approach, a transaction output is conceptualised as the most up-to-date entry of the General Ledger Account - the `GL_Entry` class, a design choice that bridges the gap between blockchain technology and the principles of financial accounting.

Each `GL_Entry` class, as an individual entry of the `GeneralLedgerAccount`, combined with `currentBalance`of the `GeneralLedgerAccount` (see Section 2) and the last columns in Tab. 5.3 - Tab. 5.5 for visual reference) within the BFS conceptualises the UTXO inspired transaction output, by encapsulating the evolving financial implications of ongoing transactions within the general ledger repository of an economic entity. This class is designed to record the value change that results from the transformation economic value, produced by completed business transaction, into a journal entry record within the general journal ledger, thus serving as the transaction output (see Fig. 2.1 steps 1-4 of the accounting cycle flow).

The structure and attributes of the `GL_Entry` class were described in Section 6.1.2,2,2, and illustrated in Fig. 6.7. The elements of the `GL_Entry` that are designed to link `GL_Entry` to a transaction output are:

1. `gL_Entryhash` - ensures the integrity and non-reputability of the transaction output;
2. `glAccountID` - linking transaction output directly to specific ledger accounts;
3. `sourceAccountIDs` - showcasing the hierarchical nature of accounting records that reflects consolidation of outputs of the related transactions;
4. `entryInstant` - which marks the chronological placement of the transaction output within the financial records
5. `debitValue` and `creditValue` - reflecting the financial impact of the transcriptional amount;
6. `sourceAccountingTransactionHashes` and `sourceBusinessTransactionHashes` - provide a comprehensive audit trail.

Lastly, the `currentBalance` of the relevant `GeneralLedgerAccount` referencing the most recent update from `GL_Entry`, represents the final monetary value utilised in the transaction output; this value is the resulting impact on the overall value of the account after debiting or crediting the impact of transaction (see last columns in Tab. 5.3 - Tab. 5.5 for visual reference).

Consensus

In the ecosystem of a diverse blockchain implementation, consensus mechanisms play an important role in maintaining the integrity and security of distributed no trust networks. In general, consensus process flow represents the platform-specific mechanisms for transaction relaying, verification and confirmation, time-ordering, smart contract execution and commitment (European Central Bank and Bank of Japan (2018)). Traditional consensus models, such as those employed by Bitcoin and Corda, showcase diversity of approaches to achieving network agreement on the validity of transactions. Bitcoin utilises *Proof of Work (PoW)* consensus mechanism, where miners compete to solve complex cryptographic puzzles. The first miner to solve the puzzle gets the right to add a new block of transactions to the blockchain. This process ensures security and decentralisation but at the cost of significant energy consumption and slower transaction speeds (Nakamoto (2008)). Unlike Bitcoin, Corda, designed for enterprise use, employs a unique consensus mechanism involving notaries - *Corda's Notary Consensus*. These notaries validate the uniqueness and finality of transactions without the need for energy-intensive mining. Corda's approach focuses on ensuring transaction integrity and preventing double-spending within its business-oriented network (Brown et al. (2016)).

Contrasting these, the “funds verification consensus” within the BFS introduces a distinct novel approach to consensus that deviates from traditional existing mechanisms. This consensus model is specifically designed to verify funds availability for transactions within a blockchain-integrated accounting framework. At its core, the “funds verification consensus” is built upon traditional consensus ideas of providing an agreement mechanism between ecosystem participants to trust each other's claims and enable them to proceed to the settlement of their transactions. However, this consensus is uniquely integrated with accounting principles, utilising internal, up-to-date accounting data of the economic entity to verify to the counterparty if the transactional obligations are capable of being settled. Furthermore, unlike the broad network consensus sought in Bitcoin and Corda, the “funds verification consensus” focuses on the bilateral verification between transacting parties, leveraging concepts from Zero-Knowledge Proofs (ZKP) for privacy preservation idea, where it only communicates to the transacting counterparty if the monetary statement made is true or false, without revealing any information beyond the validity of the statement itself. In the context of the BFS, this means that entities can verify the availability of funds for a transaction without disclosing the actual balance or details of their financial accounts.

Validator

```
...
boolean verifyTransactionInputValues (BusinessTransactionDataInterface<?> businessTransactionDataInterface){
    boolean ennoughFunds;
    EconomicEvent ee = businessTransactionDataInterface.getEconomicEvent(BFS.getCompanyInformation().getCompanyNumber());
    TransactionInputVerificationInternalData tvd = new TransactionInputVerificationInternalData(ee);
    double transactionValue = businessTransactionDataInterface.getTotalSettledtAmount();
    double avaliableFunds = 0;
    double debVal = 0;
    for (Map.Entry<AccountID, TransactionInputValue> v: tvd.getDebitedAccountsTransactionInputValue().entrySet()) {
        debVal += v.getValue().getValue();
    }
    double crVal = 0;
    for (Map.Entry<AccountID, TransactionInputValue> v: tvd.getCreditedAccountsTransactionInputValue().entrySet()) {
        crVal += v.getValue().getValue();
    }
    if ((debVal - crVal)==0) {
        avaliableFunds = debVal;
        if(transactionValue <= avaliableFunds) ennoughFunds = true;
    }
    return ennoughFunds;
}
...
```

TransactionInputVerificationInternalData

```
economicEvent: EconomicEvent
debitedAccountsTransactionInputValue:TreeMap <AccountID, TransactionInputValue>
creditedAccountsTransactionInputValue:TreeMap <AccountID, TransactionInputValue>
```

```
TransactionInputVerificationInternalData(EconomicEvent economicEvent) {
    this.economicEvent = economicEvent;
    this.debitedAccountsTransactionInputValue =
        new TreeMap<AccountID, TransactionInputValue>(new AccountIdComparator());
    for (AccountID id: Accountant.getDebitedAffectedAccounIDs(economicEvent)) {
        TransactionInputValue tiv = new TransactionInputValue(id);
        this.debitedAccountsTransactionInputValue.put(id, tiv);
    }
    this.creditedAccountsTransactionInputValue = new TreeMap<>(new AccountIdComparator());
    for (AccountID id: Accountant.getCreditedAffectedAccounIDs(economicEvent)) {
        TransactionInputValue tiv = new TransactionInputValue(id);
        this.creditedAccountsTransactionInputValue.put(id, tiv);
    }
}
```

TransactionInputValue

```
glEntryTimeStamp: Instant
sourceGlEntryHash: String
sourceBusinessTransactionHashes: List <String>
accuntID: AccuntID
normalBalance: AccountSide
value: double
hasInputValues: boolean
enoughFundsFlug: Boolean
```

```
TransactionInputValue(AccountID accountID) {
    GL_Entry lastE = GeneralLedger.getGeneralLedgerAccount(accountID).getLastGL_Entry();
    if (lastE != null) {
        this.glEntryTimeStamp = lastE.getEntryInstant();
        this.sourceGlEntryHash = lastE.getHash();
        this.sourceBusinessTransactionHashes = new ArrayList<String>();
        this.sourceBusinessTransactionHashes = lastE.getSourceBusinessTransactionHashes();
        this.value = Accountant.getGeneralLedgerAccount(accountID).getCurrentBalance();
        this.hasInputValues = true;
        this.enoughFundsFlug = null;
    } else {
        this.sourceBusinessTransactionHashes = null;
        this.hasInputValues = false;
        this.enoughFundsFlug = null;
    }
}
```

Figure 6.26: Enough Funds Consensus.

Fig. 6.26 illustrates simplified logic of `verifyTransactionInputValues` method of `Validator` class. Within the BFS, the utilisation of the `verifyTransactionInputValues` method represents an alternative novel approach to ensuring transaction integrity and funds verification within a blockchain-based accounting system. This method illustrates at the high-level practical, executable verification process that aligns with the BFS's overarching aim to integrate blockchain technology with traditional accounting principles. The principles that underpin the design of this method are as follows:

- *Integration with Accounting Framework.* It utilises transaction data and economic events to ascertain the availability of funds for transactions. It seamlessly integrates with the BFS's accounting framework, utilizing transaction data and economic events to ascertain the availability of funds for transactions.
- *Utilization of Zero-Knowledge Proofs (ZKP) Concepts.* While not implementing ZKP in its pure form, the method embodies the spirit of ZKP by ensuring that verification of funds does not compromise the privacy of the account's balance or detailed financial information.
- *Efficiency and Accuracy.* By directly comparing the transaction value against available funds within the relevant accounts, the method ensures both the efficiency of the verification process and the accuracy of its outcomes.
- *Enhancement of Transaction Security and Integrity.* By verifying that transactions are supported by sufficient funds, it supports trust in the transactional interactions between transacting counterparties, thus preventing fraudulent or erroneous transactions. the

The `Validator` utilises `TransactionInputVerificationInternalData` and `TransactionInputValues`, which are derived for transactional inputs and outputs as described earlier. These constructs ensure that every transaction input is backed by a verifiable output, thus maintaining the integrity of the financial ledger (see Fig. 6.26). The `TransactionInputValues` reflect the outcome of transactions, including the updated account balances, without exposing the underlying financial data. In Fig. 6.26, the implementation of the `verifyTransactionInputValues` method demonstrates the practical application of these design principles. The key features there are:

- `Economic Event Processing` - method begins by extracting the `EconomicEvent` associated with the transaction, demonstrating the integration with the accounting framework and ensuring that each transaction is evaluated within its specific economic context.
- *Calculation of Available Funds* - calculates available funds based on the `TransactionInputVerificationInternalData`, which combines both debited and credited accounts' values, reflecting the method's adherence to accounting principles of balancing.
- *Funds Availability Verification.* The core of the method lies in comparing the transaction value against the available funds. It ensures that enough funds are available to cover the transaction, adhering to the principle of economic validity. This feature directly reflects the consensus's goal to ensure transactions are financially sound and backed by adequate funds without revealing sensitive financial data.

The design and implementation of the `verifyTransactionInputValues` method of the `Validator` class address the practical challenges of funds verification and transaction integrity and provides innovation scope for future advancements in blockchain-based financial systems, where security, privacy and efficiency are paramount. The method enhances the financial system's integrity by ensuring every transaction is backed by sufficient funds while simultaneously preserving the privacy of financial data, a critical consideration for businesses. By enabling immediate verification of transactions based on current financial states, it supports real-time auditability and financial oversight, aligning with the broader objectives of the BFS to improve financial reporting and management practices.

Novelty of the BFS (UTXO and Funds Verification)

The integration of blockchain technology and its selected components with traditional financial accounting practices, as illustrated in the BFS, presents a novel approach to addressing problems of the financial reporting and liquidity management. This section will articulate how this process responds to the research challenges and contributes to the theoretical and practical advancements in integration of blockchain with accounting. The mechanisms for design and the implementation of the transaction input and output demonstrated in this report are instrumental in providing solutions to the research questions posed, specifically targeting the enhancement of real-time auditability, fraud prevention and the establishment of secure liquidity distribution mechanisms.

1. *Enhancing Real-Time Auditability and Fraud Prevention*: The design and implementation methodology for transaction input and output within the BFS directly confronts the challenge of fraudulent financial activities and unreliable financial reporting borne from lack or real-time auditability mechanisms in the financial accounting domain.
   (a) *Real-Time Financial Reporting against Fraud*: By generating transaction inputs and outputs through the `GL_Entry` class and instantaneously reflecting changes in the `urrentBalance` of this general ledger account, the BFS addresses this issues. By offering nearly real-time visibility into the financial status of accounts, enhances auditability and financial oversight. Furthermore, the immediate reflection of transactional impacts on account balances aids in detecting discrepancies and preventing fraudulent activities.
   (b) *Economic Validity of Funds Availability Claims*: The process also validates the economic validity of funds availability claims by leveraging the `TransactionInputValue` class. This class, by referencing the most current ledger entries, confirms the availability of funds in real-time, thereby providing a trustworthy foundation for financial statements and liquidity claims.
   (c) *Audit Processes Are Streamlined*: The close to real-time update of the current balance for the general ledger account facilitates near-instantaneous auditability, significantly reducing the window for undetected fraudulent activities.

   This ensures the reliability and integrity of the aggregated history of economic data, effectively addressing the real-time auditability problem.
2. *Establishing Direct, Secure Connections for Liquidity Support*: The integration of transaction input and output generation within the BFS presents how blockchain technology and its communication models can be utilised to develop a liquidity distribution system based on the verifiable auditability mechanisms, that are both secure and aligned with the needs of modern financial systems.
   (a) *Funds Availability Claims Are Verifiable*: The `currentBalance` in the `GeneralLedgerAccount`, updated with each `GL_Entry`, provides a reliable measure for verifying the availability of funds, ensuring that liquidity support is distributed based on accurate financial claims.
   (b) *Compliance With Distribution Rules Is Automated*: The transaction input generation mechanism enables automated compliance with liquidity distribution rules, though truth verification response provided by `Boolean enoughFundsFlug`, an integral element of the `TransactionInputValue` class. The blockchain-based framework for the BFS and for the generation of the input and output values automates this compliance with the arbitrary rules set for liquidity distribution, where each `GL_Entry`, the `Boolean enoughFundsFlug` and the `currentBalance` value together act as a verifiable record of compliance.
3. *Leveraging Blockchain for Secure Liquidity Distribution Mechanisms*: The overall BFS's design philosophy aligns with the necessity for a secure, direct and tamper-evident connection between heterogeneous economic entities and liquidity providers, such as governments or central banks. This connection facilitates:
   (a) *Financial Claims Are Tamper-Evident*: the cryptographic hash value, generated based on the constituent data of the `GL_Entry` and stored within it, serves as an immutable record of financial activity, making any attempts at fraud immediately evident and traceable. This adaptation of blockchain cryptographic practices, with its inherent tamper-evident ledger capabilities, strengthens these controls, making every transaction

and its historical lineage verifiable and immutable.

(b) *Verifiable Authenticity Controls*: By integrating the transaction input and output generation with the latest entries in the general ledger, the BFS provides a verifiable authenticity control mechanism for monetary trust between BFS ecosystem participants. By embedding these mechanisms within the blockchain's consensus mechanism and utilizing ZKP verification logic, the BFS system ensures that only transactions meeting predefined criteria are processed. This approach guarantees the integrity of transactional data and its chronological validity.

(c) *Post-Reporting Period Reconciliation Mechanism*: The close to real-time nature of the verification of financial claims made provides this needed mechanism that can be utilised between reporting periods. This most up-to-date state of the financial health of an organisation is less evident, as more time elapses since the last public financial report. The BFS incorporates such a post-reporting period reconciliation mechanism within its consensus protocol, which authenticates the validity of financial claims made by business entities. This way, the funds verification consensus ensures a tamper-evident transactional truth by automation of this verification process, addressing the need for a reliable authentication mechanism for financial claims.

By integrating blockchain technology with established accounting practices, the BFS addresses the research questions posed and sets a new standard for the future of financial reporting and liquidity management. It provides a reliable, verifiable and tamper-evident mechanism focused on ensuring the integrity of financial statements and the validity of funds availability claims.

The BFS therefore illustrates the realisation of promised potentials of blockchain technology in designing and developing secure and trusted financial systems, paving the way for innovative solutions in the distribution of financial support.

6.3 Implementation of Entity Sub-Domain

This section explores a framework within which business entities exist and operate in the BFS ecosystem. This exploration takes into account distinct aspects that contribute to an entity's capacity to exist, engage in economic activities, management of liquidity and overseeing asset strategies fundamental to the economic structure. This is where the BFS as an entity begins to take form, marking its entry into the economic landscape through processes such as incorporation and by defining its operational blueprint that lays out the execution of its unique business functions. The activities and elements within this sub-domain are not merely administrative. They reflect a series of strategic design decisions that drive BFS as an entity towards fulfilling its economic objectives, defined by its use-case. These activities are not limited to defining and orchestrating transactions but also include the essential task of arranging agreements that facilitate liquidity flow, illustrated by the "Commercial Paper Facility" of the APF. It is within this context that the BFS entity engages in securing financial resources through mechanisms like the "Lending Agreement" with the BoE, thereby ensuring the fluidity of its operations and the viability of its financial undertakings.

This sub-domain also outlines the entity's role in business-specific activities. It provides a repository for information critical to operational success, including the registration of members and transactional counterparts, each playing a designated role within the BFS ecosystem.The business entity sub-domain under Domain-Driven Design (DDD) is a cohesive unit encompassing the entity's delineation of responsibilities across various bounded contexts, from the foundational elements of corporate existence to the orchestration of business transactions.

6.3.1 Company Incorporation Context

This section describes the practical steps involved in registering and establishing an economic entity within the BFS framework.It lays out the procedural mechanisms for engaging the BFS for company operations, as demonstrated through the use-case of this research, adopted for APF::BFS. The process of establishing a new entity within the BFS framework, illustrates the initial step enabling the economic entity to exist and engage in business activities. The inception of an entity like the APF is a systematic process initiated by the completion of a registration form, referred to as `IN10` (*Form IN10. Register a private or public company. Application to register a company* (2024)), which is simulated in this research, mapping to the forms required by Companies House UK. The founding directors of the APF are responsible for providing accurate information on this form and submitting it for processing to the `Registrar` of the BFS.

In this research, following the steps of adaptation of real life use-case (Companies House (2023*b*)), the incorporation implementation for the APF is illustrated as a "private limited by shares" LLP, with 2 directors and 1 shareholder, with initial share issuance and transfer transaction implemented at instantiation of the APF::BFS, outlined earlier in Section 6.2.2. In accordance with public historic records (Companies House (2023*b*)), the incorporation of APF took place on 30 January 2009. Next this report provides a narrative description of the steps for the incorporation process for the automated APF "Incorporation protocol" implemented in this research (see (Fig. 5.3)).

1. Two directors of the APF fill out all required information in the IN10 form *Form IN10. Register a private or public company. Application to register a company* (2024);
2. They submit the `IN10` form to the `Registrar` (Companies House (2024*c*));
3. The `Registrar` authenticates and processes submitted documentation Companies House (2024*c*) and generates an instance of the `Private Register` that contains:
   (a) Incorporation documentation such as: `Certificate of Incorporation`, `Memorandum of Association` and so on, and
   (b) Genesis Block. In BFS prototype it is the Genesis `Block` that will be the initial `Block` in the `APF :: BFS :: FilingHistory`
   (c) When `PrivateRegister` instance is generated by the `Registrar`, as part of this process, the Genesis block for the BFS is generated. This Block is shared with the company directors and a copy of this `PrivateRegister` instance is permanently stored in the hypothetical Companies House for security, validity and future reference.
4. The `Registrar` shares this `Private Register` instance with the directors of the APF and stores a copy of it in local database, such as "Companies House" in the UK;

As illustrated in Fig.5.3 - "Company Incorporation Process Flow", the prospective directors of the APF, upon receiving the `PrivateRegister` instance, are now able to instantiate their `APF::BFS` as a "private company limited by shares" Companies House (2023*a*) and commence their economic activity. As part of BFS instantiation process, automation of the following processes is enabled:

1. Instantiation of initial `DirectorWallets`.
   The sensitive identity data and other relevant metadata for these wallets is derived from the form `IN10`, submitted to the `Registrar` by the directors and contained in the `PrivateRegister`. In the BFS prototype two `DirectorWallets` are generated, with functionality in `RelationToCompany` as a `CompanyMember`, and `WalletControlType` as an `ExecutiveDirector`.
2. Instantiation of `BusinessTransactionRegister`.
   It is created at BFS instantiation (see Section 6.2.2). It is an important chronological, off-ledger repository for all ongoing and completed business transactions that the APF's member wallets take part in over its business lifecycle.
3. Instantiation of "Initial `AuthorisedShareCapital`".
   Based on the company type (private company limited by shares) and other relevant metadata from the `IN10` form, at instantiation `APF::BFS`, initial share capital (`Share` tokens) of the APF are generated. This process involves (see Section 6.2.2):

   (a) instantiation of 100 Ordinary Shares for the APF (tokens),
   (b) depositing these into the APF's' `ShareStock` :: `authorisedShares` repository of the `BFS`.
4. Instantiation of an initial `ShareholderWallet`.
   The sensitive identity data and other relevant metadata for this wallet is derived from the form `IN10` (share subscriber part). In the BFS prototype, this wallet acts as a counterparty to the APF in the initial share purchase transaction outlined in Section 7.2. This wallet exercises `FullDirectControle` over the data internal to the APF, because it purchases 100% of the authorised share stock issued at incorporation Companies House (2023*a*,*b*). It is owned by the BoE.
5. Implementation of the 1st Business Transaction.
   Implementation of this transaction is automated, as all required metadata is encapsulated in the `PrivateRegister` :: `IN10` form. The initiator of this transaction is a `ShareholderWallet`, which is owned by the BoE. The specified transaction type for this is: "Initial Share Issuance Transaction". At successful completion of this transaction, a record of it is made in the `BusinessTransactionRegister` of each of the participating counterparty BFS and copied into each of the wallets involved. The process flow for this transaction is presented next.

6.3.2 Business-Specific Components Context

The BFS architecture adopts a novel approach to integrating technological innovation with the complex requirements of transactional and accounting domains. The architecture's foundation is laid on a selected design approach of Domain-Driven Design (DDD), which prioritizes a comprehension of the domain's complexity and its operational logic, as described in Chapter 4.

In this section, attention is on the elaboration of a Business-Specific Context, introduced to the foundational elements that underpin the BFS's design and its operational efficacy. This bounded context encapsulates use-case specific procedural nuances or mechanisms by winch business activity can be enabled. Use-case specific contexts for the implementation of the BFS demonstrates transactional interactions of 3 economic entities, the APF, the BoE and the hypothetical market participant - "Commercial Paper Issuer - 1" (CPI-1). This use-case centered mechanism enables the APF to engage in QE, i.e., to purchase private sector assets, is a hypothetical lending agreement between the APF and the BoE that provides necessary funding for the APF's market activity. The scenario for establishment of the landing agreement is based on the foundational role of the APF's as the BoE's legal counterparty for market transactions - implementation of QE on behalf of the BoE. This arrangement is essential for facilitating the APF's mandate to purchase high-quality, private sector assets, including sterling investment-grade commercial paper, thereby injecting liquidity into the market during times of tightening of credit conditions.

In the initial phase of program, first purchases were financed by the issue of the Treasury Bills by the HMT that were lent to the APF though the BoE (Bank of England (2009*b*)).These funds were utilised to manage "near-term cash flow requirements" (Bank of England (2010*b*) of APF. These were advanced (deposited) to the APF by the BoE upon receiving a notification of the APF's intention to make a draw down under the loan (Bank of England (2010*b*).This establishes a systematic process for the APF to purchase high-quality assets from the private sector, ensuring liquidity management and economic stability. The utilisation of this lending agreement is critical for the business lifecycle of the APF and it is used in Sections 7.4 and 7.5.1 in implementation of the business transaction smart contract between the APF and the CPI-1. Next, the implementable and the data structure for the Lending Agreement between the APF and the BoE is described.

Loan Agreement

In the BFS implementation, the mechanism that enables the APF to engage in QE, i.e., to purchase private sector assets is a hypothetical lending agreement established between the APF and the BoE.

The terms for this funding are based on the use-case of this research and the funding is secured through the issuance of Treasury Bills lent to the APF via the BoE.

The pre-conditions for this funding process are based on the use-case and set-up as follow:

- In order to enable APF to purchase Commercial Paper from CPI-1, financial resources are needed to carry out its asset purchase program effectively.
- These resources are provided to the APF by the BoE in the form of tokenized Treasury Bills issued upon a drawdown request.
- This drawdown request communicates to the BoE the value and the maturity of the tokenized Treasury Bills determined by the value and the maturity of the tokenized Commercial Paper, offered to the APF by the CPI-1, i.e. to cover "near-term cash flow requirements" of the APF.
- This funding mechanism is critical for managing the APF's financial obligation to the heterogeneous market participants, such as CPI-1.

In this Thesis, to enable funding flow for the APF, thus enabling APF to swap illiquid CP issued by private sector (a hypothetical market participant - "Commercial Paper Issuer - 1" (CPI-1)) for highly liquid Treasury Bills borrowed from BoE an illustrative lending agreement (or lending facility) between the APF and the BoE has to be established. An example of the sequential step by step implementation of this lending agreement is provided in Section 7.4. Below the description of the data structure for the `LoanAgreement` class is outlined. The establishment of this lending agreement enables the APF to manage its liquidity requirements of the APF by making drawdowns on this loan by receiving transfers of Treasury Bills from the BoE.

In the BFS prototype, the `LoanAgreement` is another smart contract mechanism that serves as an illustrative representation of this agreement between a single lender (the BoE) and a single borrower (the APF). This contract is designed to implement and automate terms of a generalised lending facility that can facilitate establishment and digitisation of contractual rules for verification and approval (or refusal) of drawdown requests made by a hypothetical borrower, such as the APF, to the hypothetical lender such as the BoE. This approach ensures the automated, efficient execution of agreement terms, aligning with the BFS's overarching objective of leveraging blockchain technology to streamline complex financial transactions. The `LoanAgreement` class within the Java prototype serves this purpose, encapsulating the terms, conditions and operational logistics of the lending agreement between the APF and the BoE.

As part of this process, two additional specialised wallets are designed and implemented in the BFS. These are the `BorrowerWallet` and the `LenderWallet` representing specific critical roles necessary for execution of the `LoanAgreement`. The elaboration on these wallets will be provided further in Section 6.3.6 and 6.3.6, respectively.

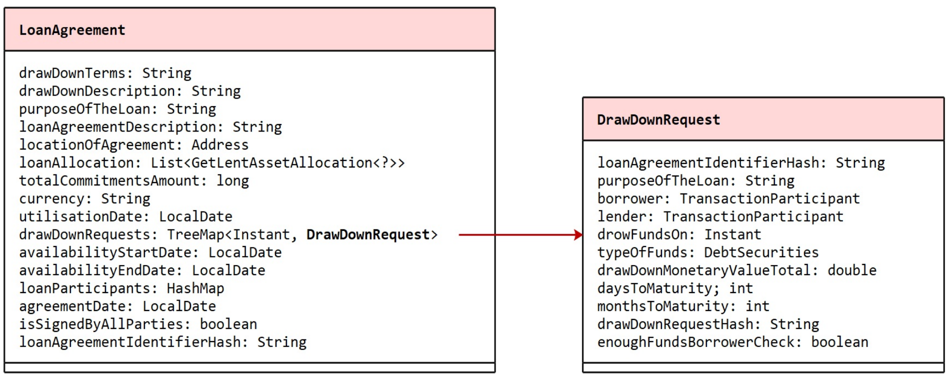


Figure 6.27: Loan Agreement and Draw-Down Request.

In Fig. 6.27, the data structure of the two main components that enable set-up and the execution of the lending agreement contract and the subsequent drawdowns of funds is described. These are

LoanAgreement and DrawDownRequest Java classes. LoanAgreement is set up between the APF as the BorrowerWallet and the BoE as the LenderWallet. An identical copy of the LoanAgreement is saved by both parties of this agreement. The DrawDownRequest is contracted between by the APF as the BorrowerWallet and sent to the the LenderWallet where it is evaluated and acted upon in accordance with verification conditions set-out within the LoanAgreement.

The attributes of the LoanAgreement class (see Fig. 6.27) are as follows:

- String drawDownTerms- describe free form human readable contractual conditions under which drawdowns can be made. These can be utilised for referencing.
- String drawDownDescription - describes the rationale behind each drawdown, such as immediate cash flow needs, or use-case specific“near-term cash flow requirements”;
- String purposeOfTheLoan - details the reasons for the loan, which must align with each drawdown request;
- String loanAgreementDescription - describes the nature of the loan agreement;
- Address locationOfAggeement - indicates the jurisdiction and address for the agreement, which is often defaults to the lender’s address, such as the BoE;
- List<GetLentAssetAllocation<?>> loanAllocation - outlines what type of funds or assets are lent. These are mapped to the enumerator DebtSecurities, which covers use-case specific tokenized debt assets such as: TreasuryBill, Gilt, CommercialPaper, CorporateBond, and OtherDebtSecurities.
- long totalCommitmentsAmount - represents the total amount the facility is authorised to lend;
- String currency - the currency in which the loan is denominated;
- LocalDate utilisationDate - marks the date on which the loan can be utilised;
- TreeMap<Instant, DrawDownRequest> drawDownRequests - tracks the history and details of each drawdown request;
- LocalDate avaliabilityStartDate and LocalDate avaliabilityEndDate - define the loan availability period;
- HashMap <PublicKey, LoanParticipant> loanParticipants - contains the cryptographic identities and additional details of all participants in the loan;
- LocalDate agreementDate - records the date the agreement was signed;
- boolean isSignedByAllParties - a confirmation flag marking that all parties have signed the contract;
- String loanAgreementIdentifierHash - provides a unique cryptographic identifier for the loan agreement.

The DrawDownRequest class also contain methods to validate, execute, calculate and report on the drawdown request. These methods ensure that each request adheres to the predetermined conditions of the loan agreement and accurately reflects the transaction’s details.

- String loanAgreementIdentifierHash - this unique cryptographic hash ties the drawdown request to its specific loan agreement. It is a copy of the LoanAgreement:: loanAgreementIdentifierHash;
- String purposeOfTheLoan - the purpose must be specified and should match the agreed-upon terms in the LoanAgreement;
- TransactionParticipant borrower and TransactionParticipant lender - objects represent the APF and the BoE, respectively;
- Instant drowFundsOn - indicates the date on which the APF intends to draw funds.
- DebtSecurities typeOfFunds - denotes the type of security, such as Treasury Bills, involved in the transaction. This type is then mapped to the LoanAgreement:: loanAllocation;
- double drawDownMonetaryValueTotal - specifies the total monetary value intended to be drawn down;
- int daysToMaturity - provide the number of days for requested (to be borrowed) debt security’s maturity;

- ❖ `int monthsToMaturity` - provide the number of months for requested (to be borrowed) debt security's maturity;
- ❖ `String drawDownRequestHash` - serves as a unique cryptographic identifier for each draw-down request;
- ❖ `boolean enoughFundsBorrowerCheck` - a boolean indicator that confirms whether the borrower has enough available funds in the loan facility to fulfil this request.

This implementation model for the `LoanAgreement`

ensures that the BFS's smart contract mechanism accurately models the complex interaction of financial operations, securing the systematic and automated execution of financial obligations within the BFS ecosystem participants. This automation streamlines compliance process with contractual lending rulers and demonstrates how financial support rules, such as these outlined in the problem description Section 1.1 (*Problem 1*), can be designed and implemented by the BFS.

The high-level implementation and functionality of the "Loan Agreement" smart contract progresses through a series of phases, including:

- ❖ The establishment of the lending agreement's terms.
- ❖ The configuration of the smart contract to match the lending facility's operational requirements.
- ❖ Establishes allocation of the total amount and the type of the lent funds - `LentAssetAllocation` (such as `DebtSecurities.TreasuryBill`, in this example);
- ❖ Verifies counterparties of this lending facility - `LoanParticipant`; ;
- ❖ Keeps track of successful draw-downs in the local repository of `DrawDownRequest`;
- ❖ Verifies if requested `drawDownMonetaryValue` is still available by the lending facility;
- ❖ Provides `loanAgreementIdentifierHash` for query;
- ❖ The integration of the contract into the BFS framework to allow APF to manage liquidity by drawing down on the loan and receiving Treasury Bills from the BoE.

After agreeing the terms of the lending facility, the `LoanAgreement` contract is stored in the `LoanAgreementRegister` - an off-ledger repository of each BFSs: the BoE and the APF, with copies in participating wallets. The `LoanAgreementRegister` is the repository for all lending facilities, global to each company. The wallet's copy of this repository stores only agreements that this `Wallet` is party to.

The "Loan Agreement" smart contract simulation is a strategic component of the BFS infrastructure, enabling the APF to fulfil its QE operations. It demonstrates how components of blockchain functionality can be adapted and harnessed to facilitate complex financial agreements and enforce financial obligations, offering transparency, security and efficiency in financial processes. The successful implementation of this mechanism within the BFS framework showcases the feasibility and effectiveness of smart contracts in automating and managing financial agreements within the realm of digital finance.

6.3.3 Business Transaction Elements Context

The Use-case specific element of the APF::BFS is THE `ShareStock shareStock` repository. It is a business type specific repository for the private or public company limited by shares. This repository records all share issuance and sale related activity of the entity. This is particularly relevant for the design and implementation of the use-case specific BFS, Based on the empirical data outlined in the Section 2.4.3, the first transaction implemented between BFS entities, the APF and the BoE is the sale to the BoE of all 100 authorised shares issued by the APF. In accordance with the use-case, once the APF::BFS is instantiated, as per IN10 specifications, the APF::BFS issues 100 Ordinary `Share` tokes with par value £1.00 per share, destined to its initial member (the BoE). These Shares are stored into the `BFS::ShareStore::authorisedShares` repository of the APF::BFS class (see Section 6.2.2 on BFS instantiation, together with Fig. 6.22). The `ShareStock` element of the BFS is a business type specific repository for the private or public company, limited by shares. This repository records all share issuance and sale related activity of the entity.

As part of this process, a `ShareCertificate`, which represents the transfer of the `Share` token's ownership from the APF to the BoE is issued and included in the data of both transaction proposals. This `ShareCertificate` is not signed yet by any of the relevant counterparties (the issuer/seller, the shareholder/buyer and the witness (Your Company Formations (2024)).and thus is not authenticated as a proof of ownership transfer for the `Share` token from its issuer (the APF) to the prospective shareholder (the BoE). The copy of such an unsigned `ShareCertificate` is stored in the `ShareStok` repository for future reference and until either: the successful completion or the cancellation of whole ongoing business transaction. Execution of such a transaction is complex and requires multiple stages to facilitate atomic DvP settlement business transaction.

Tokenized Assets and Transactional Data

In Fig. 6.28, the overall elements and their data structure are illustrated. These elements are important parts of the BFS's design and the implementation (that is, use-case specific).

In the BFS, the concept of shares as tokenized assets plays a important role in the execution of Delivery versus Payment (DvP) share transactions between the APF::BFS entity (the share issuer) and another BoE::BFS entity (the shareholder). The report now describes the design and implementation of the Share class, highlighting its contribution to the blockchain tokenization of shares and its impact on enhancing the transparency, efficiency and security of share transactions.

In Fig. 6.28, the `Share` class contains all the essential attributes and rights associated with a specific issued share, reflecting the complexities and legal requirements of share issuance and ownership. These were provided by the prospective directors of the APF in the form `IN10`, `Part3_StatementOfCapital` during company incorporation. The design of this class is focused on accurately representing shares in a digital, tokenized form, facilitating their integration into blockchain transactions. The Java implementation of the `Share` class is designed to provide a framework for the creation, management and transaction of shares within the BFS ecosystem.

Key attributes of the Share class include:

- `shareClassName`: The name or designation of the share class, distinguishing it from other types of shares within the same entity.
- `PrescribedParticulars`: Detailed particulars of the rights attached to the shares, including voting rights, dividend participation rights, capital distribution rights and redemption options.
- `currency`: The currency in which the share's par value is denominated.
- `parValue`: The nominal value of the share, representing its minimum issue price.
- `hareParticularsHash`: A cryptographic hash generated from the concatenation of the share's attributes, ensuring the immutability and uniqueness of each share's digital representation.

The tokenization of shares, as demonstrated in the `Share` class, contributes to the BFS's broader goal of leveraging blockchain technology for financial transactions. Key contributions include:

- *Enhanced Security and Transparency.* The cryptographic hash of share particulars ensures that each tokenized share is unique and immutable, enhancing the security and transparency of share ownership and transactions.
- *Streamlined Share Transactions.* By representing shares as tokenized assets on the blockchain, the BFS enables DvP where shares are exchanged with digital funds, transactions between entities, reducing the complexity and time required for traditional share trading processes
- *Regulatory Compliance.* The encoding of share rights and particulars within the tokenized asset helps ensure compliance with legal and regulatory requirements for share issuance and trading.

The next element of Fig. 6.28 is the `ShareCertificate` class, instantiated during the sale of shares from the issuer to a shareholder. Unlike `Share`s that remain with the issuer and are recorded in the "outstanding shares" repository of the `ShareStock` class, the sale of shares from the issuer to a shareholder results in the issuance of a "Share Certificate". This certificate also

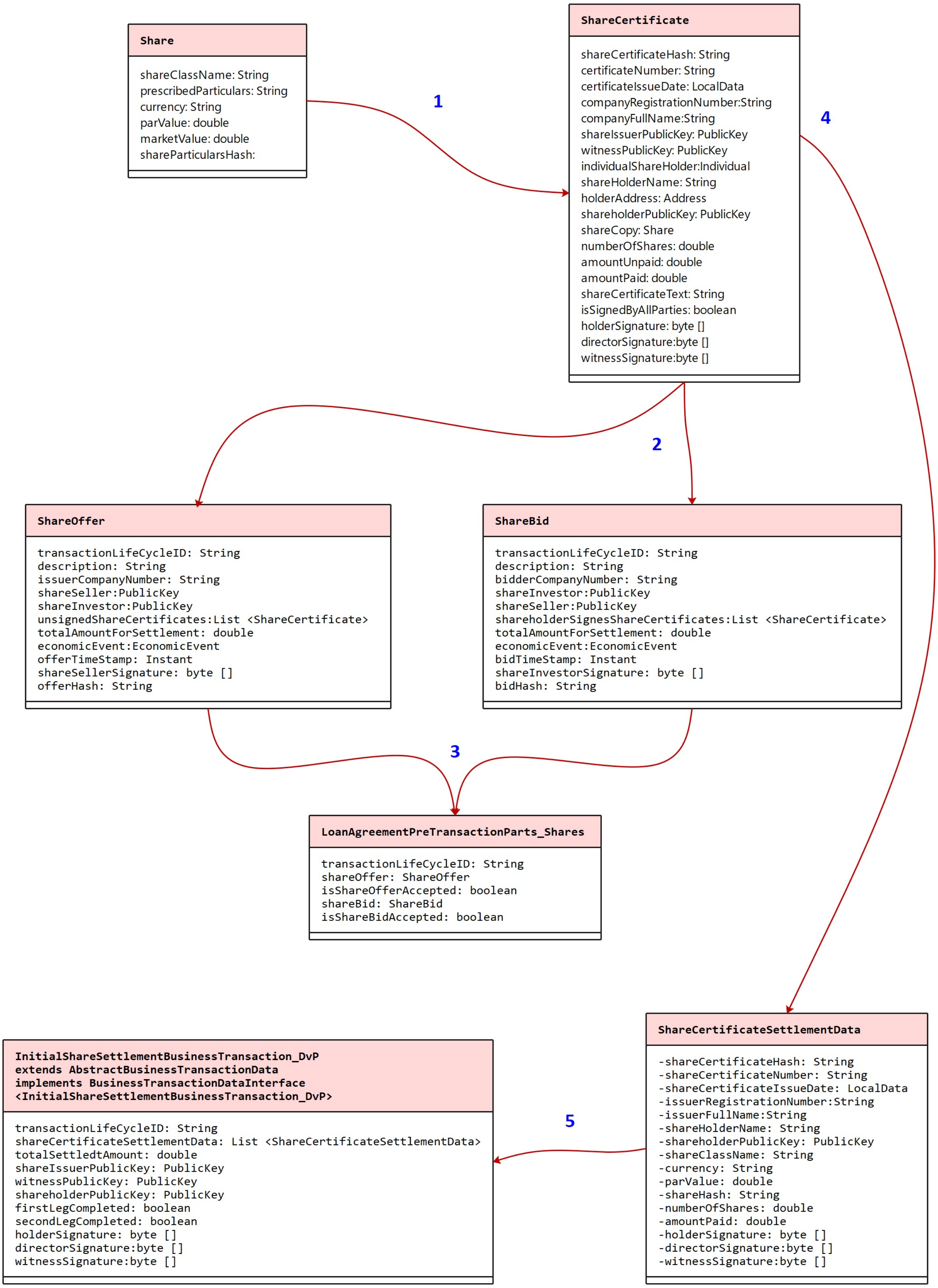


Figure 6.28: Components of Share Transaction Execution.

serves as a tokenized asset, encapsulating the rights and entitlements of its shareholder. It is a another important component that differentiates the possession of shares from the entitlements provided by the purchase of these shares. This Share Certificate is what is fiscally exchanged between share issuer and the shareholder during a share sale transaction. The `ShareCertificate` class within the BFS is designed to incorporate all essential information regarding share ownership in a tokenized form. In compliance with traditional practices, this includes details such as the certificate number, company name, shareholder name and the quantity, nominal value and type of share issued. Additionally, the certificate outlines the payment status of the shares (partly paid, fully paid, or unpaid) and features authorizing signatures and `PublicKeys` from three transaction participants - the parties of this transaction: the share issuer, a witness and the shareholder. These signatures, together with the public keys of these signers ensures the legitimacy and mutual acknowledgment of the share transaction.

The data structure for `ShareCertificate` class is illustrated in Fig. 6.28 and the elements of this class are self-explanatory. As the generalised concept, this class incorporates all relevant details related to the shares and the shareholder, ensuring that each certificate is uniquely identified and securely tied to its rightful owner through cryptographic hashing. The tokenization of "Share Certificates" within the BFS ecosystem demonstrates the innovative application of blockchain technology to traditional financial instruments. In similarity to Shares, this approach enhances:

- *Security and Verifiability.* The cryptographic hash of the share certificate ensures its authenticity and prevents tampering, providing a secure and verifiable record of share ownership.
- *Efficiency in Share Transactions.* Tokenizing share certificates on the blockchain streamlines the process of transferring share ownership, reducing the need for physical documentation and simplifying the settlement process.
- *Regulatory Compliance and Transparency.* The detailed recording of share certificate data, including the authorizing signatures, aligns with regulatory requirements for share transactions, ensuring transparency and accountability in corporate governance.

Transactional Parts

The next part of the process is description of the design and implementation of the essential components, designed in the BFS for the execution process of the DvP share transaction. In complex business transactions, such as use-case specific sale of initial share issuance, some additional digital form of pre-settlement agreements are required. In traditional share trading, bids and offers for a share trade execution must be matched in order for a trade to take place. Among other elements, these bids and offers must match on the price that one party is willing to sell for with the price that counterparty is willing to pay, including the number of exchanged shares. This share trading process divides a share sale transaction into 2 high-level stages:

1. *transaction proposal*, where bids and offers are submitted for matching.
2. *transaction settlement*, where the actual exchange of of matched shares and payment takes place.

In the context of the design of the components necessary for execution of complex business transactions in the BFS, the definition of the transactional parts is required.

Transaction Proposal - represents the initial stage of a business transaction, where parties engage a free-form pre-settlement transaction electronic agreements, outlining the conditions for the upcoming settlement. This phase is crucial for complex business transactions, such as the sale of initial share issuance, where precise coordination and agreement on terms are necessary before proceeding to the transaction settlement.

Transaction Settlement - is the phase where the previously agreed free-form conditions are incorporated into the settlement transaction and settlement is executed, culminating in the contractual exchange of securities and/or funds. It represents the formal completion of the business transaction, where all parties fulfil their obligations as per the agreement established during the transaction proposal phase.

In the design of the transactional components fourth BFS, the only settlement transaction implementation follows the transactional framework. The example of implementation and the data structure for such translation will be outlined later in the Section 7.2. This settlement business transaction is also the one that is added to the `BusinessTransactionRegister` repository, described in Section 6.3.4 and illustrated in Fig. 6.30. On the contrary, the transaction proposal part is a free-form interaction specific digital document designed to be standardised for a specific pre-settlement transaction agreement. In the context of traditional share trading, the transaction proposal phase involves the matching of bids and offers. This matching process ensures that the price one party is willing to sell for aligns with the price the counterparty is willing to pay, including the number of shares to be exchanged. The proposal must achieve consensus on several elements, including but not limited to, the price, number of shares and conditions under which the transaction can proceed. Next, in order to execute sale of these authorised shares, the concept of transaction proposal is adapted in the design of the process for execution business transaction between the APF and the BoE, where each of the parties, before settling transaction exchange documented offer and the bid with each other. These are the `ShareOffer` and the `ShareBid` (see second part of Fig. 6.28). The sequential flow for these components will be demonstrated in Section 7.2. The data structure for the `ShareOffer` transaction proposal is as follows:

- ❖ `String transactionLifeCycleID` - is generated and utilised as the unique identifier of the whole life-cycle of the business transaction, including all transaction proposals and the subsequent settlement transaction. It is a hash value of the data structure of this initial transaction proposal which also includes the offerHash;
- ❖ `String description` - a human readable description for this transaction proposal, e.g., “Initial Share Sale”;
- ❖ `String issuerCompanyNumber` - the unique company number, assigned to the share issuer entity during incorporation;
- ❖ `PublicKey shareSeller` - the public key of the entity’s wallet, who sells the shares - the `DirectorWallet`;
- ❖ `PublicKey shareInvestor` - the public key of the entity’s wallet, who purchases the shares - the `ShareholderWallet`;
- ❖ `List <ShareCertificate> unsignedShareCertificates` - the set of the draft `ShareCertificates`, reflecting the offer that is not fully signed, i.e., it is only signed by the offering entity - the APF directors in this case;
- ❖ `double totalAmountForSettlement` - total amount of digital funds asked for these shares;
- ❖ `EconomicEvent economicEvent` - will be specific to the entity’s accounting rules. This will be used by the `Accountant` of `APF::BFS` to map completed business transaction to the relevant smart contract for accounting transaction;
- ❖ `Instant offerTimeStamp` - time stamp of the share issuer signature;
- ❖ `byte [] shareSellerSignature` - the cryptographic signature of the share issuer;
- ❖ `String offereHash` - the cryptographic one-way hash function (e.g. SHA256 hash) of this offer.

In the share business transaction, described later in this report, such `ShareOffer` is designed as the 1st transaction proposal (the transaction initiating event) and it well be instantiated in the `DirectorWallet`. As the high-level process, the APF’s `DirectorWallet` sends completed (signed, dated and hashed) `ShareOffer` to the BoE’s `ShareholderWallet`. Upon inspection and agreement with this offer, the counterparty - the BoE’s `ShareholderWallet` will generate `ShareBid` - the 2nd transaction proposal. In this proposal, the `shareholder` outlines: The data structure for the `ShareBid` transaction proposal is as follows:

- ❖ `String transactionLifeCycleID` - is copied from the `ShareOffer`
- ❖ `String description` - a human readable description for this transaction proposal, e.g., “Initial Share Purchase”;
- ❖ `String bidderCompanyNumber` - - the unique company number, assigned to the share issuer entity during incorporation;

- ❖ `PublicKey shareInvestor` - - the public key of the entity's wallet, who purchases the shares - the `ShareholderWallet`;
- ❖ `PublicKey shareSeller` - the public key of the entity's wallet, who sells the shares - the `DirectorWallet`;
- ❖ `List <ShareCertificate> shareholderSignesShareCertificates` - `ShareCertificate`, only signed by the `ShareholderWallet`
- ❖ `double totalAmountForSettlement` - the total expected monetary value of all `ShareCertificate` from `shareholderSignedShareCertificates`;
- ❖ `EconomicEvent economicEvent-` - will be used by the `Accountant` of BoE::BFS to map completed business transaction to the relevant smart contract for accounting transaction;
- ❖ `Instant bidTimeStamp` - time stamp of the `shareholder`'s signature;
- ❖ `byte [] shareInvestorSignature` - `shareholder`'s signature of this `ShareBid digitalFundsOffer`;
- ❖ `String bidHash` - one-way hash function (e.g., SHA256 hash) of this `ShareBid digitalFundsOffer`

Next, the essential element of data structure of the wallets, participants in the share transaction, is the `PreTransactionParts_Shares` class (see third part of Fig. 6.28). This class enables tractability of relevant parts of the finalised and agreed upon transaction proposals that precedes the final settlement event. The data elements of this class are self-explanatory.

