SEMANTIC DISCOVERY AND REUSE OF BUSINESS PROCESS PATTERNS

A thesis submitted towards the degree of

Doctor of Philosophy

by

Laden Aldin

School of Information Systems, Computing and Mathematics

Brunel University

September 2010

ABSTRACT

In modern organisations business process modelling has become fundamental due to the increasing rate of organisational change. As a consequence, an organisation needs to continuously redesign its business processes on a regular basis. One major problem associated with the way business process modelling (BPM) is carried out today is the lack of explicit and systematic reuse of previously developed models. Enabling the reuse of previously modelled behaviour can have a beneficial impact on the quality and efficiency of the overall information systems development process and also improve the effectiveness of an organisation's business processes. In related disciplines, like software engineering, patterns have emerged as a widely accepted architectural mechanism for reusing solutions. In business process modelling the use of patterns is quite limited apart from few sporadic attempts proposed by the literature. Thus, pattern-based BPM is not commonplace. Business process patterns should ideally be discovered from the empirical analysis of organisational processes. Empiricism is currently not the basis for the discovery of patterns for business process modelling and no systematic methodology for collecting and analysing process models of business organisations currently exists.

The purpose of the presented research project is to develop a methodological framework for achieving reuse in BPM via the discovery and adoption of patterns. The framework is called Semantic Discovery and Reuse of Business Process Patterns (SDR). SDR provides a systematic method for identifying patterns among organisational data assets representing business behaviour. The framework adopts ontologies (i.e., formalised conceptual models of real-world domains) in order to facilitate such discovery. The research has also produced an ontology of business processes that provides the underlying semantic definitions of processes and their constituent parts. The use of ontologies to model business processes represents a novel approach and combines advances achieved by the Semantic Web and BPM communities. The methodological framework also relates to a new line of research in BPM on declarative business processes in which the models specify what should be done rather than how to 'prescriptively' do it. The research follows a design science method for designing and evaluating SDR. Evaluation is carried out using real world sources and reuse scenarios taken from both the financial and educational domains.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

DEDICATION

I dedicate this PhD thesis to my parents, Mum and Dad, the two most special persons in my life. They, not only gave me life, but also fill it with all the love and affection one can wish for. In all stages, my mother has been the best listener and advisor for me. Thank you my parents for your never-ending love, support and encouragement throughout my entire life and for providing me the high-quality education and the enthusiasm for higher studies. They are my source of strength and without their countless support this thesis would have never been started.

My special gratitude also is to my dearest brother Dr. Zaid and to my precious sisters Dalia and Dr. Dina and their families who tirelessly offered encouragement when it was most needed and for their loving support, concern and heartening through all these years. All my work would have not been possible without the blessings and support of my family. I am thankful to God for blessing me with such wonderful family.

By succeeding in this work, I am proud to show them that their faith in me is well justified. For this faith, I thank you.

ACKNOWLEDGEMENTS

The realisation of this work was only possible due to the collaboration of several people, toward whom I wish to express my gratefulness. Researching at the School of Information Systems, Computing and Mathematics at Brunel University has been a great pleasure and a wonderful privilege. I was very lucky to have Dr. Sergio de Cesare and Professor Mark Lycett as my supervisors. Their passion for good research and leadership styles inspired me to always try my best. So, thank you for believing in me and providing such a rewarding work environment.

In the first place, I would like to express my sincere appreciation and deep gratitude to my first supervisor, Dr. Sergio de Cesare, for his exceptional support, encouragement and guidance during all stages of my research. Dr. de Cesare instilled a thirst for excellence in me, taught me how to do high quality research, and helped me think independently and creatively. His truly incredible academic excellence, scientific intuition, and beautiful mind have made him a constant oasis of ideas, which has inspired and enriched my growth as a student, and a researcher. I gratefully also thank my second supervisor Professor Mark Lycett for his outstanding and insightful comments on my work and friendly support. Prof. Lycett is an excellent forward thinker who inspired me in the conception of my research work.

Professor Ray Paul has been another protagonist in the realisation of this PhD thesis. I was fortunate to meet him few times, for he has the empathy and quickness of thought to counsel PhD students in just about any topic.