Lastly, the final stage of the execution of the share trade transactions between two BFS entities, specifically between the APF - as the share issuer and the BoE - as the shareholder, is a key process within the BFS. The settlement phase of the business transaction that, upon success, represents the formal completion of the business transaction, where all parties fulfil their obligations as per the agreement established during the transaction proposal phase. Application of this settlement transaction atomically organises the Delivery versus Payment (DvP) exchange of share certificates tokens with digital funds. This transaction model ensures the simultaneous exchange of shares and payment, enhancing the security and efficiency of financial transactions on the blockchain.

In the bottom Section of Fig. 6.28, the `InitialShareSettlementBusinessTransaction_DvP` class and the `ShareCertificateSettlementData` are illustrated. To provide future tractability of what the `ShareCertificate` was delivered during this business transaction, but to preserve privacy of the business sensitive information within it, the design decision was necessary to abstract private information recorded in the `ShareCertificate`. This is achieved by implementation of the `ShareCertificateSettlementData` (see step 4 of Fig. 6.28). This class is what is stored within the data of the settlement transaction, which, on successful completion, will be recorded in the `BusinessTransactionRegisters` of the transaction counterparties. Furthermore, the economic data produced by this business transaction will be transformed into accounting data and processed by the accounting cycle of the BFS entity.

Next in Fig. 6.28 - step 5, the `ShareCertificateSettlementData` becomes part of the `InitialShareSettlementBusinessTransaction_DvP`. The detailed sequential steps for the execution demonstration of this transaction will be described in Section 7.2 of this report. Then, the `InitialShareSettlementBusinessTransaction_DvP` class will be described representing the implementation the business transaction smart contract framework described in Section 6.3.3. Design of the `InitialShareSettlementBusinessTransaction_DvP` class extends `AbstractBusinessTransactionData` and implements the `BusinessTransactionDataInterface` (see Section 6.3.3). This structure illustrates implementation of the smart contract framework for capturing the complexities of share settlement transactions within the BFS. During initialisation, the transaction captures the essential details from the share offer and bid, setting up the framework for the settlement process. This `InitialShareSettlementBusinessTransaction_DvP` transaction execution is atomic and consists of 2 settlement legs, where, during the first leg, the transfer of share certificates from the APF to the BoE happens; during the 2nd settlement leg, the payment is drawn for these shares.

The Settlement Process:

1. *Transfer of Share Certificates.* The first leg of the settlement involves the transfer of signed share certificates from the issuer to the shareholder. This process is secured by verifying that all share certificates are duly signed by the issuer, witness and shareholder.
2. *Payment Settlement.* The second leg entails the settlement of payment corresponding to the transferred shares. The successful completion of this phase is contingent upon the secure transfer of funds from the shareholder to the issuer
3. *Transaction Finalisation.* Upon successful completion of both legs, the transaction state is updated to COMPLETED, and the transaction is signed off by all parties, ensuring its immutability and recording it within the BFS and the `BusinessTransactionRegister` illustrated in Fig. 6.30 of this report,

Key data components of `InitialShareSettlementBusinessTransaction_DvP` class include:

- ❖ `String transactionLifeCycleID` - the unique identifier for tracking the transaction through its lifecycle stages, generated during first transaction proposal event;
- ❖ `List <ShareCertificateSettlementData> shareCertificateSettlementData` - a list of data elements representing the settlement details of each share certificate involved in the transaction;
- ❖ `double totalSettledtAmount` - the cumulative amount settled in the transaction, representing the financial value of the shares exchanged;
- ❖ `PublicKey`s for `shareIssuerPublicKey`, `witnessPublicKey` and for `shareHolderPublicKey` - distinct cryptographic public keys for the share issuer, witness and shareholder, ensuring secure identification and participation in the transaction;
- ❖ `boolean` flags to indicate `firstLegCompleted` and `secondLegCompleted`;
- ❖ `byte []`s for the `holderSignature`, `directorSignature`, and the `witnessSignature` - cryptographic digital signatures from the holder, director and witness, validating the transaction's authenticity and mutual agreement among all parties.

`InitialShareSettlementBusinessTransaction_DvP` class is illustrated and demonstrated as a smart contract designed to enable atomic DvP share trade, By doing so, the BFS addresses existing challenges in financial transactions and sets a new framework for extended potential of conditional payments, investigated for the future of decentralised financial services. These classes contribute to the field of blockchain research by:

- ❖ *Showcasing Atomic Swaps in Action.* It provides a practical example of atomic swaps in a non-cryptocurrency context, expanding an understanding of how this technology can be applied to traditional asset classes.
- ❖ *Advancing Tokenization:.* The class advances the study of tokenization by demonstrating how tokenized share certificates can be effectively used in DvP transactions, adding to the body of knowledge on blockchain-based asset management.
- ❖ *Innovating Financial Settlements.* The class represents innovation in financial settlements, highlighting how blockchain can streamline and secure complex transactions through smart contracts.

By leveraging blockchain methodologies for tokenization of share and share certificates, and the automated execution of settlement processes, this class enhances the transparency, security and efficiency of execution of complex transactions in current digitised economy. Furthermore, the digital signatures and cryptographic verification mechanisms employed underscores the commitment to regulatory compliance and trustworthiness in financial transactions.

Business Transactions

An economic ecosystem/network consists of heterogeneous economic entities (i.e., businesses or business entities). These entities are interconnected via fundamental units of interaction - business transactions. These interactions are diverse economic events of the BFS ecosystem that have or will have monetary implications, i.e., they will result in exchange of digital funds or digital assets at some point. A transaction represents the movement of value (digital funds or digital assets) from

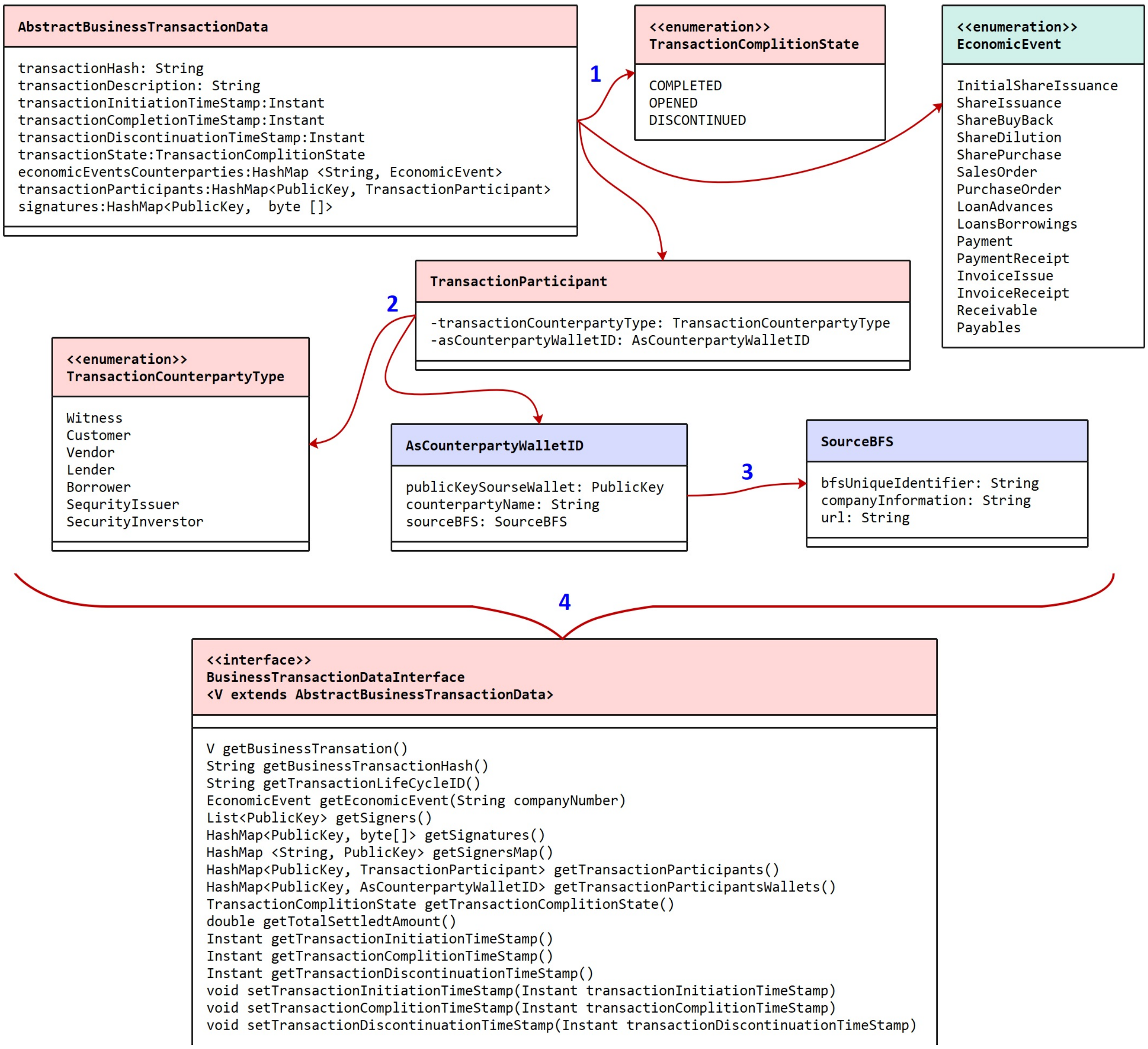


Figure 6.29: Business Transaction building blocks.

some BFS wallet addresses to some other BFS wallet addresses. To provide a standardised way of executing diverse and ever evolving set of economic interactions, the BFS provides a solution in the way of a framework for business transaction smart contracts. The components of this framework generate and execute business transaction smart contracts within the BFS ecosystem as illustrated in Fig. 6.29. It consists of the following elements:

1. `AbstractBusinessTransactionData` - the abstract class that contains a default set of elements, necessary for any business transaction generation, agreement, settlement, recording and future querying. This class and its building blocks are extended by any settlement-stage business transaction. The data components of this class are as follows:
   - ❖ `String transactionHash` - a string of the unique cryptographic hash that provides data integrity verification mechanism that include the data elements from this `AbstractBusinessTransactionData` together with the child class that extends it;
   - ❖ `Instants` for `transactionInitiationTimeStamp`, `transactionCompletionTimeStamp`, and `transactionDiscontinuationTimeStamp` - these are the time stamps for the initiation, completion or cancellation of the settlement business transaction flow;
   - ❖ `TransactionComplitionState transactionState` - an enumerator class whose elements inform accounting smart contracts about the state of this business transaction.

When the TransactionComplitionState is marked as COMPLETED in the BusinessTrtansactionRegister, the economic data produced by this business transaction is extracted, transformed into accounting data and propagated through the stages of accounting cycle;

- ❖ HashMap <String, EconomicEvent> economicEventsCounterparties - these are the distinct EconomicEvents that will be used by the accounting smart contracts to govern the accounting rules for data transformation. These EconomicEvents are unique to every entities accounting practices and rules and are pre-established by an Accountant of the company. Lastly, the String key for each entry of this map is the unique “company number” assigned to each newly incorporated entity by the Registrar of Companies House UK.
- ❖ HashMap<PublicKey, TransactionParticipant> transactionParticipants - this is the map that stores the extended data on the counterparties of the business translation.
- ❖ HashMap<PublicKey, byte []> signatures - these are the signatures of the transacting counterparties of this business transaction.

2. TransactionParticipant - used in Business Transaction to represent counterparties of the transaction. Instead of only a providing public key of a counterparty, this class houses extended identification data on those counterparts, but still preserving sensitive or confident information.
   - ❖ TransactionCounterpartyType transactionCounterpartyType - represent the side of the financial obligation within a business transaction:
     - Witness - type use to e.g., witness the signing of the agreement of a witness signer in a share sale;
     - Customer - payer = pays money
     - Vendor- payee = receives money
     - Lender - payer = pays money
     - Borrower - payee = receives money
     - SequrityIssuer - payee = receives money
     - SecurityInverstor - payer = pays money
   - ❖ AsCounterpartyWalletID asCounterpartyWalletID - an abstracted wallet data stored within business translation to enable future tractability of the economic entity and the wallet that was part of this transaction. This abstraction of the wallet data is critical to provide future tractability of the transactional chains of events. At the same time, this abstraction facilitates privacy of transaction participants by removing any sensitive data from the business transactional records.
     - PublicKey publicKeySourseWallet - a public key of the owner of the originating wallet;
     - String counterpartyName - the distinct human readable name of the owner of the originating wallet;
     - SourceBFS sourseBFS - the identification of the counterparty BFS that this wallet is a member of.
3. SourceBFS - the host BFS for which this counterparty is a company member, such as a director, employee etc.
   - ❖ String bfsUniqueIdentifier - a unique string representation designated to each BFS within a BFS ecosystem, e.g., the company number;
   - ❖ CompanyInformation companyInformation - relevant and unique publicly available information to identify each economic entity within a ecosystem;
   - ❖ String url - e.g., “www.transactioncounterparty.co.uk”
4. BusinessTransactionDataInterface implementation is designed to <V extends AbstractBusinessTransactionData>. This is to enable standardised storage and extraction of diverse impersonations of business transactions (as current and the future designed) from the pre-defined data repositories of the BFS, such as

BusinessTransactionRegister or Wallet local transaction stores outlined in the next section. The methods of this interface enable necessary communication of data relevant to the accounting of each entity.

6.3.4 Business Transactions Register Context

This section aims to encapsulate the essential role and functions of the Business Transactions Register within the BFS, highlighting its significance in the broader context of blockchain-based financial reporting. Drawing parallels from the "Corda Transaction Vault" architecture and its functionality (Brown (2018)), this class serves as a chronological archive of economic activities that stores and manages transaction records. In Corda "the vault is a database containing ledger data that is considered relevant to the wallet owner, stored in a chronological model and the form that can be easily queried and worked with" (Brown (2018)).

Another justification for the innovative design of the Business Transaction Register within the BFS is that it provides a solution for the event broadcasting communication methods of the traditional Bitcoin style blockchain. Design of the Business Transactions Register represents a response to privacy concerns posed by such blockchain communication methods. The BFS counters this by integrating Corda's Flow Framework (R3 (2023)) and Transaction Vault architecture (Brown (2018)), offering point-to-point messaging and an off-ledger transaction repository. This design ensures that business-sensitive information is only accessible to relevant parties, maintaining confidentiality while also allowing for transaction traceability. Such an approach preserves privacy and supports compliance and transparency, addressing critical challenges within the BFS framework.

In the BFS, the BusinessTransactionRegister class is conceptualised as an implementation of the application and entity local repository to store all business transactions executed between BFS ecosystem participants, i.e., the BusinessTransactionRegister is unique to each economic entity within the BFS ecosystem.

Fig. 6.30 represents the data structure for the BusinessTransactionRegister class. Drawing parallels from the Corda Transaction Vault (Brown (2018)) this class acts as a repository, capturing the essence of each business transaction, including its associated economic events, parties involved and the unique hashes that will link these business transactions to the corresponding elements within reports of filing history blockchain and with entries of accounting journals. The components of this class are as follows:

- the LocalDate for the begenningPeriodDate and the endingPeriodDate track the begging and end of the reporting period;
- the HashMap<String, BusinessTransactionDataInterface<?>> businessTransactionsRegister maintains a record of all business transactions between said economic entity and its transactional counterparties, within an accounting period.
  - the BusinessTransactionDataInterface, manages storage, retrial of business transactions, together with standardised format for inquiring about relevant parts of the data about such business transactions. Each business transaction reposited within BusinessTransactionRegister is associated a unique cryptographic hash of such transaction, together with a set of public keys belonging to the parties of such transaction. Using these elements of the blockchain cryptographic mechanisms allows for a clear audit trail of the economic data, extracted from completed business transactions and allows for communication with connected parties, enabling data provenance tracking between them.
- the ArrayList <NewBusinessTransactionEvent> connectedParties element of the BusinessTransactionRegister is another key repository that facilitates communication mechanism with every new Wallet added or registered with this economic entity. It is an additional registry for all member and counterparty wallets that perform transactional activity relevant to this economic entity.
  - the NewBusinessTransactionEvent interface is implemented by all Wallets (see Sec-

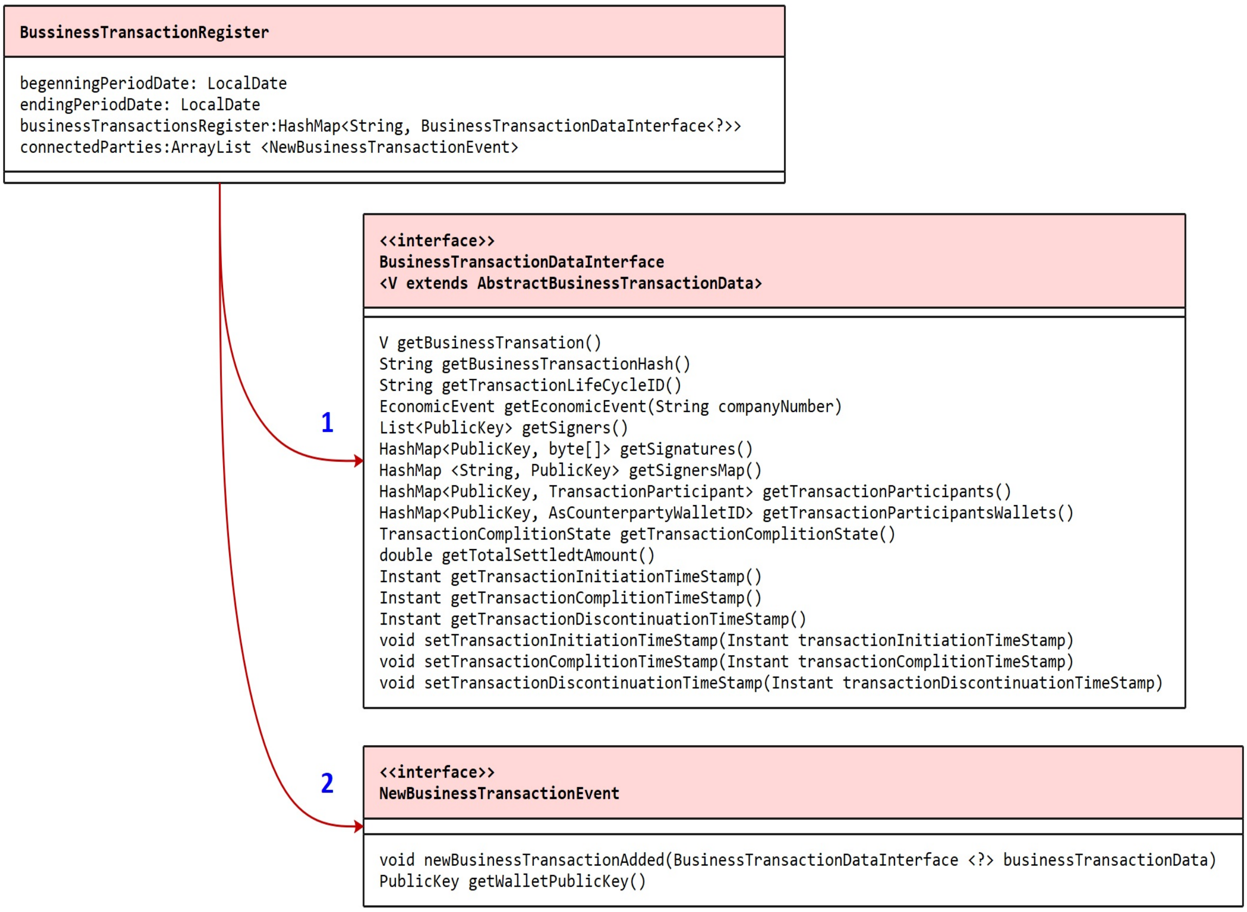


Figure 6.30: Data structure for the Business Transaction Register class.

tion 6.2.4), utilises `PublicKey getWalletPublicKey()` method to notify connected parties within the BFS ecosystem about successful completion of relevant transaction and addition of it to the `businessTransactionsRegister`.

Overall, the design and implementation of the `BusinessTransactionRegister` class in the BFS ensures maintenance, integrity and traceability of transactional data and address the privacy concerns associated with Bitcoin's broadcast communication method. It does this by verifying cryptographic signatures, which secures the authenticity and prevents repudiation of recorded business transactions. This class also serves as a communication hub, notifying connected wallets about new transactions relevant to them and thereby ensuring transparency and chronology in transaction history. This level of interactivity is essential in maintaining a transparent and up-to-date transaction history which is important for real-time financial reporting and auditing within the BFS ecosystem. The approach mirrors Corda's method of managing transactions—chronologically capturing business interactions secured by cryptographic proof, ensuring data integrity and traceability. Much like in Corda's secure and decentralised transaction management system, such a combination of chronological recording of the relevant business interactions is underpinned by the cryptographic data integrity and provenance tractability. By utilizing Corda's Flow Framework for point-to-point messaging, BFS ensures that sensitive transactional information is shared only between the involved parties, enhancing privacy. Additionally, the adaptation of Corda's Transaction Vault architecture allows BFS to maintain an off-ledger repository for business transactions, preserving their confidentiality while ensuring traceability for compliance and transparency.

6.3.5 Participants Registration Context

The BFS project, through its architectural design integrates the principles of Domain-Driven Design (DDD). This approach is taken to address complex business processes and diverse financial transactions between business entities within economy and to meet diverse needs of financial accounting within a blockchain framework. The BFS architecture, underpinned by the DDD approach, emphasises the importance of modelling complex domain logic centered around the business's core functionalities. One part of this integration is the implementation of business-specific components that capture the varied interactions and relationships between individual entities within the BFS ecosystem.

This section focuses on the design and Java implementation of the TransactingPartyWalletRegister class and the WalletRegistration<V extends Wallet> interface, designed to manage and facilitate relationships between the company and its internal members, related parties and external counterparties through a well-defined wallet registration and management system. It focuses on members and counterparties relation to the APF::BFS, to enable recording and recognition of entities interacting within the BFS domain.

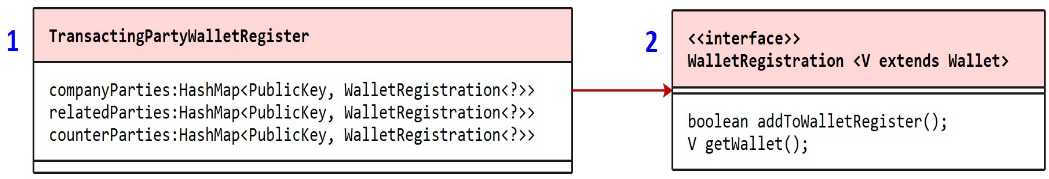


Figure 6.31: Wallet Registration.

In Fig. 6.31, the second item on the right is the inteface WalletRegistration <V extends Wallet>. This interface is defined with a generic type <V extends Wallet>, allowing for flexibility in handling various types of Wallets that engage in business and transactional activity with this company. The motivation for this interface is driven by the need to standardise the wallet registration process - as this interface must be implemented by all wallets relevant to this entity, such as internal members, related parties and as counterparties in any business transactions. At instantiation of each of such wallet, the wallet's data is recorded in the TransactingPartyWalletRegister. This is done to ensure that each wallet role, relevant to the business and transactional lifecycle of this entity is queryable.

- boolean addToWalletRegister() - ensures that every participating wallet is registered within the BFS system, making it identifiable and accessible for transaction validation and processing.
- V getWallet() - provides a mechanism to retrieve the specific wallet instance, facilitating queries and interactions with the wallet's data and functionalities.

Next, continuing onto the TransactingPartyWalletRegister, illustrated on the left of Fig. 6.31. This class acts as the registry for every wallets associated with different categories of BFS participants. This registry contains wallets belonging to all: internal members, related parties and external counterparties, ensuring a coherent and secure framework for transaction processing and record-keeping within every separate BFS. Every Wallet, as the superclass to all current and future distinct wallets, differentiated by their role in relation to the organisation, at its instantiation is assigned a data element for WalletControlType (see Section describing Wallet class - 2). An important part of that data element is the enumerator RelationToCompany, which is used here for the logical categorisation of the inner repositories within TransactingPartyWalletRegister. These are:

- HashMap<PublicKey, WalletRegistration<?>> companyParties - manages wallets created for members and employees of the company, facilitating the registration of every wallet associated with company members, thereby enabling the tracking and management of internal transactions.

- `HashMap<PublicKey, WalletRegistration<?>> relatedParties` - handles wallets designated as counterparties that transact as related parties with the company (e.g., parent or subsidiary entities), ensuring a comprehensive register of wallets involved in inter-company transactions.
- `HashMap<PublicKey, WalletRegistration<?>> counterParties` - registers wallets for any type of counterparty engaging with the company, supporting a broad spectrum of transactional relationships beyond the internal ecosystem.

The Java implementation of these components is tailored to ensure an efficient management and seamless integration and registration process for Wallets within the BFS's infrastructure. Key implementation strategies include:

- *Dynamic Registration.* This process allows for on-the-fly dynamic addition (registration) and retrieval (enables communication with) of wallets through the `addToWalletRegister` and `getWallet` methods, respectively. This flexibility supports the evolving communication needs, together with nature of business transactions and participant roles within the BFS ecosystem.
- *Security and Identity Verification.* By keying the registries with `PublicKey`, the system embeds cryptographic security measures, ensuring that communication between wallets and in transactions are secure and participants are verifiably authentic.
- *Modularity and Extensibility.* The use of generics in the `WalletRegistration` interface allows for a high degree of modularity and extensibility. This design choice enables the system to accommodate various types of wallets with specialised functionalities and diverse roles, without compromising the coherence of the registration mechanism.

Within DDD principles these components, such as `WalletRegistration` interface and `TransactingPartyWalletRegister` class, encapsulate complexities of financial relationships within a single organisation and provide a mechanism for managing wallets across different contexts — from internal members to external counterparties — reflecting the domain's specifics and ensuring seamless transactional processes. Through this integration of business-specific components, the BFS project demonstrates a application of domain-driven design principles, paving the way for advanced blockchain-based financial systems. These components simplify the process of integrating new participants within each business entity's BFS, and within the overall BFS ecosystem and enhance the system's ability to adapt to changing and complex business needs, ensuring that the overall system remains adaptable, secure and aligned with the specific needs of its users. By providing a structured and secure framework for wallet registration and management, these components significantly contribute to the BFS project's overarching goal of leveraging blockchain technology for innovative financial accounting and reporting solutions.

6.3.6 Participants Role Context

In the BFS, wallets play an important part in encapsulating identity, authority and transactional capabilities of participants within the BFS ecosystem of economic entities. Each wallet, by its design and functional requirements (see Section 3.2.7), serves a specific role mapped with its owner's responsibilities towards the business and the business' interactions within the BFS ecosystem. To satisfy functional requirements of the stakeholders of the BFS, identified in Section 3.2.7, the wallets are designed and implemented to encapsulate digital identity and business authority of its owners. It ensures secure and verifiable identification of entities participating in the BFs transactional ecosystem, enhancing trust and integrity. Furthermore, it serves as the repository for these cryptographic identities and as a comprehensive transactional enabler, transactional history orchestrator, a controller of access permissions and a secure vault for cryptographic secrets. The business-role based implementation of these wallets governs access rights to the internal economic data of the BFS ledger that belongs to an entity.

In this Section, six wallet roles have been designed and communicated. These are defined based on the specific use-case requirements, and are:

1. Director Wallet;
2. Shareholder Wallet;
3. Borrower Wallet;
4. Lender Wallet;
5. Facility Wallet;
6. Counterparty Wallet.

To be operational within the BFS, each of these wallets must extend its superclass `Wallet` described in Section 6.2.4 and illustrated Fig. 6.23. Each specialised wallet demonstrates the BFS's ability to cater for the diverse roles within a business entity and the overall heterogeneous ecosystem, ensuring that each participant has the tools necessary to fulfil their responsibilities effectively.

Director Wallet

One such specialised wallet is the `DirectorWallet` the data structure for which is illustrated in Fig. 6.32. It is designed to embody the role and responsibilities of a company director within the BFS, specifically focusing of the use-case implementation of the APF::BFS. This section describes the design and implementation of such `DirectorWallet`, illustrating its significance in facilitating director-specific transactions and governance activities. The primary use-case specific responsibility of this wallet is to act as the initiating party in the share sale transaction described further in Section 7.2 - responsible for selling shares to the BoE and receiving £100 of digital funds on behalf of the APF. The wallet serves as a digital extension of a director's authority, encapsulating their capability to influence company operations and financial decisions, all within the secure and transparent BFS environment.

The `DirectorWallet` extends the foundational `Wallet` class (see Section 6.2.4 and Fig. 6.23), incorporating additional functionalities and attributes further tailored to the needs of company directors of the APF. These functionalities enable directors of the APF to execute their governance duties. The use-case specific and generalised design considerations for the `DirectorWallet` within the APF::BFS are focused on enabling the APF directors to:

- ❖ *Participate in Share Transactions.* As APF is a private company, limited by shares (Companies House (2023*a*)), an additional responsibility of the directors of PF includes orchestration of the initial share issuance (Companies House (2023*a*)). In addition, at incorporation of the APF, the initial authorised share capital is sold to its parent company (the BoE) another responsibility of APF directors is conduct of a business transaction on the sale/transfer of these shares into the possession of its shareholder - the BoE.
- ❖ *Interact Securely within the BFS.* Utilization of cryptographic keys and signatures for secure transactions, documentation and identity verification, safeguards the integrity of directorial actions within the BFS entity in general. This enables directors to ensure that all transactions and interactions are cryptographically secure, verifiable and aligned with diverse company's policies and regulatory requirements.
- ❖ *Execute Governance Activities.* Facilitation of the execution of business decisions made at the board level, including e.g.., future approvals for major expenditures, strategic initiatives and compliance matters, by utilising individual cryptographic signatures for relevant digital documentation and interactions.

In Fig. 6.32, this use-case specific functionality is reflected in the `DirectorWallet` attributes, designed for this purpose. These are:

- ❖ `AsCounterpartyWalletID asConterparty` - as was illustrated in Fig. 6.29 of Section 6.3.3, this is an abstracted data of this wallet that will be utilised and stored as internal data of a business translation. This abstraction enables future tractability of economic entity participating in any business transaction, without exposing sensitive private data, but ensuring

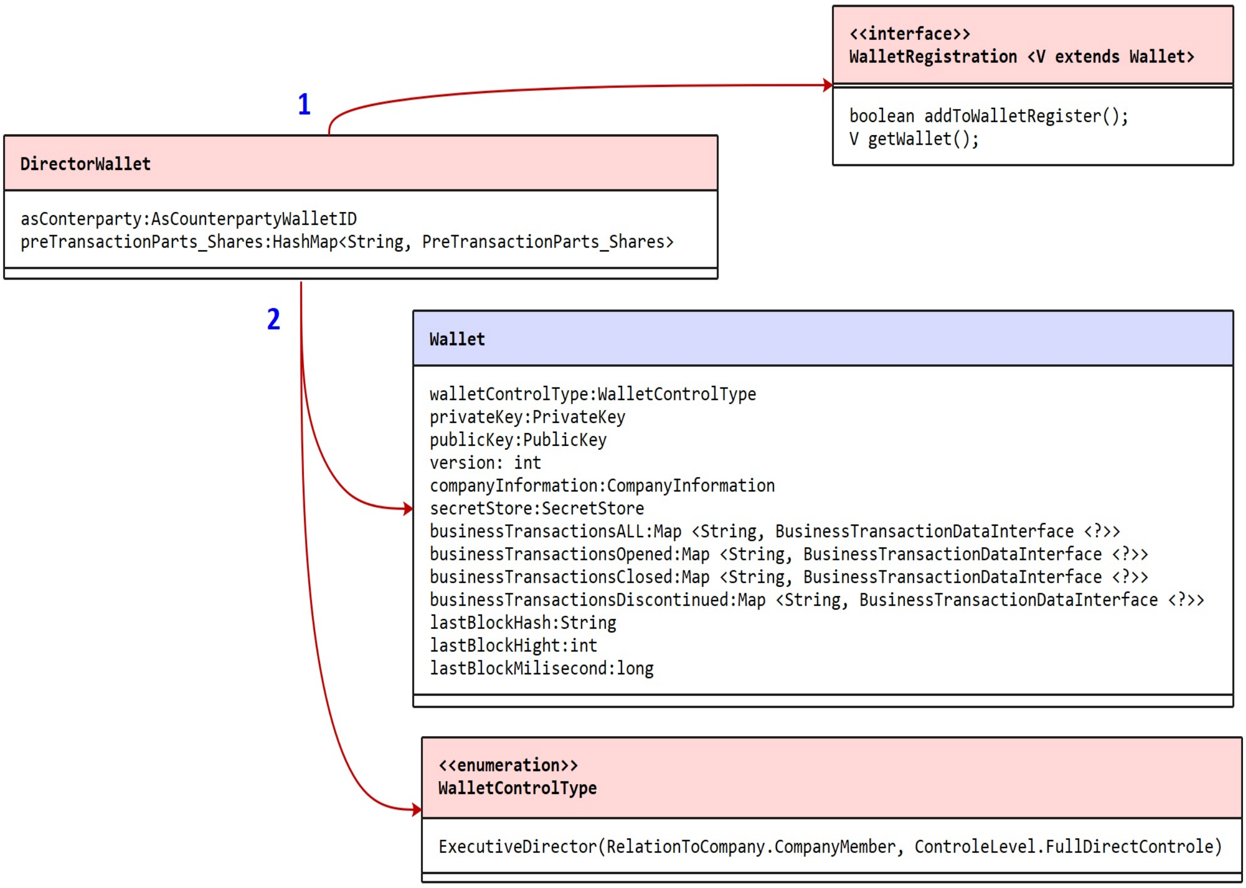


Figure 6.32: Director Wallet.

extended identification capability in complex business interactions.

- `HashMap<String, PreTransactionParts_Shares> preTransactionParts_Shares` - transaction proposal parts for each share trade transaction (see Section 6.3.3).

The `Wallet` superclass of the `DirectorWallet` is initialised within its constructor with the individual-specific and role specific information and a set of public and private keys, uniquely defining its cryptographic identity Such information includes a unique identifier for the individual (e.g., name and surname), company information, ensuring that the wallet's transactions are directly attributable to the specific director and their governance role. In the use-case description section (see Sections 2.4 and 2.4.1), at incorporation and instantiation of the APF, two prospective directors of the APF filled-out electronic versions of the form `IN10` and submitted this form to the `Registrar` (see Fig. 5.3 for company incorporation flow). Upon receiving the `PrivateRegister` instance from said `Registrar`, containing incorporation documentation, the directors are able to instantiate their `APF::BFS` as a "private company limited by shares" Companies House (2023*a*) and commence their economic activity. As the next step of the process, instantiation of `DirectorWallets` is performed. The metadata for these wallets is derived from the form `IN10` copy, contained within the `PrivateRegister` instance. In the BFS prototype, two `DirectorWallets` are generated with functionality in `RelationToCompany` as a `CompanyMember` and `WalletControlType` as an `ExecutiveDirector`.

The `DirectorWallet` is designed to represent the interests and actions of company directors within the BFS framework, while ensuring that each wallet is uniquely linked to a director's identity and their role within the company. Its implementation is instrumental in integrating blockchain technology with traditional business governance and financial management practices. By providing a dedicated tool for interaction with the wider ecosystem to company directors, the BFS enhances the efficiency, security and transparency of decision-making processes for orchestration of business activities. The `DirectorWallet` class is an example of the BFS's domain-driven design approach that encapsulates the subdomain of business entity with focus on context of the business role, tailored to meet specific needs of company directors. Through its design, the `DirectorWallet` facilitates integration of blockchain capabilities with the governance and financial management activities of directors, reinforcing the BFS's potential to transform traditional business operations with modern technological solutions.

Shareholder Wallet

The use-case specific role for the implementation for BFS wallet is the counterparty role encapsulated by the shareholder of the APF - the BoE (Companies House (2023*a*), Companies House (2023*b*)). In BFS PoC implementation, this wallet acts as a counterparty to the initial share purchase transaction outlined in Section 7.2, where the BoE is an entity that receives the `ShareCertificates` form the APF and confirms ownership of 100 Ordinary Shares from the APF. The BoE as `ShareholderWallet` pays £100.00 of digital funds to the APF for the purchase of these shares. The design and instantiation of the `ShareholderWallet` class encapsulates the functionality required for the BoE to engage in transactions with the APF, whilst providing verifiable cryptographic identity and role defined access to the relevant sensitive transactional data; this enables the BoE to execute and track transactions (see functional requirements outlined in Section 3.2.7). All necessary use-case specific identity information for this wallet is derived from the form `IN10` (share subscriber part).

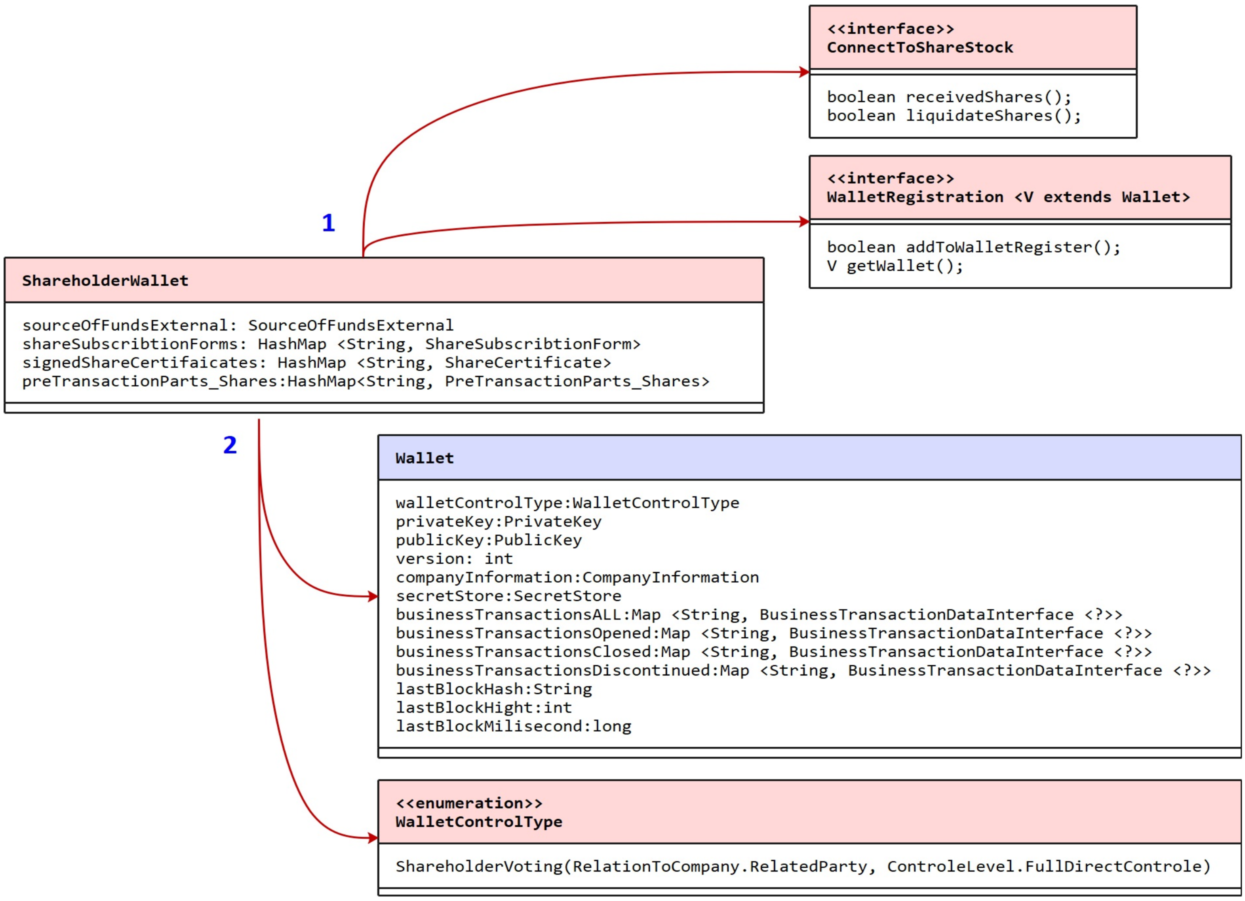


Figure 6.33: Shareholder Wallet.

The data structure for the `ShareholderWallet` class is illustrated in Fig. 6.33. A BFS implementation `ShareholderWallet` extends the generic `Wallet` class (see Section 6.2.4 and Fig. 6.23), where its constructor initialises this wallet public and private keys, uniquely defining its cryptographic identity. `ShareholderWallet` class also implements additional interfaces `ConnectToShareStock` and `WalletRegistration <ShareholderWallet>`, highlighting its specialised role and business purpose. This wallet exercises `FullDirectControle` over the data internal to the APF, because it purchases 100% of the authorised share stock issued at APF incorporation (Companies House (2023*a*,*b*)), but as a related party role in relation to the APF member `RelationToCompany.RelatedParty`. In keeping with the `DirectorWallet` and in line with the specifics of the use-case for this research, such a design enables the `ShareholderWallet` to interact seamlessly with other components of the BFS infrastructure, exercising control over the authorised share stock issued at incorporation of the APF.

Fig. 6.33 illustrates implementation of the role specific data design for this wallet:

- ❖ `SourceOfFundsExternal sourceOfFundsExternal` - facilitates connectivity with its host BFS (the BoE::BFS), and enables the movement of funds necessary to cover its transactional obligations. This provides direct communication channel with the wallet's owning BFS entity - the BoE.
- ❖ `HashMap <String, ShareCertificate> signedShareCertifaicates` - stores for signed share certificates, reflecting the completion of share transfer transactions;
- ❖ `HashMap<String, PreTransactionParts_Shares> preTransactionParts_Shares` - transaction proposal parts for each share trade transaction (see Section 6.3.3).

The `ShareholderWallet`'s design as counterparty to APF is a conceptual parallel to a customer account on an e-commerce platform. It serves as the digital representation of another entity (the BoE) within APF application, whilst functioning as a "technological footprint", facilitating transactional execution and movement of funds from the BoE::BFS - the source of such digital funds (see Fig. 5.5). However, unlike typical e-commerce accounts, this wallet has specialised functionalities that align with the BFS's objectives.

- ❖ *Identity Authentication and Authorization:* The wallet incorporates cryptography mechanisms for secure identity verification and authorization. This ensures that only authorised representatives of the BoE::BFS can access and manage this wallet.
- ❖ *Access Control and Permission Management:* Through this wallet, the entity is enabled with role defined access permissions, allowing for granular control over who can view or interact with its financial data and transactions. This feature is essential for maintaining confidentiality while ensuring that sensitive information is safeguarded against unauthorised access and manipulation.
- ❖ *Transaction Participation.* The wallet acts as a receiver of Share Certificates, confirming the ownership of shares transferred from the APF. It also sends funds to the APF, completing the financial aspect of the transaction.
- ❖ *Record-Keeping.* The wallet records transactions it participates in to ensure transparency and accuracy, a crucial feature for financial record-keeping within the BFS ecosystem.

This wallet's design and functionalities align with the overall goals of the BFS by ensuring secure, transparent and efficient participation of counterparties in the BFS ecosystem. It is a BFS tool tailored to meet the unique requirements of blockchain-based financial transactions and interactions, supporting the integrity, privacy and accountability of the economic activities it facilitates.

Borrower Wallet

The facilitation of Quantitative Easing (QE) — specifically, the acquisition of private sector assets by the APF — is orchestrated through a simulated lending agreement with the BoE. To operationalize QE process effectively, the APF requires substantial financial backing, fulfilled through these tokenized Treasury Bills. This funding is secured by initiation of a drawdown request process to the BoE, mirroring the practical needs of asset purchasing and ensuring the APF's financial commitments are met. This crucial arrangement is supported by the illustrative adaptation of the use-case that formed a scenario for simulation of issuance and subsequent lending of tokenized Treasury Bills to the APF by the BoE. This lending process is represented in the BFS prototype via the `LoanAgreement` smart contract mechanism. The technological construct articulates the lending terms between the APF (as the single borrower) and the BoE (as the lender), embodying the digital automatisation and execution of subsequent business transaction, demonstrated in Section 7.4.

Instrumental and essential to execution of this mechanism are the `BorrowerWallet` and the `LenderWallet` (described in Section 6.3.6). The wallets' design and implementation within BFS demonstrate another two business-specific roles, necessary for execution of the `LoanAgreement` smart contract. The `BorrowerWallet`(described in this section) serves firstly as a digital embodiment of a hypothetical employee of the APF, operating within the BFS ecosystem, facilitating the APF's business objective of obtaining funding, required for QE implementation. It acts as a repository for cryptographic identity for that employee and as a dynamic participant in relevant business transactions, encapsulating the operational authority of its owner. This wallet is designed to manage its owner's access permissions and secure repositories for cryptographic transactional HTLC secrets (see description is the Section 2.3.2, and implementation in the Chapter 7), ensuring reliable implementation of atomic business transactions, and the APF's operational efficacy within the BFS framework.

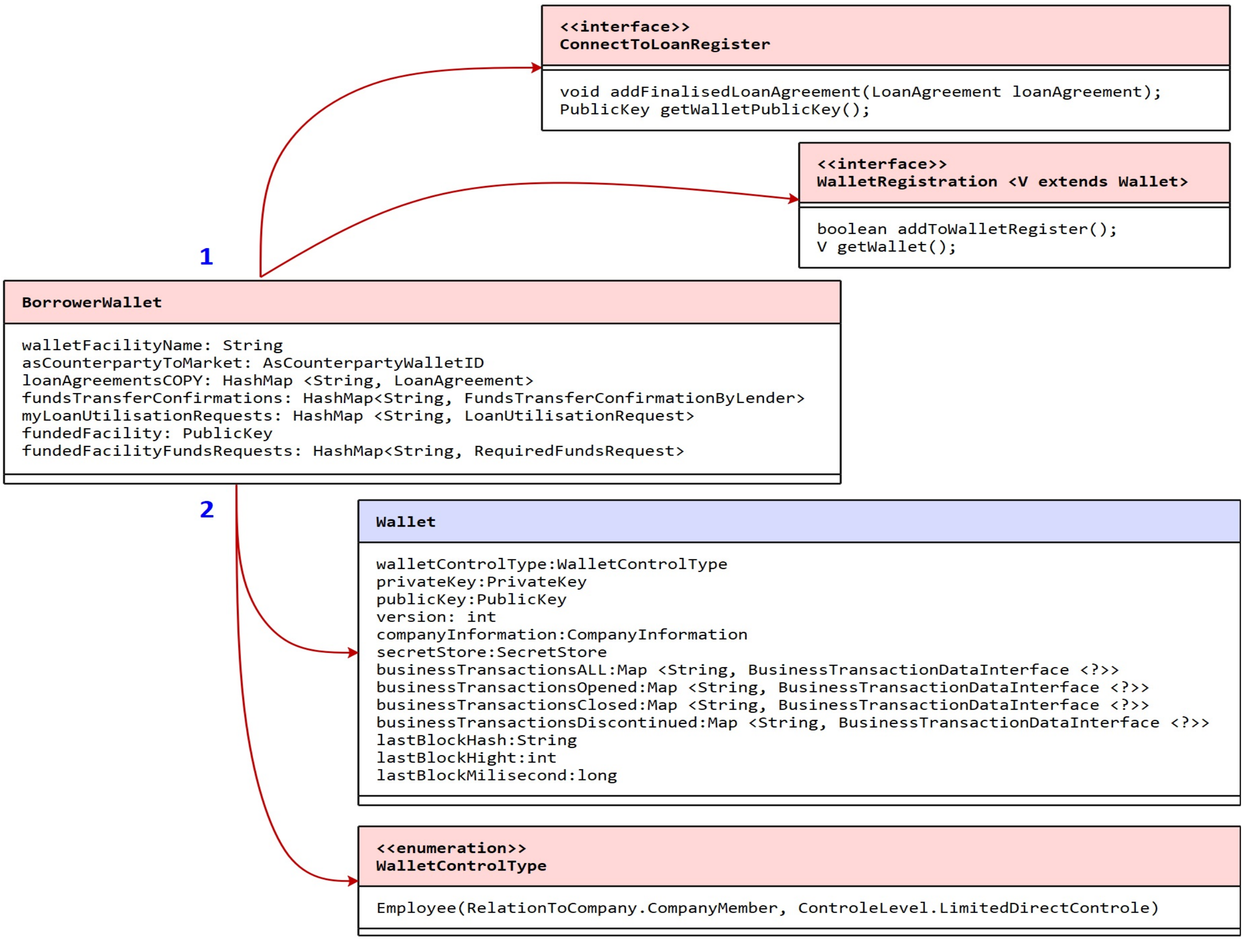


Figure 6.34: Borrower Wallet.