Finally, I am also thankful for the stimulating atmosphere at the Fluidity group and at the Department of Information Systems and Computing (DISC) at Brunel University for supporting my work throughout my PhD.

DECLARATION

A number of publications have documented the research at various points:

JOURNAL

Aldin, L., and de Cesare, S., M. "A Literature Review on Business Process Modelling: New Frontiers of Reusability", *Enterprise Information Systems Journal* (Under review April 2010).

CONFERENCES AND WORKSHOPS

Aldin, L., de Cesare, S. & and Lycett, M. 2009 c, "A Semantic-Based Framework for Discovering Business Process Patterns", *6th International Workshop on Ontology-Driven Software Engineering (ODiSE): ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA),* Orlando, Florida, USA, Oct. 25-29, pp. 26-40. [Available from: http://bura.brunel.ac.uk/handle/2438/4080]

Aldin, L. & de Cesare, S. and Lycett, M. 2009 b, "Semantic Discovery and Reuse of Business Process Patterns", *4th Annual Mediterranean Conference on Information Systems,* Athens University of Economics and Business (AUEB), Greece, September 25-27, pp. 1-6. [Available from: http://bura.brunel.ac.uk/handle/2438/4079]

Aldin, L. & and de Cesare, S. 2009 a, "A Comparative Analysis of Business Process Modelling Techniques", *Proceedings of the 14th Annual UK Association of Information Systems Conference (UKAIS)*, Oxford University, Oxford, U.K, 31st March - 01st April, pp. 8-28. [Available from: http://bura.brunel.ac.uk/handle/2438/4078]

ACRONYMS

AI: Artificial Intelligence

BP: Business Process

BPEL: Business Process Execution Language

BPEL4WS: Business Process Execution Language for Web Services

BPM: Business Process Modelling

BPML: Business Process Modelling Language

BPMN: Business Process Modelling Notation

BPMO: Business Process Modelling Ontology

BPO: Business Process Ontology

BWW: Bunge Wand Weber

CS: Content Sophistication

DE: Domain Engineering

DFD: Data Flow Diagrams

DISC: Information Systems and Computing

DOLCE: Descriptive Ontology for Linguistic and Cognitive Engineering

DR: Design Research

ebXML: Electronic Business using eXtensible Markup Language

EPC: Event-driven Process Chain

ERP: Enterprise Resource Planning

IDEF3: Integrated DEFinition

IS: Information Systems

ISD: Information Systems Development

IT: Information Technology

LIS: Legacy Information Systems

OCL: Object Constraint Language

OWL: Web Ontology Language

PD: Pattern Documentation

PLA: Preparation of Legacy Assets

POLA: Procurement and Organisation of Legacy Assets

RBAC: Role Based Access Control

REA: Resources Events Agents

READY: Dynamic REA

SA: Semantic Analysis of BP Models

SE: Semantic Enhancement of BP Models

sEPC: Semantic Event-driven Process Chain

SDL: Semantic Discovery Lifecycle

SDR: Semantic Discovery and Reuse of Business Process Patterns

SLA: Segmentation of Legacy Assets

SRL: Semantic Reuse Lifecycle

SUPER: Semantic Utilised for Process Management within and between Enterprises

TOVE: Toronto Virtual Enterprise

UML: Unified Modelling Languages

W3C: World Wide Web Consortium

WfM: Workflow Management

WSCI: Web Service Choreography Interface

WSMO: Web Services Modelling Ontology

CHAPTER 1 – INTRODUCTION: SETTING THE SCENE

1.1 Introduction

This chapter sets the scene of the research to define and develop a methodology for discovering generic business process patterns from organisational knowledge sources. The achievement of greater agility and flexibility within business process modelling (BPM) represents a key goal for organisations. One of the reasons that impede BPM to achieve this goal is the lack of systematic discovery and reuse of business process models. The developed methodology for pattern-based business process modelling will improve the productivity of modellers as well as help to achieve improved levels of traceability between business requirements and software systems. Both benefits are a consequence of the experience embedded in the patterns derived from organisational assets.