Fig. 6.34 illustrates primary data elements and additional functionality of the `BorrowerWallet` class. As an extension of the superclass `Wallet`, it facilitates BFS's commitment to standardise and manage distinct role-specific functionalities of diverse economic entities within the BFS ecosystem. By implementing `WalletRegistration <BorrowerWallet>` interface and `ConnectToLoanRegister` interface, the `BorrowerWallet` class is capable of registering within its BFS application (the APF::BFS - as a member wallet) and connecting to the relevant loan register that reflects existing lending arrangements of the APF. Its functionalities are designed to encompass the requirements of the APF, particularly in managing and initiating transactions related to funding acquisition. The data elements of the `BorrowerWallet` class (see Fig. 6.34) comprise of elements required to facilitate, track and execute lending agreements and transfers of funds required to operationalize QE implementation by the APF. Key components of this class are as follows:

- `String walletFacilityName` - identifies this wallet within the BFS ecosystem and the specific facility within APF, ensuring transactions are correctly attributed;
- `AsCounterpartyWalletID asCounterpartyToMarket` - represents the unique cryptographic identity as the counterparty to market transactions, specifically when interfacing with entities such as simulated CPI-1.
- `HashMap <String, LoanAgreement> loanAgreementsCOPY` - stores copies of loan agreements for reference;
- `HashMap<String, FundsTransferConfirmationByLender> fundsTransferConfirmations` - records transaction proposals, detailing confirmations of funds to be transferred from the lender - the BoE;
- `HashMap <String, LoanUtilisationRequest> myLoanUtilisationRequests` - records transaction proposals detailing all loan utilization requests initiated by this wallet (the APF) and sent to the lender (the BoE);
- `PublicKey fundedFacility` - public key of the `FacilityWallet` (see Section 6.3.6) for which the funds are borrowed from the lender;
- `HashMap<String, RequiredFundsRequest> fundedFacilityFundsRequests` - manages funding requests specific to that facility's operation (`FacilityWallet` described further in Section 6.3.6). Utilisation of this element is demonstrated in the sequential steps of the Section 7.5 - steps 10 - 13.

In summary, the `BorrowerWallet` encapsulates a set of behaviours for the APF to function as an active economic entity within the BFS ecosystem. It is designed to meet the stakeholder's functional requirements for the BFS (see Section 3.2.7), providing the tools necessary to capture digital identity and role-based business authority, designed to regulate access rights to internal company's data. The overall concept for the `BorrowerWallet` extends beyond specifics of its functionality; it encapsulates the nature of the BFS's ability to cater to diverse and complex roles and operational responsibilities of different business entities within a broad economic ecosystem. Through this approach to digital wallet design and implementation, the BFS addresses the functional requirements identified by stakeholders and demonstrates advanced transactional capabilities of the BFS, providing direction for future innovations in blockchain technology and business orchestration.

Lender Wallet

The `LenderWallet` is a specialised component within the BFS that encapsulates the functionality required by the BoE to act as a lender and represent a counterparty in the funds acquisition business transaction that will be demonstrated in Section 7.4. In aliment with the use-case of the Thesis, this wallet's role is essential to operationalize implementation of the QE process, particularly in the facilitation of private sector asset acquisitions by the Asset Purchase Facility (APF) (described later in Section 7.5). This function is realized through a conceptual lending agreement between the APF and the BoE, demonstrated through the use-case specific issuance and lending of tokenized Treasury Bills to the APF from the BoE (demonstrated in Section 7.4).

Fig. 6.35 illustrates primary data elements and additional functionality of the `LenderWallet` class. In keeping with the `BorrowerWallet`, the `LenderWallet` extends the `Wallet` superclass (see Section 6.2.4 and Fig. 6.36), illustrating the standardised approach to management of distinct role-

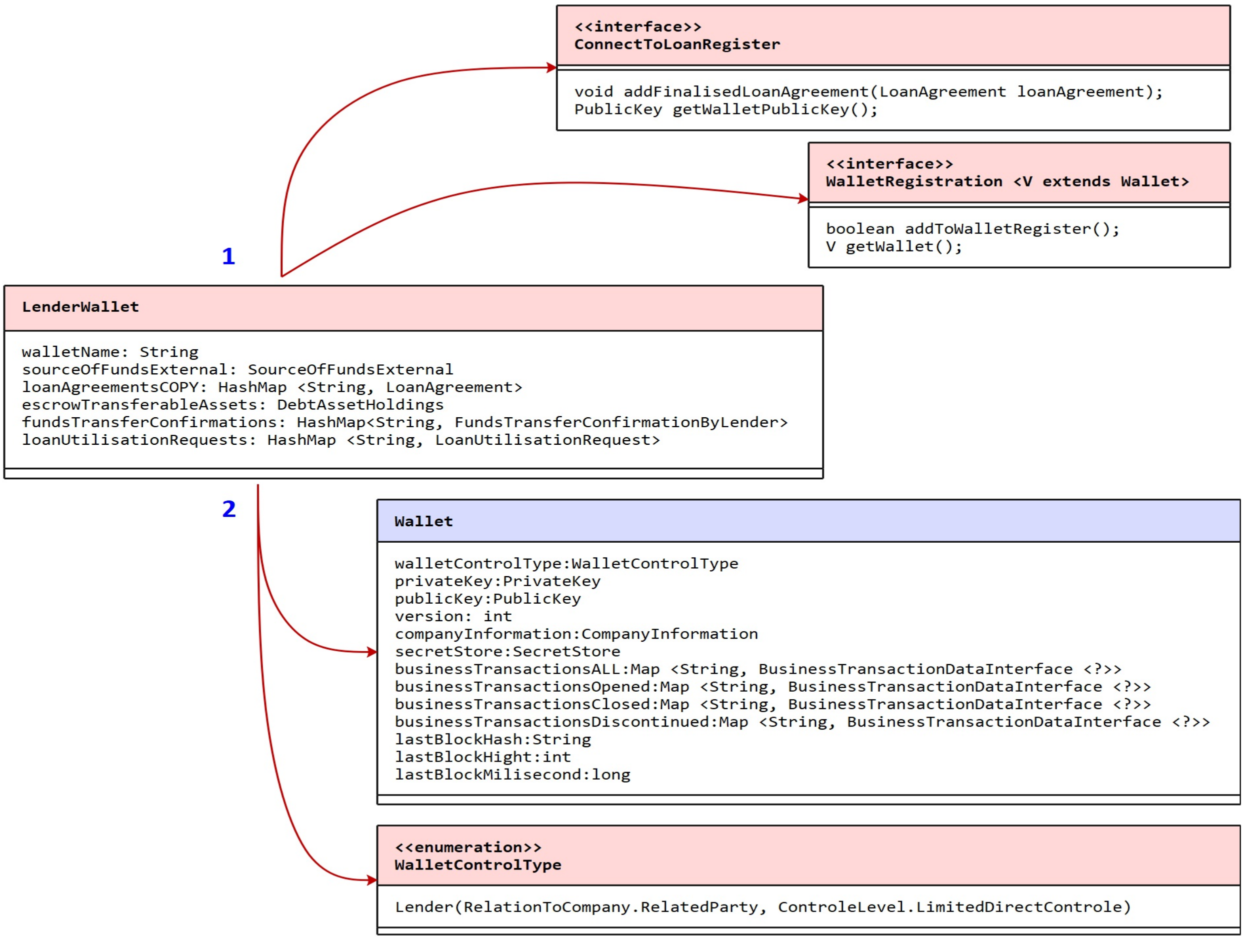


Figure 6.35: Lender Wallet.

specific functionalities for each economic entity within its ecosystem and embodying the digital identity and operational authority of the BoE within the APF::BFS application. By implementing the `WalletRegistration <LenderWallet>` interface and `ConnectToLoanRegister` interfaces, the `LenderWallet` class is capable of registering within host BFS application (the APF::BFS - as a counterparty wallet) and connecting to relevant loan registers to reflect the ongoing lending arrangements. It is designed to receive and process drawdown requests from the APF, evaluate these requests against established lending terms and facilitate secure transfer of tokenized Treasury Bills. This process underscores the wallet's capability to manage complex business transactions, ensuring operational success. The data elements of the `LenderWallet` class (see Fig. 6.35) comprise of elements required to facilitate, track and execute lending arrangements and transfers of funds required to operationalize QE implementation by the APF. Key components of this class are as follow:

- `String walletName` - identifies this counterparty wallet within the APF::BFS as the source of QE funding, e.g., "Deposit from the Bank Of England";
- `SourceOfFundsExternal sourceOfFundsExternal` - external funding source for provision of tokenized Treasury Bills; connection channel between this wallet and its legal owner - the BoE::BFS;
- `HashMap <String, LoanAgreement> loanAgreementsCOPY` - a map containing copies of `LoanAgreement` instances, ensuring the lender can validate terms against borrower requests.
- `DebtAssetHoldings escrowTransferableAssets` - the debt assets that the BoE::BFS is ready to transfer to the APF application, but with ownership retained until business trans-

action atomic settlement conditions are met. This repository is utilised as follows: the lender (BoE::BFS) *blocks the securities* to be delivered by *transferring the securities to so called* ***escrow account*** *or* ***relay point*** within the `LenderWallet` of the APF. However, although these transferred assets are inside the wallet that is part of the APF application, the ownership of this assets is still with the BoE. This is enabled to facilitate control and coordination of the *One-way-payment business transaction* (see Section 7.4) the two BFS ledgers of the APF and the BoE.

- `HashMap <String, LoanUtilisationRequest> loanUtilisationRequests` - tracks requests from borrowers to utilise loans for asset purchases (from the borrower to the lender);
- `HashMap<String, FundsTransferConfirmationByLender> fundsTransferConfirmations` - tracks confirmations of approvals for the `LoanUtilisationRequest` (from the lender to the borrower).

In the broader context of the BFS, the `LenderWallet`'s role is instrumental in managing liquidity requirements, facilitating the execution of contractual lending agreements, and embodying the BFS's overarching objectives. It acts as a digital repository for the BoE's cryptographic identity and as an active participant in the lending transactions, encapsulating the operational authority granted to it. This wallet is instrumental in managing access permissions and securely storing cryptographic HTLC secrets, ensuring integrity of the lending process within the BFS framework. Through this innovative approach, the BFS addresses the functional requirements of its stakeholders and encourages future advancements in blockchain technology and its application in complex financial ecosystems.

Facility Wallet

The Commercial Paper Facility (CPF) represents a strategic initiative implemented by the APF, established to directly channel funds to the corporate sector and facilitate their access to capital markets - the QE. Operational from 13 February 2009, the CPF facilitates credit (liquidity) for wider financial ecosystem corporations, seeking to finance their operations especially in times of financial distress. In accordance with historic chronological events, the CPF commenced its first transaction on 13 February 2009 with the primary goal of purchasing investment-grade Commercial Paper (CP) directly from issuers (Bank of England (2009*c*)) - the hypothetical Commercial Paper Issuer - 1 (CPI-1) used in this Thesis. This initiative was designed to support the liquidity needs of the corporate sector by providing an alternative funding source during periods when traditional capital market access was constrained as the consequence Global Financial Crisis (GFC) of 2007-2008. The operational mechanics of the CPF adopted for the simulation scenario of the BFS functional demonstration are outlined in Section 7.5 and involve:

1. *Direct Purchase of CP*. The APF engages in transactions to purchase tokenized CP in the primary market at the time of issue of CP. This approach ensures the provision of funds to the corporate CP issuers, such as CPI-1.
2. *Financing Through Treasury Bills*. The purchase of CP by the APF is financed by tokenized Treasury Bills (TBs) provided by the BoE (One - way Payment Part of the Business Transaction demonstrated in Section 7.5).
3. *Settlement of CP vs. TB*. The APF then swaps these TBs for the illiquid CP issued by market participants such as CPI-1. It is a Payment vs. Payment (PvP) transaction type, executed as an atomic swap transaction that implements HTLC functionality designed for secure assets swap in multi-blockchain ecosystem such as BFS.

As a critical mechanism, CPF is designed to function as a channel to provide funds directly to the corporate sector, enhancing their access to capital markets. In the context of the BFS project, this functionality is encapsulated within the `FacilityWallet` entity role designed to facilitate business transactions related to the CPF operations. The `FacilityWallet`, in particular, serves as the digital embodiment of another member of the APF::BFS entity - a hypothetical employee of the APF, operating within the BFS ecosystem, facilitating the APF's role in QE implementation.

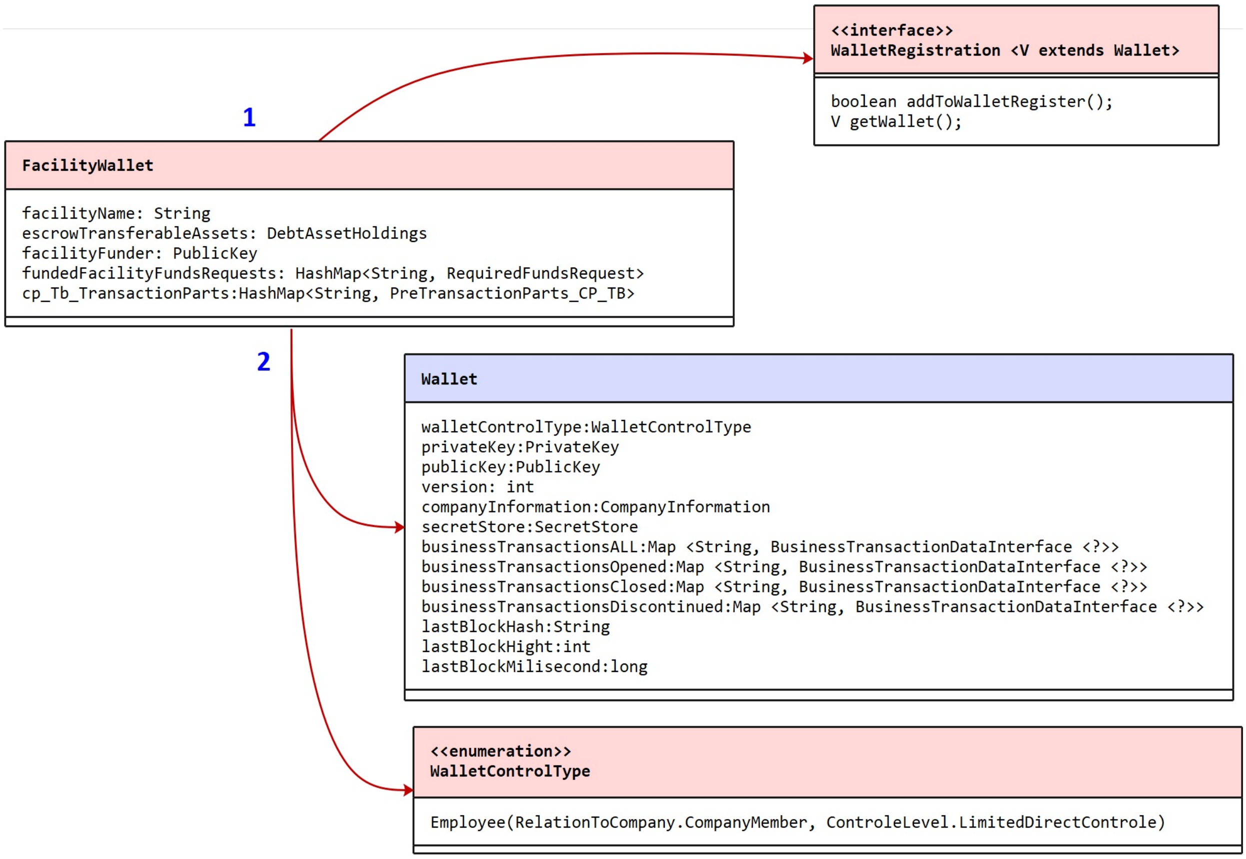


Figure 6.36: Facility Wallet.

Fig. 6.36 illustrates Java implementation of the `FacilityWallet` extends `Wallet` class (see Section 6.2.4 and Fig. 6.36), incorporating additional functionalities and attributes further tailored to the requirements of CPF. By implementing the `WalletRegistration <FacilityWallet>` interface, the `FacilityWallet` class is capable of registering within its BFS application (the APF::BFS - as a member wallet. Key components of the implementation allowing the `FacilityWallet` to inherit standard wallet functionalities while adding specific features required for CPF transactions include (see Fig. 6.36):

- ❖ `String facilityName` - uniquely identify the CPF and aligning with the APF's operational framework, e.g., "Commercial Paper Facility";
- ❖ `DebtAssetHoldings escrowTransferableAssets` - handle a temporary custody of debt assets, facilitating secure and compliant asset transfers performed prior and at the point of settlement (see 7.5.1). This wallet local repository is a transaction specific functionality that orchestrates the HTLC model for atomic swap. This is enabled to facilitate control and coordination of the *One-way-payment business transaction* to the two BFS ledgers of the APF and the BoE. The master repository for all debt assets of an entity, such as "Treasury Bills" and "Commercial Paper" is located within BFS class (see Section 6.2.2 and illustrated in Fig. 6.22 (item 7)). `escrowTransferableAssets` - is an escrow repository where debt assets agreed for settlement are transferred from the `BFS` :: `purchacedAssetsDebtAssetHoldings` - locked for a set time period, i.e., unavailable for other transactions until the set time expires or settlement fails (then transferred back to the master `BFS` :: `purchacedAssetsDebtAssetHoldings`) or liquidated during successful settlement.
- ❖ `PublicKey facilityFunder` identifies the source of funding for CPF business transactions. In this use-case it is set to the `PublicKey` of the `BorrowerWallet`, described in Section

6.3.6 as that wallet's role is to acquire funding from the BoE, necessary to cover APF's QE obligations executed by this `FacilityWallet`;

- ❖ `HashMap<String, RequiredFundsRequest> fundedFacilityFundsRequests` - manages funding requests specific to `FacilityWallet`'s operation. Use of this element is demonstrated in the sequential steps of Section 7.5 - step 10 - to fulfil matching obligations that were a priori agreed earlier in step 3 and to proceed to the settlement of this business transaction, the transaction proposal request for matching security (tokenized TBs) is generated and sent to the `BorrowerWallet`.
- ❖ `HashMap<String, PreTransactionParts_CP_TB> cp_Tb_TransactionParts` - manages pre-transaction parts (transaction proposals) for CP and TB exchanges, ensuring that each step of the transaction is recorded and processed accurately.

In keeping with previous wallet designs, `FacilityWallet` is enabled to manage access permissions and secure repositories for cryptographic transactional HTLC secrets (see description is the Section 2.3.2, and implementation in the Chapter 7), ensuring secure implementation of atomic business transactions and the APF's operational efficacy within the BFS framework. Furthermore, it acts as a repository for cryptographic identities and as a dynamic participant in business transactions, encapsulating the operational authority of its owner - the hypothetical employee of APF tasked with CPF operations. This specialised wallet ensures secure, verifiable interactions within the BFS transactional ecosystem, enhancing both trust and integrity. The CPF's primary operation involved the direct acquisition of investment-grade CP from issuers, symbolised in this Thesis by the simulated entity CPI-1 (described next in Section 6.3.6), thereby providing an alternative funding source during capital market constriction. This operation was financed through tokenized TBs provided by the BoE - `LenderWallet` (see Section 6.3.6), with the APF's `FacilityWallet` facilitating the swap of these tokenized TBs for tokenized CP in a secure and efficient manner, showcasing the BFS's capacity to streamline complex financial transactions. The `FacilityWallet`'s implementation showcases the BFS's capability for streamlining and secure, complex financial transactions through blockchain technology. The integration of the `FacilityWallet` within the BFS addresses the functional requirements of its stakeholders and sets a precedent for the application of blockchain technology in facilitating critical financial mechanisms like the CPF.

The `FacilityWallet` within the BFS plays a foundational role in enabling execution of the CPF's strategic initiative by the APF, aimed at injecting liquidity directly into the corporate sector and facilitating their access to capital markets. The use-case of CPF's operational mechanics, encapsulated within the BFS project's `FacilityWallet` are essential for understanding the BFS's contribution to this financial innovation. Specifically, the design an implementation of the `FacilityWallet` and its role within BFS serves as a critical element enabling illustration of the BFS' potential to enable of direct, top-down approach for liquidity provision from central banks to the wider non-bank economic entities. This direct channel of funds enhances access to public money, particularly in times of financial distress, showcasing the BFS's potential to contribute positively to economic stability.

Counterparty Wallet

The CPF was designed to facilitate the flow of capital to the corporate sector, significantly during the financial turmoil triggered by the Global Financial Crisis (GFC). Through the CPF, the APF aims to acquire investment-grade commercial paper (CP) directly from issuers, like the simulated entity Commercial Paper Issuer - 1 (CPI-1), thus providing an alternative funding source and enhancing liquidity in the financial ecosystem. The `CounterpartyWallet` within BFS embodies the role of **Commercial Paper Issuer - 1 (CPI-1)** acting as a market counterparty to `FacilityWAllet` in QE business transactions (see Section 7.5). This wallet functionality is governed by the specifics of the use-case and it is designed as a direct dealer for CP issues; it functions as an component within APF::BFS application to facilitate issuing and sale CP on behalf of CPI-1 into the APF::BFS. The wallet facilitates the issuance of CP into its escrow transferable assets within the APF's `CounterpartyWallet`, where the CP is held in custody until transferred to the APF. Within its operational role, the `CounterpartyWallet` serves several key functions:

- ❖ Facilitates the issuance and custody of CP by acting as an authorised representative for CPI-1 within the APF::BFS application;
- ❖ Manages and stores important documents such as admission letters and company information;
- ❖ Links to external funding sources, ensuring the availability of funds for asset purchases ;
- ❖ Holds CP in escrow, ensuring it remains in custody until it can be transferred into the APF.
- ❖ Processes transaction parts to prepare for CP issuance and TB exchange.

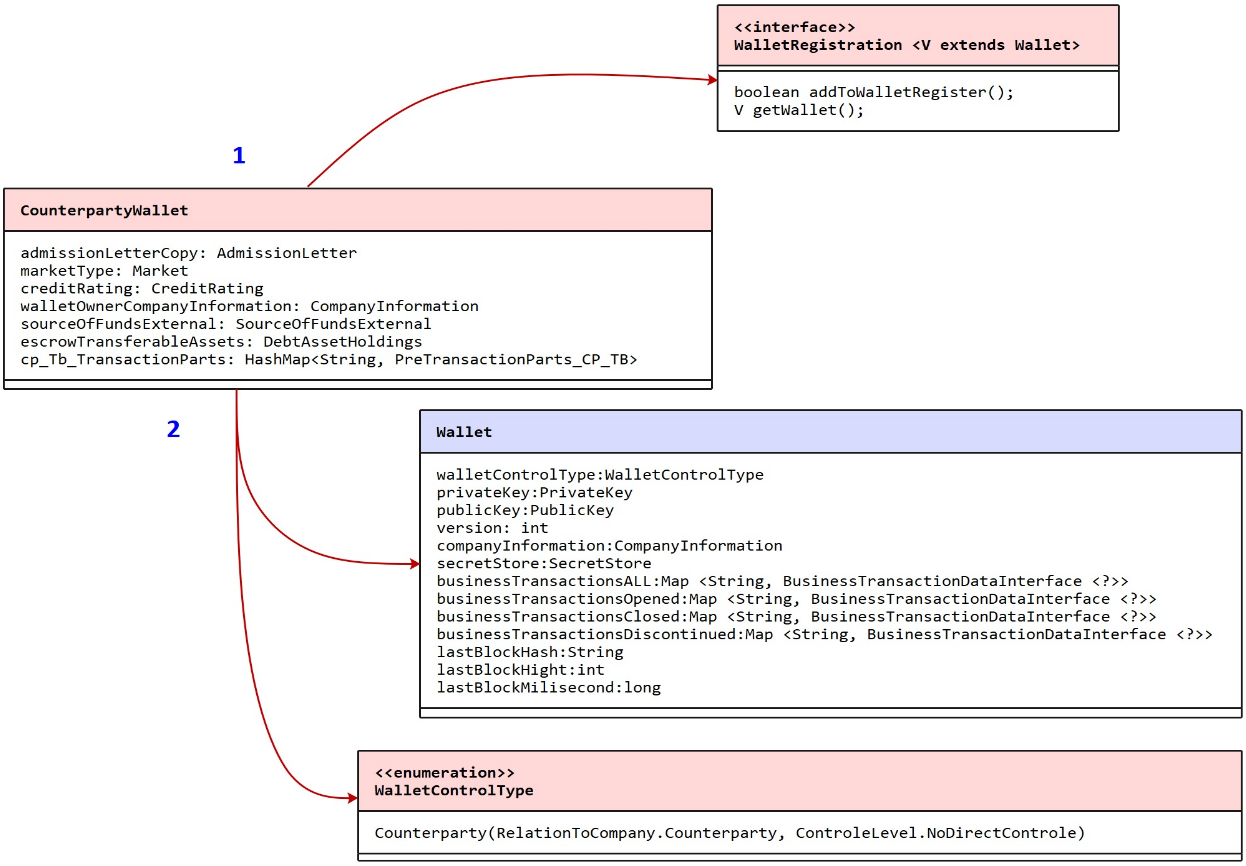


Figure 6.37: Counterparty Wallet.

Fig. 6.37 illustrates data elements and additional functionality of the `CounterpartyWallet` class. As an extension of the superclass `Wallet`, it embodies the BFS's commitment to standardise and manage distinct role-specific functionalities for each economic entity within its ecosystem. By implementing the `WalletRegistration <CounterpartyWallet>` interface, the `CounterpartyWallet` class is capable of registering within host BFS application (the APF::BFS - as a counterparty wallet). Key components of the implementation that allow the `CounterpartyWallet` to inherit standard wallet functionalities while adding specific features required to act as a market counterparty in CPF transactions, include (see Fig. 6.37):

- ❖ `AdmissionLetter admissionLetterCopy` - confirmation of the acceptance to participate in CPF. To be able to sell CP to the AP, the issue of the CP first has to apply and to be accepted to the CPF. This element is modelled on "Participation in the Bank of England's Asset Purchase Facility: Corporate Bond Purchase Scheme (the Scheme) – Admission Letter." ;
- ❖ `Market marketType` - an enumerator indicating what type of market (PRIMARY or SECONDARY) in relation to issuance of CP this participant belong to.
- ❖ `CreditRating creditRating` - credit rating of this economic entity. An example could

be Standard & Poor's (S&P) - the rating scale ranges from AAA (highest) to D (lowest); CreditRating. AAA("extremely strong capacity to meet financial commitments");

- CompanyInformation walletOwnerCompanyInformation - details about the company that owns this wallet;
- SourceOfFundsExternal sourceOfFundsExternal - external funding source for provision of tokenized Treasury Bills; connection channel between this wallet and its legal owner - the CPI-1::BFS;
- DebtAssetHoldings escrowTransferableAssets - the debt assets that the CPI-1::BFS is ready to transfer to the APF::BFS FacilityWallet, but with ownership retained until business transaction atomic settlement conditions are met. This escrow repository is utilised as follow: the counterparty (CPI-1::BFS) *blocks the securities* to be delivered by *transferring the securities to so called* ***escrow account*** *or* ***relay point*** within the CounterpartyWallet of the APF application. However, although these transferred assets are inside of the wallet that is part of the APF application, the ownership of this assets is still with the CPI-1. This is enabled to facilitate control and coordination of the *PvP business transaction* (see Section 7.5) the two BFS ledgers of the APF and the CPI-1.
- HashMap<String, PreTransactionParts_CP_TB> cp_Tb_TransactionParts - manages pre-transaction parts (transaction proposals) for CP and TB exchanges, ensuring that each step of the transaction is recorded and processed accurately.

The CPI-1, through the Counterparty Wallet, initiates the first transaction proposal — the OfferToSellCommercialPaper offer. This offer includes the sale of investment-grade, 3-month maturity "Unsecured Commercial Paper" tokens to the APF's FacilityWallet in exchange for maturity-matched "Treasury Bill" tokens. This transaction demonstrates the Counterparty Wallet's critical role in facilitating the secure and efficient exchange of financial instruments, thereby supporting the APF's liquidity provision objectives under the CPF initiative. The design and the implementation of the CounterpartyWallet class underscores the BFS's dedication to establishing and managing distinct, role-specific functionalities for each participant within BFS ecosystem. This wallet, in its role as a CPI-1 representative, facilitates the direct acquisition of CP by the APF and enables execution complex business transactions, such as that described in Section 7.5 and illustrated in Figures 7.4 - 7.6 . Lastly, by representing an opposite end of the direct channel for liquidity provision from central banks to the broader non-bank economic entities, BFS enables execution of critical financial mechanisms like the CPF. In this way, BFS sets the stage for a deeper understanding of the BFS's operational efficiency and its contribution to improving liquidity distribution methodologies in the digital age.

Architectural Characteristics of Wallets.

The BFS introduces a suite of specialised wallet roles, each tailored to facilitate distinct transactional and operational requirements within its ecosystem. The design and functional implementation of these wallets - DirectorWallet, Shareholder Wallet, BorrowerWallet, LenderWallet, FacilityWallet and CounterpartyWalle - illustrates key architectural characteristics vital for a resilient and adaptable blockchain-based system. These characteristics include **adaptability**, **extensibility**, **portability** and **flexibility**, which collectively ensure the BFS's longevity, scalability and robustness in a rapidly evolving technological landscape.

Adaptability of the BFS wallets are designed with a forward-looking approach, acknowledging the inevitability of technological advancements and shifts in financial regulatory frameworks. This adaptability is achieved through abstracting wallet functionalities to a level where future changes in blockchain technology or financial practices can be integrated with minimal modifications. For example, the FacilityWallet can accommodate new forms of liquidity support mechanisms as they emerge, without requiring an overhaul of the BFS architecture. This adaptability ensures that the BFS remains relevant and functional even as external conditions evolve.

Extensibility is at the heart of the BFS wallet designs, allowing for the addition of new functionalities as the need arises. This is particularly evident in the system's ability to incorporate new wallet roles or expand existing ones to address emerging financial operations or compliance requirements. The `CounterpartyWallet` for instance can be extended to facilitate additional types of financial instruments beyond Commercial Paper, such as bonds or equities, without disrupting existing functionalities. This extensibility supports the continuous growth and enhancement of the BFS, enabling it to serve a broader range of financial transaction scenarios over time.

Portability is a critical feature of the BFS wallets, designed to ensure seamless operation across different blockchain platforms or financial environments. This is achieved by adhering to industry-standard protocols and practices in the wallets' development, facilitating their integration into diverse technological ecosystems. The `BorrowerWallet` and `LenderWallet` for example, are constructed with compatibility in mind allowing for their use within different blockchain frameworks or financial systems with minimal adaptation. This portability enhances the BFS's utility and adoption potential across various contexts.

Flexibility within the BFS is demonstrated through the wallets' ability to adapt to changes in their operating environment or functional requirements. This flexibility is enabled by modular wallet designs that allow for dynamic configuration of functionalities according to specific transactional needs or regulatory mandates. All wallets, including `DirectorWallet` and `ShareholderWallet` illustrate this flexibility, offering customisable access controls and transaction approval mechanisms to align with evolving corporate governance structures or shareholder agreements. This ensures that the BFS can efficiently respond to internal changes or external pressures, maintaining its effectiveness and compliance.

Such design and functional implementation of the BFS wallet roles collectively underpin the BFS's adaptability, extensibility, portability and flexibility. These architectural characteristics are fundamental to the BFS's ability to navigate and thrive amidst the complexities of modern financial systems and technological innovation. By embedding these principles within each wallet's architecture, the BFS ensures a robust, scalable and forward-compatible solution capable of addressing the current and future needs of its stakeholders.

Research Contributions from Wallet Design and Implementation.

Each wallet, through its defined role and corresponding responsibilities, ensures that every participant within the BFS can fulfil their functions effectively, thus enhancing the overall integrity and efficiency of the BFS architecture. This specialised wallet ensures secure, verifiable interactions within the BFS transactional ecosystem, enhancing both trust and integrity. The design and implementation of all specialised wallets such as the `DirectorWallet`, `Shareholder Wallet`, `BorrowerWallet`, `LenderWallet`, `FacilityWallet` and `CounterpartyWalle` - within the BFS contribute to the research by demonstrating practical applications of blockchain technological components in orchestration of business operations and execution of transactions. These wallets present a tailored approach to managing and recording business events and sporting documentation transactional activities within the BFS environment.

- ❖ *Role-Based Access Control (RBAC).* These wallets illustrate the implementation of RBAC within a blockchain system, allowing for a nuanced control over data and transaction, critical for maintaining the integrity of corporate actions.
- ❖ *Smart Contract Integration.* The wallet's ability to interact with smart contracts for corporate-related transactions demonstrates the potential for automating governance processes from dividend issuance to voting on corporate resolutions.
- ❖ *Tokenization of Financial Assets.* The incorporation of the tokenization of shares, commercial paper and treasury bills allows for efficient transfer and recording of ownership on the blockchain. This facilitates the liquidity in a digital economy.
- ❖ *Source of Funds.* Incorporating a direct communication channel between different entities of the BFS ecosystem, provides a model for ensuring the provenance of funds used in transac-

tions, critical for compliance and anti-money laundering (AML) procedures.

- *DvP Transactions for Shares.* By managing the transfer of shares and funds through DvP (see Section 7.2), or swap of commercial paper for treasury bills (see Section 7.5), these wallets offer a secure method for settling complex conditional transactions that can be adopted in broader financial practices.
- *Enhanced Security and Efficiency.* The wallets contribute to the BFS by showing how blockchain components can be adapted to facilitate secure transactional interactions and streamlined business operations, reducing the need for intermediaries and cutting down on settlement times and costs.
- *Innovation in Financial Products.* The wallets serve as prototypes for developing new financial products on the blockchain, such as tokenized gilts or digital corporate bonds, opening avenues for further financial innovation.

The demonstration of the `DirectorWallet`, `Shareholder Wallet`, `BorrowerWallet`, `LenderWallet`, `FacilityWallet` and `CounterpartyWalle`, together with their superclass `Wallet`, within the BFS provide valuable overall research contributions by demonstrating how blockchain technology can be adapted to complex business operations. Their design and functionality enhance current transactional systems and provide the basis for innovative applications of blockchain to business execution, contributing to the ongoing evolution of blockchain as a transformative force in the economic domain.

Chapter 7

BFS Smart Contract Data Flow

Following the insights from the previous Chapter 6 on BFS Implementation and Proof of Concept, where granular exploration was presented on the PoC implementation for the BFS architecture. This demonstration was provided through the lens of Domain-Driven Design approach, with its sub-division into core sub-domains and related bounded contexts. Progressing to Chapter 7, where detail examination will be communicated for the *BFS Smart Contracts Data Flow* between participants of the BFS ecosystem. This section of the Thesis will focus on the demonstration of the practical application and functionality of the BFS implementation within Java environment.

Here, the discussion revolves around detailed processes involved in BFS *smart contracts* as a self-executing terms of the agreement directly written into lines of code and their central role in automating business and accounting transactions, together with lending arrangements, within the BFS framework. This chapter sets out detailed examination of the data flow and operational processes within the BFS, specifically through the prism of business and accounting transaction smart contracts. Smart contracts, integral to the BFS framework, automate and secure financial transactions within the blockchain ecosystem, embodying the fusion of technology with traditional financial processes. It is structured to provide a comprehensive overview of:

❖ **Introduction to Atomic Swap and HTLC** applicability to the APF::BFS use-case implementation: A brief on what HTLC is and its importance in the BFS ecosystem, setting the stage for its utilisation within complex business transactions smart contracts of the BFS (see Section 7.1).

❖ **Data Flow and Operational Processes within Smart Contracts**. A thorough demonstration of the data flow and the logical sequence of operational processes encapsulated in the business and accounting transactions smart contracts of the BFS PoC. Key areas of focus include:

1. ***Initial Share Settlement Business Transaction (DvP with HTLC)***: An exploration of the first business transaction smart contract executed between the APF and the BoE, showcasing how the APF issues and sells tokenized Shares to the BoE (see Section 7.2).
2. ***Accounting Transaction Smart Contract***: Details on how economic data from the Initial Share Settlement Business Transaction is recorded into the APF::BFS's accounting systems, ensuring accurate documentation in the General Journal and General Ledger (see Section 7.3).
3. ***Drawdown Request on Loan Agreement with HTLC***: Although not a stand-alone transaction, this part illustrates how a drawdown request can be implemented within the BFS framework to facilitate subsequent QE activities by APF. It is a One-Way-Payment, execution of which is implemented as a sequential step of the complex atomic three-party QE business transaction, described next (see Section 7.4).
4. ***Atomic Four-Party QE with HTLC***: A section on the implementation of the Quantitative Easing (QE) process, highlighting how the BFS facilitates asset acquisitions by

the APF through smart contract automation. It is a Delivery vs. Delivery (DvD) and One-Way-Payment transaction, demonstrating the BFS's smart contract's capability to streamline automate QE process, especially in asset acquisitions by the APF (see Section 7.5). All designed and implemented transactional components, demonstrated within complex transactional chain, together with their relationships and operational processes to orchestrate this complex flow, are modelled on the existing APF's *Operating Procedures for the Asset Purchase Facility* (Bank of England (2024*a*)), *Terms and Conditions of the Asset Purchase Facility* (Bank of England (2024*b*)), *Bank of England: Settlement Procedures for the Asset Purchase Facility* (Bank of England (2010*c*)) and *Bank of England: Market Notice 24 April 2009* (Bank of England (2009*d*)).

❖ **Process Steps and Diagrams**. Detailed descriptions of the operational steps for each outlined smart contract, complemented with diagrams to visually represent the transactions and data flow within the BFS prototype.

Through this detailed examination of smart contract data flow and processes, this chapter aims to shed light on the BFS's operational efficacy in a controlled laboratory environment. The primary objective is to demonstrate the BFS's capabilities in automating complex financial transactions, thereby offering significant insights into the potential of blockchain technology to redefine financial reporting and liquidity management. As one navigates through the chapter, the mechanisms for automation introduced by the BFS's smart contracts are demonstrated, offering a comprehensive overview of their functionality.This exploration highlights the technical capabilities of the BFS prototype and its practical implications for the future of financial technologies.

7.1 Atomic Swap DvP for Share Settlement Business Transaction.

This section of the report outlines applicability of the atomic swap mechanism with HTLC in business transaction smart contracts executed in the BFS ecosystem. Because the BFS ecosystem assumes a multi-blockchain environment, where each economic entity owns its own individual BFS to guarantee atomicity for complex transactions whose operations span several chains Lu et al. (2024) and to ensure seamless coordination of atomic value exchange, this project employs Hashed Time Lock Contract (HTLC) functionality outlined in Section 2.3.2. In the BFS ecosystem, atomic swaps would be implemented as part of the broader processes of all business transaction smart contracts, where these two concepts are adapted for the exchange of issued shares for digital funds. Below, a narrative elaboration on the functional sequence of the processes for the Atomic Swap Mechanism for DvP transaction in Section 7.2 and Fig. 7.1 is provided and at high-level it involves:

1. *Initiation of Transaction.* On the APF's side of the atomic swap, the APF initiates a DvP transaction by creating a first transaction proposal. This proposal contract specifies the terms of the share sale, including the number of shares, share price and the identity of the involved parties. The same process will be repeated by the BoE side of this swap.
2. *Locking Mechanism.* During this process, both parties will employ funds verification consensus and HTLC a locking mechanism that will hold tokenized Shares and the digital funds in the corresponding escrows (wallet local repositories) of each of the entities. The shares from the APF and the funds from the BoE are locked against any other spend, preventing either party from accessing them until certain conditions are met, such as successful settlement, failure of cryptographic verifications or expiration of time.
3. *Secret Generation and Hashing.* Both parties generate cryptographic secrets hashed and shared between them. These hashes serve as proofs of commitment to this transaction.
4. *Verification and Confirmation.* The implementation of the atomic swap model then requires verification by each of the parties that the hashes received match the secrets provided. Upon successful verification, each party automatically confirms that the transaction can proceed.
5. *Settlement Execution.* Upon mutual verification, the two legs of the business transaction smart contract execute the exchange atomically (transfer of share - 1st leg; transfer of digital

funds - 2nd leg). This means that the shares are transferred to the BoE's wallet and the corresponding funds are transferred to the APF's wallet almost simultaneously.

6. *Record-Keeping.* The Business Translation Register of each of the counterparties records completed business transaction, providing an audit trail for both parties and any regulatory observers.
7. *Completion Failure.* If either of the legs of the transaction are not fulfilled, the atomic swap logic must ensure that the assets revert to their original owners; this means the shares go back to the APF and the funds return to the BoE. As this PoC demonstrates only a *happy path* implementation scenarios, all other transaction execution scenarios are left for the design and implementation of future research.

The implementation of atomic swaps for complex multi-blockchain transactions in the BFS ecosystem ensures secure, efficient and transparent execution and settlement of business transactions between heterogeneous business entities and liquidity providers like central banks, by marrying traditional financial procedures with the innovative capabilities of blockchain technology.

7.2 First Business Transaction: Initial Share Settlement Business Transaction (DvP with HTLC).

In this Section, the central point for the exploration of the procedural operations within business transaction smart contract mechanism is the "Initial Share Settlement Business Transaction (DvP with HTLC)", which marks the first business transaction executed between the APF and the BoE. This transaction illustrates the BFS's adeptness at facilitating complex financial operations, demonstrating how the APF issues and sells tokenized shares to the BoE in a seamless and secure manner. At the heart of this transactional narrative is the exchange of 100 Ordinary Shares, each with a par value of £1.00, from the APF to the BoE, paired with a reciprocal payment of £100 from the BoE to the APF. This exchange is initiated immediately following the instantiation of the APF::BFS, showcasing the system's capability to execute predefined transactions though automation of the value exchange process in alignment with the transaction's predefined conditions. This initial transaction presents the BFS artefact's operational capabilities and sets the stage for subsequent implemented and future transactions within the BFS framework.

In developing a smart contract for this transaction, the focus is on the articulation of its core components, including roles of the participating actors and the critical elements that underpin this transaction. The "web" of interactions and dependencies among these components is guided by the principles of Domain-Driven Design (DDD) and the Design Science Research Methodology (DSRM). These methodologies facilitate a structured approach to conceptualising and implementing complex technological systems Wieringa (2014), ensuring that the application of designed smart contracts reflect a deep understanding of the domain and are grounded in a rigorous, iterative design process. DDD emphasises the importance of a model that accurately represents the nuanced relationships and concepts inherent in business operations (Evans (2004)). When combined with DSRM's focus on the methodical creation and refinement of innovative artefacts, this approach ensures the smart contract's theoretical soundness and practical viability. Through this lens, the "First Business Transaction: Initial Share Settlement Business Transaction (DvP with HTLC)" section aims to illuminate the BFS's potential to improve digital transactions, underlining the overall system's adaptability in handling complex financial exchanges.

Next, a taxonomy of participating actors and key elements that underpin this transaction is presented:

❖ **Actors** - *Participants in the Transaction.* This business transaction involves three primary participants

   1. `DirectorWallet` - an APF `DirectorWallet` responsible for selling the shares to the BoE and handling the receipt of £100 on behalf of the APF.

2. `DirectorWallet` - another APF `DirectorWallet` responsible for witnessing and authorising the transfer of `ShareCertificate` token to the BoE.
3. `ShareholderWallet`- BoE as an entity acting as both the receiver of `ShareCertificate` that confirms ownership of 100 Ordinary Shares from the APF and the sender of £100.00 to the APF.

❖ **Fundamental Elements** and their data:

1. `IN10 in10` - the data for this form is used to provide all contractual specifications for this business transaction. By its design, `IN10` form facilitates all information for issues and tokenizes Ordinary Shares with a nominal value, representing the initial equity stake of the BoE in the APF. The publicly available data utilised for `IN10` form is synthesised from the incorporation documentation of the APF in Companies House's UK public repository.
2. The `BFS::ShareStock shareStock` - is an element of the BFS class for an entity such as APF. It is a business type specific repository for the private or public company limited by shares. This repository records all share issuance and sale related activity of the APF. It is designed to play a role in recording and tracking all share-related activities, ensuring every issuance, transaction and adjustment is accounted for and transparent.
3. The `BFS::ShareStock::HashMap <Share, Double> authorisedShares` - is the repository where all issued `Share` tokens are stored for the APF. Inline with the contractual conditions of the APF us-case, all 100 `Share` tokens issued at instantiation are earmarked for the BoE, the designated initial member of the APF and are awaiting the completion of the DvP atomic swap (see Section 6.2.2 on BFS instantiation, together with Fig. 6.22).
4. The `ShareCertificate` - in compliance with traditional practices, it is transferred from the share issuer to the shareholder. It contains all essential information regarding share ownership in a tokenized form. This includes details such as the certificate number, company name, shareholder name and the quantity, nominal value and type of share issued. Additionally, the certificate outlines the payment status of the shares (partly paid, fully paid, or unpaid) and features authorizing signatures and `PublicKeys` from all transaction participants - the share issuer, a witness and the shareholder. These signatures, together with the public keys of these signers ensure the legitimacy and mutual acknowledgment of the share transaction.
5. Information for transaction proposals, such as
`ShareOffer shareOffer` and the `ShareBid digitalFundsOffer`, described in Section 6.3.3 are the *pre-settlement agreements and arrangements*. The data parameters for these proposals are captured through a verb—IN10— form, serving as the basis for this transaction. This pre-transaction data is essential as it initialises precise automation of legally binding terms and conditions, including the number of shares, their nominal value and the identities of the transacting parties.
6. `SecretStore` - manages the secure storage and retrieval of cryptographic secrets, utilised in the HTLC implementation by the transaction wallets. Its transaction specific elements, such as `Secret String` and `SecretHash` elements are as follows:
   (a) For the APF side - the `Secret String` X_{APF} and its `SecretHash`, generated as: $Y_{APF} = \mathrm{H}(X_{APF})$.
      i. the `SecretHash` Y_{APF} will share with the BoE prior to sending the first Transaction proposal;
      ii. the `Secret String` X_{APF} will be utilised by the APF to claim a payment for the Shares from the BoE.
   (b) For the BoE side - the `Secret String` X_{BOE} and its `SecretHash`, generated as: $Y_{BOE} = \mathrm{H}(X_B A_{OE},)$:
      i. the `SecretHash` Y_{BOE} will be shared with the APF prior to sending the 2nd transaction proposal;

ii. the `Secret String` X_{BOE} will be utilised by the BoE to claim Shares from the APF.

7. `InitialShareSettlementBusinessTransaction_DvP` - this is the settlement transaction - the *business transaction smart contract*, the Java implementation of which `extends abstract class AbstractBusinessTransactionData` and `implements interface BusinessTransactionDataInterface` `<V extends AbstractBusinessTransactionData>` described in Section 6.3.3 and illustrated in Fig. 6.29. Following successful completion of this business transaction, in the `APF::BFS::ShareStore` - 100 ordinary shares will transition from the authorised share stock repository into the outstanding share stock repository, reflecting the successful execution of the share transfer. Lastly, the relevant settlement data from the share certificates and the transfer of digital funds amounting to £100.00 is recorded within the metadata of this business transaction, ensuring transparency and accuracy in financial record-keeping.
8. `BusinessTransactionRegister` - an APF application local repository, storing this business transaction when completed.

Now this research will explore how the economic data stemming from this completed business transaction undergoes transformation into accounting data. This transformation is achieved through the utilization of accounting smart contracts ultimately aggregating the data into the relevant accounts within the accounting system.

7.2.1 Sequential Process Flow of the Initial Share Settlement Business Transaction (DvP with HTLC).

This section presents a detailed walkthrough of the first business transaction — a Delivery versus Payment (DvP) atomic swap—between the Asset Purchase Facility (APF) and the Bank of England (BoE). This part of the Section will provide a detailed look into how the DvP model is tailored for the BFS, demonstrating the issuance and exchange of shares versus digital funds between the APF and the BoE. It provides granular descriptions of the step-by-step procedures, illustrated in Fig. 7.1, which starts with the APF instantiating the transaction and culminates with the successful settlement of tokenized shares.

The aim of this section is to demonstrate the practical execution and advantages of implementing a DvP transaction within the BFS framework, leveraging blockchain's unique capabilities for financial operations. This demonstration offers insights into the smart contract's role in simplifying and securing the exchange of financial instruments. The BFS's approach to managing and documenting this fundamental business transaction sets the stage for future discussions on transforming financial processes through advanced technological solutions.

The sequential process flow, illustrated in Fig. 7.1 consist of 17 primary steps and is as follows:

1. At instantiation of the APF::BFS,hinging of the data provided in the form `IN10`, 100 Ordinary `Share` tokens, with par of £1.00 per share, are issued and stored in the `BFS::ShareStock::authorisedShares` repository.
2. The `DirectorWallet director1` of the APF (as the issuer and the original holder of the `Shares`) utilises a `IN10` form from `PrivateRegister` to initialise the `ShareOffer shareOffer` - the 1st transaction proposal. As part of this process, a `ShareCertificate` that represents the transfer of the `Share` token's ownership from the APF to the BoE is issued and included in the data of this 1st transaction proposal. This `ShareCertificate` is not signed yet by any of the relevant counterparties (the issuer/seller, the shareholder/buyer and the witness [... provide reference]), and thus is not authenticated as a proof of ownership transfer for the `Share` token from its issuer (the APF) to the prospective shareholder (the BoE). The copy of such an unsigned `ShareCertificate` is stored in the `ShareStok` repository, for future reference and until either: the successful completion or the cancellation of whole ongoing business transaction. As an aspect of the HTLC framework and following the affirmative confirma-

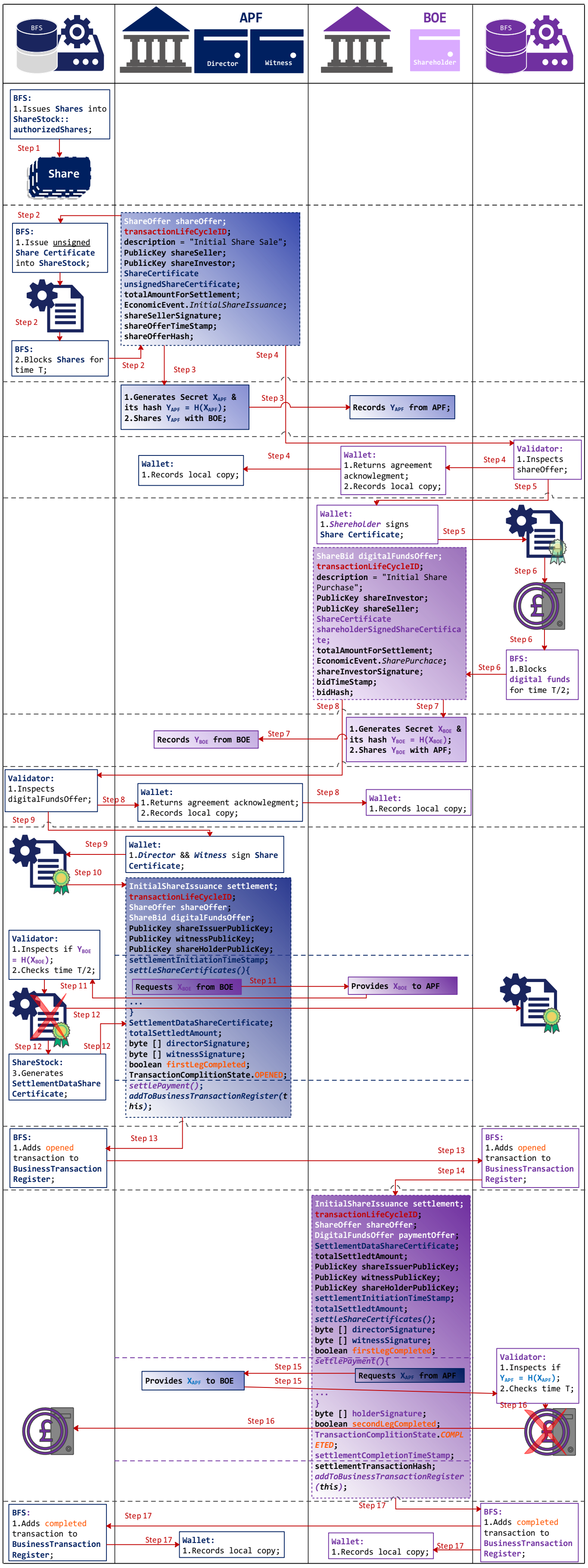


Figure 7.1: Delivery-vs-payment (DvP) transaction flow for shares vs payment multi-ledger atomic swap implementing HTLC.

tion of the consensus's `verifyTransactionInputValues(...)` verification mechanism, 100 Ordinary `Share` tokens are blocked in the `ShareStock` for a time interval `T`. The ownership of this `Shares` will be released to:

(a) the BoE if, within time interval `T` the BoE provides `Secret String` X_{BOE} that satisfies `SecretHash` $Y_{BOE} = \mathrm{H}(X_{BOE})$. The `Secret String` X_{BOE} and the `SecretHash` Y_{BOE} will be communicated to the APF in due course of this transaction (see part 7 below).

(b) the APF, if the time interval `T` passes, or Secret verification is not successful.

In the `ShareOffer shareOffer` - the 1st transaction proposal, the `DirectorWallet director1` outlines:

(a) `String transactionLifeCycleID` - is generated and used as the unique identifier of this `InitialShareSettlementBusinessTransaction_DvP` life-cycle by all its constituent parts.

(b) `String description = "Initial Share Sale"`;

(c) `PublicKey shareSeller` = `DirectorWallet director1`'s `PublicKey`;

(d) `PublicKey shareInvestor` = `ShareholderWallet shareholder`'s `PublicKey`;

(e) `ShareCertificate unsignedShareCertificate` - a draft `ShareCertificate` that is not signed by any of the relevant signatories;

(f) `double totalAmountForSettlement` = £100.00 - the total expected monetary value of all `ShareCertificate` from `unsignedShareCertificates`;

(g) `EconomicEvent.InitialShareIssuance` - will be used by the `Accountant` of `APF::BFS` to map completed business transaction to the relevant smart contract for accounting transaction;

(h) `byte []shareSellerSignature` - `director1`'s signature of this `ShareOffer shareOffer`;

(i) `Instant shareOfferTimeStamp` - time stamp of the `director1`'s signature;

(j) `String shareOfferHash` - one-way hash function (e.g., SHA256 hash) of this `ShareOffer shareOffer`;

3. Next, prior to submitting completed `ShareOffer shareOffer` (1st transaction proposal) to the BoE's `ShareholderWallet shareholder`, the APF's `DirectorWallet director1` generates a `Secret String` X_{APF} and its `SecretHash` that contains $Y_{APF} = \mathrm{H}(X_{APF})$. The `director1` wallet communicates the `SecretHash`, containing the Y_{APF} with the BoE and stores the `Secret String` X_{APF} and its `SecretHash` locally in the `Wallet::SecretStore ::HashMap <String, SecretHash> mySecretHashStore`. This `Secret String` X_{APF} will be utilised by the APF to claim the payment from the BoE for the `Shares` sold.
4. The APF's `director1` wallet sends the completed (signed, dated and hashed) `ShareOffer shareOffer` to the BoE's `ShareholderWallet shareholder`, whose `Validator` inspects its data integrity and verifies its content. Upon confirmation of acceptance from the `shareholder` to the APF, both parties save a copy of this proposal locally for future reference.
5. The `ShareholderWallet` of the BoE, upon affirmative return from its `Validator`, proceeds to the 2nd transaction proposal component - the `ShareBid digitalFundsOffer`. Ahead of composing the `digitalFundsOffer`, the `shareholder` inspects unsigned `ShareCertificate` supplied with the `ShareOffer`. Upon agreement, the `ShareholderWallet` signs `ShareCertificate` and stores its copy locally for future reference. This `ShareCertificate` which is only signed by the `shareholder` will be appended to the data of the `ShareBid digitalFundsOffer`. Once more, as this `ShareCertificate` token is not signed yet by all relevant counterparties (missing signatures of: issuer/seller, and the witness, it is not yet authenticated as a proof of ownership transfer for the `Share` tokens between the issuer (the APF) to the prospective shareholder (the BoE).
6. As an aspect of the HTLC framework, outlined in ..., and following the affirmative confirmation of the consensus's `verifyTransactionInputValues(...)` funds verification mechanism, the agreed upon amount of digital funds to cover financial obligation to the APF (£100.00) is blocked for the time interval `T/2`. These digital funds will be released to:

(a) the APF if, within the time interval `T/2`, the APF provides `Secret String` X_{APF} that satisfies SecretHash $Y_{APF} = \mathrm{H}(X_{APF})$ that is in possession of the Shareholder Wallet (see 3).

(b) the BoE, if the time interval T/2 passes, or the Secret verification is not successful.

The BoE's ShareholderWallet shareholder generates ShareBid digitalFundsOffer - 2nd transaction proposal. In this proposal, the shareholder outlines:

(a) String transactionLifeCycleID - is copied from the ShareOffer;
(b) String description = "Initial Share Purchase";
(c) PublicKey shareInvestor = ShareholderWallet shareholder's PublicKey;
(d) PublicKey shareSeller = DirectorWallet director1's PublicKey;
(e) ShareCertificate shareholderSignedShareCertificate - ShareCertificate that is only signed by the ShareholderWallet shareholder;
(f) double totalAmountForSettlement= £100.00 - the total expected monetary value of all ShareCertificate from shareholderSignedShareCertificates;
(g) EconomicEvent.SharePurchace - will be used by the Accountant of BoE::BFS to map completed business transaction to the relevant smart contract for accounting transaction;
(h) byte [] shareInvestorSignature - shareholder's signature of this ShareBid digitalFundsOffer;
(i) Instant bidTimeStamp - time stamp of the shareholder's signature;
(j) String bidHash - one-way hash function (e.g., SHA256 hash) of this ShareBid digitalFundsOffer;

7. Next, prior to submitting completed ShareBid digitalFundsOffer (2nd transaction proposal) to the APF's DirectorWallet director1, the BoE's ShareholderWallet generates a Secret String X_{BOE}, and its SecretHash, that contains $Y_{BOE} = \mathrm{H}(X_{BOE})$. The shareholder1 wallet communicates the SecretHash containing Y_{BOE} with the APF and stores the Secret String X_{BOE} and its SecretHash locally in the Wallet::SecretStore ::HashMap <String, SecretHash> mySecretHashStore. This Secret String X_{BOE} will be used by the BoE to retrieve the agreed upon ShareCertificate token from the APF.
8. The BoE's shareholder wallet sends completed (signed, dated and hashed) ShareBid digitalFundsOffer to the APF's director1, whose Validator inspects its data integrity and verifies its content. Upon confirmation of acceptance from the director1, both parties save a copy of ShareBid digitalFundsOffer locally for future reference.
9. The DirectorWallet director1 of the APF, upon affirmative return from its Validator, proceeds to the concluding component of this business transaction life-cycle - the InitialShareSettlementBusinessTransaction_DvP settlement (see 7).
   → Ahead of initiating settlement, the director1 wallet inspects the ShareCertificate received with the ShareBid, signed by the shareholder.
   → Upon an overall agreement, the director1 wallet signs this ShareCertificate and employs DirectorWallet director2, as the witness signatory on the ShareCertificate.
   → At this stage, as the ShareCertificate is signed by all required parties and is dated, it authenticates that the ownership of existing Share tokens is transferable from the APF to the BoE, i.e., that from the date of this ShareCertificate, the shares in question are granted and made available to the new shareholder
   → The copy of this issued ShareCertificate is kept by the APF for their records locally in ShareStock:: HashMap <String, ShareCertificate> issuedShareCertificatesActiveCOPY for future reference.
   → Furthermore, during settlement, the ShareCertificate token will be transferred (settled) directly between the APF::BFS::ShareStock to the BoE. The ShareCertificateSettlementData, which comprises of non-sensitive ShareCertificate's data, will be generated and appended to the InitialShareSettlementBusinessTransaction_DvP settlement.
10. Commencing here, the DvP business transaction is ready for the final settlement, i.e., the

atomic swap of the `ShareCertificate` token against the agreed amount of digital cash (£100.00). On the pre-agreed trade date at the trade time, the APF's `DirectorWallet director1`, utilising form `IN10`, earlier saved `ShareOffer shareOffer` and `ShareBid digitalFundsOffer`, and the `ShareCertificate` signed by all parties and dated, instantiates `InitialShareSettlementBusinessTransaction_DvP settlement`. In this `settlement` transaction the `director1` specifies:

(a) `String transactionLifeCycleID` - is copied from the `ShareBid`;
(b) `ShareOffer shareOffer` - a copy of agreed on `ShareOffer` from local repository;
(c) `ShareBid digitalFundsOffer` - a copy of agreed on `ShareBid` from local repository;
(d) `PublicKey shareIssuerPublicKey` = `DirectorWallet director1`'s `PublicKey`;
(e) `PublicKey witnessPublicKey` = `DirectorWallet director2`'s `PublicKey`, who acts as the witness signatory on `ShareCertificate` token;
(f) `PublicKey shareHolderPublicKey` = `ShareholderWallet shareholder`'s `PublicKey`;
(g) `Instant settlementInitiationTimeStamp` - the time stamp of the call to `boolean settleShareCertificates(...)` method outlined below in 10h;
(h) Furthermore, this DvP `settlement` transaction will consist of 2-legs of the atomic assets transfers (implementations of which are described next):

1st leg - `boolean settleShareCertificates(...)` functionality is employed by the APF's `director1` wallet and performs relocation of `ShareCertificate` token into the ownership of the BoE. Only successful completion of this leg allows this DvP transaction to proceed;

2nd leg - `boolean settlePayment(...)` functionality is employed by the BoE's `shareholder` wallet and performs transfer of the digital funds into the ownership of the APF. Successful implementation of this method is the 2nd leg of this atomic `settlement` transaction, which leads to full completion.

(i) together with `boolean firstLegCompleted` flag and `boolean secondLegCompleted` flag. These flags will coordinate both legs of settlement completion for this business transaction.

11. As initiator of the settlement, the APF's `director1` wallet calls `boolean settleShareCertificates(...)` method (see 10h). During this process, the wallet verifies secret:
    (a) the APF's `DirectorWallet` requests X_{BOE} from the BoE's `ShareholderWallet`;
    (b) the BoE's `ShareholderWallet` provides X_{BOE} to the APF's `DirectorWallet`;
    (c) the APF verifies `if` (Y_{BOE} = H(X_{BOE}) `and` checks the time is within `T/2`);
12. Upon successful HTLC verification, the `director1` wallet completes the 1st leg of settlement:
    (a) transfers the `ShareCertificate` token directly from `APF::BFS::ShareStock` to the BoE's `ShareholderWallet`. As part of this process, the `Shares` that have been sold during this business transaction become outstanding and are relocated to the `APF::BFS::ShareStock::outstandingShares`;
    (b) generates `SettlementDataShareCertificate settlementDataShareCertificate` from local copy of transferred `ShareCertificate` (from `issuedShareCertificatesActiveCOPY`), and appends this `settlementDataShareCertificate` data to the business transaction;
    (c) together with `double totalAmountForSettlement` = £100.00 - the total monetary value of all `ShareCertificate` from `settlementDataShareCertificate`;
    (d) then sets `boolean firstLegCompleted` = `true`, to indicate successful transfer of `ShareCertificate` to its `shareholder`;
    (e) `byte [] directorSignature` - `DirectorWallet director1`'s signature of this `settlement` transaction;
    (f) `byte [] witnessSignature` - `DirectorWallet director2`'s signature of this `settlement` transaction;
    (g) `TransactionComplitionState.OPENED` - as the payment leg of this business transaction is not completed yet, the state of this transaction remains `OPENED` and therefore is

unavailable to be processed by the accounting transaction smart contract until both legs are completed;

13. At successful completion completion of the 1st settlement leg, the APF's `director1` wallet calls `boolean addToBusinessTransactionRegister(this)` and the `OPENED settlement` business transaction is added to the `BusinessTransactionRegister`s of each of the counterparties.
14. As the counterparty to this settlement, the BoE's `shareholder` wallet is automatically notified by its `Consensus` of the opened transaction in its `BusinessTransactionRegiste`. This business transaction is pushed from the BoE's `BusinessTransactionRegister` into the `shareholder` wallet.
15. As responder of the settlement, the BoE's `shareholder1` wallet calls `boolean settlePayment(...)` method (see 10h). During this process, the wallet verifies secret:
    (a) the BoE's `ShareholderWallet` requests X_{APF} from the APF's `DirectorWallet`;
    (b) the APF's `DirectorWallet` provides X_{APF} to the BoE's `ShareholderWallet`;
    (c) the APF verifies `if` $Y_{APF} = \mathrm{H}(X_{APF})$ `and` checks the time is within `T`);
16. Upon successful HTLC verification, the `shareholder` wallet completes 2nd leg of settlement:
    (a) transfers it financial obligation of pre-agreed amount of digital funds from the BoE's `ShareholderWallet` directly to the APF's `BFS`. The digital funds are available to the BoE's `ShareholderWallet` though its connection to the BoE's BFS via `ShareholderWallet::SourseOfFunds` link.
    (b) adds `byte [] holderSignature` - `ShareholderWallet shareholder`'s signature of this `settlement` transaction;
    (c) then sets `boolean secondLegCompleted = true`, to indicate successful settlement of digital funds obligation to APF;
    (d) sets `TransactionComplitionState.COMPLETED` - as both legs of this business transaction are completed;
    (e) `Instant settlementCompletionTimeStamp;` - the time stamp of the successful return from the `boolean settlePayment(...)` method outlined in 10h;
    (f) generates the `String settlementTransactionHash` - one-way hash function (e.g., SHA256 hash) of the whole `InitialShareSettlementBusinessTransaction_DvP settlement`;
17. At successful completion completion of both settlement legs, the BoE's `shareholder` wallet calls `boolean addToBusinessTransactionRegister(this)`, and the `COMPLETED settlement` business transaction is added to the `BusinessTransactionRegister`s of each of the counterparties and to each of the participating `Wallet`s respectively for future reference. At the end of the accounting cycle, or on demand, the economic data produced by the explicitly completed transactions from the `BussinessTransactionRegister` is available to be transformed into accounting data and propagated though the stages of the accounting cycle.

This Section navigated through the BFS Smart Contracts Data Flow, illustrating and describing functional application of smart contract business transaction in the BFS architecture. The sequential process flow of the Initial Share Settlement Business Transaction (DvP with HTLC) between the APF and the BoE has served as a concrete demonstration of how blockchain technological components can be used to facilitate secure and complex financial interactions. The automation of trust established through this smart contract highlights the transformative potential of blockchain in the domain of digital transactions. As one moves forward, this exploration sets a precedent for how financial entities might engage with each other in this digitised and decentralised future, leveraging blockchain technology for enhanced efficiency, transparency and security.

This transformation represents a critical nexus between the initial financial exchange and the broader financial reporting process, encapsulating the essence of how “accounting smart contracts” facilitate the accurate and timely updating of financial records. By elaboration on this process, the next Section will not only deepen our understanding of the BFS's capabilities, but also highlight the system's role in driving innovation within the domain of financial accounting.

7.3 First Accounting Transaction Smart Contract.

The initiation of the first *accounting transaction smart contract* within the BFS artefact, specifically for the Initial Share Settlement Business Transaction, implemented between the `APF::BFS` and the `BOE::BFS` is essential in demonstrating the integration of business activity of an economic entity with blockchain-enabled accounting practices.

The `journalInitialShareSettlementBusinessTransactions_DvP()` method of the `Accountant` class serves as an initiation interface for implementation of this first *accounting transaction smart contract* within the BFS artefact. It is designed to automate generation of accounting entries specific to the share settlement transaction (the use-case of this study), facilitating accuracy and transparency of financial recording. This process is an illustration of the project's aim of adapting blockchain capabilities for enhancing financial transactions' reliability and clarity, thereby providing an example of verifiable and accurate financial management within blockchain architecture adaptation.

At this part of the project's journey, at explicit completion of business transaction, the `Accountant` of the APF::BFS implements its functionality - the `Accountant.journalInitialShareSettlementBusinessTransactions_DvP();` method - the *accounting smart contract* call to process data from first business transaction, described in the previous Section 7.2.1. As a pre-requisite of the process, the `EconomicEvent.InitialShareIssuance` was already assigned to the `AllowableAccountingTransaction` class by the `Accountant` of the `APF::BFS`. A mandatory step to automate rules-based double-entry accounting process for every accounting smart contract, each `AllowableAccountingTransaction` that embodies this double entry automation is set up by the`Accountant` in advance is compliance with economic entity specific accounting requirements. `AllowableAccountingTransaction` is instantiated with relevant `EconomicEvent` flag, together with relevant set of `Accounts` within the `AffectedAccounts` class. These digital arrangements will govern which particular financial `Accounts` will be debited and/or credited during execution of *accounting smart contract* - in accordance with double-entry accounting. The accounting smart contract method, described in this Section, is specifically design to demonstrate the adaptive implementation of the APF use-case to requirements of current research project, such as examination of adaptation of blockchain's procedural elements to fulfil accounting process. The publicly available data for this scenario is retrieved from incorporation documentation of the APF in Companies House's public repository (Companies House (2024*b*)). In this orchestrated scenario, the hypothetical accountant of APF is tasked with processing a initial business transaction into the APF's internal accounting records. This accountant is assumed to have knowledge of all necessary preconditions for processing executed business transaction into internal accounting records of the APF (Companies House (2024*b*)). These preconditions are:

- the business transaction was executed at APF incorporation;
- it is a sale of initial share issuance from the APF, the issuer and the parent company of the APF, the BoE and the shareholder;
- transaction marks the sale of the full authorised share capital of the APF, underlining the significance of this financial activity within the entity's operational framework;
- the `EconomicEvent.InitialShareIssuance` was assigned to the business transaction and it was mapped with with the APF's company number within `HashMap <String, EconomicEvent> economicEventsCounterparties` of the `AbstractBusinessTransaction` - the superclass of the settlement smart contract transaction, highlighting its importance in classifying business activity. When retrieved during accounting smart contract execution, this `EconomicEvent.InitialShareIssuance` will be used to engage relevant

`AllowableAccountingTransaction` described earlier in this Section.
- ❖ the amount of digital funds paid for the Shares by the BoE is expected to be the full amount agreed by the parties, i.e., there are no payments left outstanding;
- ❖ the transaction type is DvP atomic swap, that has settled successfully, i.e., the `TransactionComplitionState` is marked as `COMPLETED`.

The step-by-step code implementation for the execution of the accounting smart contract in the BFS, specifically the `Accountant.journalInitialShareSettlementBusinessTransactions_DvP();` method, represents a detailed and sequential process that involves coordination of multiple sub-domains of the BFS's architecture. This step-by-step process is illustrated in Fig. 7.2, and it ensures the integration of transactional data into the accounting records.The execution of this accounting smart contract is synchronous, where the implementation of the next step depend on success of the previous, providing reliability, transparency and integrity of the overall process.

`Accountant.journalInitialShareSettlementBusinessTransactions_DvP()` method is called to process the Initial Share Settlement Business Transaction. Below is a detailed explanation of each step of this `Accountant.journalInitialShareSettlementBusinessTransactions_DvP();` method :

1. Retrieving Business Transaction. The specific business transaction is fetched from the `BussinessTransactionRegister's businessTransactionsRegister`, a `HashMap<String, BusinessTransactionDataInterface<?>>` designed to hold all business transactions of the APF.
2. Verification of Preconditions. Before processing, the method verifies if the transaction meets following mandatory pre-conditions:
   (a) verifies for that `TransactionComplitionState.COMPLETED`, and
   (b) the `EconomicEvent.InitialShareIssuance` is assigned to the correct (APF) company number;
3. Extracting Share Certificate Settlement Data. The method retrieves the `ShareCertificateSettlementData`, which contains all relevant information about the tokenized Shares involved in the transaction, such as monetary value; the total settled amount of digital funds is paid by the shareholder - the BoE to the share issuer - the APF. This value of £100.00 is set as a new transactional value that will be assigned to the debited and credited accounting entries (in compliance with double-entry accounting principles).
4. To digitally enforce correct accounting rules for the execution of this accounting smart contract the method identifies the specific `AllowableAccountingTransaction` object related to this initial investment which dictates the rules for debiting and crediting accounts. The `EconomicEvent.InitialShareIssuance` is mapped to the entry within `AccountingTransactionRegister`'s `HashMap <String, AllowableAccountingTransaction> allowableAccountingTransactions`. The earlier described instance of `AllowableAccountingTransaction initialInvestment` class is utilised here (see Section 6.1.4, Fig. 6.13 for "Allowable Accounting Transaction class instantiation").
5. Initiation of Accounting Transaction. `new AccountingTransaction (...)` object is instantiatedThis object is created with detailed parameters including `AllowableAccountingTransaction allowableAccountingTransaction`, transaction description `String transactionDescription` ,`LocalDate recordDate` - the Share Certificate issue date, `double absolutTansactingAmountsettled` monetary amount and `String sourceBusinessTransactionHash` - the transaction hash of the settlement transaction - the source for this accounting record.
6. When the execution is passed to the constructor of this `new AccountingTransaction (...)` all general purpose field of the class are set to represent parameters passed, such as:
   (a) `this.transactionDescription = transactionDescription;`
   (b) `this.recordTimeStamp = DateFormating.localDateToInstant(recordDate);`

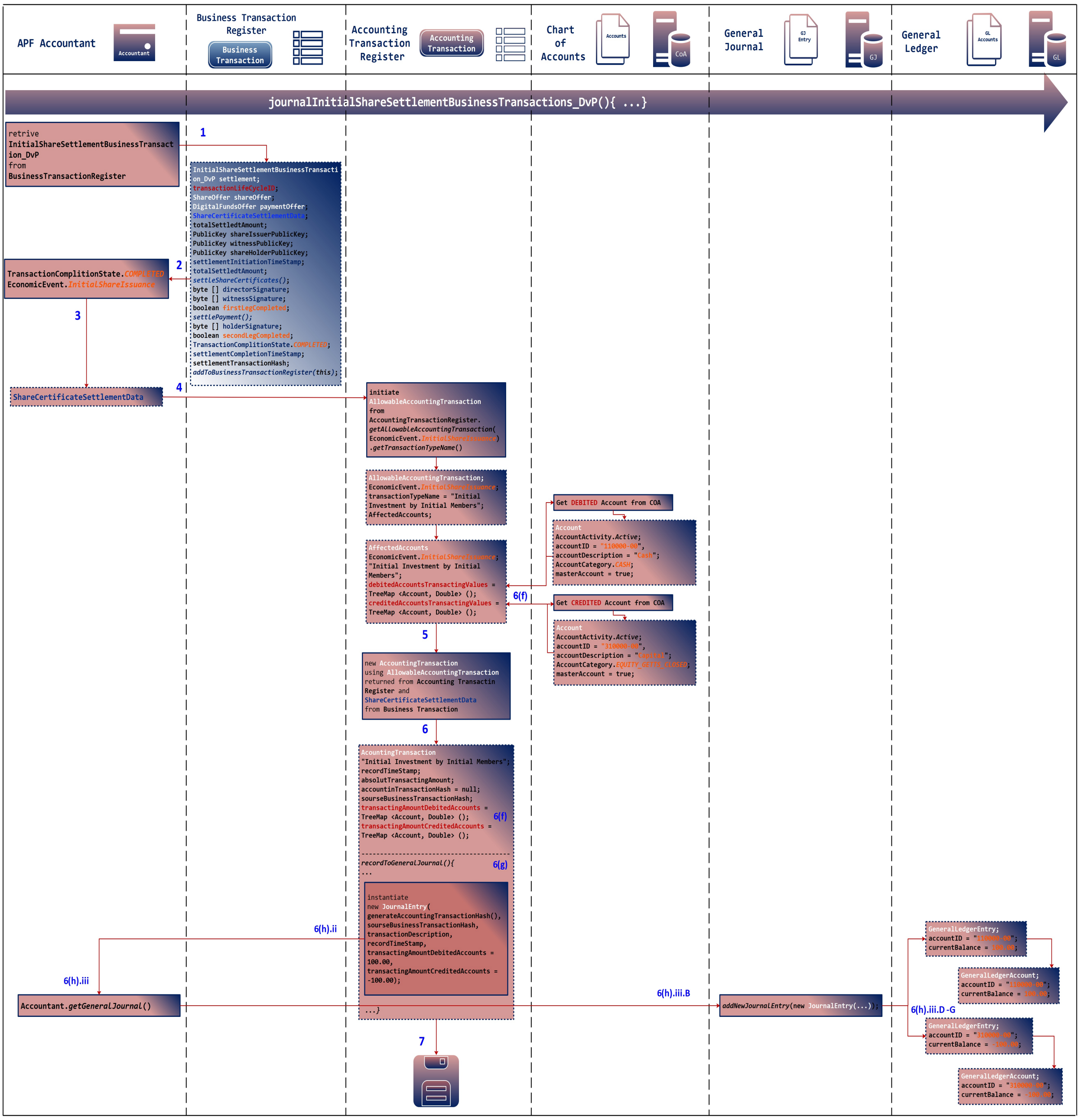


Figure 7.2: First accounting transaction smart contract sequential data flow.

(c) `this.absolutTransactingAmount = absolutTansactingAmount;`

(d) `this.accountinTransactionHash = null` - to be calculated at successful completion of this accounting transaction;

(e) `this.sourseBusinessTransactionHash = sourseBusinessTransactionHash;`

(f) The next part of the execution within `AccountingTransaction` constructor is to identify `Accounts` to be used in the accounting journals and ledgers, where the new value extracted from completed business transaction will be posted. Utilising `AffectedAccounts`, debited and credited accounts from the `AllowableAccountingTransaction` passed in constructor of this `AccountingTransaction`, the specific `Accounts`, mapped with the absolute transactional value of £100.00 of digital funds and are set to the `transactingAmountDebitedAccounts` and `transactingAmountCreditedAccounts TreeMaps` within its instance.

(g) The final execution step of the `AccountingTransaction` constructor is to call a private method of its class: `recordToGeneralJournal();`

(h) From this point onwards, the execution of the *accounting smart contract* moves to generate accounting records within internal accounting journals and ledgers of the APF::BFS.

i. Inside of the private method of `AccountingTransaction` class ::`recordToGeneralJournal() -` this process facilitates automation of necessary steps of the accounting cycle that will generate and post a new entries to: in General Journal (Fig. 6.6) and the General Ledger (Fig. 6.7).

ii. Within this method, a new journal entry is instantiated that records required accounting information such as: `JournalEntry newJournalEntry = new JournalEntry( generateAccountingTransactionHash(), this.sourceBusinessTransactionHash, this.transactionDescription, this.recordTimeStamp, this.transactingAmountDebitedAccounts, this.transactingAmountCreditedAccounts);`

iii. In the following step, the General Journal repository of the APF::BFS `Accountant` class is engaged, where this new journal entry is stored: `Accountant.getGeneralJournal().addNewJournalEntry(newJournalEntry);`

A. Execution is passed to the General Journal Repository:

B. There, a new journal entry is added to the chronological repository of General Journal, called "journal records": `journalRecords.put(newJournalEntry.getRecordDate(), newJournalEntry);`

C. Upon successful addition to the `journalRecords`, next execution moves to updating General Ledger of the APF::BFS. A method `postToGeneralLedger(newJournalEntry)` is called, in which the copy of this posted journal entry is used to instantiate new debit and credit entries for General Ledger repository. In accordance with double-entry accounting principles, new General Ledger Accounts must be updated for the debited and credited side of the accounting equation.

D. First, the method generates the debited side `GL_Entry glEntry = new GL_Entry(debitsCopy.getKey().getAccountID(), newEntry.getRecordDate(), debitsCopy.getValue(), 0, newEntry.getAccountingTransactionHash(), newEntry.getSourseBusinessTransactionHash());`

E. Then post this debited general ledger entry to its general ledger account: `Accountant.getGeneralledger().postGL_Entry(glEntry);`

F. Last, the method generates and posts credited general ledger entry to its general ledger account:`GL_Entry glEntry = new GL_Entry (creditsCopy.getKey().getAccountID(), newEntry.getRecordDate(), 0, creditsCopy.getValue(), newEntry.getAccountingTransactionHash(), newEntry.getSourseBusinessTransactionHash());`

(i) Upon successful postings and updates, the execution of the `AccountingTransaction` constructor concludes and the call returns app-the-stack to the *accounting smart contract* - `accountant.journalInitialShareSettlementBusinessTransactions_DvP();` method.

7. Within `accountant.journalInitialShareSettlementBusinessTransactions_DvP();` method, the last procedure is left to implement.The finalised successful `AccountingTransaction at1` is reposited by the `Accountant` the `AccountingTransactionRegister` repository (see Section 6.1.5) for future reference and verification if/when necessary. This is achieved by calling the `AccountingTransactionRegister.addToAccountingTransactionRegister(...)` method.

The conclusion of the *accounting transaction smart contract* process marks a significant milestone in exploration of the BFS's capabilities. It illustrates transition of economic data from a completed business transaction from sub-domain of entity into the sub-domain of accounting. The automated generation of accounting entries specific to the share settlement transaction showcases the BFS's ability to facilitate precision in financial recording. This process, enabled by initiation interface for the accounting transaction within BFS artefact - the `Accountant.journalInitialShareSettlementBusinessTransactions_DvP()` method - ensures that each entry is automated and reflects all necessary detail and accuracy required for financial reporting. By defining permissible types of `AllowableAccountingTransactions` in advance(Stolowy & Ding (2019)), relevant to specific business activities and aligning these with corresponding flags of the `EconomicEvents` within related business transactions, the BFS accounting process automation minimises the potential for post-event manual manipulation of the accounting records. This methodical approach, anchored in predefined rules, enhances the integrity and accuracy of the financial data within the BFS. It enhances trust in accounting data, but also gives an example of adaptability of BFS architecture to various financial management requirements. This Section has highlighted some of the key aspect of the BFS: its ability to ensure that every economic event can be accurately and instantaneously captured, processed and integrated into an entity's accounting records, underscoring BFS's efficacy in providing post accounting period basis for verifiable monetary claims.

In transitioning to the next Section, the focus will shift once more to the demonstration of another business transaction data flow and its processes for orchestration of the strategic utilization of draw-down on an illustrative lending agreement. The next section will describe the key elements and roles of actors involved in this process, where the APF will act as the borrower and the BoE as its lender, with following implementation logic demonstrated from the perspective of the APF.

7.4 Drawdown Request on Loan Agreement with HTLC

The focus now is on the demonstration of a next sequential step of the APF use-case: acquisition of funding necessary for the APF to implement QE. BFS executes this mechanism through simulated lending agreement between the APF and the BoE, a draw-down request from the APF to the BoE, and the simulation of issuance and subsequent lending of tokenized Treasury Bills to the APF by the BoE. This lending processes is represented in the BFS prototype via the `LoanAgreement` smart contract mechanism. This technological construct articulates the lending terms between the APF (as the single borrower) and the BoE (as the lender), embodying the digital automatisation and execution of subsequent business transaction This section of the Thesis will explore key roles of the involved actors, primarily from the vantage point of the APF, as it engages in the drawdown process. It will also describe fundamental components necessary for this smart contract to complete successfully. The following narrative will provide granular view of the implementation of this lending agreement, a one-way payment transaction essential for the APF's liquidity management enabling subsequent asset purchase activities within the framework of QE. The upcoming demonstration offers further insights into the BFS's capability to facilitate complex financial processes, maintaining efficiency and integrity throughout.

❖ **Actors.** There are two participating BFS `Wallets` in the one-way payment transaction that enables loan draw-downs of Treasury Bills by the APF from the BoE:

1. `BorrowerWallet apfAsSubsidiaryBorrower -` an APF::BFS hypothetical employee role. Its functionalities are designed to facilitate the requirements of the APF in managing and initiating transactions related to funding acquisition.
2. `LenderWallet boeTreasuryLender` - is the counterparty wallet within APF::BFS application, whore role encapsulates the functionality required by the BoE to act as a lender by acting a counterparty in the funds acquisition business transaction, demonstrated in Section 7.5. In the sequential transactional data flow of this transaction, this wallet acts as the receiver of the 1st transaction proposal from the APF, where it retrieves the `DrawdownRequest` included and verifies it against pre-existing `LoanAgreement` terms.

Successful outcome of such verification will automate subsequent issuance and lending (transfer) of tokenized Treasury Bills to the APF.

❖ **Fundamental Elements.** The primary elements of the demonstration of the execution of this business transaction are:

- → `LoanAgreement` - this is a digitalised document designed as a complex smart contract that sets and automates hypothetical legal contractual arrangements between the APF and the BoE.In this Thesis, the lending agreement data (variables) and its rules (functionalities) are specified through adaptation and simplification for illustrative purposes of the use-case specific requirements. These requirements are drawn from the public data sources outlined previously.
- → `DrawDownRequest` - is another digitalised document designed component part that of the complex `LoanAgreement` smart contract. It is initiated by the APF. This request is attached as a data element within first transaction proposal of the flow, where it is sent to the BoE. Upon receipt of this request, the BoE, as the lender, verifies this request against existing, preliminary stored `LoanAgreement`, by invoking method of this agreement `enoughAllocatedFunds(DebtSecurities.TreasuryBill, amount)`.
- → `TreasuryBill` - is a tokenized representation of the `DebtSecurities.TreasuryBill` issued by the BOE. Upon successful verification of the `DrawDownRequest`, BoE issues a `TreasuryBill` token and transfers it into the ownership of the APF as the sequential step of the `SettlementTransactionLoanUtilisationTreasuryBills` business transaction smart contract. When a `TreasuryBill` token is transferred from the BoE to the APF, it is sequentially deleted from the BoE's `DebtAssetHoldings` repository after addition to the APF's `DebtAssetHoldings` repository.
- → `SettlementDataTreasuryBill` - The abstracted data record of `TreasuryBills` transferred during settlement. It contains all necessary data elements from these transferred TBs, for accounting and future reference when required, but with concentration protection of some private information about these TBs. This element persists within `SettlementTransactionLoanUtilisationTreasuryBills`.
- → A business transaction comprised of three components:
  - → `LoanUtilisationRequest` - 1st transaction proposal - request to draw funds by the borrower.
  - → `FundsTransferConfirmationByLender` - 2nd transaction proposal - approval notification by the lender.
  - → `SettlementTransactionLoanUtilisationTreasuryBills` - settlement transaction smart contract that facilitates transfer of tokenized `TreasuryBills` from ownership of the BoE to the ownership of the APF:BFS.

In the next section, the description proceeds with the granular processes involved in implementing a drawdown request within a loan agreement. This next segment will articulate how such requests are not simply documented financial operations but are a part of a digitalized smart contract system - a part of the BFS's liquidity management toolkit.

7.4.1 Sequential Process Flow of the Business Transaction for Drawdown Request on Loan Agreement (OwP with HTLC)

In this section of the report, the logic and mechanics from the perspective of the APF acting as a TBs borrower is described. The Section will illustrate the step-by-step implementation of public money liquidity request process, emphasising automation that underpins this objective of the BFS. This exploration will provide a detailed representation of the lending agreement's functional execution and frame the broader context of how blockchain can innovate liquidity provision in the economy.

In Fig. 7.3, the sequential steps of the successful loan drawdown is illustrated. As the precondition to this business transaction, the `LoanAgreement` that coordinates contractual basis

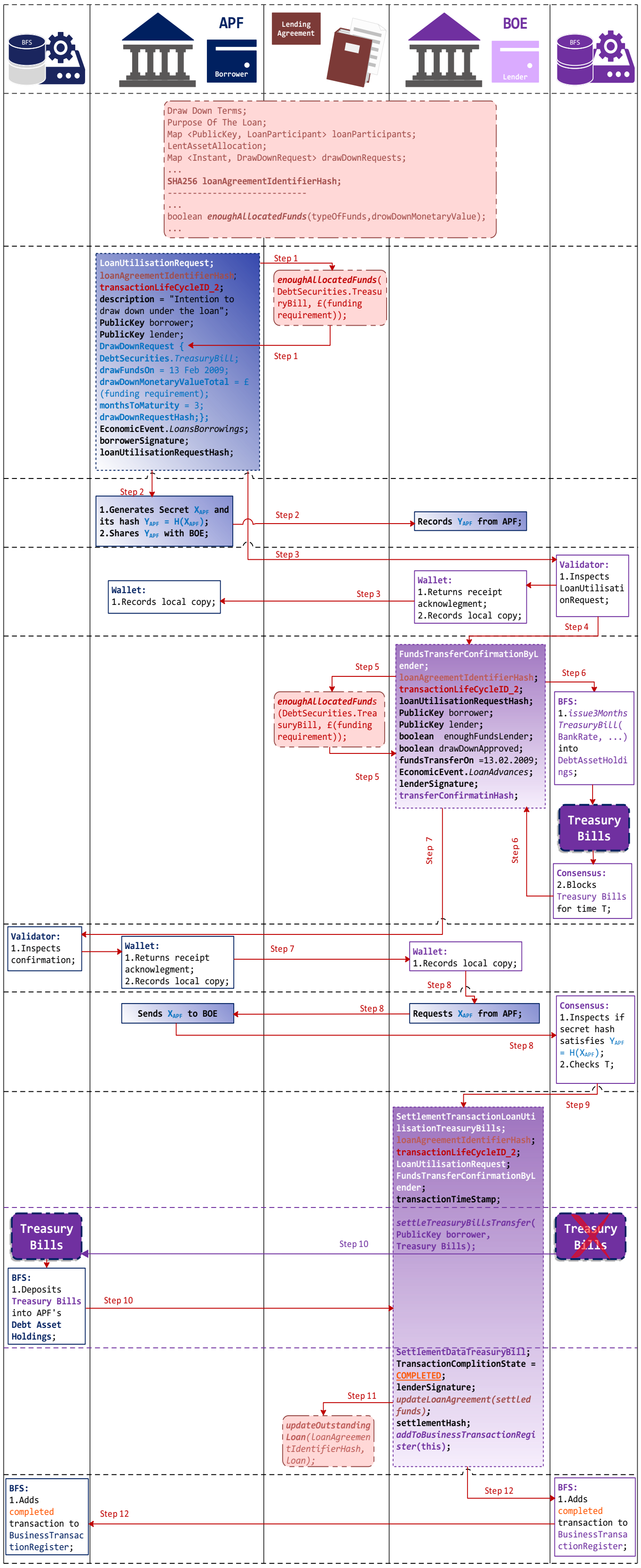


Figure 7.3: One-Way Payment Transaction Process Flow with HTLC: Loan Drawdown

for this transaction is assumed to be pre-arranged between the APF and the BoE and its latest copy is stored in the wallets' local repositories of all the parties to this agreement - in the `LoanAgreementRegisters`. The narrative description of the steps and the mechanisms of this process are as follows:

1. APF's employee wallet - `BorrowerWallet apfAsSubsidiaryBorrower` initiates `LoanUtilisationRequest requestToDrawTBills` - the 1st transaction proposal - to borrow `TreasuryBills` from BoE to cover APF's funding requirements. As part of this process, the amount and type of the requested funds is verified against the wallet local copy of the `LoanAgreement`:
   (a) The `LoanAgreement` is identified by its `loanAgreementIdentifierHash` from `BorrowerWallet`'s copy of the `LoanAgreementRegister`;
   (b) As part of this request, the APF's `BorrowerWallet` employs `LoanAgreement`'s functionality of `enoughAllocatedFunds(...)` to confirm locally that the requested `drawDownMonetaryValue` is still available by the lending facility;
   (c) As part of this process, an instance of `DrawDownRequest` is returned and appended to the APF's `LoanUtilisationRequest` data.
2. Following the affirmative confirmation of the `LoanAgreement`'s `enoughAllocatedFunds(...)` verification mechanism, the `apfAsSubsidiaryBorrower` wallet generates a secret X_{APF} and its hash Y_{APF} = H(X_{APF}). The `apfAsSubsidiaryBorrower` wallet shares Y_{APF} with the BoE.
3. The `BorrowerWallet` then sends a signed and dated `LoanUtilisationRequest` to the lender - the `LenderWallet boeTreasuryLender`, and saves a local copy of it in its wallet. The BoE as the lender of funds verifies the contents of the `LoanUtilisationRequest` and sends a confirmation of its receipt back to the APF.
4. At this stage, the verification of the `LoanUtilisationRequest` by the BoE's `Validator` for transaction integrity and the BoE's copy of the `LoanAgreement`'s validation of the borrower's identity (outlined in 6.3.2-6.3.2) performed. Upon its affirmative outcome, the BoE's `LenderWallet` proceeds to the generation of the `FundsTransferConfirmationByLender` - the 2nd transaction proposal.
5. As part of this confirmation, the BoE's `LenderWallet` also locally employs `LoanAgreement`'s functionality of `enoughAllocatedFunds(...)` (outlined in 6.3.2-6.3.2) to confirm that requested `drawDownMonetaryValue` is still within the remaining balance of the `LentAssetAllocation` of the lending facility.
6. the BoE then issues the requested `TreasuryBills` and blocks these in the escrow of the `LenderWallet boeTreasuryLender` for the time `T`.
7. `FundsTransferConfirmationByLender` 2nd transaction proposal is signed, hashed and sent to the the APF's `Validator` for inspection. This inspection includes: validation of the integrity of the transactional data. Both counterparties record locally in their respective wallets, the copy of this `FundsTransferConfirmationByLender` 2nd transaction proposal.
8. On the agreed settlement date and time, in order to settle the funds (i.e., the requested, approved and blocked amount of `TreasuryBills`), the BoE's `LenderWallet` requests the secret X_{APF} from APF. If APF provides the correct secret to the BoE that satisfies Y_{BOE} = H(X_{BOE}) and the time since the blocking of `TreasuryBills` by the BoE is still within `T`, the BoE can initiate the settlement part of this transaction.
9. The `SettlementTransactionLoanUtilisationTreasuryBills` is initiated by the BoE's `LenderWallet`, where both transaction proposals are recorded for future reference together with `SettlementDataTreasuryBill`. Generates `SettlementDataTreasuryBills` as a non-private metadata for the blocked `TreasuryBills` that remain a permanent record in this business transaction.
10. During this process, transfer of blocked `TreasuryBill` tokens is performed from the BoE's ownership to the APF's, i.e., the Treasury Bills are permanently deleted from the BoE's wallet local `DebtAssetHoldings` escrow repository after addition to the

APF::BFS `DebtAssetHoldings` repository.