This chapter is organised as follows. Section 1.2 presents the motivation of the research. Section 1.3 defines the research aim and objectives based on the established definition of the research problem and the motivations. Section 1.4 explains the Design Research method used for conducting this work. In Section 1.5 an overview of the thesis is presented. Section 1.6 shows a diagram for simplifying the reading of the thesis and finally the chapter summary is presented in Section 1.7.

1.2 Research Motivations

The modelling of business processes and their subsequent automation, in the form of workflows, constitutes a significant part of information systems development (ISD) within large modern enterprises. Business processes (BP) are designed on a regular basis in order to align operational practices with an organisation’s changing requirements (Azoff et al., 2007). A fundamental problem in the way business process modelling is carried out today is the lack of explicit and systematic reuse of previously developed models. Although all business processes possess unique characteristics, they also do share many common traits making it possible to classify business processes into generally recognised patterns of organisational behaviour.

Patterns have become a widely accepted architectural technique in software engineering. Patterns are general solutions to recurring problems. A pattern generally includes a generic definition of the problem, a model solution and the known consequences of applying the pattern. In business process modelling the use of patterns is quite limited. Apart from a few sporadic attempts proposed by the literature (Ericksson & Penker, 2000; Malone et al., 2003), pattern-based business process modelling is not commonplace. The benefits of adopting patterns are numerous. For example, as the academic literature and industry reports document, the adoption of design patterns in software engineering projects improves reuse of shared experiences, reduces redundant code, reduces design errors and accelerates the learning curve (Cline, 1996). As a consequence, it is conceivable that patterns in BPM can produce similar advantages, thus reducing both time and cost of generating business process models and their subsequent transformation into software designs of enterprise applications.

However, the systematic adoption of patterns in BPM cannot be a simple transposition of the experience acquired by the design patterns community in software engineering. This is due to some essential differences between business modelling and software design. While the latter involves the representation of an engineered artefact (i.e., software), the former concerns the representation of behaviour of a real world system (i.e., the business organisation). As such business process patterns should ideally be discovered from the empirical analysis of organisational processes. The discovery of real world patterns should resemble the process of discovery of scientific theories; both must be based on empirical data of the modelled phenomena. Empiricism is currently not the basis for the discovery of patterns for BPM and no systematic methodology for collecting and analysing process models of business organisations currently exists. This research aims at developing such a methodology. The main driver for the discovery of business process patterns will be ontologies. Ontologies are formalised semantic models of real world systems or domains and as such are well suited to uncover and represent the meaning of data underlying business processes.

This research is called Semantic Discovery and Reuse of Business Process Patterns (SDR). The broad findings emerging from the research are that business organisations will be capable of more flexibly adapting themselves to changing operational practices thanks to the generalised nature and semantic expressiveness of ontology-based business process patterns. In particular, the novel contribution of this research is a methodological framework (SDR) that enables business modellers to empirically and ontologically discover business process patterns and to reuse such patterns in future development projects.

1.3 Research Aim and Objectives

The aim of this research is to *investigate the effectiveness of business process patterns in business process modelling by developing and evaluating a methodological framework for empirically deriving ontological patterns of business processes from organisational knowledge sources.* This methodological framework seeks to provide a systematic way of generating and utilising reusable patterns within an organisation. The synthesis of empiricism and ontological based patterns are incorporated in the Semantic Discovery and Reuse (SDR) methodological framework. Figure 1-1 demonstrates the research aim and objectives plan of this research. In attempting to provide effective solutions the following objectives direct the aim of this research:

Objective I: Evolve a methodological framework for identifying patterns of business processes from enterprise knowledge sources (i.e., documentation, systems, domain experts, etc.).

Objective II: Derive a set of generic business process patterns by applying the developed methodology and to organise the discovered patterns within the repository.

Objective III: Investigate the development of an ontology of business process to be used for the derivation of semantic process models.

Objective IV: Evaluate the methodological framework and derived patterns in terms of their significance to theory and practice through recognised evaluation techniques.