11. After successful settlement operation, the `LoanAgreement` is updated to reflect outstanding available funding facility.
12. The business transaction state is marked as `completed` and this business transaction is added to the `BusinessTransactionRegister` of each of the counterparties. At the end of the accounting cycle, or on demand, economic data from explicitly completed transactions from the `BussinessTransactionRegister` are propagated through the relevant stages of the accounting cycle.

In closing this Section, one can observe the strategic and technical steps that the APF::BFS can employ to secure funding from the BoE through an automated `LoanAgreement` and `DrawDownRequest` smart contract. The process flow provided detailed progression of the APF's drawdown request, showcasing the BFS's capacity to accommodate complex financing needs within its digital framework. The smart contract's role in automating such financial arrangement reaffirms the BFS's contribution to enhancing the financial operations' efficiency and integrity. This Section has successfully demonstrated an automated and rule-based distribution of liquidity support, providing a system where validity and chronological integrity of financial claims are verifiably authenticated. The integration of a drawdown request within the BFS's smart contract infrastructure showcases an innovative approach to transparency and immediacy in financial transactions, offering a method for economic entities to demonstrate the validity of their financial claims without manual intervention.By successfully navigating the APF through this transactional journey, this research has reinforced the BFS's value as a utility for economic entities requesting public money funding and securing access to such liquidity through the validation of digital requests, effectively bridging the gap between verifiable financial needs and the provision of public money liquidity.

7.5 Second Business Transaction: Atomic Four-Party QE with HTLC

In the previous section, the exploration of the *Drawdown Request on Loan Agreement with HTLC* provided a contribution to understanding of how financial entities like the APF can interact with liquidity providers through adaptation of blockchain mechanisms, such as smart contracts. It demonstrated how such smart contracts could be designed to orchestrate handling complex financial scenarios, ensuring that, for example, the APF could access the necessary funds in an automated, efficient and reliable manner. This process lays the groundwork for subsequent asset purchases remit of the APF, integral to its market operations. The mechanisms developed in this Thesis are theoretically significant and instrumental in actualizing a digitally-enhanced economic environment where transactions are processed with improved efficiency and integrity. This Section transitions into another critical component of the demonstration of the BFS functionality, the actual use-case model of business transaction for an illustrative *Atomic Four-Party QE implementation*. This section will detail the complexities within the transactional orchestration among the APF, the BoE and the hypothetical Commercial Paper Issuer - 1 (CPI-1), by outlining the strategic implementation of a simulated CPF, an initiative by the APF to navigate the economic aftermath of a global financial crisis. Here, the report will demonstrate and describe the BFS prototype executing next business transaction, an atomic swap between tokenized Treasury Bills and tokenized Commercial Paper, showcasing an advanced application of another distinct BFS's smart contract in streamlining financial support to the corporate sector.

All designed and implemented transactional components, demonstrated within complex payment chain for this BFS transaction, together with relationships between them and operational processes to orchestrate these are modelled on the APF's *Operating Procedures for the Asset Purchase Facility* (Bank of England (2024*a*)), *Terms and Conditions of the Asset Purchase Facility* (Bank of England (2024*b*)), *Bank of England: Settlement Procedures for the Asset Purchase Facility* (Bank of England (2010*c*)) and *Bank of England: Market Notice 24 April 2009* (Bank of

England (2009*d*)).

In this section, the granular description of this business transaction processes will be provided, which at high-level will cover: the direct purchase of CP from the primary market by the APF; the linked automated financing of this purchase through simultaneous issue and transfer of TBs and the subsequent 2-legs atomic settlement mechanism that embodies the BFS's ability to handle complex, multi-party financial transactions with precision and assurance. In doing so, this research will highlight how the BFS, through its innovative use of smart contracts, can facilitate liquidity provision, ultimately contributing to a resilient economic infrastructure. Next this report will give a description of the essential components, which include the primary actors and fundamental elements. The interplay and relationships among these components are orchestrated through the application BFS smart contract logic, which serves as the guiding framework for their interactions within the BFS architecture. This logic is adaptable, ensuring that the smart contracts' functionality aligns with the overarching objectives of the BFS and functional requirements of BFS's stakeholders. The taxonomy of participating actors and key elements that underpin this transaction is presented next:

❖ **Actors.** There total of 4 `Wallet`s participating in this transaction, divided between 3 economic entities, such as: 1) the Commercial Paper Issuer 1 (CPI-1), 2) the APF and 3) the BoE:

1. `CounterpartyWallet issuerOfCP` - a counterparty role to te APF::BFS, a hypothetical market participant, a Commercial Paper issuing entity requesting public funding from the APF. It is a wallet that will initiate the business transaction. The wallet facilitates the issuance of CP into its escrow transferable assets within the APF's `CounterpartyWallet`, where the CP is held in custody until transferred to the APF during final 2-leg settlement.
2. `FacilityWallet commercialPaperFACILITY` - a role of the APF's hypothetical employee is modelled as an interface between the APF and the wider market participants. As a critical mechanism, this wallet is designed to function as a channel to provide funds directly to the corporate sector, enhancing their access to capital markets. In accordance with historic chronological events of the APF use-case, the primary goal of this wallet is to purchase investment-grade Commercial Paper (CP) directly from issuers (Bank of England (2009*c*)) - the hypothetical Commercial Paper Issuer - 1 (CPI-1). Furthermore, to obtain funding to match its transactional obligation against CPI-1, communicated to this wallet in the *trade input order transaction proposal* from CPI-1 (steps 6-8 of Fig.**??**), this wallet will instantiate `RequiredFundsRequest` to another APF::BFS employee, the `BorrowerWallet apfAsSubsidiaryBorrower`
3. `BorrowerWallet apfAsSubsidiaryBorrower` - in addition to the role and functionality described in the previous Section, this APF's employee wallet receives `RequiredFundsRequest` from APF's `FacilityWallet` and transforms the requested information into a `LoanUtilisationRequest` to be immediately forwarded to the lender - the `LenderWallet boeTreasuryLender`
4. `LenderWallet boeTreasuryLender` - is the counterparty type wallet within APF::BFS application, whore role encapsulates the functionality required by the BoE to act as a lender in this transaction's data flow. In the sequential transactional data flow of this transaction, this wallet acts as the receiver of the `DrawDownRequest` from the APF, where it retrieves and verifies it against pre-existing `LoanAgreement` terms. Successful outcome of such verification will automate subsequent issuance and lending (transfer) of tokenized Treasury Bills to the APF.

❖ **Fundamental Elements.** Taxonomy of the key elements required for the execution of this business transaction:

1. `transactionLifeCycleID` - a correlation ID for the whole life-cycle of this transaction;
2. `transactionHash` - a cryptographic hash of the internal transactional data;
3. `SecretStore` - a `Wallet` local repository for the secrets, used by this `Wallet`'s owner during complex business transactions. Secret as a concept - is the part of the HTLC

model. It is required for verifiable release of transactional debt assets such as tokenized TreasuryBills and CommercialPaper. The primary items that are constant to the secret are: 1) secret string (X), 2) secret hash (Y), and 3) a one-way hash function (H) that converts secret string X into the secret hash Y: Y = H(X). An entity that is requesting an economic item generates unique Secret. This entity than shares its unique hash of that secret Y with its counterparty - the economic item provider. When it is time for settlement of the economic item, an entity that expects to receive the payment provides a secret string X to its counterparty to verify and to draw on the expected economic item. The elements within SecretStore repository are:

   (a) Secret String - X - is a String generated by the Wallet and is stored in the wallet's SecretStore::HashMap<Secret String, SecretHash> mySecretHashsHashStore. One way SHA256 hash function is applied in this string to generate Y (String secretHashString_Y) which is the variable of the SecretHash class.
   (b) SecretHash - does not contain Secret String. Its variables are:
      - String transactionLifeCycleID - correlation ID of the transactional chain;
      - String sourceTransactionHash - the hash of the transactional part in the chain that required generation of this secret;
      - String secretHashString_Y; - secret hash (Y), where Y = H(X);
      - PublicKey secretProviderPublicKey - Public Key of the Wallet, who generated this secret;

4. DebtAssetHoldings - as described in the previous sections of this report, it is a BFS application local repository that persist debt assets of an entity;
5. EconomicEvent of each of the counterparty wallets that at successful completion of this transaction will be used by internal accounting smart contracts of the business entity. These are: PaymentReceipt, Payment, LoansBorrowings, and LoanAdvances.
6. DebtSecurities
   (a) CommercialPaperUnsecured - as a tokenized asset within BFS framework
   (b) TreasuryBill - as a tokenized asset within the BFS.
7. SettlementDataCommercialPaper -a non-private metadata for the blocked CommercialPaperUnsecureds that will remain a permanent record in this business transaction.
8. SettlementDataTreasuryBill - a non-private metadata for the blocked TreasuryBills, that will remain a permanent record in this business transaction.
9. LoanAgreement and DrawDownRequest - as described in the previous Section 7.4.
10. A business transaction that comprises of 9 (nine) components. These transactional components, relationships between them and operational processes to orchestrate these is modelled on the APF's *Operating Procedures for the Asset Purchase Facility* (Bank of England (2024*a*)) and *Terms and Conditions of the Asset Purchase Facility* (Bank of England (2024*b*)).
   (a) OfferToSellCommercialPaper offer - 1st transaction proposal sent from the CPI-1 to the APF;
   (b) Response_OfferAcceptance response - 2nd transaction proposal sent from the APF to the CPI-1;
   (c) TradeOrderFinancialSecurity tradeINPUT - part 1 of the settlement translation sent from the CPI-1 to the APF;
   (d) RequiredFundsRequest internalRequest - an APF internal request to obtain funding required to match its obligations to the CPI-1, specified in the trade input order;
   (e) LoanUtilisationRequest - transaction proposal - request to draw funds by the borrower.
   (f) FundsTransferConfirmationByLender - transaction proposal - approval notification by the lender.

(g) `SettlementTransactionLoanUtilisationTreasuryBills` - settlement transaction
(h) `TradeOrderFinancialSecurity tradeMATCH`
(i) `TradeTransactionCommercialPaperTreasuryBills settlement`

❖ **Functionality.** Some of the methodologies designed to enable execution of this business transaction:

1. `verifyNominal(...)` - a method that verifies nominal value of the debt asset;
2. `getDiscountRate(issuerCreditRating,oisRate3Month);` - a method that estimated discount rate applied to corresponding debt asset, such as `CommercialPaper` token;
3. `issue3MonthsCommercialPaper(...);` - the method that instantiates issuance of `CommercialPaperUnsequred` token, in using the `verifyNominal(...)` and `getDiscountRate(issuerCreditRating,oisRate3Month);` methods.
4. `settleTreasuryBillsTransfer(PublicKey borrower, Treasury Bills)` is part of the `SettlementTransactionLoanUtilisationTreasuryBills` - settlement transaction - a OwP part of the transactional chain where the settlement of `TreasuryBills` is first performed between the BoE and the APF::BFS in line with the process outlined in Section 7.4.
5. `settleTreasuryBills(...)` - the 1st leg of the final atomic settlement transaction, that transfers `TreasuryBills` from the APF to the CPI-1 within final `TradeTransactionCommercialPaperTreasuryBills settlement`. This method is used by the `FacilityWallet commercialPaperFACILITY`, the provider of the `TreasuryBill` token to the `CounterpartyWallet issuerOfCP`. This call transfers `TreasuryBill` token from the ownership of one APF::BFS into the ownership of CPI-1, by inserting the token into the `BFS::DebtAssestHoldings` repository of the later and permanently removing this token from `BFS::DebtAssestHoldings` of the APF::BFS.
6. `settleCommercialPaper(...)` - the 2nd leg of the final atomic settlement transaction that transfers a `CommercialPaperUnsequred` token as past of the `TradeTransactionCommercialPaperTreasuryBills settlement` business transaction. This method is used by the CPI-1, which is the provider of the `CommercialPaperUnsequred` token. It transfers this token from the ownership of one CPI-1 (escrow wallet local repository `DebtAssestHoldings`) into the ownership of APF's `BFS::DebtAssestHoldings` repository, by inserting this token into it and permanently removing the token from `BFS::DebtAssestHoldings` of the `CounterpartyWallet issuerOfCP`.

In conclusion of the narrative description of the taxonomy of components, actors and functionality that constitutes the foundation of this complex business transaction, demonstrates advanced design and implementation of the smart contract capabilities within BFS artefact. By modelling and implementing current transactional components that mirror the APF's operational procedures and terms, the design and implementation of this artefact has provided a window into how the BFS can orchestrate complex payment chains, describing relationships and operational processes that ensure the system's efficiency and effectiveness.

7.5.1 Sequential Process Flow of the Atomic Four-Party QE Business Transaction (DvD with HTLC)

In this section, the BFS prototype's sequential execution of another complex business transaction: *an atomic swap between tokenized Treasury Bills and tokenized Commercial Paper* will be presented. This segment aims to showcase another layer of the evolution in complexity within BFS's smart contract functionality.

Through a granular depiction and narrative description of the *atomic 4-party QE implementation*, this section will illustrate the transactional interactions among the APF, the BoE, and the hypothetical Commercial Paper Issuer - 1 (CPI-1). All designed and implemented transactional components, demonstrated within complex payment chain for this BFS transaction, together with relationships between them and operational processes to orchestrate these are modelled on the

APF's *Operating Procedures for the Asset Purchase Facility* (Bank of England (2024*a*)), *Terms and Conditions of the Asset Purchase Facility* (Bank of England (2024*b*)), *Bank of England: Settlement Procedures for the Asset Purchase Facility* (Bank of England (2010*c*)), and *Bank of England: Market Notice 24 April 2009* (Bank of England (2009*d*)).

This discussion will illuminate the strategic execution of a simulated CPF by the APF, devised to mitigate the economic repercussions of financial crises, thereby illustrating the BFS's comprehensive capabilities in facilitating complex financial transactions and enhancing the resilience of economic infrastructures.

In Fig. 7.4 to Fig. 7.6, the sequential steps of the successful execution for this atomic 4-party QE business transaction are illustrated. The description of all these procedural steps is as follows:

1. CPI-1 `CounterpartyWallet issuerOfCP` initiates 1st transaction proposal - the `OfferToSellCommercialPaper offer` to sell: investment grade, 3 months maturity, "Unsecured Commercial Paper" token to the APF's `FacilityWallet commercialPaperFACILITY` at issuance (on the primary market) in exchange for the maturity matched "Treasury Bill" token. In this `offer` to sell (see Fig. 7.4 step 1), the CPI-1:
   (a) generates `transactionLifeCycleID`. This is a correlation ID that will be used by all subsequent parts of this business transaction as it is a unique identifier for the whole transactional chain within this *business transaction smart contract*.
   (b) Furthermore, in this 1st transaction proposal, the `issuerOfCP` specifies the nominal amount of individual security offered. In accordance with "Facility Documentation" (see Functionality: 1), the nominal amount should be no less that £1 million and be expressed in increments of £0.1 million. For simplicity, in this project £ 1 million is assigned as the nominal value of the Commercial Paper to be issued.
   (c) The CPI-1 also defines the type of the offered security - `DebtSecurities.CommercialPaper`;
   (d) the type of the market (primary);
   (e) current credit rating of the issuer of the paper. Based on the "Facility Documentation", this rating then used in the trade input order (see below: 7) at agreed issue date to calculate the discount rate and the pricing for the Commercial Paper (Bank of England (2009*d*)) (see Functionality:2 3). For simplicity, "AAA" is assigned in this project as a hypothetical credit rating to the CPI-1;
   (f) that the seller of the Commercial Paper will be CPI-1 and the receiver of the Commercial Paper, the buyer, will be APF;
   (g) then signers, dates and hashes the 1st transaction proposal - the `offer`.
2. CPI-1 sends its `offer` to the APF's `commercialPaperFACILITY` wallet, whose `Validator` inspects and verifies its contents and the integrity of the data. Upon approval, the `commercialPaperFACILITY` wallet returns acceptance acknowledgement back to the CPI-1 and both parties record this `offer` locally.
3. Concurrently to successful verification, by accepting this `offer`, the APF's `FacilityWallet commercialPaperFACILITY` generates the 2nd transaction proposal `Response_OfferAcceptance response`. This `response`, confirms amongst other things:
   (a) the transaction date - 13th of February 2009;
   (b) the security type that the APF is agreeing to purchase - Commercial Paper;
   (c) the type of the matching security - Treasury Bills,
   (d) the nominal value of each - £1 million;
   (e) the maturity of both - 3 months for each (i.e., the trade date + 3 months);
   (f) transfers `transactionLifeCycleID` from the `offer` transaction proposal.

   The `commercialPaperFACILITY` signers, dates and hashes this `response`.
4. Next, the APF's `FacilityWallet` sends completed confirmation of the offer acceptance - the `response` - back to the CPI-1, who in turn: inspects this 2nd transaction proposal, acknowledges receipt back to the APF and both parties record a copy of it locally.
5. On the day of the agreed transaction date (or a trade date), trade orders must be input by

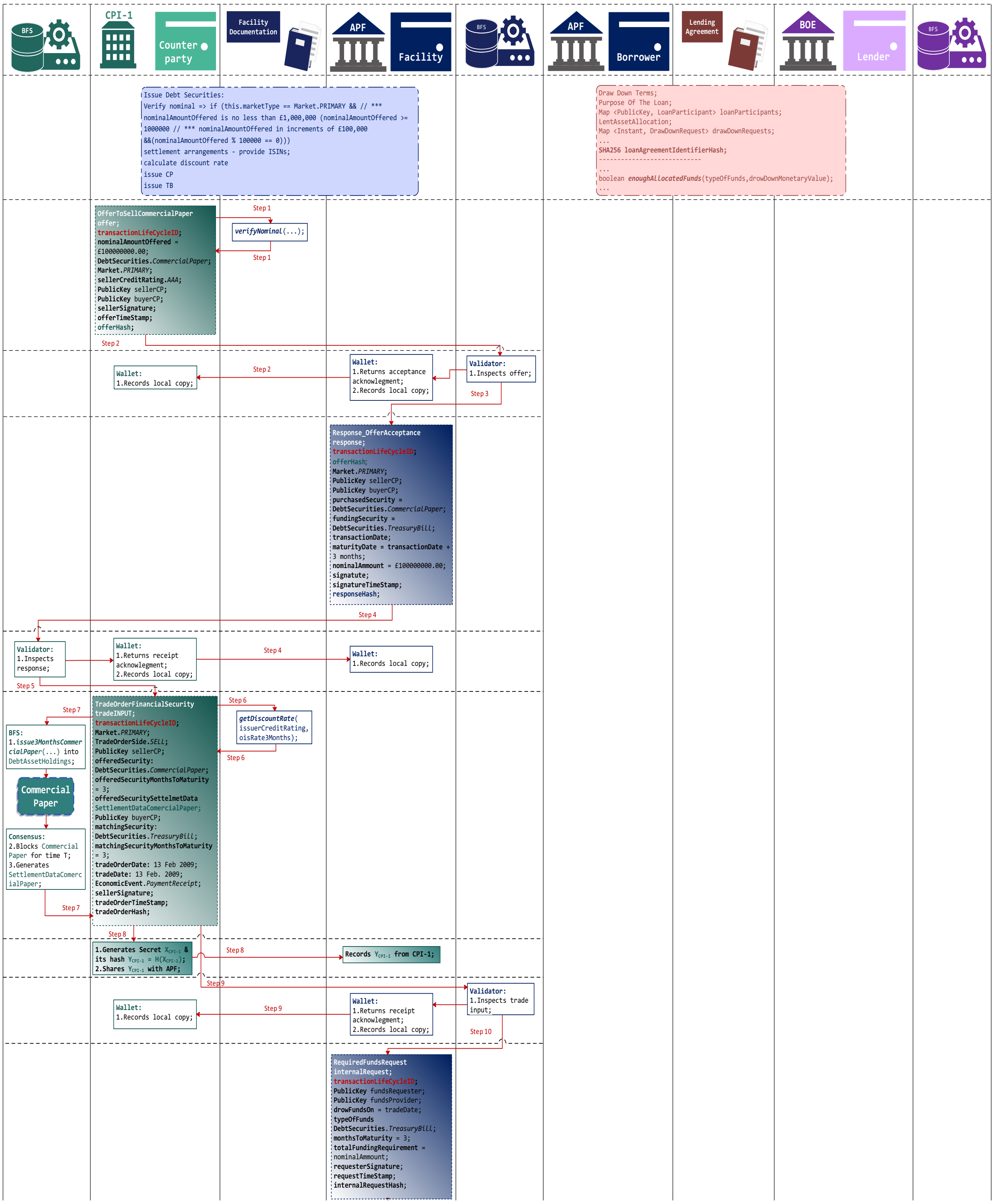


Figure 7.4: Top part of the sequential flow of the atomic 4-party QE business transaction (DvD with HTLC).

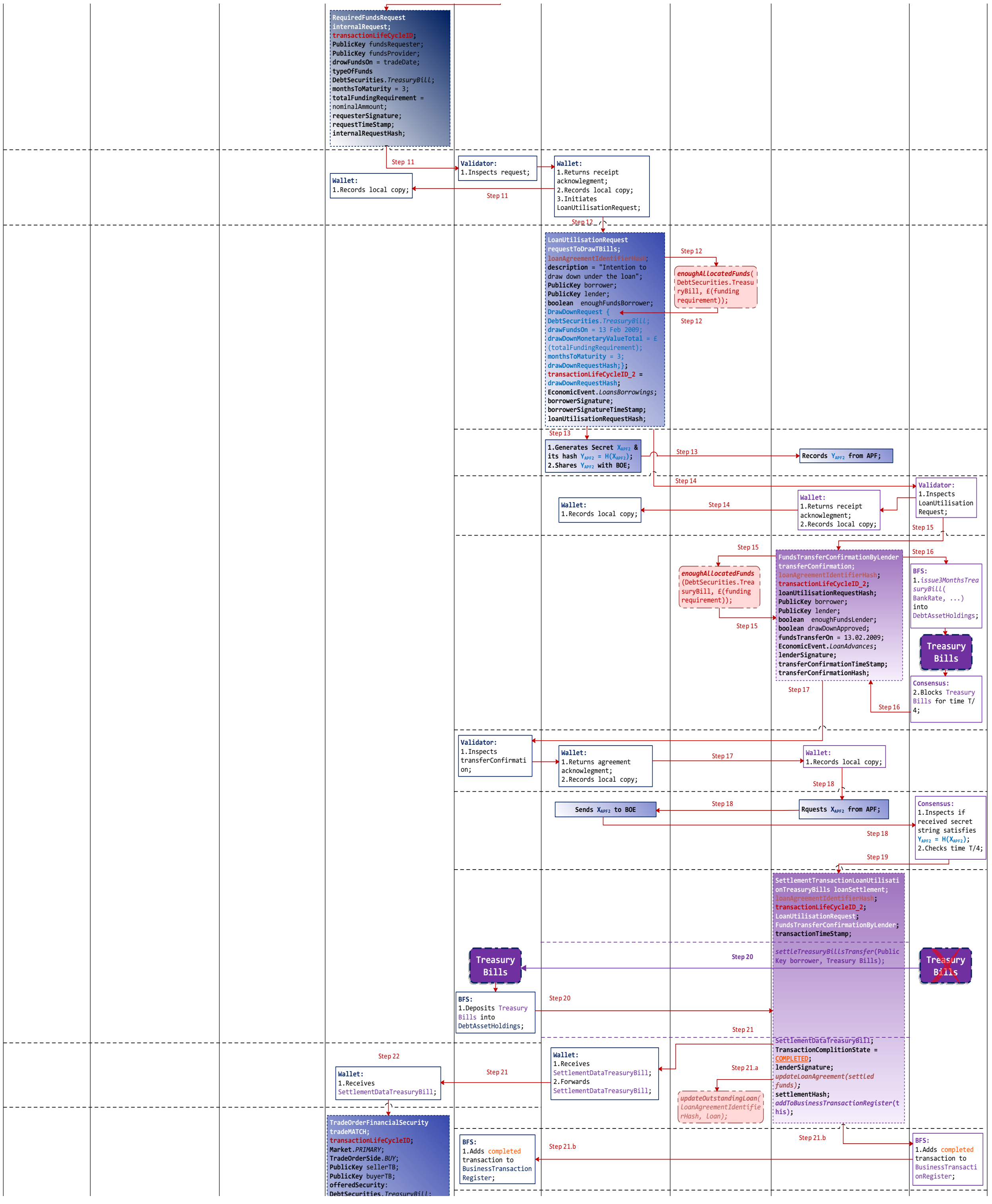


Figure 7.5: Middle part of the sequential flow of the atomic 4-party QE business transaction (DvD with HTLC).

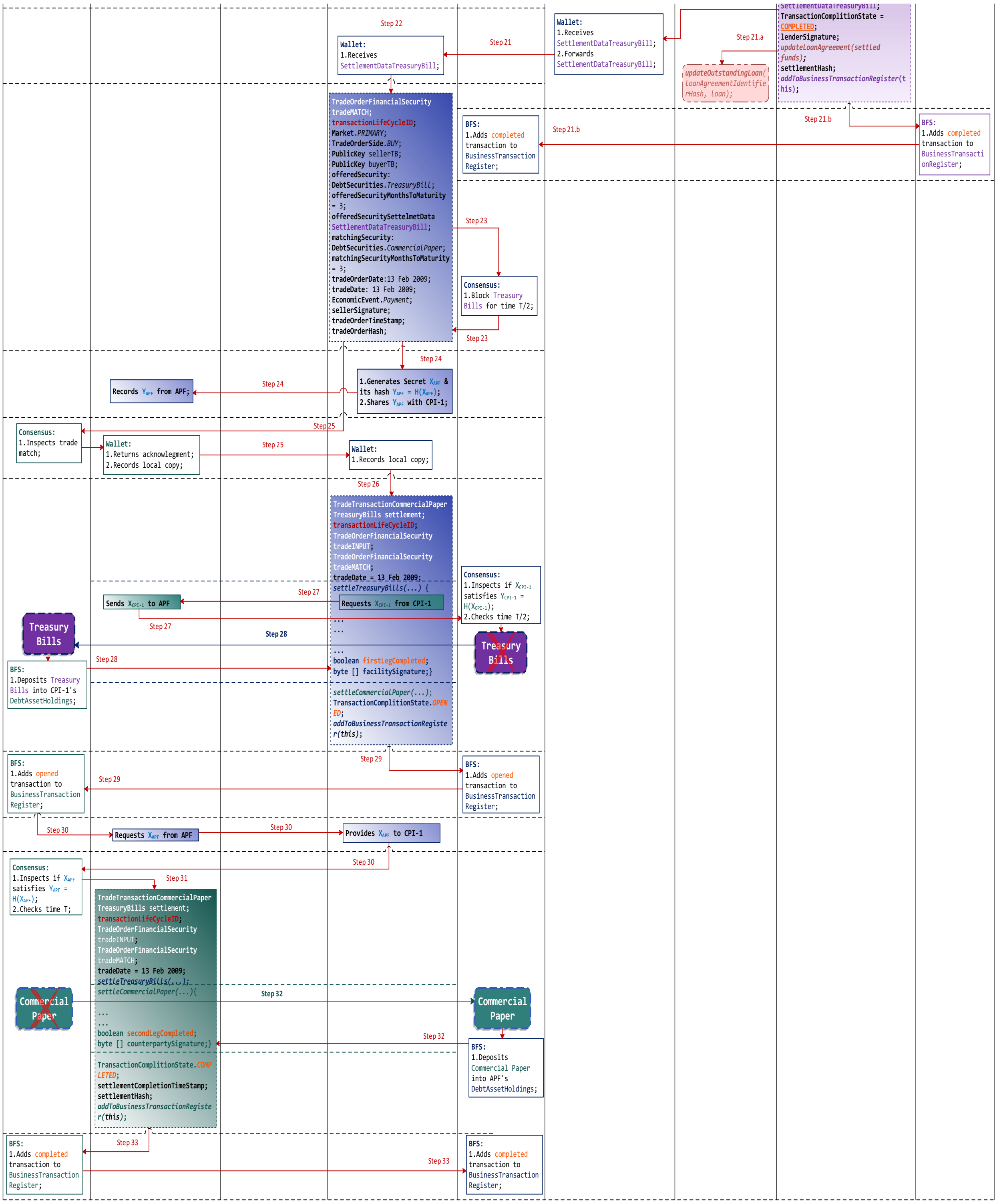


Figure 7.6: Bottom part of the sequential flow of the atomic 4-party QE business transaction (DvD with HTLC).

both counterparties, specifically:

1st - a "trade input" from the seller (CPI-1), and

2nd - a "trade match" from the buyer (APF).

Although, in accordance with the trade input and trade match orders must be completed on the day, before the trade date; this project illustrates the same day implementation for trade input, trade match and settlement. Input and match trade orders will become an integral part of the settlement of our DvD transaction. As part of this arrangements, the CPI-1 `CounterpartyWallet issuerOfCP` initiates `TradeOrderFinancialSecurity tradeINPUT`. In this trade order, the CPI-1 specifies:

(a) the `transactionLifeCycleID` from the `response` transaction proposal.
(b) the type of the market: `Market.PRIMARY`;
(c) the order side: `TradeOrderSide.SELL`;
(d) the `PublicKeys` of the counterparties;
(e) input date and date of the trade: 13 Feb 2009;
(f) what security is on offer: `CommercialPaperUnsecured`;
(g) settlement data summarising non-sensitive information about offered security: `offeredSecuritySettelmetData SettlementDataComercialPaper;`;
(h) what matching security is agreed upon: `matchingSecurity: DebtSecurities.TreasuryBill`;
(i) their respective maturities: 3 months from the `tradeDate`;
(j) Furthermore, CPI-1's wallet adds `EconomicEvent.PaymetReceipt` to this trade input order. Upon successful completion of the whole transaction this variable will be used by the `Accountant` to map the economic data, produced by this business transaction to the corresponding smart contract of the Accounting Cycle.

6. As part of this process, to price the issue of agreed `CommercialPaperUnsecured` token, in accordance with, CPI-1's wallet estimates the "discount rate" , based on the"spread above the risk-free rate" (for the February of 2009 - the use-case):
   (a) credit rating of the issuer (for CreditRating.AAA it is 75bp), and
   (b) maturity matched overnight index swap (OIS) rate (`oisRate3Months`). This research assumes this rate to be 1.53%

   The discount rate is obtained by querying `getDiscountRate(issuerCreditRating, oisRate3Month);` functionality of 2.
7. Next, as part of the trade input order, on the trade date, the CPI-1 `CounterpartyWallet issuerOfCP` is able to initiate the issue of the agreed upon `CommercialPaperUnsecured` token. This operation is performed by the CPI-1 `BFS` in its `DebtAssetHoldings` repository utilising `issue3MonthsCommercialPaper(...)` functionality. Using the model of the HTLC, this token is subsequently blocked in the repository for the time `T`. In Accordance with the HTLC, `SecretHash` for which is generated in the next Step 8, this `CommercialPaperUnsecured` token is blocked in the `DebtAssetHoldings` repository either:
   (a) until successful transfer of the `CommercialPaperUnsecured` token to the APF, if this transfer is performed within time `T` and if during transfer the APF's `FacilityWallet commercialPaperFACILITY` provides a `Secret String` that satisfies `SecretHash` shared by CPI-1, or
   (b) if time `T` expires, the `CommercialPaperUnsecured` token stored indefinitely (maturity limited) in the CPI-1's `DebtAssetHoldings` repository, i.e., the receiver of the token is CPI-1. From this point the `CommercialPaperUnsecured` token is available for onwards transactions to the CPI-1.

   Lastly, `SettlementDataCommercialPaper` is generated and added to the `tradeINPUT` part of this business transaction. This is to enable for pre-settlement provision of the ISIN(s) of the securities as per requirements of the (Bank of England (2010*c*)). Finally, this `TradeOrderFinancialSecurity tradeINPUT` is signed by the CPI-1 `CounterpartyWallet`, timestamped and hashed.

8. Before sending this "trade input order" to the counterparty (to the APF), CPI-1 generates a `Secret String` X_{CPI-1} and its `SecretHash` that contains $Y_{CPI-1} = \mathrm{H}(X_{CPI-1})$. The CPI-1's `CounterpartyWallet issuerOfCP` shares Y_{CPI-1} with the APF and stores this `Secret String` X_{CPI-1} and its `SecretHash` locally in the `Wallet::SecretStore ::HashMap <String, SecretHash> mySecretHashsHashStore`.
9. The following procedure is to send this `TradeOrderFinancialSecurity tradeINPUT` to the APF's `FacilityWallet commercialPaperFACILITY`. APF's `Validator` inspects and verifies the content and integrity of the received trade input and, upon approval, the `FacilityWallet` returns acknowledgement back to the CPI-1. Both counterparties record a local copy of this trade order.
10. In order for the APF's `commercialPaperFACILITY` wallet to fulfil its matching obligations that were pre-agreed earlier in the `response` (step 3) and to proceed to the settlement of this business transaction, matching security is required - the corresponding 3-months `TreasuryBill` token.
    Here, the following hypothetical assumptions are made:
    1st - the APF does not hold `TreasuryBills` on its books in advance. The `TreasuryBill` token is issued upon request of APF on the trade date (i.e., the trade date of the DvD transaction);
    2nd - issue of the `TreasuryBill` token is performed by the BoE not HMT. This assumption is taken for the purpose of the simplification of the illustrative transaction flow.
    To obtain required funding to cover its outstanding matching liability, the APF relies on the earlier established `LoanAgreement`, outlined in Section 6.3.2 between the APF's `BorrowerWallet apfAsSubsidiaryBorrower` and the BoE's `LenderWallet boeTreasuryLender`. According to this agreement, required funds, categorised as `TreasuryBills`, can be drawn from the BE to manage "near-term cash flow requirements" of the APF upon request. As the `FacilityWallet` is not a party to the `LoanAgreement` for the funding provision from the BoE, in order to get the required funds, the `commercialPaperFACILITY` wallet requests another, known to it APF member - the `PublicKey facilityFunder` = `BorrowerWallet apfAsSubsidiaryBorrower` who facilitates the provision of required liquidity, thus enabling the fulfilment of the matching obligations. To achieve this, the APF's `commercialPaperFACILITY` wallet initiates `RequiredFundsRequest internalRequest`. In this `RequiredFundsRequest internalRequest`, the `FacilityWAllet` specifies:
    (a) the member's `Wallet`'s `PublicKey`;
    (b) the trade date for the liquidity to be delivered by (13th Feb. 2009);
    (c) the type of required funds (`TreasuryBill`);
    (d) its maturity (trade date + 3 months), and
    (e) the nominal value of the obligation to fulfil (£100000000.00).
    (f) The request is signed, dated and hashed by the APF's `commercialPaperFACILITY`.
    When this `internalRequest` will be then sent to the APF's `BorrowerWallet`, it will be used to initiate and to settle the `DrawDownRequest` (see 9) on the `LoanAgreement` with the BoE. This loan draw down transaction will be implemented concurrently (on the same trade date) as the current ongoing DvD transaction, i.e., it is an intermediate part of the DvD transaction. Additionally, the successful atomic settlement of the DvD transaction relies on the successful atomic settlement of the loan draw-down. However, the settlement of loan draw-down precedes settlement of DvD transaction.
    During this process, the BoE's `LenderWallet` will be notified of the nominal value and the maturity for the issue of the `TreasuryBill` token. The token is issued and settled between the APF and the BoE using a "one-way-payment" architecture outlined below in the Step 12. As soon as this part of the transaction is settled (Step 12) and the `TreasuryBill` token is transferred to the APF's `DebtAssetsHoldings`, the `FacilityWallet` will be able

to commence the `TradeOrderFinancialSecurity tradeMATCH` - the "trade match" order that is outlined further in Step 22 and to facilitate conclusion of the settlement of "Initial purchases of private sector assets" market transaction.

11. Consequently, the finalised `RequiredFundsRequest internalRequest` is sent from the APF's `commercialPaperFACILITY` wallet to the APF's member - the `BorrowerWallet apfAsSubsidiaryBorrower`. The APF's `Validator` inspects and validates request's data. The `BorrowerWallet` returns acknowledgement and a local copy of this `internalRequest` is recorded by both member `Wallets`.
12. The APF's member - the `BorrowerWallet apfAsSubsidiaryBorrower` uses received `RequiredFundsRequest internalRequest` to instantiate `LoanUtilisationRequest requestToDrawTBills` to obtain the `TreasuryBil` token from the BoE. This `TreasuryBil` token will be:
    - 1st - deposited into the APF's `DebtAssetHoldings` utilising the loan draw-down procedure outlined below and in Section 7.4 and
    - 2nd - used by the APF's `FacilityWallet commercialPaperFACILITY` to fulfil its matching obligations to the CPI-1 `CounterpartyWallet issuerOfCP`.

    This process will enable the APF to swap illiquid Commercial Paper issued by private sector for highly liquid Treasury Bills borrowed from the BoE.

    Outlined below are the steps involved in the draw-down process as specified in the `LoanAgreement` outlined in the Section 7.4. The loan draw-down transaction is implemented using"one-way-payment" settlement procedure architecture, where funds are only transferred from one party to another.To commence the process of provision of requested liquidity by obtaining the `TreasuryBill` token from BOE, the APF's `BorrowerWallet apfAsSubsidiaryBorrower` initiates `LoanUtilisationRequest requestToDrawTBills` - the 1st transaction proposal of the loan draw-down transaction.

    To proceed, the latest version of the `LoanAgreement` is identified by its `loanAgreementIdentifierHash` and retrieved from `BorrowerWallet`'s local copy of the `LoanAgreementRegister` repository.

    As part of the `LoanUtilisationRequest`, the `BorrowerWallet` includes:
    (a) the `loanAgreementIdentifierHash`;
    (b) transaction description = "Intention to draw down under the loan";
    (c) `PublicKeys` of the `BorrowerWallet` and the `LenderWallet`;
    (d) As part of this request, the APF's `BorrowerWallet` employs `LoanAgreement`'s functionality of `enoughAllocatedFunds(...)` to confirm locally that requested `drawDownMonetaryValue` is available, by setting flag `enoughFundsBorrower = true`.
    (e) Following this, `DrawDownRequest` is instantiated, where `BorrowerWallet`, amongst others, specifies:
        i. the type of funds requested to draw (`DebtSecurities.TreasuryBill`);
        ii. requested draw-down monetary value (i.e., the nominal value: `drawDownMonetaryValueTotal` = £ 1000000.00);
        iii. months to maturity of the expected security (`monthsToMaturity` = 3);
        iv. date of the anticipated settlement (`drawFundsOn` = `internalRequest.getTradeDate()` = 13 Feb 2009);
        v. generates draw-down hash `drawDownRequestHash`. This hash becomes the `transactionLifeCycleID` for the whole loan draw-down transaction.
    (f) inserts `transactionLifeCycleID` = `drawDownRequest.getDrawDownRequestHash()`;
    (g) adds `EconomicEvent.LoansBorrowings`. Upon successful completion of this business transaction, on demand, or at the end of the "accounting period", the `Accountant` of the APF will utilise this `EconomicEvent`, to map this transaction to the corresponding `AccountingTransaction` smart contract.
    (h) Finally, the `BorrowerWallet` signs, timestamps and hashes completed transaction pro-

posal.

13. Before submitting LoanUtilisationRequest to the BoE's LenderWallet boeTreasuryLender, the apfAsSubsidiaryBorrower wallet generates Secret String X_{APF2} and its SecretHash that contains Y_{APF2} = H(X_{APF2},). The apfAsSubsidiaryBorrower wallet shares Y_{APF2} with the BoE and stores this Secret String X_{APF2} and its SecretHash locally in the Wallet::SecretStore ::HashMap <String, SecretHash> mySecretHashsHashStore.
14. The BorrowerWallet then sends the completed LoanUtilisationRequest to the BoE's LenderWallet boeTreasuryLender, whose Validator inspects its data integrity and verifies its content. Upon confirmation of acceptance from the boeTreasuryLender, both parties save a copy of it locally.
15. Upon its affirmative response from BOE's Validator, the BOE's LenderWallet proceeds to the generation of the FundsTransferConfirmationByLender transferConfirmation - the 2nd transaction proposal. As part of this confirmation, the BOE's LenderWallet employs LoanAgreement's functionality of enoughAllocatedFunds(...) (outlined in 6.3.2-6.3.2) to confirm locally that requested drawDownMonetaryValue is still within the remaining balance of the LentAssetAllocation of the lending facility. If approved, the LenderWallet completes transferConfirmation, where it includes:
    (a) loanAgreementIdentifierHash;
    (b) transactionLifeCycleID_2;
    (c) loanUtilisationRequestHash;
    (d) PublicKeys of the borrower and the lender;
    (e) boolean enoughFundsLender = true is assigned;
    (f) boolean drawDownApproved = true is assigned;
    (g) fundsTransferOn = 13.02.2009;
    (h) EconomicEvent.LoanAdvances. Upon successful completion of this business transaction, on demand, or at the end of the accounting period, the Accountant of the BoE will use this EconomicEvent, to map this transaction to the corresponding AccountingTransaction smart contract.
    (i) lenderSignature;
    (j) transferConfirmationTimeStamp;
    (k) transferConfirmatinHash;
16. As this process is happening on the trade date of the DvD transaction (13th of February 2009), the BFS of the BoE issues the requested TreasuryBill token into its DebtAssetHoldings repository.Utilising the architecture of the HTLC, this token is subsequently blocked in the repository for the time T/4. This TreasuryBill token is blocked in the DebtAssetHoldings repository either:
    (a) until successful transfer of the TreasuryBill token to the APF, if this transfer is performed within time T/4, and if during transfer the APF's FacilityWallet commercialPaperFACILITY provides Secret String X_{APF2} that satisfies SecretHash shared by APF, or
    (b) if time T/4 expires, the TreasuryBill token stored indefinitely (maturity limited) in the BoE's DebtAssetHoldings repository, i.e., the receiver of this token is the BoE. From this point the TreasuryBill token is available for onwards transactions to the BoE.
17. FundsTransferConfirmationByLender (the 2nd transaction proposal) is sent to the APF's Validator for inspection and upon affirmation, the APF's BorrowerWallet returns acknowledgement and both parties record a copy of this step locally.
18. At this point, the loan draw-down business transaction is ready for settlement. The settlement is initiated by the BoE's LenderWallet by requesting the Secret String X_{APF2} from the APF. If APF FacilityWallet commercialPaperFACILITY provides Secret String X_{APF2} that satisfies SecretHash Y_{APF2} = H(X_{APF2},) shared by APF earlier in the Step 13 and, if the time since blocking of TreasuryBill by the BoE is still within

T/4, the LenderWallet proceeds to instantiation of SettlementTransactionLoanUtilisationTreasuryBills loanSettlement business transaction.

19. The LenderWallet adds following data to the loanSettlement:
    (a) loanAgreementIdentifierHash;
    (b) transactionLifeCycleID_2;
    (c) both transaction proposals are recorded for future reference (LoanUtilisationRequest together with FundsTransferConfirmationByLender);
    (d) the transactionTimeStamp of the initiation of this settlement transaction;
20. During this process, relocation of blocked TreasuryBill token is performed from the BoE to the APF by calling on settleTreasuryBillsTransfer(PublicKey borrower, Treasury Bills) functionality of this transaction. Settlement is achieved by migration of the token from the BoE's DebtAssetHoldings repository into APF's DebtAssetHoldings repository. Upon confirmation from the APF of successful receipt, the BoE permanently removes this token from its ownership.
21. At the point of successful transfer confirmation from the APF, SettlementDataTreasuryBills is also communicated to the APF's BorrowerWallet. This data is required by the APF's FacilityWallet as a pre requisite of the trade match order (provision of ISINs of the securities. This SettlementDataTreasuryBills is therefore will be passed internally from the APF's BorrowerWallet to the APF's FacilityWallet who is then able to proceed with DvD transaction completion, outlined below from Step 22. From this point, completion of the loan draw-down transaction and continuity of the DvD transaction progress concurrently. To finalise the loan draw-down transaction, the SettlementDataTreasuryBill is recorded as part of the loanSettlement transaction data; the business transaction state is marked as completed and the transaction is signed by lender.
    The final steps for the completion of loan draw-down transaction are as follows:
    (a) After successful loanSettlement, the LoanAgreement is updated to reflect outstanding available funding facility and new version of the LoanAgreement is communicated between participants;
    (b) The completed loanSettlement business transaction is added to the BusinessTransactionRegister of each of the counterparties. At the end of the accounting cycle, or on demand, economic data from explicitly completed transactions from the BussinessTransactionRegister will be propagated through the relevant stages of the accounting cycle.
22. Upon receipt of the SettlementDataTreasuryBill from APF's member (the BorrowerWallet) by the the APF's FacilityWallet commercialPaperFACILITY, the FacilityWallet instantiates TradeOrderFinancialSecurity tradeMATCH part of the DvD business transaction. To match earlier input trade input, outlined in Step 5, the FacilityWallet specifies:
    (a) the transactionLifeCycleID from the tradeInput (Step 5);
    (b) the type of the market: Market.PRIMARY;
    (c) the order side: TradeOrderSide.BUY;
    (d) the PublicKeys of the counterparties;
    (e) match order date and date of the trade: 13 Feb 2009;
    (f) what security is on offer on this side of the order: offeredSecurity: DebtSecurities.TreasuryBill;
    (g) settlement data summarising non-private (non-sensitive) information about offered security: offeredSecuritySettelmetData SettlementDataTreasuryBill;
    (h) what matching security is agreed on: matchingSecurity: DebtSecurities.CommercialPaper;
    (i) their respective maturities: 3 months from the tradeDate;
    (j) Furthermore, APF's wallet adds EconomicEvent.Payment to this trade input order.

Upon successful completion of the whole transaction this variable will be used by the `Accountant` to map the economic data, produced by this business transaction to the corresponding smart contract of the Accounting Cycle.

23. During this process, utilising architecture of the HTLC, the `FacilityWallet` instructs the APF's consensus to block corresponding amount of `TreasuryBill` token in the `DebtAssetsHoldings` repository of the APF for the time `T/2`. The block is:
   (a) until successful transfer of the `TreasuryBill` token to the CPI-1, `if` this transfer is performed within time `T/2`, `and if` during transfer the CPI-1 `CounterpartyWallet` provides a `Secret String` that satisfies `SecretHash` shared by APF in the next Step 24, or
   (b) if time `T/2` expires, the `TreasuryBill` token stored indefinitely (maturity limited) in the APF's `DebtAssetHoldings` repository, i.e., the receiver of the token is the APF. From this point, the `TreasuryBill` token is available for onwards transactions to the APF.