The objectives respond to the proposed aim of the research to develop and evaluate the developed methodological framework called Semantic Discovery and Reuse of business process patterns. Initially, the Semantic Discovery Lifecycle (the first lifecycle) is the focus to discover reusable business process patterns. Subsequently, the discovered business process patterns and their reusability are to support the methodology.

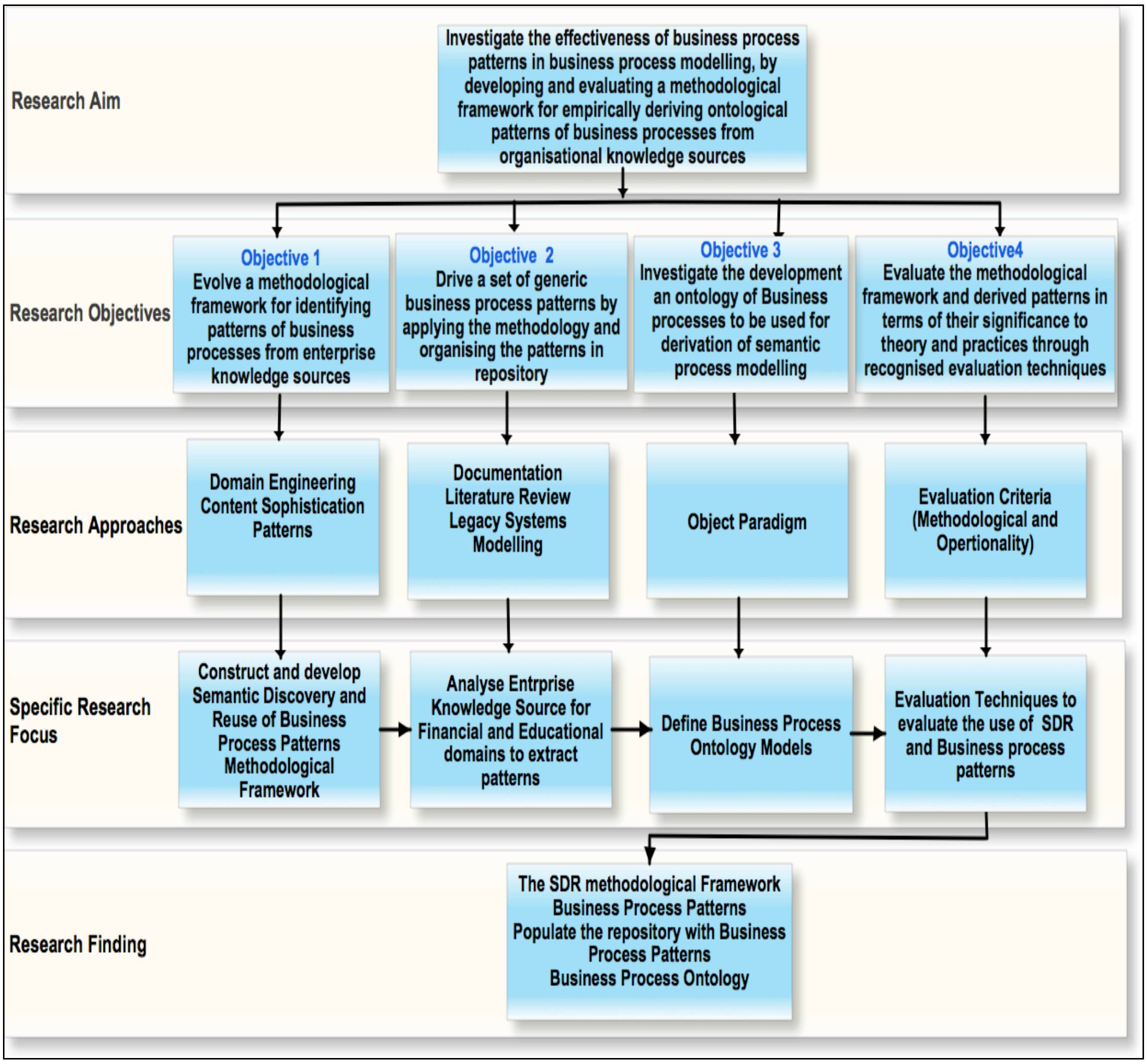

Research Aim
Investigate the effectiveness of business process patterns in business process modelling, by developing and evaluating a methodological framework for empirically deriving ontological patterns of business processes from organisational knowledge sources
Research Objectives
Objective 1
Evolve a methodological framework for identifying patterns of business processes from enterprise knowledge sources
Objective 2
Drive a set of generic business process patterns by applying the methodology and organising the patterns in repository
Objective 3
Investigate the development an ontology of Business processes to be used for derivation of semantic process modelling
Objective4
Evaluate the methodological framework and derived patterns in terms of their significance to theory and practices through recognised evaluation techniques
Research Approaches
Domain Engineering
Content Sophistication
Patterns
Documentation
Literature Review
Legacy Systems
Modelling
Object Paradigm
Evaluation Criteria
(Methodological and Opertionality)
Specific Research Focus
Construct and develop Semantic Discovery and Reuse of Business Process Patterns Methodological Framework
Analyse Entrprise Knowledge Source for Financial and Educational domains to extract patterns
Define Business Process Ontology Models
Evaluation Techniques to evaluate the use of SDR and Business process patterns
Research Finding
The SDR methodological Framework
Business Process Patterns
Populate the repository with Business Process Patterns
Business Process Ontology

Figure 1-1: Research Aim and Objectives Plan

In this research data sources from two domains are utilised in order to apply the SDR methodological framework and to produce business process patterns across multiple domains. These two domains are: (1) Financial services - the range of financial solutions offered by this domain covers areas of retail banking, insurance and mortgages systems, and (2) Education - information about this domain is provided by two handbooks for staff and students of the Department of Information Systems and Computing (DISC) at Brunel University. Each source of data used in this research has been developed over time as guidance to address the respective needs of the financial and the educational domains.

1.4 Design Research Method