   Finally, the `FacilityWallet` signers, timestamps and hashes this trade match order.
24. Next, the APF's `commercialPaperFACILITY` wallet generates new `Secret String` X_{APF} and its `SecretHash` $Y_{APF} = \mathrm{H}(X_{APF})$. The `FacilityWallet` shares this `SecretHash` that contains Y_{APF} with the CPI-1, and stores this `Secret String` X_{APF} and its `SecretHash` locally in the `Wallet::SecretStore ::HashMap <String, SecretHash> mySecretHashsHashStore`.
25. Afterwards, finalised `TradeOrderFinancialSecurity tradeMATCH` is sent to the CPI-1 `CounterpartyWallet issuerOfCP` consensus for inspection, verification, agreement and record keeping (by both counterparties).
26. Commencing here, the DvD business transaction is ready for the final settlement, i.e., the atomic swap of the `CommercialPaperUnsecured` token against the `TreasuryBill` token. On the pre-agreed trade date at the trade time, the APF's `commercialPaperFACILITY` wallet initiates `TradeTransactionCommercialPaperTreasuryBills settlement`, where it specifies:
   (a) the `transactionLifeCycleID` from the `tradeMatch` (from Step 22);
   (b) the `TradeOrderFinancialSecurity tradeInput` and the `TradeOrderFinancialSecurity tradeMatch` are added to the settlement data for future referencing;
   (c) the timestamp of the settlement initiation: `tradeDate`: 13 Feb 2009;
   (d) furthermore, this DvD settlement transaction will consist of 2-legs of the atomic assets transfers (implementations of which will be described further):
   1st leg - `settleTreasuryBills(...)` functionality (see Functionality: 5) is employed by the APF's `commercialPaperFACILITY` wallet and performs relocation of `TreasuryBill` token into the ownership of the CPI-1. Only successful completion of this leg allows this DvD transaction to proceed;
   2nd leg - `settleCommercialPaper(...)` functionality (see Functionalitty: 6) is employed by the CPI-1's `issuerOfCP` wallet and performs relocation of `CommercialPaperUnsecured` token into the ownership of the APF. Successful implementation of this method is the 2nd leg of this atomic `settlement` transaction that leads to full completion.
   (e) together with `boolean firstLegCompleted` flag, and `boolean secondLegCompleted` flag. These flags will coordinate both legs of settlement completion for this business transaction.
27. As initiator of the settlement, the APF's `commercialPaperFACILITY` wallet calls `settleTreasuryBills(...)` method. During this process, the wallet requests `Secret String` X_{CPI-1} from the CPI-1's wallet. When provided, the X_{CPI-1} is hashed and verified by the Consensus of the APF against earlier received Y_{CPI-1}, i.e. if $\mathrm{H}(X_{CPI-1}) = Y_{CPI-1}$. If the time interval is still within `T/2`, the APF's `commercialPaperFACILITY` wallet progresses to the settlement of the 1st leg.
28. Upon successful verification, the `FacilityWallet` transfers`settlmentAssetTreasuryBills`

TreasuryBill token to the CPI-1 Wallet and simultaneously shares its Secret String X_{APF} with CPI-1. The token is deposited into the CPI-1's DebtAssetsHoldings repository, with acknowledgement communicated to the APF. During this process:
   (a) the TreasuryBIll token is permanently removed from the ownership of APF;
   (b) the boolean firstLegCompleted flag is set to true;
   (c) the FacilityWallet signs transaction.

29. APF's commercialPaperFACILITY wallet sets TransactionComplitionState.OPENED and this not completed settlement business transaction is added to the BusinessTransactionRegisters of both counterparties.
30. As the counterparty to this settlement, the CPI-1's issuerOfCP wallet is automatically notified by its Consensus of the opened transaction that is retrieved from CPI-1's BusinessTransactionRegister into the issuerOfCP wallet. Secret String is requested from the APF and is inspected against the provided SecretHash Y_{APF} .
31. If Consensus determines that X_{APF} satisfies $Y_{APF} = \mathrm{H}(X_{APF})$ and the time is still within T, CPI-1 proceeds to the 2nd leg of the settlement business transaction.
32. CPI-1's issuerOfCP wallet calls settleCommercialPaper(...) method. During this process, the CPI-1 transfers
    settlmentAssetCommercialPaperUnsecured
    CommercialPaperUnsecured token to the APF's FacilityWallet. This token is deposited into the APF's DebtAssetsHoldings repository, with acknowledgement sent back to the CPI-1. During this process:
    (a) the CommercialPaperUnsecured token is permanently removed from the ownership of CPI-1;
    (b) the secondLegCompleted flag is set to true;
    (c) the CounterpartyWallet signs transaction;
    (d) the CounterpartyWallet sets TransactionComplitionState.COMPLETED;
    (e) finally, dates completion of settlement, and hashes the whole transition.
33. At this point, both legs of the settlement are concluded and the completed transaction is added to the BusinessTransactionRegisters of all counterparties. From this point forward, at the end of the accounting period or on demand, economic data, produced by this transaction is available to be transformed and propagated though the stages of the accounting cycle.

The Chapter 7 of the Thesis demonstrated another novel capability of the BFS, utilised for automation of a complex logic for its *business transaction smart contract*; it was implemented to execute *Atomic Four-Party QE with HTLC* and its sequential process flow, following a transactional landscape within the BFS architecture.

The direct purchase of Commercial Paper by the APF, facilitated through the strategic issuance and transfer of Treasury Bills, illustrates advanced orchestration for public liquidity provision mechanisms. This, paired with the multi-leg atomic settlement process, underscores the system's ability to navigate and manage financial interactions that involve multiple stakeholders with varying business objectives and roles. Through granular examination of these transactional steps, this research revealed the BFS's novel application of smart contracts. The detailed mapping of transactional components against past and existing APF's operational protocols and settlement procedures reaffirms the BFS's commitment to innovating within established financial frameworks, offering a window into the future of economic transactions powered by blockchain technology. Such innovation can enhance provision of public liquidity and contribute to a more resilient and responsive economic infrastructure. As report advances to the next Chapter, the focus shifts from the transactional interactions to the broader implications and contributions of these smart contract mechanisms.

By articulating these contributions, the next Chapter aims to bridge the gap between the detailed transactional understanding presented in previous Chapters and the broader research and practical implications of such technological advancements. This will underscore the BFS's role in driving forward future advancements within financial technology and illustrate the potential of smart contracts to reshape the future of financial transactions, accounting and management of liquidity.

Chapter 8

BFS Validation

Chapter 8 describes evaluation of the BFS, employing a rigorous validation process of DSRM to assess its efficiency and practicality against established benchmarks. This critical examination aims to ensure that the BFS not only adheres to its conceptual framework but also effectively addresses the complexities encountered in real-world applications. The validation methodology encompasses a comprehensive review of the BFS within a simulated environment, closely replicating specifics of the operational conditions it is designed to navigate. This approach allows for an in-depth analysis of the system's interactions within its intended context, evaluating its functionality and utility through the lens of stakeholder expectations and requirements.

The framework for validation is structured around several core components, including the alignment with design theory principles, the interaction dynamics with the problem context, the established validation criteria, and the employed validation methods. Following this structured approach, the process transitions to analytics of the execution of the BFS validation. This stage is important, as it not only assesses the results for compliance with theoretical expectations, but also engages in a discussion on the implications and insights derived from the validation exercise. Through this systematic evaluation, Chapter 8 aims to gauge the BFS's capability to innovate traditional financial reporting and liquidity management practices, marking a step forward in the integration of blockchain technology within the domain of accounting.

8.1 Introduction

This study is motivated by the potential of blockchain technology for addressing malicious manipulations of accounting data. It focuses on the BFS artefact within a specific real-world scenario, analysing its components and interactions. This Section focuses on validating the BFS artefact against stakeholder goals, requirements and research questions. The design and development journey of the technological solution, such as BFS artefact, involves multiple iteration of the design and implementation, ensuring that the final prototype meets these requirements and satisfies stakeholder expectations (Wieringa (2014)). Application of design theory of Wieringa (2014), plays a key role here, informing understanding of the BFS artefact's attributes and its interactions within its intended environmental context.This theoretical basis will enable prediction of the artefact's potential impact when implemented and evaluated in the real-world.

The purpose of *validation* in the context of design science research is to critically assess the effectiveness and applicability of the BFS artefact against a set of predefined criteria. Validation, as further outlined by Wieringa (2014), precedes the final real-world validation and includes examining the BFS prototype's interactions in its intended simulated environment, modelled by the adaptation of the use-case lifecycle. This process goes beyond mere theoretical speculation, providing a concrete evidence base for the artefact's efficacy and its capacity to meet the stated research objectives. The ambition here is to infer how the artefact would perform within its operational context without the necessity of its actual implementation in a live setting. The validation goal is to confirm its alignment with the theoretical design and to demonstrate its effectiveness in solving real-world problems. This validation reinforces the credibility of the BFS artefact, and underscores the relevance and applicability of design science research in addressing complex issues in the domain of finance and accounting.

8.1.1 Overview of Validation Approach

The validation approach for the Blockchain Financial Statements artefact is structured to ensure a comprehensive evaluation of its functionality, performance and alignment with the predefined stakeholder goals and requirements. This approach encompasses a blend of theoretical analysis and empirical testing within a simulated context, mirroring real-world scenarios to assess the artefact's operational efficacy. In the context of the Blockchain Financial Statements artefact, functional validation is not just essential for the BFS artefact's development and demonstration, but are also are considered as relevant research methodologies for design science research validations, as established by Wieringa (2014). The validation process adopts a pre-implementation evaluation strategy.

The object of this study, the BFS artefact itself, is tested within a simulated context that mirrors the complexities and nuances of its intended operational environment. This validation strategy involves inspecting the potential effects of the BFS prototype's interaction with its environment and mapping these effects against the expectations and requirements of stakeholders outlined in Section 3.2.7. This validation process is predominantly conducted within a controlled experimental lab environment, where the BFS artefact undergoes exposure to scenarios, reflecting the real-world business and accounting transactions of the APF use-case. These interactions are intentionally perspective to assesS the artefact's response and operational behaviour.

Through this modelling and simulation, this work illustrates simulation of the complex interaction of business transactions and their transformation into economic data, culminating in the generation of tamper-evident outcomes. The simulated environment for research validation serves as a crucial platform for demonstrating the BFS artefact's ability in enforcing complex transactional rules and transforming transactional data into verifiable economic outputs.

By adopting modelling and simulation of Wieringa (2014) as primary research methodology, thorough examination of the BFS artefact's anticipated behaviour in a "single case" research experiment context was necessary. This simulated exploration methodology of Wieringa (2014) allows for an empirical observation of the BFS artefact's response. Examination of the BFS artefact's anticipated behaviour in a "single case" research experiment context provides necessary insights into its potential performance and impact. Furthermore, guided by a curiosity-driven research ethos, this study aims to explore blockchain technology's ability to address challenges such as the malicious manipulation of accounting data. The BFS artefact is analysed within this simulated context of the transactional interactions. This approach validates the artefact's functional capabilities and demonstrates mechanisms that govern the associations between components within the specified use-case scenario.

8.2 Validation Framework

The foundation of the BFS artefact is based on a comprehensive design theory that addresses the artefact's structural properties, functional capabilities and its interaction within a specified problem context (Wieringa (2014)). This theoretical framework serves as a foundation for establishing a clear understanding of how the BFS artefact is expected to operate, interact with its environment and achieve the designated objectives.

Design Theory Principles

The design theory for the BFS encapsulates an innovative integration of blockchain technology into traditional accounting and financial reporting processes. It is based on the belief that blockchain's inherent features, such as decentralised ledger technology, cryptographic security and consensus mechanisms—can significantly enhance the reliability, transparency and efficiency of financial data management. This theoretical foundation proposes that by automating accounting transactions and securely recording financial data on a blockchain, entities can achieve greater assurance in the integrity of their financial statements and a reduction in the risk of fraudulent activities. Central to the BFS artefact's design theory are its key properties, which include:

- ❖ **Tamper-Evident Record-Keeping and Security** - utilizing blockchain to ensure that once financial data is recorded, it cannot be altered without detection, thus safeguarding the integrity of financial records. This immutability, coupled with robust cryptographic security measures, plays a vital role in fraud prevention and data integrity.
- ❖ **Automated Transactional and Accounting Processes** -by leveraging smart contracts to automate the execution of business and accounting transactions, reducing manual errors and increasing process efficiency and importantly enforce predefined rules and obligations to comply with.
- ❖ **Real-Time Audibility** - facilitating near-instantaneous verification of financial data, allowing stakeholders to access up-to-date financial information, facilitating trust and enhancing decision-making processes.
- ❖ **Decentralisation and Transparency** - central to the BFS artefact's design is the principle of decentralisation, ensuring that financial data is not controlled by a single entity but is instead distributed across a network of members, enhancing transparency and reducing single points of failure or malicious manipulation.

Interactions with the Problem Context

The BFS artefact is designed to interact with the economic ecosystem by providing a platform for secure and transparent financial transactions between entities. This interaction aims to address challenges of accounting fraud and data manipulation by ensuring that all financial activities are accurately and reliably recorded. Furthermore, the artefact's ability to generate and publish financial statements in real-time or on demand, addresses the gap in timely financial reporting, enabling more informed decision-making by stakeholders. Guided by this design theory, several predictions about the BFS artefact's performance and impact can be made:

Firstly, the work anticipates that the BFS artefact will reduce the incidence of accounting fraud and errors, as the blockchain's tamper-evident nature makes unauthorised alterations easily detectable. Secondly, the real-time or near-real-time recording of economic impact of the transactional interactions on the blockchain data structure predicted to increase transparency of financial information, allowing stakeholders instantaneous or on-demand access to verifiable financial data.

Thirdly, by providing a transparent and secure platform for financial reporting, the BFS is expected to foster greater overall trust among consumers, business owners, regulators, and other stakeholders in the accuracy and reliability of financial information. Fourthly, the automation of accounting processes through smart contracts is predicted to streamline accounting process, improve operational efficiency reduce processing times, streamline financial reporting workflows, reduce manual errors, expedite the generation of financial statements, and lower operational costs for businesses. Finally, the BFS is anticipated to simplify compliance with financial regulations and facilitate more straightforward auditing processes. The clear, immutable record of business activity enables easy verification by regulators and auditors, potentially transforming the regulatory landscape.

The validation framework involves testing these predictions through simulated scenarios and functional evaluation, aimed at assessing the artefact's efficacy in enhancing the accounting process and mitigating fraud within the financial reporting ecosystem. The aim of the validation framework is to demonstrate theoretical benefits of integrating blockchain technology into accounting practices, thereby contributing to the advancement of knowledge in the field and providing practical solutions.

Validation Criteria

The validation of BFS is structured around a set of clearly defined criteria that align with the research objectives (see Section 1.2), the anticipated benefits for stakeholders (see Section 3.2.7) and the specific requirements of the artefact itself (see Section 3.2.7). These criteria are essential for a systematic validation of BFS performance, its potential impact on financial reporting practices and its efficacy in mitigating challenges associated with accounting fraud and data manipulation. The primary research objectives guide the development of validation criteria, focusing on BFS's ability to mitigate accounting fraud, enhance the transparency and reliability of financial reporting and improve the efficiency of accounting processes through blockchain technology. The criteria considers how well the BFS artefact integrates blockchain to achieve these objectives, by evaluating the BFS:

- ❖ effectiveness of the BFS in reducing instances of accounting fraud through tamper-evident record-keeping and secure transaction processing;
- ❖ capacity to offer a transparent view of financial transactions and ensure the reliability of financial data presented in the financial statements;
- ❖ efficiency gains in accounting processes attributed to the automation capabilities of smart contracts and the streamlined approach to financial reporting.

The detailed functional requirements outlined for the BFS artefact are also important in forming the validation criteria. These requirements are closely examined to ensure the artefact matches predefined functional requirements, effectively executing the intended operations and delivering the expected outcomes. This involves applying the artefact in simulated scenarios reflective of real-world financial transactions and reporting processes, thereby assessing its performance and determining its efficacy in enhancing financial reporting practices. Each of these criteria is measured for the validation aims to confirm that the BFS meets the theoretical expectations and provides practical benefits to the stakeholders, ultimately contributing to the advancement of blockchain applications in financial reporting and accounting practices.

Validation Methods

The validation of BFS employs a combination of methods essential in demonstrating effectiveness of the designed system. These methods include modelling, simulation and laboratory testing. Each method plays a crucial role in this comprehensive validation strategy and was selected for its ability to provide insights into the BFS's functionality within controlled conditions that mimic the complexity of real-world operations.

Modelling serves as the initial step in validation and as a foundational method. By creating abstract representations of BFS artefact's operational environment and its interactions, enables exploration of the BFS's conceptual design and its interactions within this simulated business environment. It helps in detailing the system's architecture and the logical flow of data and processes. Modelling is selected due to its strength in providing a visual and logical representation of the artefact, facilitating an understanding of its complexity and it is instrumental in identifying potential flaws or improvements early in the design phase.

On the other hand, simulation builds upon the developed model, adding layers of complexity and dynamism that bring us closer to real-world conditions. It involves creating scenarios that mimic real-world business transaction and accounting practices, observing the artefact's responses to inputs and conditions. By simulating the behaviour of BFS, particularly in how it processes transactions and generates financial statements, it is possible to observe its performance and identify any discrepancies from expected outcomes. Simulation is an invaluable tool for validating the BFS artefact, because the scenario can be adjusted and iterated with relative ease, allowing for extensive testing of the artefact's response to a wide range of inputs and situations without the cost and risk associated with actual implementation. It allows us to test hypotheses and answer "what-if" questions in a risk-free environment.

8.3 Execution of Validation

The execution of validation for the BFS artefact involves a detailed process where each component and its interactions are tested to confirm that the artefact meets all defined requirements and performance standards. This critical stage is where the theoretical constructs of the design are rigorously tested and manifested into a tangible, operational system. Initiating with a structured agenda, this phase includes manual validation of the simulation of the real-world conditions derived from the use-case scenarios to test different aspects of BFS performance from functional operability to data accuracy and integrity. This holistic validation approach tests the artefact against practical applications and meticulously emulates the deployment within its intended operational framework. According to guidelines of Wieringa (2014), a validation exercise is experimental and is usually done in the laboratory. In validation, the BFS is exposed to various scenarios - the implementation of business and accounting transactions, functional behaviour for which is set-up by modelling from a real-world scenario derived from the use-case, to see/observe how the artefact responds and behaves.

To observe the artefact's behaviour in real-time, **manual testing** exercises were run in controlled lab environment for examination of insights into interactions within its simulated context. This approach enables real-time feedback for in-depth understanding of how BFS interacts within its simulated context, especially focusing on its ability to execute business and accounting transactions, process and transform economic data into financial reports and securely store these reports on the blockchain. To verify behaviour of the BFS in situations that closely mimic its intended operational environment, **scenario-based testing** was utilised. This step tests the artefact's response to various inputs, workflows and data types, ensuring accuracy and adaptability. This testing is essential in demonstrating the artefact's ability to handle complexities of real-world business transactions and accounting processes. This type of validation - is where the BFS's theoretical functionality confronts practical application. It involves the orchestration of real-world scenarios and the development of test cases that draw from the chosen use-case. These scenarios and tests are designed to emulate the artefact's deployment within its intended environment and assess its response to data processing tasks. The real-world scenarios chosen for validation reflect common interactions within the economic system the BFS aims to serve. These include the inception of business entities, business transactions execution between such entities and the generation and publication of financial statements. By mimicking the complexity of real-world operations, these scenarios provide a platform for evaluating the BFS's capabilities. Test cases were implemented to simulate the end-to-end process of business and accounting transactions. This is included the initial setup of business entities, the generation of transactions and the subsequent automated accounting processes. Through these test cases, the BFS was assessed on its ability to process and integrate transactional data, execute smart contract provisions and produce financial statements reflective of the entity's economic activities.

To validate a ability of the BFS to contribute to stakeholder goals (Wieringa (2014)), multiple **functional correctiveness testing** was implemented. Functional testing is essential for verifying that each function of the artefact behaves as expected. It is needed to assess the artefact's capacity to meet its intended stakeholders goals and functional requirements and to effectively address research questions within its conceptual framework, without the necessity of deploying the artefact in a real-world environment (Wieringa (2014)). This form of testing evaluates each feature of the BFS by providing scenario-based input and observing the output, verifying the BFS's operational behaviour without necessarily understanding its internal workings. The functional testing also verifies the artefact's data processing abilities. It involves executing a series of tests that apply transactional data through the artefact's processing mechanisms, observing the outcomes and comparing them against expected results based on the design specifications and stakeholder requirements. These are observed through the artefact's performance in recording transactions, updating ledgers and reflecting these actions in the financial statements. The accuracy and timeliness of the BFS artefact's output are critical markers of its effectiveness and, as such, are scrutinised against the expected results derived from the test cases.

This overall approach to validation allows for a thorough examination of how the artefact processes data, handles transactions, generates financial reports and achieves its goals of improving financial transparency and integrity, without the need for real-world deployment. This methodical approach ensures a comprehensive validation of the artefact's capabilities, supporting its theoretical foundation and confirming its potential effectiveness in addressing the challenges identified in the research. This phase is crucial as it translates the theoretical design into a verifiable, functioning system.

8.3.1 Description of the Experiments

The validation of the BFS framework is essential for determining its effectiveness and practicality in real-world applications. Following the principles outlined by Wieringa (2014), the validation of the BFS framework was conducted through a series of experiments that were intended to assess the artefact's *functionality*, *performance*, and *compliance* with stakeholder requirements.

These experiments were executed in a simulated, controlled laboratory environment that closely mimics real-world transactional and accounting operations, based on the Asset Purchase Facility use case. The main objective of these experiments was to assess how well the BFS artefact addresses the challenges it was designed to solve.

The validation was structured around the three core components of the BFS: *accounting and business transactions*, *financial reporting*, and *blockchain ledger integration.* The specific goals of these experiments were:

- ❖ To assess the BFS's ability to execute business and accounting *transactions accurately.*
- ❖ To validate the system's *capability to automatically and correctly generate financial reports.*
- ❖ To test the *integrity and immutability of financial records* stored on the blockchain ledger.

The simulated scenarios involved the generation and recording of business transactions, the initiation and run of accounting cycle, and postings of accounting data in the accounting ledgers, the issuance of financial reports, and the publication of these reports on a blockchain ledger. These experiments also included the interaction between different economic entities, such as the APF, the BoE, and the CPI-1, as per the original use case.

These experiments were conducted using Java as the programming language for implementing the BFS's smart contracts, financial transactions, and accounting processes, ensuring the platform's compatibility with blockchain integration.

8.3.2 Definition of Metrics

To quantitatively assess the BFS artefact's performance, a set of validation metrics was established. These metrics provided insight into the system's operational efficiency, accuracy, and ability to meet the requirements of stakeholders. Key metrics for these experiments were as follows:

Transactional accuracy - measures the percentage of correctly processed transactions to the total number of transactions. The purpose of this exercise is to assess the BFS's ability to execute and record business and accounting transactions without errors or inconsistencies. The metric describes the ratio of correct transactions to total transactions processed.

$$\text{Transactional Accuracy} = \frac{\text{Correct Transactions}}{\text{Total Transactions}} \times 100\%$$

Transaction processing time - measures average time taken by the BFS to complete business or accounting transaction from initiation to the final recording in the BFS Filing History ledger. The goal is to evaluate the BFS's efficiency in handling real-time business interactions and financial accounting processes. This metric describes average processing time (measured in milliseconds).

$$\text{Transaction Processing Time} = \frac{\text{End Time of Transaction} - \text{Start Time of Transaction}}{\text{Total Transactions}}$$

Smart contract compliance - measures degree (percentage) to which business and accounting transactions complied with the rules encoded in the BFS smart contracts governing them. The repurpose of this test is to ensure that business and accounting transactions comply with defined procedural requirements. This metric communicates the ratio of transactions executed in line with smart contract rules to the total transactions.

$$\text{Smart Contract Compliance} = \frac{\text{Compliant Transactions}}{\text{Total Transactions}} \times 100\%$$

Data integrity - measures the ability of the BFS to maintain data immutability after it has been stored on the BFS filing history, ensuring no tampering or alterations occur. It provides a ratio of unaltered accounting publications to the total number of records. The metric describes the number of recorded transactions that matched the original transactional data (tamper-evidence) vs. altered records.

$$\text{Data Integrity} = \frac{\text{Untampered Data Entries}}{\text{Total Data Entries}} \times 100\%$$

Financial reporting accuracy - measures the degree of accuracy in the generation of financial reports, evaluated by comparing generated reports to expected financial results. The purpose of this test is to assess the BFS's ability to accurately reflect economic events in financial reports. Metric describes the ratio of correctly generated financial reports to total reports produced.

$$\text{Financial Reporting Accuracy} = \frac{\text{Accurate Reports}}{\text{Total Generated Reports}} \times 100\%$$

Efficiency gains - measures degree to which the BFS automates, and this streamlines, accounting processes compared to traditional systems. This test evaluates the improvement in processing times following the implementation of the BFS artefact. BFS automation of accounting minimises human intervention, reducing the time needed for various accounting tasks. The goal here is to measure how much the BFS increases efficiency through this automation and blockchain integration. Metric communicates comparison of processing times and error rates between the BFS and traditional accounting systems.

$$\text{Efficiency Gain} = \frac{\text{Time Before Implementation} - \text{Time After Implementation}}{\text{Time Before Implementation}} \times 100\%$$

8.3.3 Values Obtained from Validation

The BFS artefact was subjected to a series of tests using the metrics described above. The experiments provided both qualitative insights and quantitative data, allowing for a thorough assessment of the BFS artefact's performance. The results obtained from these validation exercises are summarised below.

Transactional accuracy - 100% of the transactions were processed correctly, adhering to both accounting and business rules embedded in the smart contracts. This outcome validates the

BFS's ability to execute complex financial transactions without discrepancies, ensuring trust in its financial management system. Furthermore, the BFS demonstrated a strong ability to execute business and accounting transactions accurately, which is important in reducing fraud and ensuring compliance.

Transaction processing time - BFS completed transactions with an average processing time of 42 milliseconds.This quick processing time indicates the system's capability to efficiently handle high volumes of transactions, a critical factor for scalability in meeting real-time operational needs.

Smart contract compliance - in running BFS PoC prototype, 100% of transactions adhered to the predefined smart contract rules. This validates the BFS's ability of enforcement of business rules, ensuring that all transactions did not required manual intervention. This outcome demonstrates that all transactions executed within the BFS artefact were procedurally compliant, which is crucial for establishing trust in automated accounting systems.

Data integrity - Out of 2 executed business transactions, followed by 2 accounting transactions, none were found to have been altered after being recorded in the Business Transaction Register and published on the BFS filing history ledger respectively. Additionally 100% of the financial reports published on the BFS filing history remained immutable. This ensures that once financial data is recorded and published on the BFS, it remains unchangeable - a critical requirement for trust and transparency in reporting. This also confirms the BFS's use of blockchain to ensure the integrity of financial data, making it tamper-evident and reinforcing the system's ability to combat fraud.

Financial reporting accuracy - overall 98% of final financial reports generated by the BFS were accurate, with only minor discrepancies detected due to rounding issues in few runs of the exercise, but overall, the BFS demonstrated its ability to accurately transform transactional data into comprehensive financial reports. This consistency validates BFS reliability in automating financial reporting, which is a key objective in reducing manual errors.

Efficiency gains - in the validation exercises, the time before implementation is defined as time taken for traditional accounting processes, which typically involves a range of time-intensive activities, each of which can be impacted by human error and inefficiencies (e.g., data entry, reconciliation, auditing, and financial reporting). According to reports from professional organisations such as the PwC (2021) and ACCA (2020), these activities can consume significant amounts of time, particularly in large organisations. Reconciliation alone can account for approximately 25-30% of the total time spent on accounting tasks, while financial reporting cycles often stretch into weeks depending on the size and complexity of the organisation. data entry can take up to 10-20% of the overall time, depending on transaction volume. Auditing can take 15-20%, as auditors manually review financial records, and financial reporting itself takes 20-25%, with significant delays due to manual compilation and verification of data.
In contrast, BFS This automation leads to near-instantaneous execution of transactions and financial reporting, although the BFS processes are still sequential to ensure accurate and verified completion of each step. In the BFS artefact, the time required for these tasks is reduced to near-zero due to the automated nature of the system. The main time investment occurs during the initial setup, after which the system processes transactions in real-time.
As a result, if traditional financial reporting takes 40 hours, and the BFS artefact reduces this to 5 hours, the BFS can achieve efficiency gains of up to 87.5%, where majority of the time is dedicated to the review and oversight.

8.3.4 Quantitative Discussion of Performance

The quantitative results for the performance metrics obtained during the validation experiments demonstrate BFS's operational efficiency, accuracy, and ability to meet stakeholder goals and research objectives.

The system's ability to achieve 100% transactional accuracy and data integrity, alongside 100%

smart contract compliance, highlights its reliability in maintaining secure and tamper-evident financial records.

The transaction processing time of 42 milliseconds illustrates the BFS's capability to handle transactions efficiently in a real-time environment, which is critical for large-scale financial operations. This processing time, combined with a 87.5% improvement in efficiency over traditional systems, supports the BFS's potential for scalability in broader financial ecosystems, making it a viable alternative to traditional manual accounting, by streamlining financial reporting processes and minimising the risk of human error.

The BFS reduces the manual labour involved in accounting processes by automating key functions such as data entry, reconciliation, and financial reporting. With the initial investment in setting up and validating accounting smart contracts, the system delivers significant efficiency gains once operational. This efficiency improvement is particularly valuable in large organisations, where the volume of transactions and the complexity of financial reporting can significantly slow down the accounting cycle.

In terms of financial reporting accuracy, achieving 98% accuracy suggests that the BFS can significantly reduce human error in the accounting process. This is critical for organisations that rely on timely and accurate financial information to make informed decisions. Validation exercise for the BFS artefact thus presents a compelling case for the adoption of blockchain technology in financial accounting, offering both time savings and enhanced accuracy.

Furthermore, immutability of the BFS Filing History ledger, and metrics for data integrity, further support the BFS's role in preventing fraud and ensuring data security, which are critical objectives of the framework. By demonstrating that the BFS can accurately and securely process transactions and generate financial reports, the artefact has shown that it is capable of meeting its design objectives.

Lastly, the BFS's ability to secure data on the blockchain, ensuring immutability and compliance with smart contracts, further reaffirms its role in enhancing financial transparency and integrity. The combination of these performance metrics validates the BFS's theoretical design and confirms its practical applicability in real-world financial reporting and accounting scenarios.

The results indicate that the BFS artefact performs well under simulated conditions, showing promise for further development and real-world implementation. However, while the BFS demonstrated promising performance during validation, future research should explore its scalability and adaptability in more complex and varied financial environments. The current validation was conducted in a controlled laboratory setting, and further work is needed to assess how the BFS would perform in large-scale, real-world applications involving more diverse stakeholders and financial instruments. Additionally, future work could investigate integrating advanced technologies like artificial intelligence (AI) to further enhance the BFS's ability to identify potential discrepancies and improve efficiency in processing large data volumes.

8.4 Results Analysis and Discussion

This section provides insights into validation results demonstrated by the BFS, particularly focusing on transactional accuracy, compliance with smart contract rules, integrity of transactional data and accuracy in financial reporting. These findings are illustrative in assessing the BFS artefact's potential to transform traditional accounting practices through blockchain technology.

Transactional Accuracy.
The BFS artefact was subjected to a series of scenario-based tests designed to evaluate its ability to accurately execute, process,and record business and accounting transactions. These tests involved simulating use-case data based of scenarios for these transactions including those that required complex data transformations and interactions between different economic entities. The results indicate that the BFS artefact is able to process transactional events with a high degree of accuracy,

effectively capturing, transforming, processing and reflecting the details of each transactional event in the relevant repositories and the blockchain ledger without discrepancies.

Compliance with Smart Contract Rules.
Another critical area of evaluation was the artefact's compliance with predefined smart contract rules governing business and accounting transactions. Through the execution of validation tests, the BFS artefact demonstrated consistent adherence to the functional and procedural requirements of these rules. Smart contracts were executed as intended, automatically enforcing the conditions and obligations they encoded. This compliance ensures that transactions processed by the BFS artefact are in alignment with the established protocols and business logic, further validating the system's reliability.

Integrity of Transactional Data.
The integrity of transactional data is paramount in the BFS, particularly given its aim to mitigate challenges associated with fraudulent accounting practices. The validation process tested the artefact's ability to verify if received transactional messages were unaltered and its ability persist to and retrieve tamper-evident records from blockchain data structure. The findings revealed that once transactions were received, verified, processed and recorded, the data remained immutable. This outcome illustrates the potential of the BFS artefact to enhance transparency and trust in financial reporting by ensuring that transactional data is secure, verifiable and enhances the reliability of the financial and accounting information it generates.

Accuracy in Financial Reporting.
The ultimate test of the BFS artefact's effectiveness was its accuracy in financial reporting. The artefact successfully transformed raw transactional data into comprehensive financial statements with high accuracy. The automated accounting cycle within the BFS ensured that all economic events were correctly classified, recorded and consolidated into financial statements. This automation process significantly reduced the potential for human error, leading to financial reports that accurately reflect the financial health and activities of the entity. Moreover, the BFS artefact's capability to publish these financial statements onto a blockchain data structure – the filing history – secured the reports against unauthorised alterations and made them readily accessible for verification and audit, further enhancing transparency and accountability in financial reporting. The validation outcomes demonstrate BFS's functional accuracy in key areas critical to financial reporting and governance, validating the design choices and implementation strategies. By adopting blockchain technology, the BFS ensures the accuracy and integrity of transactional data and transforms the process of financial reporting, making it more reliable, transparent and efficient. The results illustrate the BFS artefact's potential to contribute to the modernisation of financial reporting and to provide a solid foundation for further research and development in blockchain-based accounting systems. These results guide the iterative refinement of the BFS artefact, ensuring that it continues to evolve in line with stakeholder expectations and technological advancements.

In conclusion, during validation phase of the BFS artefact the insights were synthesise and gained from manual testing, scenario-based testing and functional correctiveness testing. Drawing on Wieringa (2014) this Section emphasises the important role these testing methodologies play in ensuring the BFS artefact's alignment with its design intentions and stakeholder expectations.

Manual Testing, *Scenario-Based Testing*, and *Functional Correctiveness Testing* methods form the core of the validation strategy applied in this research, offering an understanding of the BFS artefact's operational capabilities. Manual testing provided a hands-on approach to identifying immediate issues, scenario-based testing explored the artefact's response to a variety of operational contexts and functional correctiveness testing ensured each feature performed as expected. Collectively, these approaches underscored the artefact's ability to fulfil the requirements set forth at the inception of the design process (see Section 3.2.7). Concerning *effects and requirements satisfaction*, the integration of these testing methodologies has confirmed the BFs's capability to enhance financial reporting accuracy and ensure compliance with smart contract protocols. This validation process demonstrated the artefact's compliance with its primary objectives and its potential to contribute to broader financial transparency and integrity. These results validate the artefact's theoretical design and confirm its practical effectiveness in enhancing the transparency and integrity of financial information management.

Wieringa (2014) advocates consideration of trade-off and sensitivity questions to help clarify the set of the real-world instances of the similar artefacts implemented in the similar contexts (Wieringa (2014)) Addressing these trade-off and sensitivity questions raises important considerations. Should the BFS artefact's architecture undergo modifications, the implications on its operational efficacy and stakeholder goal fulfilment would necessitate thorough evaluation to maintain or enhance its performance attributes. Similarly, altering the artefact's contextual environment prompts reassessment of its adaptability and resilience, ensuring its continued relevance and effectiveness in new or evolving conditions. Changes in regulatory environments, stakeholder needs or technological advancements necessitate an artefact capable of evolving without losing its core functionality and purpose. These outline the potential avenue for the future exploration of leveraging emerging technologies, enhancing system performance and expanding the empirical evidence base to support broader applicability and optimisation of BFS.

Future research could investigate harnessing AI to automate and enhance testing processes for blockchain-based financial systems like the BFS artefact. This includes optimising test case generation, improving anomaly detection and accelerate the development cycle and facilitate more dynamic adaptations to evolving requirements. The integration of AI could enhance the efficiency, coverage and accuracy by automating identification of patterns in data and predicting potential problem areas within the software.Further studies are also encouraged to focus on optimising the BFS artefact's performance, particularly concerning data processing and system scalability. Although the functional correctness of the BFS output is a direct reflection of its innovative architecture, there is room for improvement in its operational efficiency. The real-world applications demand rigorous stress testing to ensure scalability and responsiveness under varied and intensive operational conditions. Future work could focus on optimising the processing algorithms and exploring distributed computing frameworks to meet the demands of larger, more complex financial ecosystems.

The current validation of the BFS artefact has been primarily theoretical and conducted within a simulated environment. The BFS artefact represents a novel approach within its problem domain and its impact has been inferred rather than directly observed. Future work should aim to bridge this gap by implementing the BFS artefact in real-world settings. This approach will allow for an empirical evaluation of its impact on the problem domain of transparency, integrity and timeliness of financial reporting. This evaluation will require collaboration with stakeholders to monitor the artefact's performance in diverse operational environments and to gather feedback for iterative refinement.

While laboratory simulations provide a controlled environment to test hypotheses and predict outcomes, they may not fully capture the complexity and unpredictability of real-world applications. The context of the laboratory simulation may be less similar to the context of the real world application, because in the real world implementation there are various diverse conditions of practice may disturb or unexpectedly influence the results demonstrated in the simulation of the single case scenario (Wieringa (2014)). This simulation ignores uncertainty about inputs that drive and govern the desired outcomes and restricts the artefacts adaptability (Wieringa (2014)). Future research should consider field testing and generalisation to validate the BFS artefact's functionality in diverse operational contexts. Real-world applications may introduce variables and conditions that differ significantly from controlled environments, potentially affecting the artefact's performance. Field tests are warranted to provide support to future generalisation of the BFS architecture and optimisation of its functionality (Wieringa (2014) based on real-world feedback and interactions.

Wieringa (2014) emphasises an important aspect of conducting implementation evaluation, which leverages real-world experiences to refine the BFS design and underlying theoretical framework. Future exploration should consider evaluation research (Wieringa (2014)) notably to explore how the BFS interacts with its context domain in the field, i.e., investigate its reliability and timeliness in providing tamper-evident validity of the monetary claims. This could be achieved by application of experimental or technical action research, implementation of statistical surveys and observational case studies as the research methods, as recommended by Wieringa (2014). This research can offer insights into its practical utility and areas for improvements.

In conclusion, the validation Chapter of the BFS underscores the critical role of testing in the field of design science research. The successful execution of manual, scenario-based and functional correctiveness testing provides foundation for the artefact's future development and deployment, promising significant advancements in the field of financial reporting and blockchain technology. This Chapter reaffirms the necessity of implementation of testing methodologies in bridging the gap between theoretical design and practical functionality, setting a ground for future research and development in design science and blockchain-based financial systems. Overall, the Thesis provides a foundation for future investigations into the potential of blockchain technology to transform financial reporting and management. The future work for the BFS artefact encompasses a multidimensional approach that seeks to harness emerging technologies, optimise performance, validate real-world applicability and expand the empirical evidence base. This forward-looking agenda promises to refine and enhance the BFS artefact, ensuring its relevance and utility in an ever-evolving financial landscape.

Chapter 9

Conclusions and Further Work

This project's journey unfolded multiple layers of complexity involved in the integration of blockchain technology into financial reporting processes and public liquidity management, aiming to enhance transparency, security and efficiency. This Chapter 9 provides examination of the research findings of the Blockchain Financial Statements (BFS) project, placing focus on interpreting these outcomes within the broader context of its impact and examine their wider implications for theory, practice and future investigations. Through this introspective and forward-looking lens, this work aims to spark a continued dialogue on enhancing the financial systems that underpin our economic structures and interactions.

The BFS initiative, at its core, seeks to address pressing challenges in the domains of accounting, business management, and central banking intervention during turbulent economic conditions. These challenges include, but are not limited to, the integrity of financial reporting, the agility of liquidity provision and the overarching quest for trust and transparency between economic entities and liquidity providers. In navigating these discussions, the objective is to offer a comprehensive understanding of the BFS project's contributions to the field of blockchain, finance and accounting setting the stage for ongoing innovation and exploration in this dynamic and rapidly evolving domain.

9.1 Introduction

This Chapter of the Blockchain Financial Statements Thesis discusses significant findings and implications arising from the integration of blockchain technology within the financial reporting, business management and liquidity provision domains. This exploration is framed around assessing these findings against their alignment with the communicated research problems, objectives, functional requirement fort the BFS prototype. The examination is shaped by the four critical research problems, each probing into the viability, implications, and potential enhancements blockchain technology introduces to the conventional frameworks of financial transactions, reporting, and regulatory compliance.

The **first problem** examines the feasibility to utilise a blockchain-enabled framework of creating a direct, secure connection between heterogeneous economic entities and public money providers, such as governments or central banks.This inquiry is crucial for ensuring a tamper-evident transactional data integrity and automating compliance with liquidity support regulations, underscoring the need for a reconciliation mechanism that authenticates the validity of financial claims for post-reporting period.

The **second question** extends the discussion around possibility of the integration of financial statements into this blockchain-enabled framework. It explores the possibility of establishing an automated, direct connection between financial statements of these public money providers and diverse business entities, thereby enhancing assurance of financial drawdown requests for public money.

Problem three shifts focus to the role of blockchain within the BFS framework, examining its architectural and technological attributes, together with DLT specific process and models, by scoping and mapping its capacity to address systemic issues inherent in traditional financial accounting and reporting. It questions the extent to which blockchain technology can be harnessed to design and develop a secure and trusted mechanism for liquidity distribution and verification, leveraging its data architecture for accurate business activity recording and reporting.

Finally, **problem four** encapsulates the primary overarching goal for this research project - the creation of a direct, secure, and tamper-evident communication between economic entities and public liquidity providers. It explores the possibility of designing and developing a blockchain-based accounting based system that utilises blockchain's strengths to verify, log and report on business operations in a way that mitigates present challenges of misleading financial reporting, and facilitates secure, trustworthy liquidity distribution through transactions.

To address these challenges, the research outlines five key objectives, ranging from investigation of the state-of-knowledge about the impact from blockchain innovation on accounting and business practices, together with its prospective utilisation in central banking, to designing and developing the BFS artefact itself. Throughout this discussion on BFS's implications, the study critically evaluates how the BFS architecture, with its integration of smart contracts and novel funds verification consensus model, fulfils essential functional requirements for stakeholders. This includes ensuring verifiable transactional data and immutable financial statements, enabling real-time validation of transactions, balancing privacy with accountability, automating transactional and accounting processes, offering digital identity solutions, enabling role-based access and transactional control, and facilitating secure transactions across multiple blockchain systems. These objectives encapsulate a comprehensive approach to exploring blockchain's applications in finance and accounting, contributing to the body of knowledge, and offering practical implications and recommendations for stakeholders. Such comprehensive discussion not only underscores the BFS's significant advancements in employing blockchain for financial accounting and reporting, but also highlights its potential in fostering a more transparent, efficient, and reliable financial ecosystem.

9.2 Interpretation of Findings

This Section presents a comprehensive examination of the Blockchain Financial Statements research findings. This exploration is central to understanding the dynamic interplay between advanced blockchain functionalities and traditional financial accounting practices. The analysis looks into how the BFS aligns with and addresses the specific needs and challenges related to ensuring data integrity, enabling efficient and accurate transaction validation, enhancing privacy while maintaining accountability, streamlining accounting processes, and fostering a secure and collaborative ecosystem for all stakeholders. Through a detailed examination of each core aspect of the BFS, this segment sheds light on the practical implications, theoretical advancements, and future directions in the integration of blockchain within financial and accounting landscapes. This critical assessment not only highlights the BFS's innovative contributions but also sets the stage for future developments in the convergence of blockchain technology and financial accounting.

Blockchain in Accounting. The exploration into the impact of blockchain technology on accounting practices and business models has uncovered significant insights that align with the first research objective, offering a comprehensive understanding of how blockchain's inherent features—decentralisation, immutability, and cryptographic security—revolutionise the way businesses operate, transact, and report financial activities. The exploration into blockchain technology's potential to innovate business practices has unveiled a paradigm shift in financial reporting and fraud mitigation, aligning with the broader aim to fortify the integrity and reliability of financial statements. One of the primary findings is the potential for blockchain to ensure the creation and preservation of immutable transaction records. This ability significantly boosts the reliability of record-keeping practices, offering a steadfast solution to the problem of data manipulation and fraudulent activities within financial reporting. The literature suggests that the blockchain's transparent and decentralised nature could serve as a powerful deterrent against unethical accounting practices by providing a platform where transactional data alterations are easily detectable by all network participants. By enforcing decentralised consensus mechanisms, blockchain-based accounting systems inherently deter manipulation of financial data, offering a robust solution to one of the critical challenges facing traditional accounting practices.

In synthesizing these findings, the research contributes to a deeper understanding of blockchain's potential to redefine accounting and business models. It lays a solid foundation for further exploration into the application of blockchain technology, offering secure, transparent, and efficient financial ecosystems. This aligns with the broader goal of leveraging blockchain to enhance the financial infrastructure, ultimately contributing to the sustainable development of the global economy through improved practices in financial reporting and fraud prevention.

Blockchain in Central Banking. The systematic mapping study on "Blockchain for Central Banks" directly aligns with and fulfils the second research objective of exploring the potential application of blockchain technology within central banking systems.

This comprehensive analysis bridges the gap for this research projects between theoretical propositions and practical considerations, shedding light on the dynamic potential of blockchain in revolutionizing central banking operations, services, and regulatory frameworks. Through a methodical review of peer-reviewed publications, this study categorises the diverse array of blockchain use-cases within the realm of central banking, encompassing Central Bank Digital Currency (CBDC), Payment Clearing and Settlement (PCS) systems, assets transfer, audit trails, and regulatory compliance. The thematic overview not only maps out the current landscape of academic and practical research efforts but also highlights the opportunities and challenges that emerge from integrating blockchain technology into central banking functions. One of the critical insights derived from this study is the significant attention directed towards the development and implications of CBDCs. This focus underscores the growing recognition of blockchain's capacity to secure and streamline digital currency operations, presenting a futuristic vision for monetary transactions and policies. The exploration of PCS systems operated by central banks further promotes blockchain's potential to enhance efficiency, security, and transparency in financial settlements, potentially transforming the traditional banking infrastructure. The study's findings on assets transfer, audit trails, and regulatory compliance further underscore blockchain's broader applicability in ensuring secure, transparent, and accountable central banking operations. These use-cases demonstrate blockchain's versatility in addressing key concerns such as data integrity, auditability, and adherence to regulatory standards, reinforcing the technology's role in fortifying the financial ecosystem's foundation. This comprehensive mapping study validates the conceptual viability of a blockchain-based solution, such as BFS in central banking and reinforces the necessity for continued empirical developments to bridge the gap between theoretical potential and practical implementation of blockchain in central banking. In essence, this systematic mapping study significantly advances the understanding of blockchain's applicability and implications for central banking, aligning with the research objective to explore and understand the opportunities and challenges associated with blockchain integration. This aligns with the overarching goal of leveraging blockchain technology to improve central banking operations and support the broader economic ecosystem in times of need.

The subsequent findings discussed within this Chapter are designed to advance the fulfilment of research objectives 3, 4, and 5 (see Section 1.2), directly aligning with the functional requirements as identified by stakeholders (see Section 3.2.7). Through the conceptualisation, design, and development of the BFS, this research provides the practical implementation of blockchain technology in enhancing the reliability, verifiability, and real-time processing of financial reporting and liquidity management. These findings contribute to a body of knowledge that not only explores the theoretical potential of blockchain technology within financial systems but also provides a tangible demonstration of its application, thereby offering practical insights and recommendations for businesses, financial institutions, and policy-makers. This exploration ensures that the BFS framework serves as a reference point for stakeholders aiming to leverage blockchain technology to optimise financial infrastructure, thereby enhancing transparency, efficiency, and integrity across financial and accounting practices.

Architectural Approach. The Domain-Driven Design (DDD) methodology underpins the BFS architecture, promoting a deep understanding of complex domain logic. This approach ensures that the complexities within wide business domain of BFS and interactions of its components are accurately modelled and implemented. Through a single use-case scenario, the BFS's capability to manage financial transactions and reporting within a blockchain-enabled framework is showcased. By adhering to architectural framework of DDD, the BFS prototype demonstrates the theoretical feasibility and its practical effectiveness in developing a technically sound and domain-aligned modular solution capable of integration blockchain technology within the financial reporting domain and capable of aligning with business orchestration needs.