The research objectives are concerned with answering the research question. Therefore, a research method should be selected that matches the nature of the research objectives. This corresponds with generating knowledge through making (Purao, 2002), which is best characterised by the Design Research (DR) method (Hevner et al., 2004; March & Smith, 1995). DR provides a suitable and comprehensive framework for the design and the analysis of artificial phenomena such as organisations or information systems (IS). Also, it consists of the construction and the evaluation of the artefacts that resolve a significant and recognised problem. Therefore, the Design Research method (Hevner et al., 2004) supports the design, construction and evaluation of the Semantic Discovery and Reuse of Business Process Patterns methodological framework and of its business process patterns. Also, March and Smith's (1995) framework was chosen to support artefact design as it enables the researcher to 'do' and 'observe' design as process steps; not only observe finalised instantiations. This framework provides a basis for the practical execution of the research project by articulating artefacts in terms of constructs, models, methods and instantiations. In this work the use of Design Research is directly associated with the expected result, which is to create the methodological framework called SDR (Semantic Discovery and Reuse of Business Process Patterns) for producing business process patterns. The result of the study has been evaluated to ensure both the validity and the reliability of the methodological framework and to determine the extent to which the outcome can be used as a basis for further research. A broad outline of the developmental research stages of Vaishnavi and Kuechler (2004) were followed to direct the research finding process. In this framework an iterative process of design was used to ensure continuous improvement in the designed artefacts. Each iteration feeds back knowledge gained through construction and evaluation into the design of the following iterations. Evaluation methods proposed by Hevner et al. (2004) were adopted with metrics selected from March and Smith (1995). Further explanation of the stages and iterations are covered in Chapter 3.

1.5 Chapters Overview

Chapter 2 presents a literature review in which three intersecting fields significant to the research are discussed: business processes, patterns and ontologies. The literature is organised in three main sections. The first section provides an overview of business processes with emphasis on modelling and discovery techniques. In the second section, the antecedents of existing patterns are analysed and these lead to the finding that these existing patterns do not resolve the problems of domain reuse in modelling organisational processes. Ontology is explained in the third section to explore the feasibility of adapting the use of ontology, which allows for more accurate semantic representations of processes and their elements when modelling organisational processes.