Technological Approach - Dual Blockchain Framework Integration. At the core of the BFS's domain model lies the innovative application of blockchain technology, tailored to function as a comprehensive financial accounting system for economic entities. This model is central in transforming raw economic data into financial reports that are cryptographically secure and verifiable on demand. The dual blockchain framework integration is a testament to the project's technological innovation, presenting a solution that harmonises the strengths of two distinct blockchain architectures to overcome the limitations of each and to improve traditional financial reporting. The technological stack designed and implemented for the BFS reveals how the ecosystem functions as a network of distinct economic entities. Each entity maintains individual financial statements or BFS filing history, summarizing its unique economic engagements. This decentralised yet coherent structure for financial reporting is critical for understanding the BFS as a Proof of Concept (PoC) Java prototype operating within such a multi-ledger ecosystem.

- **Immutable and Verifiable Data:** The BFS architecture leverages blockchain's inherent tamper-evident nature, ensuring that all transactional data, reflected in financial statements is immutable. This feature directly addresses a stakeholder's need for data integrity and trust in financial reporting.
- **Privacy and Accountability:** The dual blockchain framework, incorporating elements from both Bitcoin and Corda platforms enables a balance between the need for transactional privacy with the requirement for accountability. While the Bitcoin framework provides a immutable, cryptographically linked data structure for the BFS's filing history, the Corda platform's Flow Framework and Transaction Vaults introduce a refined approach to data privacy and traceability.

BFS is envisioned and designed to facilitate continuous recording and periodic reporting on the entity's economic activity, offering verifiable, on-demand validation of financial health based on historical transactional interactions. Such a technological approach to the BFS prototype demonstrates its operational viability, addressing the need for verifiable yet privacy-preserving techniques in the secure sharing of sensitive financial information. This dual framework approach highlights the BFS's potential as a transformative tool in the realm of financial accounting and liquidity management, offering a blueprint for future innovations in the field.

Entries in General Journal and General Ledger The BFS introduces novel transformative approach to traditional accounting entries that capture generalised accounting information. In the BFS, each general journal and general ledger entry innovatively incorporates blockchain's data provenance features. Each of these entries embeds an additional layer for verification through hashes of source business transaction that created these entries. This novel integration aids in the auditing process by ensuring each record's traceability and verification. Such a unique approach to transformation of traditional accounting lays the groundwork for more transparent and reliable financial reporting by ensuring every piece of economic data can be traced back to its original transaction. By embedding blockchain traceability within each of these entry, BFS provides a tamper-evident record of all financial activities. This feature guarantees the integrity of financial records and supports the system's role in enhancing transparency in financial reporting.

- ❖ **Verifiable Transactional Data.** The BFS architecture's core is its ability to produce verifiable transactional data, a fundamental need for all stakeholders within the economic ecosystem. By leveraging blockchain technological processes, the BFS ensures that each entry in the General Journal and the subsequent postings in the General Ledger are accurate and immutable and traceable. Each transaction within the BFS, from its initiation to its final inclusion in the subsequent financial statements, carries a unique cryptographic identifier, ensuring its origin and integrity are preserved and verifiable at any point.
- ❖ **Real-time Validation of Transactions.** In addition to maintaining verifiable records, the BFS system provides real-time validation of transactions capability, particularly critical in today's fast-paced economic environment, where the timeliness of financial information can significantly impact decision-making and strategic planning. The BFS employs a dynamic validation mechanism, where transactions are recorded instantaneously in the accounting journals and can be validated in real-time as they are processed through the accounting cycle. This ensures that stakeholders have access to the most current and accurate financial data, facilitating immediate insight into the entity's financial activities and status.

Financial Statements Implementation The BFS architecture's aliment with domain-specific requirements of accounting demonstrates its capability to generate accurate and real-time financial reports. The implementation strategy for the BFS opted for a focused approach by demonstrating only the Balance Sheet's generation, based on a select number of business transactions. This decision aligns with the project's objective to provide a PoC illustration of the BFS's capabilities in real-time financial reporting and auditing, while allowing us to test and demonstrate a financial position of an entity at a given time.

- ❖ **Real-time Auditing and Verification with Privacy Preservation.** A notable innovation within the BFS framework is the report entry structure within financial statements. In similarity to the ledger entries, it diverges from traditional accounting practices by offering a more detailed view of the accounts in financial reports. Leveraging data tractability framework of Corda, BFS ensures that each transaction contributing to the final balances within report entries of the Balance Sheet can be continuously tracked, providing a level of traceability and verification for every record. By enabling this direct link back to the source of each transactional record, the BFS sets a precedent for future financial reporting standards, potentially revolutionizing the landscape of financial auditing and validation in the blockchain era.
- ❖ **Balancing Privacy with Accountability.** In addition to the source hashes, the report entries include a mapping of encrypted identities of participants of these transactions within report entry data. The inclusion of mapping of transactional hashes and encrypted identities reinforces the BFS's commitment to accountability, while facilitating privacy preservation to individual participants through encryption.

Such design and implementation of the BFS's financial statements fulfils essential functional requirements of its stakeholders, ensuring verifiable and immutable financial records. This innovation in financial reporting enables real-time transaction validation and offers a framework for balancing privacy with accountability, ultimately contributing to the construction of a more transparent, secure and resilient financial ecosystem.

Adaptive Accounting Cycle The accounting cycle within the BFS is designed to facilitate automating transactional and accounting processes and is adaptive to the unique business requirements of an organisation. It can be designed to run a full-circle following each business transaction, thereby is able to facilitate production of distinct Balance Sheet reports for comparative analysis instantaneously. This process can also be set up on demand and can be aligned to traditional reporting time frames if necessary. This methodology highlights the system's flexibility and efficiency and its potential to provide immediate or on demand insights into an entity's financial health post-transactional activities.

Block Data Structure. The BFS system's block data structure is designed to house the diverse types of data focused on comprehensive financial reporting requirements. This structure is critical for integrating mandatory reporting elements, as required by regulatory entities such as Companies House. The block data structure's versatility is showcased through the encapsulation of various data types, ranging from financial statements to incorporation documentation and more. This diversity ensures that the BFS can serve as a singular, authoritative source for all financial and regulatory information pertaining to a business entity. The inclusion of specific data types such as ACCOUNTS, CAPITAL, CHARGES, and INCORPORATION documents within the block structure is not merely a technical achievement but a strategic alignment with real-world financial reporting and compliance standards. This alignment guarantees that the BFS adheres to and actually enhances the standards set by traditional financial reporting systems.

The BFS Filing History Blockchain. Central to the BFS system is the filing history blockchain, a novel implementation that serves as a digital ledger for all mandatory reporting of an economic entity. The filing history blockchain illustrates BFS's capacity to digitise and secure a wide array of financial and operational data, thus providing a reliable foundation for transparent and verifiable record-keeping. By mirroring the organisational structure and data categories of the Companies House's reporting standards, the BFS ensures its utility and relevance in real-world financial ecosystems.

- ❖ **Ensuring Immutable and Verifiable Data** The core of the BFS system's design philosophy is the guarantee of data immutability and verifiability. By leveraging blockchain technology, the BFS system ensures that once a transaction is recorded within the block data or the filing history blockchain, it cannot be altered or deleted. This immutability, paired with the system's inherent ability to verify the authenticity and accuracy of the recorded data, provides stakeholders with a reliable foundation for trust in the financial statements and reports generated by the BFS system.
- ❖ **Real-time Auditing and Verification with Privacy Preservation.** Moreover, the BFS's blockchain structure facilitates secure transaction and information exchange across multiple blockchain systems. By employing a dual blockchain framework, the BFS leverages the strengths of different blockchain technologies to ensure the security and privacy of transactions while maintaining an auditable record of financial activities.
- ❖ **Balancing Privacy with Accountability.** BFS architecture addresses and fulfils the fundamental requirements of stakeholders by providing immutable and verifiable transactional data and financial statements, maintaining a balance between privacy and accountability, and facilitating secure transactions across multiple blockchain systems.
- ❖ **Secure Transactions Across Multiple Blockchains.** The interoperability and security of transactions across different blockchain systems is achieved by integrating dual blockchain frameworks and employing advanced cryptographic techniques and HTLC model, the BFS system facilitates secure and seamless transactions within its ecosystem. This capability ensures that financial transactions, asset transfers and record updates are conducted in a secure environment, safeguarding the integrity of the financial data.

UTXO and Consensus In the BFS framework, UTXO and consensus mechanisms are innovatively tailored to fit the nuanced requirements of financial reporting and liquidity management, diverging from their conventional roles in cryptocurrency transactions. This bespoke adaptation aligns with the BFS's objective to provide a solution for the real-time validation of financial transactions and the verification of monetary claims, thereby fostering a secure and trustworthy environment for all ecosystem participants. Within the BFS, the concepts of UTXO and Consensus evolve into a uniquely designed mechanism for validating transactional input through internal query of accounting data of an entity, using the most up-to-date accounting records without disclosing proprietary information. By integrating UTXO generation into the BFS's accounting processes, the system can offer verifiable assurances to transacting parties regarding the legitimacy and sufficiency of claimed funds, all while maintaining the privacy of sensitive business information. This innovative application of UTXO and consensus reduces the risk of fraudulent activities and streamlines the process of transaction validation, ensuring that all network participants adhere to a unified standard of truth. The BFS's strategic application of both UTXO and consensus mechanisms illustrates a bridge between the domains of blockchain technology and traditional accounting. This fusion reinforces the system's role in advancing the reliability and verifiability of financial documentation. Through these adaptations, the BFS has established a pioneering framework that meets critical stakeholder requirements, including immutable transactional records, instantaneous transaction validation and a balance between privacy and accountability. This interpretation of findings from the BFS demonstrates a significant leap forward in leveraging blockchain technology to foster secure, efficient and transparent financial management practices. As such, it sets a new benchmark for blockchain applications in financial reporting and opens avenues for future exploration and innovation in blockchain-based financial systems.

Wallets and Role Based Access. The exploration of wallet implementations within the BFS offers a nuanced view of how digital identities and transactional capabilities are managed and optimised in a blockchain-enabled BFS ecosystem. This strategic approach to wallet design and functionality underscores the BFS's commitment to fostering a secure, efficient and user-centric environment for financial transactions and interactions.

- **Role-based Digital Identity and Access Control.** The BFS wallets serve a dual purpose: they are custodians of digital identity and arbiters of transactional authority. This dual functionality is pivotal in maintaining a secure and verifiable network of participants, thereby reinforcing the trust and integrity of the BFS ecosystem. The wallet's design as a digital identity repository and a transactional command center ensures that access to an entity's BFS ledger is meticulously governed, aligning with the overarching goals of transparency and security.
- **Facilitating Multi-Blockchain Transactions.** The adaptability of wallet implementations within the BFS extends to their capacity to facilitate transactions across diverse blockchain architectures. This capability is necessary in ensuring that the BFS can operate seamlessly within a broader blockchain ecosystem, enabling interoperability and enhancing liquidity flows between different networks. The wallets' design to support multi-blockchain transactions reflects the BFS's vision of a interconnected financial landscape where assets and information can flow freely and securely.

This alignment of wallet functionalities with the overarching objectives of the BFS—emphasising secure transactions, real-time data validation and stakeholder privacy—highlights the system's innovative approach to addressing contemporary challenges in financial reporting.

Loan Agreement Smart Contract The implementation of the Loan Agreement Smart Contract within the BFS architecture represents a significant advancement in streamlining and securing the process of liquidity provision between economic entities and liquidity providers. This smart contract embodies a programmable agreement which automates the terms and conditions under which financial support is extended and utilised, ensuring both the integrity and efficiency of transactions.

- **Automation and Efficiency in Liquidity Provision.** The Loan Agreement Smart Contract is specifically designed to automate the complex legal and financial arrangements traditionally associated with loan agreements. By encapsulating these terms within executable code, the BFS significantly reduces the administrative overhead and potential for human error in the management of loan agreements. This automation facilitates a more direct and efficient mechanism for liquidity provision, enabling economic entities to rapidly access financial resources in accordance with predefined criteria and conditions.

- ❖ **Real-time Validation and Compliance.** The Loan Agreement Smart Contract also introduces the capability for real-time validation of compliance with the terms of the agreement. Through the use of blockchain technology, transactions can be automatically validated against the terms codified in the smart contract, ensuring that each drawdown request, fund transfer and repayment adheres to the agreed-upon conditions. This real-time validation serves as a powerful tool for risk management, minimizing the potential for disputes and enhancing the overall integrity of financial transactions within the BFS ecosystem.
- ❖ **Streamlining Financial Operations.** The integration of the Loan Agreement Smart Contract within the BFS exemplifies a strategic approach to streamlining financial operations. By leveraging blockchain technology to automate and secure loan agreements, the BFS facilitates a more agile financial landscape, where liquidity can be rapidly mobilized to meet the operational needs of economic entities. This approach optimises financial management practices and contributes to the resilience and adaptability of the broader economic ecosystem.

Business Transaction Smart Contracts This Thesis examines the integration of smart contracts to automate key business transactions, each embodying distinct aspects of financial operations within the blockchain framework. These transactions, from share settlements to asset acquisitions and liquidity management, demonstrate the BFS's capability to align with traditional financial processes while leveraging blockchain's inherent benefits of security, transparency and efficiency.

1. **Initial Share Settlement Business Transaction (DvP with HTLC)** The implementation of the Initial Share Settlement Business Transaction showcases the BFS's capacity to execute secure and transparent financial transactions. By employing a Delivery versus Payment (DvP) model with Hashed Timelock Contracts (HTLC), the BFS ensures that the issuance and sale of tokenized shares between the APF and the BoE occur simultaneously and conditionally. This mechanism mitigates risk and enhances trust between parties, illustrating the potential of smart contracts to streamline equity transactions in compliance with existing financial protocols and standards.
2. **Drawdown Request on Loan Agreement with HTLC** The drawdown request mechanism within the BFS framework demonstrates the system's flexibility and responsiveness in managing liquidity. Though part of a broader transactional framework, this feature highlights the BFS's capacity to adapt and respond to immediate financial needs. While functioning as part of a broader transactional framework, this smart contract automates the process of drawdown requests, facilitating efficient and secure access to financial resources necessary for asset purchases. This integration illustrates the BFS's comprehensive approach to managing liquidity provision, from initial funding agreements to the execution of financial transactions essential for QE activities.
3. **Atomic Four-Party QE with HTLC** The Quantitative Easing process implementation via BFS's smart contracts illustrates how the integration of blockchain technology with central banking functions can offer innovative approaches to monetary policy implementation, particularly in terms of liquidity provision and economic stimulation (see Section 2.1.2). Such demonstration is reinforced by the BFS's innovative approach to facilitating acquisitions of financial assets. The amalgamation of Delivery versus Delivery (DvD) and One-Way-Payment transactional models with blockchain native process for implementing atomic assets swap with HTLC, underpins the BFS's ability to handle complex, multi-party transactions efficiently. This functionality is crucial for the APF's

operational effectiveness, enabling support of diverse monetary policy objectives and regulatory goals. The design and operationalization of this smart contract are deeply rooted in the existing past and present operational procedures of the APF's, factual terms and conditions for market participants in CPF, and actual procedural rules AF's settlement processes for QE enabled assets purchases. Such real-world specification driven model for a BFS's smart contract ensures that the BFS's functionalities are in alignment with tangible financial governance structures. This alignment validates the BFS's capabilities in handling exceptionally complex transactional arrangements and reinforces its potential to transform liquidity management practices.

Accounting Transactions Smart Contracts The implementation of Accounting Transactions Smart Contract within BFS marks a significant advancement in the integration of blockchain technology with conventional accounting processes. This innovative approach facilitates the automation of accounting transactions, bridging the gap between the execution of financial activities and their representation in financial records.

- ❖ **Automation and Verification** At its core, the Accounting Transactions Smart Contract automates the generation of accounting entries from economic events, ensuring that each financial activity is accurately captured and reflected in the entity's accounting records. By automating the generation, classification and recording of accounting entries, the BFS delivers on its promise of real-time financial data processing. This automation streamlines accounting workflows, drastically reducing the potential for human error and the risk of fraudulent financial reporting. This automation extends beyond mere data entry to include the validation and verification of transactional data against predefined accounting rules and principles.
- ❖ **Real-time Financial Reporting** Another pivotal aspect of the Accounting Transactions Smart Contract is its contribution to real-time financial reporting. Traditional accounting processes often entail a significant lag between the occurrence of an economic event and its reflection in financial statements. However, the automated and instantaneous nature of smart contract execution within the BFS minimises this delay, enabling near real-time updates to financial records. This capability is instrumental in providing stakeholders with timely and relevant financial information, thereby improving decision-making and enhancing transparency across the financial ecosystem.
- ❖ **Integration with Blockchain Security Features** Furthermore, the integration of Accounting Transactions Smart Contract with blockchain's inherent security features—such as immutability, transparency and cryptographic verification—fortifies the integrity of financial records. Each accounting entry generated through these smart contracts is securely recorded on the off-ledger repository of an entity, creating verifiable trail of financial activities. This aids in the prevention of fraudulent alterations to financial records and facilitates a more robust and trustworthy financial audit process.
- ❖ **Facilitating Compliance and Transparency** Moreover, the Accounting Transactions Smart Contract implementation facilitates added layer of financial control and oversight. By codifying accounting rules and principles within the smart contract logic, the BFS enforces strict adherence to accounting standards, ensuring that all financial transactions are processed in accordance with best practices and legal requirements. This meticulous approach to transaction processing underscores the BFS's role as a robust framework for financial governance and compliance.

In summary, the Accounting Transactions Smart Contract implementation within the BFS exemplifies the potential of blockchain technology to revolutionise accounting practices. Through automation, real-time reporting, enhanced security, and regulatory compliance, these smart contracts address critical needs within the financial reporting domain, offering a glimpse into the future of financial management and accountability. The BFS, through this implementation, fulfils essential functional requirements for stakeholders and sets a new benchmark for innovation in accounting processes.

9.3 Implications

The exploration and implementation of the Blockchain Financial Statement within this thesis yield significant implications across theoretical, practical and policy domains. These implications are central to understanding of how financial data is managed, reported and regulated, and of the potential impact and future directions enabled by integrating blockchain technology within financial systems.

9.3.1 Theoretical Implications

This research enriches the academic discourse surrounding blockchain applications in financial systems by demonstrating a tangible framework where blockchain's inherent capabilities are directly applied to financial reporting and accounting processes. It introduces a novel framework that embodies the principles of Domain-Driven Design (DDD) coupled with the robustness of blockchain technology to automate and secure financial transactions and reporting. The BFS architecture showcases a practical application of blockchain beyond its traditional boundaries, illustrating its potential in addressing complex financial accounting processes. It also introduces a novel perspective on the utilization of smart contracts and blockchain infrastructure to automate and secure financial transactions and reporting, thereby offering a comprehensive model for future studies in this domain.

This work lays a foundational theory for the development of decentralised financial systems that are secure and transparent and adaptive to the dynamic nature of financial markets. It provides a conceptual and operational roadmap for further academic exploration into the integration of blockchain in various facets of financial management and reporting.

9.3.2 Practical Implications

From a practical standpoint, the BFS system demonstrates a significant leap towards enhancing financial transparency and efficiency within business operations. By demonstrating the feasibility and benefits of blockchain-based financial reporting, the BFS serves as a prototype that can inspire the development of similar systems across various sectors. It showcases how real-time transaction validation, coupled with immutable and verifiable record-keeping, can significantly reduce the risks of fraud and errors in financial statements. Furthermore, By automating accounting transactions and ensuring the immutability of financial records, BFS offers a solution to the perennial challenges of financial misreporting and fraud. Businesses can leverage this system to streamline their financial processes, ensuring real-time reporting and validation of transactions, which, in turn, can foster trust among stakeholders. Financial institutions, on the other hand, can benefit from the system's ability to provide a secure and verifiable platform for managing transactions, thereby reducing operational risks and compliance costs. The BFS system exemplifies how blockchain technology can be tailored to meet the specific needs of financial reporting, offering a blueprint for businesses and financial institutions aiming to adopt blockchain for financial management. For businesses, this means improved trust with stakeholders and reduced costs associated with financial auditing and compliance. Financial institutions can leverage these insights to streamline their operations, enhance customer trust and adapt to the evolving regulatory landscape with greater agility.

9.3.3 Policy Implications

The findings from this thesis hold profound implications for policy-making, especially in the realm of regulatory standards for blockchain systems in financial reporting. The project reinforces the need for regulatory frameworks that can accommodate the unique characteristics and benefits of blockchain technology, while ensuring financial stability and consumer protection. Policymakers can draw from the BFS's architecture and implementation to understand the potential regulatory challenges and opportunities presented by blockchain in financial systems. Moreover, the BFS architecture offers a model for how blockchain systems can be designed to align with regulatory requirements, facilitating a constructive dialogue between regulators, businesses and technology providers. This understanding can inform the development of policies that encourage innovation in financial technologies while safeguarding the integrity of financial markets. Specifically, the BFS model can serve as a reference point for establishing standards related to transactional transparency, data integrity and system security in blockchain-based financial reporting systems. In summary, the BFS system embodies a multidimensional contribution to the theoretical knowledge, practical application and policy development surrounding blockchain technology in financial systems. Its implications extend beyond the immediate scope of financial reporting, offering insights and frameworks that can guide future research, operational strategies and policy formulations in the evolving landscape of blockchain and financial technology.

9.4 Limitations of the Study and Recommendations for Future Research

This exploration into the Blockchain Financial Statement (BFS) system, while offering valuable insights and contributions to the field, is accompanied by several limitations that require further research and development. These limitations frame the current scope of the study and suggest directions for future work to enhance the BFS's applicability, robustness and user experience.

- **Generalisation Beyond the APF Use-Case.** The BFS system, as currently designed, is tailored specifically to the Asset Purchase Facility (APF) use-case. This focus has provided a detailed and concentrated examination of the potential for blockchain within a narrowly defined scope of financial transactions and reporting. However, this specificity limits the direct applicability of the BFS system to other financial contexts without further modification and customisation. Future iterations of the BFS project would benefit from a more generalised framework that can easily adapt to a variety of financial and operational scenarios and extended requirements, thereby broadening the system's utility across different sectors and transaction types.

- **Inter-Blockchain Networking.** The theoretical architecture of the BFS proposes an innovative approach to inter-blockchain networking to facilitate seamless interactions between distinct blockchain systems. However, the practical implementation of this inter-blockchain networking component remains in the conceptual phase. Actualising this aspect of the BFS architecture is critical for enabling the system to operate within a multi-blockchain ecosystem, thereby expanding its utility and applicability. Addressing this limitation involves the development and integration of networking code that can efficiently manage cross-chain transactions and data sharing.

- **Accounting Use-Cases Expansion.** While the BFS demonstrates significant advancements in automating and securing accounting processes through blockchain technology, its scope of accounting use-cases remains limited. The current system primarily addresses financial reporting and transaction validation and primarily focused on the generation of balance sheets, leaving out other accounting functions that could benefit from blockchain integration. To fully harness the potential of blockchain technology in revolutionizing financial reporting, it is essential to expand the BFS to include a wider range of accounting functionalities. This expansion would involve the integration of additional financial statements, such as income statements and cash flow statements and the incorporation of diverse accounting scenarios and transactions. Expanding the BFS to cover a wider array of accounting practices, such as cost accounting, management accounting and others, would enhance the system's comprehensiveness and relevance to a broader audience within the financial sector.

- ❖ **Comprehensive Transactional Scenarios.** The BFS prototype illustrates transactional flows that represent successful scenarios, often referred to as the "happy path". This focus has limited the exploration of how the BFS handles exceptions, errors and unusual transactional scenarios. This limitation restricts the system's ability to handle exceptions, errors and unusual transactional patterns that are likely to occur in real-world financial environments. Future developments should focus on designing and implementing comprehensive transactional scenarios, including those involving transaction failures, disputes and reversals, to ensure the BFS's robustness and reliability.

- ❖ **User Interface Development.** The current proof of concept (PoC) for the BFS is concentrated on the back-end logic, lacking a user interface that facilitates interaction with the system for non-technical users. The absence of a user-friendly interface limits the accessibility of the BFS, potentially hindering its adoption by a wider range of stakeholders who could benefit from its capabilities. Developing a comprehensive and intuitive user interface is essential for bridging this gap, enabling users to efficiently manage and interact with financial data and transactions within the BFS ecosystem.

- ❖ **Scope and Test NFRs** The current PoC BFS focuses on design and implementation of the functional specifications for the prototype. However, an area that warrants acknowledgment is the absence of a comprehensive evaluation against non-functional requirements (NFRs) and the subsequent need to test these requirements to validate the system's overall performance and resilience. Non-functional requirements represent criteria not directly concerned with the specific functions or features of the system, such as performance, scalability, security and usability. These aspects are crucial for ensuring that the BFS meets its functional objectives but also adheres to broader expectations of reliability, efficiency and user satisfaction. The current phase of the BFS project has primarily focused on demonstrating functional capabilities and the conceptual framework, leaving a detailed assessment of NFRs for future consideration. Future research and development should aim to systematically assess and enhance these aspects, ensuring that the BFS is robust, secure, scalable and user-friendly, thereby fulfilling its promise as a transformative tool for the financial industry.

In addressing these limitations, future research and development efforts can significantly enhance the BFS system's functionality, scalability and applicability, making it a more versatile tool for blockchain-based financial reporting and transactions.

9.5 Concluding Remarks

Reflecting on the broader impact of the BFS research, it is evident that the findings contribute significantly to the discourse on the integration of blockchain technology in financial systems, whilst highlighting a need for a forward-looking perspective on the role of blockchain in financial reporting and liquidity management. The BFS exemplifies how blockchain can improve traditional financial reporting constraints, offering a future where financial operations are more secure, efficient and transparent. In summary, this thesis encapsulates a comprehensive journey from conceptualisation to the realization of the BFS, underscoring the realistic improvements from integration blockchain technological components in financial reporting and liquidity management. Looking ahead, the implications of this study extend beyond accounting and transactions. The BFS model presents a blueprint for future blockchain applications in finance, where financial transactions and reporting are more secure, transparent, and efficient. As the landscape of blockchain in finance continues to evolve, the insights garnered from this study offer valuable perspectives on the direction of future research and development. The BFS project showcases the practical application of blockchain in financial systems and sets a precedent for further exploration into the role of blockchain technology in shaping the future of financial reporting. This study contributes significant insights into the application and benefits of blockchain in finance and prompts further exploration into its expansive potential. Looking ahead, the landscape of blockchain in finance appears ripe for innovation, poised to redefine traditional financial systems in ways previously unimagined.

Appendix A

Research Ethics Letter

College of Engineering, Design and Physical Sciences Research Ethics Committee
Brunel University London
Kingston Lane
Uxbridge
UB8 3PH
United Kingdom

www.brunel.ac.uk

21 February 2024

LETTER OF CONFIRMATION

Applicant: Mrs Natalia Dashkevich

Project Title: Blockchain Financial Statements (BFS): Transaction to Financial Statements Accounting System for Central Bank to Business Liquidity

Reference: 47586-NER-Feb/2024- 50101-1

Dear Mrs Natalia Dashkevich,

The Research Ethics Committee has considered the above application recently submitted by you.

This letter is to confirm that, according to the information provided in your BREO application, your project does not require full ethical review. You may proceed with your research as set out in your submitted BREO application, using secondary data sources only. You may not use any data sources for which you have not sought approval.

Please note that:

- **You are not permitted to conduct research involving human participants, their tissue and/or their data. If you wish to conduct such research (including surveys, questionnaires, interviews etc.), you must contact the Research Ethics Committee to seek approval prior to engaging with any participants or working with data for which you do not have approval.**
- The Research Ethics Committee reserves the right to sample and review documentation relevant to the study.
- If during the course of the study, you would like to carry out research activities that concern a human participant, their tissue and/or their data, you must submit a new BREO application and await approval before proceeding. Research activity includes the recruitment of participants, undertaking consent procedures and collection of data. Breach of this requirement constitutes research misconduct and is a disciplinary offence.

Good luck with your research!

Kind regards,

Simon Taylor

Professor Simon Taylor

Chair of the College of Engineering, Design and Physical Sciences Research Ethics Committee

Brunel University London

Appendix B

All Included Papers

B.1 Database Search Results

Database Search Results: D* - Database Name; R* – Results; I/E* – application of inclusion / exclusion criteria on title, keywords and abstract (automated, if available via database, or manual). Table B.1

B.2 Full List of Headers of the Data Collection Form

Table B.2

B.3 All Included Papers

Table B.3

B.4 Publication Venues and Publishers

Table B.4 and Table B.5

B.5 Matrix of the Research

Table B.6 - the complete matrix of the research of all included papers.

Table B.1: Database Search Results. (Appendix A)

D*	**Database Applicable Search String**	**R***	**I/E***
IEEExplorer	(“banking” OR “bank” OR “central bank” OR “reserve bank” OR “monetary authority” OR “monetary” OR “financial Intermediary” OR “financial Intermediation” OR “clearing” OR “clearinghouse” OR “settlement” OR “financial institution” OR “FinTech” OR “financial technology” OR “inter-bank” OR “IBPS” OR “real-time gross settlement” OR “RTGS” OR “payment settlement” OR “CBDC” OR “money supply” OR “monetary policy” OR “technocracy”) AND (“blockchain” OR “distributed ledger technology” OR “DLT” OR “smart contracts”)	995	30
Scopus	(“banking” OR “bank” OR “central bank” OR “reserve bank” OR “monetary authority” OR “monetary” OR “financial Intermediary” OR “financial Intermediation” OR “clearing” OR “clearinghouse” OR “settlement” OR “financial institution” OR “FinTech” OR “financial technology” OR “inter-bank” OR “IBPS” OR “real-time gross settlement” OR “RTGS” OR “payment settlement” OR “CBDC” OR “money supply” OR “monetary policy” OR “technocracy”) AND (“blockchain” OR “distributed ledger technology” OR “DLT” OR “smart contracts”) AND (LIMIT-TO (PUBYEAR , 2020) OR LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2018) OR LIMIT-TO (PUBYEAR , 2017) OR LIMIT-TO (PUBYEAR , 2016) OR LIMIT-TO (PUBYEAR , 2015) OR LIMIT-TO (PUBYEAR , 2014) OR LIMIT-TO (PUBYEAR , 2013) OR LIMIT-TO (PUBYEAR , 2012) OR LIMIT-TO (PUBYEAR , 2011) OR LIMIT-TO (PUBYEAR , 2010) OR LIMIT-TO (PUBYEAR , 2009) OR LIMIT-TO (PUBYEAR , 2008)) AND (LIMIT-TO (ACCESSTYPE(OA)))	1002	7
SSRN	bank blockchain	99	40
	banking blockchain	90	
JEL E4	blockchain	301	29
JEL E5	blockchain	198	13
JEL G01	blockchain	40	5
ScienceDirect	blockchain OR “distributed ledger technology” OR DLT OR “smart contracts” search within: banking OR bank OR “central bank” OR “reserve bank” OR “monetary authority” OR clearing OR clearinghouse OR settlement OR “real-time gross settlement” OR RTGS OR “inter-bank” OR CBDC OR “money supply” OR “monetary policy”	493	4
arXiv	bank* AND blockchain*	35	4
Web of Science	(bank* OR “central bank” OR “reserve bank” OR “monetary authority” OR monetary OR “financial Intermedia*” OR clearing OR clearinghouse* OR settlement* OR “financial institution*” OR FinTech OR “financial technology” OR “inter-bank*” OR IBPS OR “real-time gross settlement” OR RTGS OR “payment settlement” OR CBDC OR “money supply” OR “monetary policy” OR technocracy) AND (blockchain* OR “distributed ledger*” OR DLT OR “smart contract*”)	690	24
ACM	(+bank* +blockchain)	431	2
	Total	**4,374**	**158**
	Removing Duplicates		**142**

Table B.2: The Full List of Headers of the Data Collection Form. (Appendix B)

Classification Schema	**Data Types**	**Column Headers**
Topic-Independent Classification Schema	Basic metadata	ID (Paper ID)
		Publication Year
		Publication Venue Name
		Publisher
		Literature Type
	Type of Empirical Research	Evaluation
		Validation
	Type of Non-Empirical Research	Solution Proposal
		Philosophical Paper
		Opinion Paper
		Experience Paper
	Research Contributions	Protocol
		Model
		PoC
		Framework
		Taxonomy
		New Knowledge
Topic-Specific Classification Schema	Blockchain-Based Use-Cases for Central Banks	CBDC
		PCS
		Assets
		Audit
		Regulation

Table B.3: All Included Papers. (Appendix C)

Reference ID	Document Title	Authors
Hileman & Rauchs (2017)	2017 Global Blockchain Benchmarking Study	Garrick Hileman and Michel Rauchs
Arthur et al. (2018)	A 9-Dimension Grid for the Evaluation of Central Bank Digital Currencies	José Parra-Moyano and Arthur Macherel and Adrien Treccani
Han et al. (2019)	A Blockchain-based Framework for Central Bank Digital Currency	Xuan Han and Yong Yuan and Fei-Yue Wang
Zhai & Zhang (2018)	A Cash Flow Blockchain Based Privacy-Preserving	Xiaojun Zhai and Chongyang Zhang
Didenko et al. (2020)	After Libra, Digital Yuan and COVID-19: Central Bank Digital Currencies and the New World of Money and Payment Systems	Arner, Douglas W. and Buckley, Ross P. and Zetzsche, Dirk Andreas and Didenko, Anton
Tsai, Zhao, Zhang, Yu & Deng (2018)	A Multi-Chain Model for CBDC	Wei-Tek Tsai and Zihao Zhao and Chi Zhang and Lian Yu and Enyan Deng
Chiu (2017)	A New Era in Fintech Payment Innovations? A Perspective from the Institutions and Regulation of Payment Systems	Iris H-Y Chiu
Wu et al. (2019)	A Regulated Digital Currency	Yanbing Wu and Haining Fan and Xiaoyun Wang and Guangnan Zou
Tsai et al. (2016)	A System View of Financial Blockchains	Wei-Tek Tsai and Robert Blower and Yan Zhu and Lian Yu
Kavassalis et al. (2018)	An Innovative RegTech Approach to Financial Risk Monitoring and Supervisory Reporting	Petros Kavassalis and Harald Stieber and Wolfgang Breymann and Keith Saxton and Francis Joseph Gross
Tsai, Deng, Ding & Li (2018)	Application of Blockchain to Trade Clearing	Wei-Tek Tsai and Enyan Deng and Xiaoqiang Ding and Jie Li
Shah & Jani (2018)	Applications of Blockchain Technology in Banking and Finance	Tejal Shah and Shailak Jani
Andolfatto (2018)	Assessing the Impact of Central Bank Digital Currency on Private Banks	David Andolfatto
Geva (2018)	Banking in the Digital Age: Who is Afraid of Payment Disintermediation?	Benjamin Geva
Cocco et al. (2017)	Banking on Blockchain: Costs Savings Thanks to the Blockchain Technology	Luisanna Cocco and Andrea Pinna and Michele Marchesi
Borgonovo et al. (2018)	Between Cash, Deposit and Bitcoin: Would We Like a Central Bank Digital Currency? Money Demand and Experimental Economics	Emanuele Borgonovo and Alessandra Cillo and Stefano Caselli and Donato Masciandaro
Nguyen (2016)	Blockchain – A Financial Technology For Future Sustainable Development	Quoc Khanh Nguyen
Guo & Liang (2016)	Blockchain Application and Outlook in the Banking Industry	Ye Guo and Chen Liang
Yoo (2017)	Blockchain Based Fnancial Case Analysis and its Implications	Soonduck Yoo
Ozili (2019)	Blockchain Finance: Questions Regulators Ask	Peterson K Ozili
Seretakis (2019)	Blockchain, Securities Markets and Central Banking	Alexandros L. Seretakis
Chen et al. (2018)	Blockchain-Based Implementation for Financial Product Management	Bihuan Chen and Zhixiong Tan and Wei Fang
Chiu & Koeppl (2019)	Blockchain-based Settlement for Asset Trading	Jonathan Chiu and Thorsten V. Koeppl
Lipton (2018)	Blockchains and Distributed Ledgers in Retrospective and Perspective	Alexander Lipton
Meaning et al. (2018)	Broadening Narrow Money: Monetary Policy with a Central Bank Digital Currency	Jack Meaning and Ben Dyson and James Barker and Emily Clayton
Khiaonarong & Humphrey (2019)	Cash Use Across Countries and the Demand for Central Bank Digital Currency	Tanai Khiaonarong and David Humphrey
Griffoli et al. (2018)	Casting Light on Central Bank Digital Currency	Tommaso Mancini Griffoli and Maria Soledad Martinez Peria and Itai Agur and Anil Ari and John Kiff and Adina Popescu and Celine Rochon
Bech & Garratt (2017)	Central Bank Cryptocurrencies	Morten L. Bech and Rodney Garratt
Cœuré & Loh (2018)	Central Bank Digital Currencies	Benoît Cœuré and Jacqueline Loh
Kumhof & Noone (2018)	Central Bank Digital Currencies: Design Principles and Balance Sheet Implications	Michael Kumhof and Clare Noone
Fung & Halaburda (2016)	Central Bank Digital Currencies: A Framework for Assessing Why and How	Ben S. C. Fung and Hanna Halaburda
Nabilou (2019)	Central Bank Digital Currencies: Preliminary Legal Observations	Hossein Nabilou
Chiu et al. (2019)	Central Bank Digital Currency and Banking	Jonathan Chiu and Seyed Mohammadreza Davoodalhosseini and Janet Hua Jiang and Yu Zhu
Bordo & Levin (2017)	Central Bank Digital Currency and the Future of Monetary Policy	Michael D. Bordo and Andrew T. Levin
Koumbarakis & Dobrauz-Saldapenna (2019)	Central Bank Digital Currency: Benefits and Drawbacks	Antonios Koumbarakis and Guenther Dobrauz-Saldapenna
Engert & Fung (2017)	Central Bank Digital Currency: Motivations and Implications	Walter Engert and Ben S. C. Fung
Pfister (2019)	Central Bank Digital Currency: One, Two or None?	Christian Pfister
Furche & Sojli (2018)	Central Bank Issued Digital Cash	Andreas Furche and Elvira Sojli
Nabilou & Prum (2019)	Central Banks and Regulation of Cryptocurrencies	Hossein Nabilou and André Prüm
Murray (2019)	Central Banks and the Future of Money	John Murray
Danezis & Meiklejohn (2015)	Centrally Banked Cryptocurrencies	George Danezis and Sarah Meiklejohn
Lutz (2018)	Coexistence of Cryptocurrencies and Central Bank Issued Fiat Currencies - A Systematic Literature Review	Julia K. T. Lutz
Milne (2018)	Cryptocurrencies from an Austrian perspective	Alistair Milne
Hayes (2016)	Decentralised Banking: Monetary Technocracy in the Digital Age	Adam Hayes
Agur et al. (2019)	Designing Central Bank Digital Currencies	Itai Agur and Anil Ari and Giovanni Dell'Ariccia
Caytas (2016)	Developing Blockchain Real-Time Clearing and Settlement in the EU, U.S., and Globally	Joanna Caytas
Raskin & Yermack (2018)	Digital Currencies, Decentralised Ledgers and the Future of Central Banking	Max Raskin and David Yermack
Pinna & Ruttenberg (2016)	Distributed ledger technologies in securities post-trading: Revolution or evolution?	Andrea Pinna and Wiebe Ruttenberg
Priem (2020)	Distributed Ledger Technology for Securities Clearing and Settlement - Benefits, Risks and Regulatory Implications	Randy Priem
BIS (2017)	Distributed ledger technology in payment, clearing and settlement - An analytical framework	Benoît Cœuré
Mills et al. (2016)	Distributed Ledger Technology in Payments, Clearing, and Settlement	David C. Mills and Kathy Wang and Brendan Malone and Anjana Ravi and Jeffrey Marquardt and Anton I. Badev and Timothy Brezinski and Linda Fahy and Kimberley Liao and Vanessa Kargenian and Max Ellithorpe and Wendy Ng and Maria Baird
Auer (2019)	Embedded Supervision: How to Build Regulation into Blockchain Finance	Raphael Auer
Ben Dhaou & Rohman (2018)	Everything and its opposite: Socio-economic implications of Blockchain technology: Case of monetary policy	Soumaya I. Ben Dhaou and Ibrahim Kholilul Rohman
Wu & Liang (2017)	Exploration and Practice of Inter-bank Application Based on Blockchain	Tong Wu and Xiubo Liang
Thakor (2020)	Fintech and Banking: What Do We Know?	Anjan V. Thakor
Wang et al. (2018)	Inter-Bank Payment System on Enterprise Blockchain Platform	Xin Wang and Xiaomin Xu and Lance Feagan and Sheng Huang and Limei Jiao and Wei Zhao
Dow (2019)	Monetary Reform, Central Banks and Digital Currencies	Sheila Dow
Shirai (2019)	Money and Central Bank Digital Currency	Sayuri Shirai
Kang & Lee (2019)	Money, Cryptocurrency, and Monetary Policy	Kee-Youn Kang and Seungduck Lee
Sun et al. (2017)	Multi-Blockchain Model For Central Bank Digital Currency	He Sun and Hongliang Mao and Xiaomin Bai and Zhidong Chen and Kai Hu and Wei Yu
Brunnermeier & Niepelt (2019)	On the Equivalence of Private and Public Money	Markus K. Brunnermeier and Dirk Niepelt
Chapman et al. (2017)	Project Jasper: Are Distributed Wholesale Payment Systems Feasible Yet?	James Chapman and Rodney Garratt and Scott Hendry and Andrew McCormack and Wade McMahon
Grody (2018)	Rebuilding Financial Industry Infrastructure	Allan D. Grody
Micheler & Whaley (2019)	Regulatory Technology: Replacing Law with Computer Code	Eva Micheler and Anna Whaley
Kahn & Wong (2019)	Should the Central Bank Issue E-Money?	Charles M. Kahn and Francisco Rivadeneyra and Russell Wong
Berentsen & Schar (2018)	The Case for Central Bank Electronic Money and the Non-Case for Central Bank Cryptocurrencies	Aleksander Berentsen and Fabian Schar
Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)	The Impact of the Distributed Ledger Technology on the Single Euro Payments Area Development	Elżbieta Jantoń-Drozdowska and Alicja Mikołajewicz-Woźniak
Barrdear & Kumhof (2016)	The Macroeconomics of Central Bank Issued Digital Currencies	John Barrdear and Michael Kumhof
Ducas & Wilner (2017)	The Security and Financial Implications of Blockchain Technologies: Regulating Emerging Technologies in Canada	Evangeline Ducas and Alex Wilner
Auer & Böhme (2020)	The Technology of Retail Central Bank Digital Currency	Raphael Auer and Rainer Boehme
Benos et al. (2017)	The Economics of Distributed Ledger Technology for Securities Settlement	Evangelos Benos and Rod Garratt and Pedro Gurrola-Perez
Rio & César (2017)	Use of Distributed Ledger Technology by Central Banks: A Review	César A Del Río

Table B.4: All Publication Venues.

Publication Venue Name	Number of Papers	Paper Reference ID
SSRN Journal of Economic Literature	4	Chiu & Koeppl (2019), Milne (2018), Hileman & Rauchs (2017), Benos et al. (2017)
Bank of England Working Papers	3	Kumhof & Noone (2018), Barrdear & Kumhof (2016), Meaning et al. (2018)
Federal Reserve Bank of St. Louis Research Paper Series	3	Andolfatto (2018), Berentsen & Schar (2018), Kahn & Wong (2019)
SSRN Electronic Journal	3	Arthur et al. (2018), Furche & Sojli (2018), Koumbarakis & Dobrauz-Saldapenna (2019)
Bank of Canada Staff Discussion Papers	2	Engert & Fung (2017), Fung & Halaburda (2016)
BIS CPMI Papers	2	BIS (2017), Cœuré & Loh (2018)
Financial Innovation	2	Priem (2020), Guo & Liang (2016)
IMF Working Papers	2	Khiaonarong & Humphrey (2019), Agur et al. (2019)
Journal of Risk Finance	2	Lipton (2018), Kavassalis et al. (2018)
NBER Papers on Monetary Economics	2	Raskin & Yermack (2018), Bordo & Levin (2017)
Asia Pacific Journal of Innovation and Entrepreneurship: 'Blockchain on business and entrepreneurship'	1	Yoo (2017)
Asian Development Bank Institute Working Paper Series	1	Shirai (2019)
BAFFI CAREFIN Centre for Applied Research on International Markets, Banking, Finance and Regulation Research Paper Series	1	Borgonovo et al. (2018)
Bank of Canada Staff Working Paper	1	Chiu et al. (2019)
Banque de France Research Paper Series	1	Pfister (2019)
BIS Quarterly Review	1	Auer & Böhme (2020)
BIS Quarterly Review Special Features Series	1	Bech & Garratt (2017)
BIS Working Paper Series	1	Auer (2019)
Board of Governors of the Federal Reserve System Finance and Economics Discussion Series; Divisions of Research and Statistics and Monetary Affairs	1	Mills et al. (2016)
C.D. Howe Institute - Monetary Policy Research	1	Murray (2019)
CESifo Working Paper Series	1	Brunnermeier & Niepelt (2019)
Columbia Journal of European Law	1	Caytas (2016)
Enfoque UTE (Universidad Tecnológica Equinoccial)	1	Rio & César (2017)
Equilibrium. Quarterly Journal of Economics and Economic Policy	1	Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)
European Banking Institute (EBI) Research Paper Series	1	Didenko et al. (2020)
European Banking Institute (EBI) Working Paper Series	1	Geva (2018)
European Business Organization Law Review	1	Micheler & Whaley (2019)
European Central Bank (ECB) Research Paper Series - Occasional Papers	1	Pinna & Ruttenberg (2016)
FiDL Working Papers	1	Lutz (2018)
Financial System Review	1	Chapman et al. (2017)
Future Internet	1	Cocco et al. (2017)
IEEE International Conference on Cloud Computing (CLOUD)	1	Wang et al. (2018)
IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS)	1	Zhai & Zhang (2018)
IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)	1	Sun et al. (2017)
IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)	1	Han et al. (2019)
IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)	1	Tsai, Deng, Ding & Li (2018)
IEEE Symposium on Service-Oriented System Engineering (SOSE)	1	Tsai et al. (2016)
IMF Staff Discussion Notes	1	Griffoli et al. (2018)
International Conference on Computer Science and Education (ICCSE)	1	Wu & Liang (2017)
International Conference on Dependable Systems and Their Applications (DSA)	1	Tsai, Zhao, Zhang, Yu & Deng (2018)
International Conference on Green Technology and Sustainable Development (GTSD)	1	Nguyen (2016)
International Conference on Theory and Practice of Electronic Governance (ICEGOV)	1	Ben Dhaou & Rohman (2018)
International Finance Review	1	Ozili (2019)
International Journal (IJ)	1	Ducas & Wilner (2017)
International Journal of Political Economy	1	Dow (2019)
International Telecommunication Networks and Applications Conference (ITNAC)	1	Chen et al. (2018)
Journal of Banking Regulation	1	Nabilou (2019)
Journal of Financial Intermediation	1	Thakor (2020)
Journal of Risk Managment in Financial Institutions	1	Grody (2018)
Law, Innovation and Technology	1	Chiu (2017)
Mediterranean Conference on Information Systems (MCIS)	1	Hayes (2016)
NDSS (Network and Distributed System Security) Symposium	1	Danezis & Meiklejohn (2015)
Regulating Blockchain. Techno-Social and Legal Challenges, Oxford University Press, 2019	1	Seretakis (2019)
Research Gate	1	Shah & Jani (2018)
Review of Banking and Financial Law	1	Nabilou & Prum (2019)
Science China Information Sciences	1	Wu et al. (2019)
SNB-CIF Conference on Cryptoassets and Financial Innovation (Swiss National Bank)	1	Kang & Lee (2019)

Table B.5: All Publishers.

Publisher	Number of Papers	Paper Reference ID
IEEE	10	Tsai, Deng, Ding & Li (2018), Nguyen (2016), Wu & Liang (2017), Wang et al. (2018), Sun et al. (2017), Tsai, Zhao, Zhang, Yu & Deng (2018), Tsai et al. (2016), Zhai & Zhang (2018), Chen et al. (2018), Han et al. (2019)
Elsevier BV	9	Ozili (2019), Seretakis (2019), Hileman & Rauchs (2017), Grody (2018), Arthur et al. (2018), Furche & Sojli (2018), Lutz (2018), Thakor (2020), Nabilou (2019)
Bank of Canada	5	Chiu & Koeppl (2019), Chapman et al. (2017), Engert & Fung (2017), Fung & Halaburda (2016), Chiu et al. (2019)
BIS	5	BIS (2017), Bech & Garratt (2017), Cœuré & Loh (2018), Auer (2019), Auer & Böhme (2020)
BOE	4	Benos et al. (2017), Kumhof & Noone (2018), Barrdear & Kumhof (2016), Meaning et al. (2018)
FED	4	Mills et al. (2016), Andolfatto (2018), Berentsen & Schar (2018), Kahn & Wong (2019)
Emerald Publishing Limited	3	Yoo (2017), Lipton (2018), Kavassalis et al. (2018)
IMF	3	Griffoli et al. (2018), Khiaonarong & Humphrey (2019), Agur et al. (2019)
NBER	3	Raskin & Yermack (2018), Bordo & Levin (2017), Brunnermeier & Niepelt (2019)
SpringerLink	3	Guo & Liang (2016), **?**), Micheler & Whaley (2019)
EBI	2	Geva (2018), Didenko et al. (2020)
Taylor and Francis Group, LLC	2	Chiu (2017), Dow (2019)
ACM	1	Ben Dhaou & Rohman (2018)
ADBInstitute	1	Shirai (2019)
AISeL	1	Hayes (2016)
ArXiv	1	Danezis & Meiklejohn (2015)
BAFFI CAREFIN	1	Borgonovo et al. (2018)
Banque de France	1	Pfister (2019)
C.D. Howe Institute	1	Murray (2019)
CJEL	1	Caytas (2016)
ECB	1	Pinna & Ruttenberg (2016)
EQUILIBRIUM	1	Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)
MDPI	1	Cocco et al. (2017)
PwC	1	Koumbarakis & Dobrauz-Saldapenna (2019)
RBFL	1	Nabilou & Prum (2019)
ResearchGate	1	Shah & Jani (2018)
SAGE	1	Ducas & Wilner (2017)
SciELO	1	Rio & César (2017)
SCIENCE PRESS	1	Wu et al. (2019)
SNB	1	Kang & Lee (2019)
SSRN	1	Milne (2018)

Table B.6: Matrix of the Research. (Appendix E)

Reference ID	Publication Year	Literature Type	Evaluation	Validation	Solution Proposal	Philosophical Paper	Opinion Paper	Experience Paper	Protocol	Model	PoC	Framework	Taxonomy	New Knowledge	CBDC	PCS	Assets	Audit	Regulation
Hileman & Rauchs (2017)	2017	Journal / Magazine	*											*	*	*	*	*	*
Arthur et al. (2018)	2018	Journal / Magazine				*						*		*	*				*
Han et al. (2019)	2019	Conference Proceedings				*						*		*	*				*
Zhai & Zhang (2018)	2018	Conference Proceedings				*						*		*		*			
Didenko et al. (2020)	2020	Grey				*	*					*	*	*	*	*			*
Tsai, Zhao, Zhang, Yu & Deng (2018)	2018	Conference Proceedings	*						*	*				*	*				
Chiu (2017)	2017	Journal / Magazine					*							*		*			*
Wu et al. (2019)	2019	Journal / Magazine			*				*		*			*	*			*	*
Tsai et al. (2016)	2016	Conference Proceedings				*						*		*		*	*		*
Kavassalis et al. (2018)	2018	Journal / Magazine				*						*		*					*
Tsai, Deng, Ding & Li (2018)	2018	Conference Proceedings			*	*					*	*		*		*			*
Shah & Jani (2018)	2018	Journal / Magazine				*						*		*		*			*
Andolfatto (2018)	2018	Grey	*											*	*				
Geva (2018)	2018	Grey					*							*	*		*		
Cocco et al. (2017)	2017	Journal / Magazine					*							*					
Borgonovo et al. (2018)	2018	Journal / Magazine			*					*	*			*	*				
Nguyen (2016)	2016	Conference Proceedings				*							*	*					*
Guo & Liang (2016)	2016	Journal / Magazine					*							*		*			*
Yoo (2017)	2017	Journal / Magazine					*							*		*			
Ozili (2019)	2019	Journal / Magazine						*						*					*
Seretakis (2019)	2019	Journal / Magazine					*							*	*	*	*	*	*
Chen et al. (2018)	2018	Conference Proceedings			*	*					*	*		*			*	*	
Chiu & Koeppl (2019)	2018	Grey	*							*				*		*	*		
Lipton (2018)	2018	Journal / Magazine				*							*	*	*	*			
Meaning et al. (2018)	2018	Grey	*				*							*	*				*
Khiaonarong & Humphrey (2019)	2019	Grey			*	*	*						*	*	*				
Griffoli et al. (2018)	2018	Grey				*						*	*	*	*				
Bech & Garratt (2017)	2017	Grey				*	*						*	*	*				
Cœuré & Loh (2018)	2018	Grey				*		*				*	*	*	*	*			*
Kumhof & Noone (2018)	2018	Grey	*											*	*				
Fung & Halaburda (2016)	2016	Grey				*						*		*	*	*			*
Nabilou (2019)	2019	Journal / Magazine					*							*	*				*
Chiu et al. (2019)	2019	Grey	*							*				*	*				
Bordo & Levin (2017)	2017	Grey					*							*	*				*
Koumbarakis & Dobrauz-Saldapenna (2019)	2019	Grey				*	*					*		*	*				
Engert & Fung (2017)	2017	Grey				*	*	*				*		*	*				
Pfister (2019)	2019	Grey				*		*				*		*	*				*
Furche & Sojli (2018)	2018	Journal / Magazine					*							*	*				
Nabilou & Prum (2019)	2019	Journal / Magazine					*							*	*				*
Murray (2019)	2019	Grey					*							*	*				*
Danezis & Meiklejohn (2015)	2016	Conference Proceedings		*		*			*	*		*		*	*				
Lutz (2018)	2018	Journal / Magazine				*							*	*					
Milne (2018)	2017	Journal / Magazine				*	*					*		*		*	*	*	
Hayes (2016)	2016	Conference Proceedings				*	*					*		*					*
Agur et al. (2019)	2019	Grey	*			*				*		*		*	*				
Caytas (2016)	2016	Journal / Magazine					*							*		*			*
Raskin & Yermack (2018)	2016	Grey					*							*	*				
Pinna & Ruttenberg (2016)	2016	Grey				*	*						*	*		*		*	
Priem (2020)	2020	Journal / Magazine					*							*		*	*	*	*
BIS (2017)	2017	Grey				*	*					*	*	*		*	*	*	*
Mills et al. (2016)	2016	Grey				*	*	*					*	*		*	*	*	*
Auer (2019)	2019	Grey			*	*				*	*	*		*					*
Ben Dhaou & Rohman (2018)	2018	Conference Proceedings					*							*		*		*	*
Wu & Liang (2017)	2017	Conference Proceedings	*						*					*		*			
Thakor (2020)	2019	Journal / Magazine				*							*	*					
Wang et al. (2018)	2018	Conference Proceedings			*	*					*	*		*		*			
Dow (2019)	2019	Journal / Magazine					*							*	*				*
Shirai (2019)	2019	Grey				*	*						*	*	*				
Kang & Lee (2019)	2019	Conference Proceedings		*						*				*	*				
Sun et al. (2017)	2017	Conference Proceedings	*						*	*				*	*				
Brunnermeier & Niepelt (2019)	2019	Grey			*	*				*		*		*	*				
Chapman et al. (2017)	2017	Grey						*						*		*		*	*
Grody (2018)	2017	Journal / Magazine					*							*				*	*
Micheler & Whaley (2019)	2019	Journal / Magazine					*							*					*
Kahn & Wong (2019)	2018	Grey				*	*	*				*		*	*				*
Berentsen & Schar (2018)	2018	Grey				*	*						*	*	*				*
Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017)	2017	Journal / Magazine				*							*	*		*			
Barrdear & Kumhof (2016)	2016	Grey	*											*	*				
Ducas & Wilner (2017)	2017	Journal / Magazine					*							*		*	*	*	*
Auer & Böhme (2020)	2020	Grey				*							*	*	*				
Benos et al. (2017)	2017	Grey				*	*						*	*		*	*	*	*
Rio & César (2017)	2017	Journal / Magazine				*							*	*					

Appendix C

Technical Variables

C.1 Technical Benefits of Blockchain

C.2 Technical Limitations of Blockchain

Table C.1: Technological Benefits of Blockchain:

Technical Variable	**Opinion of the Researchers**
Access types	Permissionless Nguyen (2016), Didenko et al. (2020); Permissioned Chiu (2017), Tsai, Deng, Ding & Li (2018), Chapman et al. (2017), Chiu & Koeppl (2019), Lipton (2018), Milne (2018), Priem (2020), Mills et al. (2016), Wang et al. (2018), Sun et al. (2017), Benos et al. (2017), BIS (2017), Chapman et al. (2017), Kumhof & Noone (2018), Barrdear & Kumhof (2016), Auer & Böhme (2020), Didenko et al. (2020).
Verification Process	In the permissioned blockchains, an ultimate authoritative verification is needed to be performed by the trusted party, meaning that existing intermediaries, that perform such functions will still be required Chiu (2017).
Use of Smart Contracts (SC)	Automated modification to the rate of money creation via contingent SC could streamline monetary policy operations Raskin & Yermack (2018); SC based automation of 'priority' for payments could ease spot liquidity needs Hileman & Rauchs (2017); For securities markets: • could facilitate DvP Chiu (2017), Chiu & Koeppl (2019), Lipton (2018); • automatic execution and payment of certain derivatives Priem (2020), Didenko et al. (2020); • automate certain transfers based on pre-specified events agreed to by counterparties to a transaction Mills et al. (2016); • automate certain non-elective corporate actions Priem (2020); • automation of facilitation, execution, or enforcement of the performance of certain contract terms could significantly simplify back office processes and records management BIS (2017); • creation of dynamic transactional document - a close-to-real-time "digital doppelgänger" for each financial contract during its entire life span Kavassalis et al. (2018).
Network Effects	•A larger DLT-based settlement network could allow users to settle trades with more counterparties Benos et al. (2017), thus increasing network externalities; • Network resilience is provided through distributed data management Mills et al. (2016), where having a distributed database enables faster recovery, as well as protection at the system level Hileman & Rauchs (2017) in the event of cyberattack or failure Benos et al. (2017).
Data Provenance Traceability and Transparency on Blockchain	• Provides data auditability and immutability and higher level of transparency Priem (2020), Mills et al. (2016), Kavassalis et al. (2018), Didenko et al. (2020); •Transparent nature of global audit log on blockchain enables traceability of anything represented on the ledger, preventing manipulation through the public auditability of the system Hileman & Rauchs (2017), Kavassalis et al. (2018); •Offers immutable, tamper-resistant transactional records Milne (2018), Sun et al. (2017), Kavassalis et al. (2018), Didenko et al. (2020), by ensuring a high cost for dishonest actions Chiu & Koeppl (2019); •Reduces the possibility of data falsification and manipulation Ben Dhaou & Rohman (2018), Didenko et al. (2020) lowering the risk of fraud and enhance resolution management capabilities Ducas & Wilner (2017); •Ensures immutable records of the history of any flow of funds or securities, with traceable amendments, where the integrity of the records is ensured by the integrity of the ledger itself Benos et al. (2017), Didenko et al. (2020); •Immutability of data recorded in the ledger is crucial to the safety as it relates to data integrity BIS (2017), Didenko et al. (2020); •Increased traceability of records stored on the ledger though consensus mechanism, which ensures who can change records and how Chapman et al. (2017); •Traceability could further be provided via reliance on the user account address protocol, where a central bank could separate the user's identity and transaction information Sun et al. (2017); •Traceability could be utilised for compliance with KYC rules, AML requirements and CTF regulations BIS (2017), Didenko et al. (2020); Overall data management costs reduction Benos et al. (2017).
Use of Oracles	•Computer servlets, that are programmed to store data feeds externally – Oracles - could be utilised for automation of validation of user-provided information which could then be added to a blockchain ledger Priem (2020) •Establishment of Oracles as a DC (Data Center) layer could promote stronger supervision Sun et al. (2017)
Point-to-point Data Transmission	•An opportunity to improve, or illuminate data reconciliation processes of shared transactional data between financial institutions that could reduce data discrepancy, facilitate quicker reconciliation, improve or remove burdensome back office activities BIS (2017), Guo & Liang (2016), Benos et al. (2017) and refocus many of the hundreds of data intermediaries and financial market utilities that play a significant role in reconciling risk prone and costly standard and non-standard data Grody (2018); •For example, the inter-bank payments require reconciliation between the different databases Wu & Liang (2017). That will no longer be required, as the blockchain consensus algorithm could become a single, authoritative general ledger Wu & Liang (2017); •These will simplify operational processes and reduce the number of financial intermediaries or illuminate some (32), as DLT could eliminate the need for centrally maintaining back-up systems Benos et al. (2017); •Decentrality (44), or disintermediation of third-party settlement could lead to cost reductions Yoo (2017), ?), MacDonald et al. (2016), Pinna & Ruttenberg (2016), Priem (2020), Mills et al. (2016), Wu & Liang (2017), Hileman & Rauchs (2017), Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017), Ducas & Wilner (2017), Benos et al. (2017), Bech & Garratt (2017), as well as decreasing some of the risks Wu & Liang (2017), Mills et al. (2016), Hileman & Rauchs (2017).
Increased Automation	A key feature of DLT technology is its programmability to automate certain functions BIS (2017), that allows for: • straight-through processing of transactions Chiu (2017); • automatic recording of transaction from different locations combined with secure and cost-effective data storing solutions Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017); • automation of clearing and simplification and automation many of the back-office processes currently involved in the post-trade cycle Benos et al. (2017), Chapman et al. (2017).
Enhanced operational Efficiency through Faster Processing	•Increased operational efficiency for settlement for cross-bank or cross-border transfers Benos et al. (2017), Bech & Garratt (2017), Zhai & Zhang (2018), could reduce complexity Mills et al. (2016) and improve end-to end operational speed Tsai, Deng, Ding & Li (2018), Yoo (2017), Priem (2020), Mills et al. (2016), by, e.g., reducing duration of settlement cycle Benos et al. (2017), Chapman et al. (2017), to near-real time Milne (2018), Wu & Liang (2017), Chapman et al. (2017), or even to 24/7/365 processing Jantoń-Drozdowska & Mikołajewicz-Woźniak (2017); •Faster transfers suggest that financial market participants will receive their funds and securities more quickly, freeing up liquidity that could be tied up in collateralBIS (2017).
DLT as the Platform for CBDC	This is the most widely investigated use-case by central banks to use blockchain as the underlying platform to launch their own CBDC Hileman & Rauchs (2017), Sun et al. (2017).

Table C.2: Technological Limitations of Blockchain:

Technical Variable	**Opinion of the Researchers**
Still Evolving Technology	Currently, not one central bank has implemented a "live", operational system based on blockchain, because the technology is still "maturing" Kahn & Wong (2019), Kumhof & Noone (2018), Hileman & Rauchs (2017), **?**), Murray (2019) it still has not been properly evaluated or tested Ducas & Wilner (2017), Hileman & Rauchs (2017), Chiu (2017), Pinna & Ruttenberg (2016), Ben Dhaou & Rohman (2018), Benos et al. (2017), Didenko et al. (2020), meaning that the impact of its adaptations is still uncertain Meaning et al. (2018), Engert & Fung (2017), Berentsen & Schar (2018) and further analysis of technological feasibility and operational costs is needed Griffoli et al. (2018), Cœuré & Loh (2018), Didenko et al. (2020).
Network Effects	Difficulties in building new participant networks because of reluctance to change already established complex business processes Nguyen (2016), combined with reluctance to give up various degrees of existing control and re-alignment of incentives of different participants Mills et al. (2016), Hileman & Rauchs (2017).
Interoperability	Lack of interoperability between blockchains and legacy financial market infrastructure and/or between numerous niche blockchain protocols is an important issue that creates fragmentation, friction and raised costs due to increased complexity of connections to and use of different systems, leading to more operational and systemic risks Chiu (2017), Priem (2020), Mills et al. (2016), Hileman & Rauchs (2017), Ducas & Wilner (2017), Benos et al. (2017).
Smart Contracts (SC)	• There are challenges with self-executing SC, as those are not immune to faulty or malicious code that could cause adverse and unpredictable behavioural patterns in the financial ecosystem BIS (2017) and latency in correction of errors; • Interdependencies between SC could result in a transmission channel for unforeseen risks BIS (2017); • The legal status and nature of SC is not defined and is unclear BIS (2017), Seretakis (2019), Priem (2020), Mills et al. (2016), Geva (2018).
P2P	There are significant drawbacks with P2P networks: • the design goal of a P2P network (t avoid all regulations) is in direct contradiction to a principal design goal for any financial systems with compulsory regulation Tsai et al. (2016); • the multipath connection inherent in a P2P network creates barriers to regulation Tsai et al. (2016); • difficult to monitor and control P2P applications as operations may be autonomous and decentralised Tsai et al. (2016); • irregularities such as copyright infringement and security leaks Tsai et al. (2016); • each node in a P2P network serves as a client as well as a server, the performance of a P2P network is inherently slower than a regular network Tsai et al. (2016).
Cost of Processing	Significant cost associated with running public permissionless blockchains in terms of network bandwidth, storage and processing power Ducas & Wilner (2017), BIS (2017), Kavassalis et al. (2018)
Scalability and Operational Capacity	There are significant concerns with operational capacity and scalability of blockchain's ability to process large volumes of transactions on a daily basis or handling unexpected pick volumes in times of market volatility – a crucial requirement of current PSC systems BIS (2017), Ben Dhaou & Rohman (2018), Mills et al. (2016), Hileman & Rauchs (2017), Murray (2019). Only limited numbers of simultaneous transactions can be written in a block due to set limits on a block size Ducas & Wilner (2017), Geva (2018) and the consensus mechanism used BIS (2017).
Consensus Algorithms	Current consensus algorithms, combined with cryptographic verification and validation (44), from a technical point of view, introduce latency though complexity and limit the number of transactional transfers to financial data processing, when compared to existing PCS systems Ben Dhaou & Rohman (2018), Mills et al. (2016), Benos et al. (2017)
Mining	In a permissioned blockchain, mining is seen as irrelevant; participating banks maintain those ledgers and thus miners and the mining process are not needed Tsai et al. (2016) and seen as a dead-weight cost Chiu & Koeppl (2019)
Immutability	Irreversibility of events born from immutability of blockchain is an important issue, as it prohibits error handling, transactions reversal in case of fraud, technological misuse or other events and hampers maintenance Ben Dhaou & Rohman (2018), Hileman & Rauchs (2017), Benos et al. (2017), Chiu (2017), Mills et al. (2016), BIS (2017), Didenko et al. (2020).

Bibliography

ACCA (2020), 'Acca integrated report 2020'. Accessed: 2024-10-05.

AccountingTools (2021), 'Financial accounting basics'.

Agur, I., Ari, A. & Dell'Ariccia, G. (2019), Designing central bank digital currencies, Technical report, IMF Working Paper No. 19/252.
URL: *https://ssrn.com/abstract=3523125*

Al-Debei, M. & Avison, D. (2010), 'Developing a unified framework of the business model concept', *European Journal of Information Systems* **19**, 359–376.

Andersen, N. (2016), 'Blockchain technology a game-changer in accounting?', Deloitte, 1–5.

Andolfatto, D. (2018), Assessing the impact of central bank digital currency on private banks, Technical Report 2018-25, FRB St. Louis.

Antonopoulos, A. M. (2014), *Mastering Bitcoin: Unlocking Digital Cryptocurrencies*, O'Reilly Media.
URL: *https://archive.org/details/masteringbitcoin0000anto*

Arthur, M., Treccani, A. & Moyano, J. P. (2018), A 9-dimension grid for the evaluation of central bank digital currencies, Technical Report 1, University of Zurich.

Auer, R. (2019), 'Embedded supervision: how to build regulation into blockchain finance', *Globalization and Monetary Policy Institute Working Paper* (371).

Auer, R. & Böhme, R. (2020), 'The technology of retail central bank digital currency', *BIS Quarterly Review* .

Bains, S. (2019), *Explaining the Future: How to Research, Analyze, and Report on Emerging Technologies*, Oxford University Press.

Bank of England (2009*a*), 'Asset purchase facility', `https://www.bankofengland.co.uk/-/media/boe/files/news/2009/february/asset-purchase-facility.pdf`. Accessed: 2024-03-29.

Bank of England (2009*b*), 'Asset purchase facility quarterly report - 2009 q1', `https://www.bankofengland.co.uk/asset-purchase-facility/2009/2009-q1`. Accessed: 2024-03-29.

Bank of England (2009*c*), 'Governor letter - 29 january 2009', `https://www.bankofengland.co.uk/-/media/BoE/Files/letter/2009/governor-letter-290109.pdf`. Accessed: 2024-03-29.

Bank of England (2009*d*), 'Market notice 24 april 2009', `https://webarchive.nationalarchives.gov.uk/ukgwa/20100404173712mp_/http://www.bankofengland.co.uk/markets/marketnotice090424.pdf`. Accessed: 2024-03-29.

Bank of England (2010*a*), 'Archived data. monetary financial institutions' balance sheets, income, and expenditure'.

Bank of England (2010*b*), 'Bank of england annual report and accounts - 2010', `https://www.bankofengland.co.uk/annual-report/2010`. Accessed: 2024-03-29.

Bank of England (2010*c*), 'Settlement procedures for the asset purchase facility', `https://webarchive.nationalarchives.gov.uk/ukgwa/20100404181654mp_/http://www.bankofengland.co.uk/markets/apf/settlementprocedures.pdf`. Accessed: 2024-03-29.

Bank of England (2023*a*), 'Bank of england market operations guide'.
URL: *https://www.bankofengland.co.uk/markets/bank-of-england-market-operations-guide*

Bank of England (2023*b*), 'Bank of england statistics tables'.

Bank of England (2024*a*), 'Operating procedures for the asset purchase facility'.

Bank of England (2024*b*), 'Terms and conditions of the asset purchase facility'.

Bank of England (Year), Asset purchase facility quarterly report, Technical report.

Barontini, C. & Holden, H. (2019), Proceeding with caution-a survey on central bank digital currency, BIS Paper 101, Bank for International Settlements (BIS).

Barrdear, J. & Kumhof, M. (2016), 'The macroeconomics of central bank issued digital currencies'. Accessed: Add the access date here.

Bech, M. L. & Garratt, R. (2017), 'Central bank cryptocurrencies', *BIS Quarterly Review September* .

Beller, M. & Hejderup, J. (2019), Blockchain-based software engineering, *in* '2019 IEEE/ACM 41st International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER)', IEEE, pp. 53–56.

Ben Dhaou, S. I. & Rohman, I. K. (2018), Everything and its opposite: Socio-economic implications of blockchain technology: Case of monetary policy, *in* 'Proceedings of the 11th International Conference on Theory and Practice of Electronic Governance', pp. 631–639.

Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E. & Virza, M. (2014), Zerocash: Decentralized anonymous payments from bitcoin, *in* '2014 IEEE Symposium on Security and Privacy', IEEE, pp. 459–474.

Benos, E., Garratt, R. & Gurrola-Perez, P. (2017), 'The economics of distributed ledger technology for securities settlement'. Accessed: Add the access date here.

Berentsen, A. & Schar, F. (2018), 'The case for central bank electronic money and the non-case for central bank cryptocurrencies'. Accessed: Add the access date here.

Besançon, L., Da Silva, C. F. & Ghodous, P. (2019), Towards blockchain interoperability: Improving video games data exchange, *in* '2019 IEEE international conference on blockchain and cryptocurrency (ICBC)', IEEE, pp. 81–85.

Bird, C. (2010), 'Conway's law', *Making Software: What Really Works, and Why We Believe It* p. 187.

BIS (2017), Distributed ledger technology in payment, clearing and settlement - an analytical framework, Technical Report 157, Bank for International Settlements.
URL: *https://www.bis.org/cpmi/publ/d157.pdf*

Bonsón, E. & Bednárová, M. (2019), 'Blockchain and its implications for accounting and auditing', *Meditari Accountancy Research* **27**(5), 725–740.

Bonson, E. & Michaela, B. (2019), 'Blockchain and its implications for accounting and auditing', *Meditari Accountancy Research* **27**(5), 725–740.

Bordo, M. D. & Levin, A. T. (2017), Central bank digital currency and the future of monetary policy, Tech. rep., National Bureau of Economic Research.

Borgonovo, E., Cillo, A., Caselli, S. & Masciandaro, D. (2018), Between cash, deposit and bitcoin: Would we like a central bank digital currency? money demand and experimental economics, Technical Report 2018-75, BAFFI CAREFIN Centre Research Paper.

Brown, R. G. (2018), The corda platform: An introduction, Technical report, R3. Accessed: date-of-access.
URL: *https://www.corda.net/content/corda-platform-whitepaper.pdf*

Brown, R. G., Carlyle, J., Grigg, I. & Hearn, M. (2016), 'Corda: An introduction', `https://www.r3.com/reports/corda-an-introduction/`.

Brunnermeier, M. K. & Niepelt, D. (2019), On the equivalence of private and public money, Technical report, National Bureau of Economic Research.

Castro, M., Liskov, B. et al. (1999), Practical byzantine fault tolerance, *in* 'OSDI', Vol. 99, pp. 173–186.

Casu, B., Girardone, C. & Molyneux, P. (2006), *Introduction to Banking*, Vol. 10, Pearson Education.

Caytas, J. (2016), 'Developing blockchain real-time clearing and settlement in the eu, us, and globally', *Columbia Journal of European Law: Preliminary Reference (June 22, 2016)* .

Centobelli, P., Roberto, C., Pasquale, D. V., Eugenio, O. & Giustina, S. (2022), 'Blockchain technology design in accounting: Game changer to tackle fraud or technological fairy tale?', *Accounting, Auditing and Accountability Journal* **35**(7), 1566–1597.

Chapman, J., Garratt, R., Hendry, S., McCormack, A. & McMahon, W. (2017), 'Project jasper: Are distributed wholesale payment systems feasible yet', *Financial System* **59**.

Chen, B., Tan, Z. & Fang, W. (2018), Blockchain-based implementation for financial product management, *in* '2018 28th international telecommunication networks and applications conference (ITNAC)', IEEE, pp. 1–3.

Chiu, I. H. (2017), 'A new era in fintech payment innovations? a perspective from the institutions and regulation of payment systems', *Law, Innovation and Technology* **9**(2), 190–234.

Chiu, J., Davoodalhosseini, S. M., Jiang, J. H. & Zhu, Y. (2019), 'Central bank digital currency and banking', Available at SSRN 3331135.

Chiu, J. & Koeppl, T. V. (2019), 'Blockchain-based settlement for asset trading', *The Review of Financial Studies* **32**(5), 1716–1753.

Christidis, K. & Devetsikiotis, M. (2016), 'Blockchains and smart contracts for the internet of things', *IEEE access* **4**, 2292–2303.

Cocco, L., Pinna, A. & Marchesi, M. (2017), 'Banking on blockchain: Costs savings thanks to the blockchain technology', *Future Internet* **9**(3), 25.

Companies House (2023*a*), 'Bank of england asset purchase facility fund limited. company number 06806063. overview'.
URL: *https://find-and-update.company-information.service.gov.uk/company/06806063*

Companies House (2023*b*), 'Filing history for bank of england asset purchase facility fund limited (company number 06806063)'.
URL: *https://find-and-update.company-information.service.gov.uk/company/06806063/filing-history*

Companies House (2024*a*), 'Companies house', `https://www.gov.uk/government/organisations/companies-house`. Accessed: 2024-03-29.

Companies House (2024*b*), 'Companies house: Apf incorporation documentation. filing history for company number 06806063', `https://find-and-update.company-information.service.gov.uk/company/06806063/filing-history`.

Companies House (2024*c*), 'Company registrar's rules and powers', `https://www.gov.uk/government/publications/company-registrars-rules-and-powers`. Accessed: 2024-03-29.

Companies House (2024*d*), 'Guidance. company registers', `https://www.gov.uk/government/publications/company-registers`. Accessed: 2024-03-29.

Corporate Finance Institute (2021), 'Three financial statements'.

Cœuré, B. & Loh, J. (2018), 'Central bank digital currencies'. Accessed: Add the access date here.
URL: *https://www.bis.org/cpmi/publ/d174.pdf*

Danezis, G. & Meiklejohn, S. (2015), 'Centrally banked cryptocurrencies'.
URL: *http://arxiv.org/abs/1505.06895*

Dashkevich, N., Counsell, S. & Destefanis, G. (2020), 'Blockchain application for central banks: A systematic mapping study', *IEEE Access* **8**, 139918–139952.

David, Y. (December 2015), Corporate governance and blockchains, Technical Report 21802, National Bureau of Economic Research.

Didenko, A. N., Zetzsche, D. A., Arner, D. W. & Buckley, R. P. (2020), 'After libra, digital yuan and covid-19: Central bank digital currencies and the new world of money and payment systems'.

Domanski, D., Gambacorta, L. & Picillo, C. (2015), 'Central clearing: trends and current issues', *BIS Quarterly Review* .

Dow, S. (2019), 'Monetary reform, central banks, and digital currencies', *International Journal of Political Economy* **48**(2), 153–173.

Ducas, E. & Wilner, A. (2017), 'The security and financial implications of blockchain technologies: Regulating emerging technologies in canada', *International Journal* **72**(4), 538–562.

Engert, W. & Fung, B. S. (2017), 'Central bank digital currency: Motivations and implications'. Accessed: Add the access date here.

European Central Bank and Bank of Japan (2018), Securities settlement systems: Delivery-versus-payment in a distributed ledger environment, Technical report, Project STELLA.

Evans, E. (2004), *Domain-driven design: tackling complexity in the heart of software*, Addison-Wesley Professional.

Form IN10. Register a private or public company. Application to register a company (2024), `https://www.gov.uk/government/publications/register-a-private-or-public-company-in01`. [Online].

Fung, B. S. & Halaburda, H. (2016), 'Central bank digital currencies: a framework for assessing why and how', Available at SSRN 2994052.

Furche, A. & Sojli, E. (2018), 'Central bank issued digital cash'. Accessed: Add the access date here.
URL: *http://dx.doi.org/10.2139/ssrn.3213028*

Garanina, T., Ranta, M. & Dumay, J. (2022), 'Blockchain in accounting research: current trends and emerging topics', *Accounting, Auditing & Accountability Journal* **35**(7), 1507–1533.

Garay, J. A., Kiayias, A. & Leonardos, N. (2015), The bitcoin backbone protocol: Analysis and applications, *in* 'Annual International Conference on the Theory and Applications of Cryptographic Techniques', Springer, pp. 281–310.

Garousi, V., Felderer, M. & Mäntylä, M. V. (2016), The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature, *in* 'Proceedings of the 20th international conference on evaluation and assessment in software engineering', pp. 1–6.

Garousi, V., Felderer, M. & Mäntylä, M. V. (2019), 'Guidelines for including grey literature and conducting multivocal literature reviews in software engineering', *Information and software technology* **106**, 101–121.

Geerts, G. L. (2011), 'A design science research methodology and its application to accounting information systems research', *International journal of accounting Information Systems* **12**(2), 142–151.

Geva, B. (2018), 'Banking in the digital age-who is afraid of payment disintermediation?'. Accessed: Add the access date here.

Golub, A., Grossmass, L. & Poon, S. (2018), 'Ultra short tenor yield curves for high-frequency trading and blockchain settlement'.
URL: *https://ssrn.com/abstract=3133024*

Griffoli, M. T. M., Peria, M. M. S. M., Agur, M. I., Ari, M. A., Kiff, M. J., Popescu, M. A. & Rochon, M. C. (2018), 'Casting light on central bank digital currencies'. Accessed: Add the access date here.

Grody, A. D. (2018), 'Rebuilding financial industry infrastructure', *Journal of Risk Management in Financial Institutions* **11**(1), 34–46.

Guo, Y. & Liang, C. (2016), 'Blockchain application and outlook in the banking industry', *Financial Innovation* **2**(1), 24.

Han, X., Yuan, Y. & Wang, F. (2019), A blockchain-based framework for central bank digital currency, *in* 'IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)', pp. 263–268.

Hans, B. (2019), 'Blockchains, real-time accounting, and the future of credit risk modeling', *Ledger* **4**.

Hassani, H., Huang, X. & Silva, E. (2018), 'Banking with blockchain-ed big data', *Journal of Management Analytics* **5**(4), Taylor and Francis.

Hayes, A. (2016), *Decentralized banking: Monetary technocracy in the digital age*, Springer.

Hearn, M. & Brown, R. G. (2016), 'Corda: A distributed ledger', *Corda Technical White Paper* **2016**, 6.

Hileman, G. & Rauchs, M. (2017), '2017 global blockchain benchmarking study', *Available at SSRN 3040224* .

Holotiuk, F., Pisani, F. & Moormann, J. (2017), The impact of blockchain technology on business models in the payments industry, *in* 'Wirtschaftsinformatik 2017 Proceedings'.
URL: *https://aisel.aisnet.org/wi2017/track09/paper/6*

Hyperledger Fabric – Hyperledger (n.d.), `https://www.hyperledger.org/projects/fabric`. Accessed: Add the access date here.

James, K. L., Randall, N. P. & Haddaway, N. R. (2016), 'A methodology for systematic mapping in environmental sciences', *Environmental evidence* **5**, 1–13.

Jantoń-Drozdowska, E. & Mikołajewicz-Woźniak, A. (2017), 'The impact of the distributed ledger technology on the single euro payments area development', *Equilibrium. Quarterly Journal of Economics and Economic Policy* **12**(3), 519–535.

Judmayer, A., Stifter, N., Krombholz, K. & Weippl, E. (2022), *Blocks and chains: introduction to bitcoin, cryptocurrencies, and their consensus mechanisms*, Springer Nature.

Kahn, C. M. & Wong, T.-N. (2019), 'Should the central bank issue e-money?', *Money* pp. 01–18.

Kang, K.-Y. & Lee, S. (2019), 'Money, cryptocurrency, and monetary policy'. Accessed: Add the access date here.
URL: *http://dx.doi.org/10.2139/ssrn.3303595*

Kavassalis, P., Stieber, H., Breymann, W., Saxton, K. & Gross, F. J. (2018), 'An innovative regtech approach to financial risk monitoring and supervisory reporting', *The Journal of Risk Finance* **19**(1), 39–55.

Keele, S. et al. (2007), 'Guidelines for performing systematic literature reviews in software engineering'.

Khiaonarong, M. T. & Humphrey, D. (2019), 'Cash use across countries and the demand for central bank digital currency'. Accessed: Add the access date here.

Kitchenham, B. A., Budgen, D. & Brereton, O. P. (2010), The value of mapping studies–a participant-observer case study, *in* '14th international conference on evaluation and assessment in software engineering (ease)', BCS Learning & Development.

Kitchenham, B., Brereton, P. & Budgen, D. (2010), The educational value of mapping studies of software engineering literature, *in* 'Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1', pp. 589–598.

Koumbarakis, A. & Dobrauz-Saldapenna, G. (2019), 'Central bank digital currency: Benefits and drawbacks', *SSRN Electronic Journal* .

Kumhof, M. & Noone, C. (2018), 'Central bank digital currencies-design principles and balance sheet implications'. Accessed: Add the access date here.

Lao, L., Li, Z., Hou, S., Xiao, B., Guo, S. & Yang, Y. (2020), 'A survey of iot applications in blockchain systems: Architecture, consensus, and traffic modeling', *ACM Computing Surveys (CSUR)* **53**(1), 1–32.

Lipton, A. (2018), 'Blockchains and distributed ledgers in retrospective and perspective', *The Journal of Risk Finance* **19**(1), 4–25.

Lu, H., Jajoo, A. & Namjoshi, K. S. (2024), 'Atomicity and abstraction for cross-blockchain interactions', *arXiv preprint arXiv:2403.07248* .

Lutz, J. K. (2018), 'Coexistence of cryptocurrencies and central bank issued fiat currencies-a systematic literature review'.

MacDonald, T., Allen, D. & Potts, J. (2016), 'Blockchains and the boundaries of self-organized economies: Predictions for the future of banking'.
URL: *https://ssrn.com/abstract=2749514*

Magretta, J. (2002), 'Why business models matter', *Harvard Business Review* .
URL: *https://hbr.org/2002/05/why-business-models-matter*

Manda, V. K. & SS, P. R. (2018), Blockchain technology for the mutual fund industry, *in* 'National Seminar on Paradigm Shifts in Commerce and Management', pp. 12–17.

McCallig, J., Robb, A. & Rohde, F. (2019), 'Establishing the representational faithfulness of financial accounting information using multiparty security, network analysis and a blockchain', *International Journal of Accounting Information Systems* **33**, 47–58.

Meaning, J., Dyson, B., Barker, J. & Clayton, E. (2018), 'Broadening narrow money: monetary policy with a central bank digital currency'. Accessed: Add the access date here.

Mei, F., Ge, W., Luo, S. & Shevlin, T. (2011), 'Why do cfos become involved in material accounting manipulations?', *Journal of Accounting and Economics* **51**(1-2), 21–36.

Micheler, E. & Whaley, A. (2019), 'Regulatory technology: Replacing law with computer code', *European Business Organization Law Review* pp. 1–29.

Mills, D. C., Wang, K., Malone, B., Ravi, A., Marquardt, J., Badev, A. I., Brezinski, T., Fahy, L., Liao, K. & et al., V. K. (2016), 'Distributed ledger technology in payments, clearing, and settlement'. Accessed: Add the access date here.

Milne, A. (2018), Cryptocurrencies from an austrian perspective, *in* 'Banking and Monetary Policy from the Perspective of Austrian Economics', Springer, pp. 223–257.

Mingming, T. (2020), Research on the application of blockchain technology in accounting information system, *in* '2020 International Conference on Virtual Reality and Intelligent Systems (ICVRIS)', IEEE, pp. 330–334.

Mohanty, D., VorugantI, N., Patel, C. & Manglani, T. (2023), 'Implementing blockchain technology for fraud detection in financial management', *BioGecko* **12**, 2.

Moll, J. & Yigitbasioglu, O. (2019), 'The role of internet-related technologies in shaping the work of accountants: New directions for accounting research', *The British accounting review* **51**(6), 100833.

Murray, J. (2019), 'Central banks and the future of money', *CD Howe Institute Commentary* **540**.

Nabilou, H. (2019), 'Central bank digital currencies: Preliminary legal observations', Available at SSRN. Accessed: Add the access date here.

Nabilou, H. & Prum, A. (2019), Central banks and regulation of cryptocurrencies, Technical report, University of Luxembourg Law Working Paper No. 2019-014.
URL: *https://ssrn.com/abstract=3421417*

Nadine, R. (2017), 'Do we really want blockchain-based accounting? decentralized consensus as enabler of management override of internal controls'.

Nakamoto, S. (2008), 'Bitcoin: A Peer-to-Peer Electronic Cash System'. [Online]. Available.
URL: *https://bitcoin.org/bitcoin.pdf*

NetSuite (2022), 'Accounting cycle'.

Nguyen, Q. K. (2016), Blockchain-a financial technology for future sustainable development, *in* '2016 3rd International conference on green technology and sustainable development (GTSD)', IEEE, pp. 51–54.

Oladejo, M. T. & Jack, L. (2020), 'Fraud prevention and detection in a blockchain technology environment: challenges posed to forensic accountants', *International Journal of Economics and Accounting* **9**(4), 315–335.

Ortiz, G. (2009), Issues in the governance of central banks a report from the central bank governance group, Technical report, Bank for International Settlements - Central Bank Governance Group.

Ozili, P. K. (2019), Blockchain finance: Questions regulators ask, *in* 'Disruptive Innovation in Business and Finance in the Digital World', International Finance Review, Vol. 20, Emerald Publishing Limited, pp. 123–129.

Peffers, K., Tuunanen, T., Rothenberger, M. A. & Chatterjee, S. (2007), 'A design science research methodology for information systems research', *Journal of management information systems* **24**(3), 45–77.

Peters, G. & Panayi, E. (2016), Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money, Springer International Publishing, pp. 239–278.

Petersen, K., Feldt, R., Mujtaba, S. & Mattsson, M. (2008), Systematic mapping studies in software engineering, *in* '12th international conference on evaluation and assessment in software engineering (EASE)', BCS Learning & Development.

Petersen, K., Vakkalanka, S. & Kuzniarz, L. (2015), 'Guidelines for conducting systematic mapping studies in software engineering: An update', *Information and software technology* **64**, 1–18.

Pfister, C. (2019), 'Central bank digital currency: One, two or none?'. Accessed: Add the access date here.

Pinna, A. & Ruttenberg, W. (2016), Distributed ledger technologies in securities post-trading revolution or evolution?, Technical Report 172, ECB Occasional Paper.

Priem, R. (2020), 'Distributed ledger technology for securities clearing and settlement: benefits, risks, and regulatory implications', *Financial Innovation* **6**(1), 11.

PwC (2021), 'Global economic crime and fraud survey 2021'. Accessed: October 5, 2024.

R3 (2023), 'Corda flow framework'. [Online]. Available.
URL: *https://docs.r3.com/en/platform/corda/4.9/enterprise/cordapps/api-flows.html*

Raskin, M. & Yermack, D. (2018), Digital currencies, decentralized ledgers and the future of central banking, *in* 'Research Handbook on Central Banking', Edward Elgar Publishing.

Richards, M. & Ford, N. (2020), *Fundamentals of software architecture: an engineering approach*, O'Reilly Media.

Rio, D. & César, A. (2017), 'Use of distributed ledger technology by central banks: A review', *Enfoque Ute* **8**(5), 1–13.

Rohrer, E., Tschorsch, F. & Scheuermann, B. (2017), 'An analysis of utxo-based blockchain systems', *Transactions on Emerging Telecommunications Technologies* **28**(8).

Rückeshäuser, N. (2017), 'Do we really want blockchain-based accounting? decentralized consensus as enabler of management override of internal controls'.

Sage50 (2023), 'Sage accountants'.
URL: *https://www.sage.com/en-gb/accountants-1/*

Schmitz, J. & Leoni, G. (2019), 'Accounting and auditing at the time of blockchain technology: a research agenda', *Australian Accounting Review* **29**(2), 331–342.

Secinaro, S., Dal Mas, F., Brescia, V. & Calandra, D. (2021), 'Blockchain in the accounting, auditing and accountability fields: a bibliometric and coding analysis', *Accounting, Auditing & Accountability Journal* **35**(9), 168–203.

Seretakis, A. L. (2019), 'Blockchain, securities markets and central banking'. Accessed: Add the access date here.

Shah, T. & Jani, S. (2018), 'Applications of blockchain technology in banking and finance'.

Shirai, S. (2019), 'Money and central bank digital currency'. Accessed: Add the access date here.

Stolowy, H. & Ding, Y. (2019), *Financial Accounting and Reporting: A Global Perspective*, 5th edn, Cengage Learning EMEA, United Kingdom.

Sun, H., Mao, H., Bai, X., Chen, Z., Hu, K. & Yu, W. (2017), Multi-blockchain model for central bank digital currency, *in* '2017 18th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)', IEEE, pp. 360–367.

Swan, M. (2015), *Blockchain: Blueprint for a new economy*, " O'Reilly Media, Inc.".

Thakor, A. V. (2020), 'Fintech and banking: What do we know?', *Journal of financial intermediation* **41**, 100833.

Tinn, K. (2018), 'Smart contracts and external financing'.
URL: *https://ssrn.com/abstract=3072854*

Tsai, W.-T., Blower, R., Zhu, Y. & Yu, L. (2016), A system view of financial blockchains, *in* '2016 IEEE Symposium on Service-Oriented System Engineering (SOSE)', IEEE, pp. 450–457.

Tsai, W.-T., Deng, E., Ding, X. & Li, J. (2018), Application of blockchain to trade clearing, *in* '2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)', IEEE, pp. 154–163.

Tsai, W.-T., Zhao, Z., Zhang, C., Yu, L. & Deng, E. (2018), A multi-chain model for cbdc, *in* '2018 5th International Conference on Dependable Systems and Their Applications (DSA)', IEEE, pp. 25–34.

UK Government (2020), 'Financial support for businesses during coronavirus (covid-19)'.
URL: *https://www.gov.uk/government/collections/financial-support-for-businesses-during-coronavirus-covid-19*

UK Government (2021*a*), 'Measuring error and fraud in the covid-19 schemes'.

UK Government (2021*b*), 'New hotline launched to report covid fraudsters'.
URL: *https://www.gov.uk/government/news/new-hotline-launched-to-report-covid-fraudsters*

UK Government (2023), 'File your company annual accounts'.

Walker, M. (2017), 'Bridging the gap between investment banking architecture and distributed ledgers'.
URL: *https://ssrn.com/abstract=2939281*

Wang, X., Xu, X., Feagan, L., Huang, S., Jiao, L. & Zhao, W. (2018), Inter-bank payment system on enterprise blockchain platform, *in* '2018 IEEE 11th International Conference on Cloud Computing (CLOUD)', IEEE, pp. 614–621.

Wieringa, R. J. (2014), *Design science methodology for information systems and software engineering*, Springer.

Wieringa, R., Maiden, N., Mead, N. & Rolland, C. (2006), 'Requirements engineering paper classification and evaluation criteria: a proposal and a discussion', *Requirements engineering* **11**, 102–107.

Wirtz, B. W., Pistoia, A., Ullrich, S. & Otte, V. G. (2016), 'Business models: Origin, development and future research perspectives', *Long Range Planning* .

Wohlin, C. (2014), Guidelines for snowballing in systematic literature studies and a replication in software engineering, *in* 'Proceedings of the 18th international conference on evaluation and assessment in software engineering', pp. 1–10.

Wu, T. & Liang, X. (2017), Exploration and practice of inter-bank application based on blockchain, *in* '2017 12th International Conference on Computer Science and Education (ICCSE)', IEEE, pp. 219–224.

Wu, Y., Fan, H., Wang, X. & Zou, G. (2019), 'A regulated digital currency', *Science China Information Sciences* **62**(3), 32109.

Yoo, S. (2017), 'Blockchain based financial case analysis and its implications', *Asia Pacific Journal of Innovation and Entrepreneurship* **11**(3), 312–321.

Your Company Formations (2024), 'A guide to share certificates'. Accessed: 2024-09-28.
URL: *https://www.yourcompanyformations.co.uk/blog/a-guide-to-share-certificates*

Zhai, X. & Zhang, C. (2018), A cash flow blockchain based privacy-preserving, *in* '2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS)', IEEE, pp. 809–813.