Chapter 3 introduces the Design Research methodology and its application within this research. The aim of the chapter is to demonstrate rigour in the use of Design Research as a suitable and comprehensive framework for the design and the analysis of artificial phenomena. The broad outline of the developmental research stages is illustrated with the three stages of *Awareness of the problem and types of solution*, *Development* and *Final Evaluation*. The content of the *Development* stage involves iterations of work that *Design*, *Deploy* and *Evaluate* research artefacts, with feedback directing the research for the following iteration. The broad outline follows the general design methodology of Vaishnavi and Kuechler (2004) for constructing artefacts (constructs, models, methods and instantiations) identified by March and Smith (1995). Evaluation of each iteration and its artefacts follows methods and criteria prescribed by Hevner et al. (2004) and March and Smith (1995). Finally, this chapter provides necessary planning mechanisms at both theoretic and practical levels allowing the focus to move onto the first design iteration in Chapter 4.

Chapter 4 appraises the first design iteration. A core contribution is the evolvement of the first version of the Semantic Discovery and Reuse of Business Process Patterns (SDR) methodological framework. The focus in iteration one is on the first version of the first lifecycle the Semantic Discovery Lifecycle (SDL) of SDR and its application in the financial services domain. Therefore, this chapter lays down the necessary groundwork for the practical application of the SDL that subsequently is evaluated with the research outputs. Final feedback is provided so as to influence the design of iteration two in Chapter 5.

Chapter 5 presents a refined and a second version of the SDR methodological framework. The iteration aims to improve the SDL lifecycle by: (1) grouping and subdividing some phases of SDL to increase efficiency of the lifecycle and the flow of the phases; (2) constructing the ontological definition of business processes to incorporate the new model; (3) altering the process models to reflect new presentation guidelines for the processes using the ontological definition of business process; (4) introducing a generalised ontological model of business processes using Protégé and the FaCT++ reasoner to increase process enhancement; (5) applying SDL to the higher education domain of Brunel University; and (6) documenting discovered business process patterns. Improvements are evaluated against the previous iteration and to influence the design of iteration three in Chapter 6.

Chapter 6 extends the use of the final version of the Semantic Discovery Lifecycle (SDL) after being developed and refined in the first and second iterations. This final iteration aims at: (1) applying the Semantic Discovery Lifecycle to the retail banking, insurance and mortgage systems of the financial services domain to produce business process patterns, (2) documenting business process patterns of the financial services domain and structure a hierarchy of discovered business process patterns across domains, and (3) demonstrating SDL's ability to produce patterns of business processes across multiple systems (retail banking, insurance and mortgages) and multiple domains (financial and educational). The evaluation of iteration three artefacts is carried out against the findings of previous iterations to produce suggestions for future work. The chapter concludes with all the iterations being synthesised.

Chapter 7 presents a summary of the thesis. A brief account is provided of how each of the objectives is realised through the activities carried out within these chapters and stating the research values and contributions of the developed SDR and patterns. Finally, the chapter ends with an exposition of the research limitations and new lines of research and recommendations proposed for further research.

In addition, five appendices are provided. **Appendix A** represents extracted business processes and their BPMN models. **Appendix B** presents the ontological models of the interpreted business processes. **Appendix C** provides a list of the BP ontology with generalisation models. **Appendix D** presents the documentation of the discovered business process patterns. **Appendix E** represents the developed business process ontology in the Web Ontology Language (OWL). These appendices provide further details related to the various discoveries in different chapters.

1.6 Thesis Overview

To simplify the reading of this thesis a diagram summarising the chapters is presented in Figure 1-2. The high level flow from aim, objectives to contributions are included. In preference of clarity the flow is presented in a sequential form, not showing the iterative nature of artefact refinement that transpired.

1.7 Summary

This chapter introduced the research presented in the remainder of the thesis. Starting with the motivation for undertaking such research and the literature that reflects the current state of the art in the area of business process modelling in information systems development. The aim of the research was defined with the objectives required to fulfil the aim. A supporting Design Research methodology for conducting this research has been defined to justify the research question and to address the research contribution. Finally, the content of each chapter is summarised.

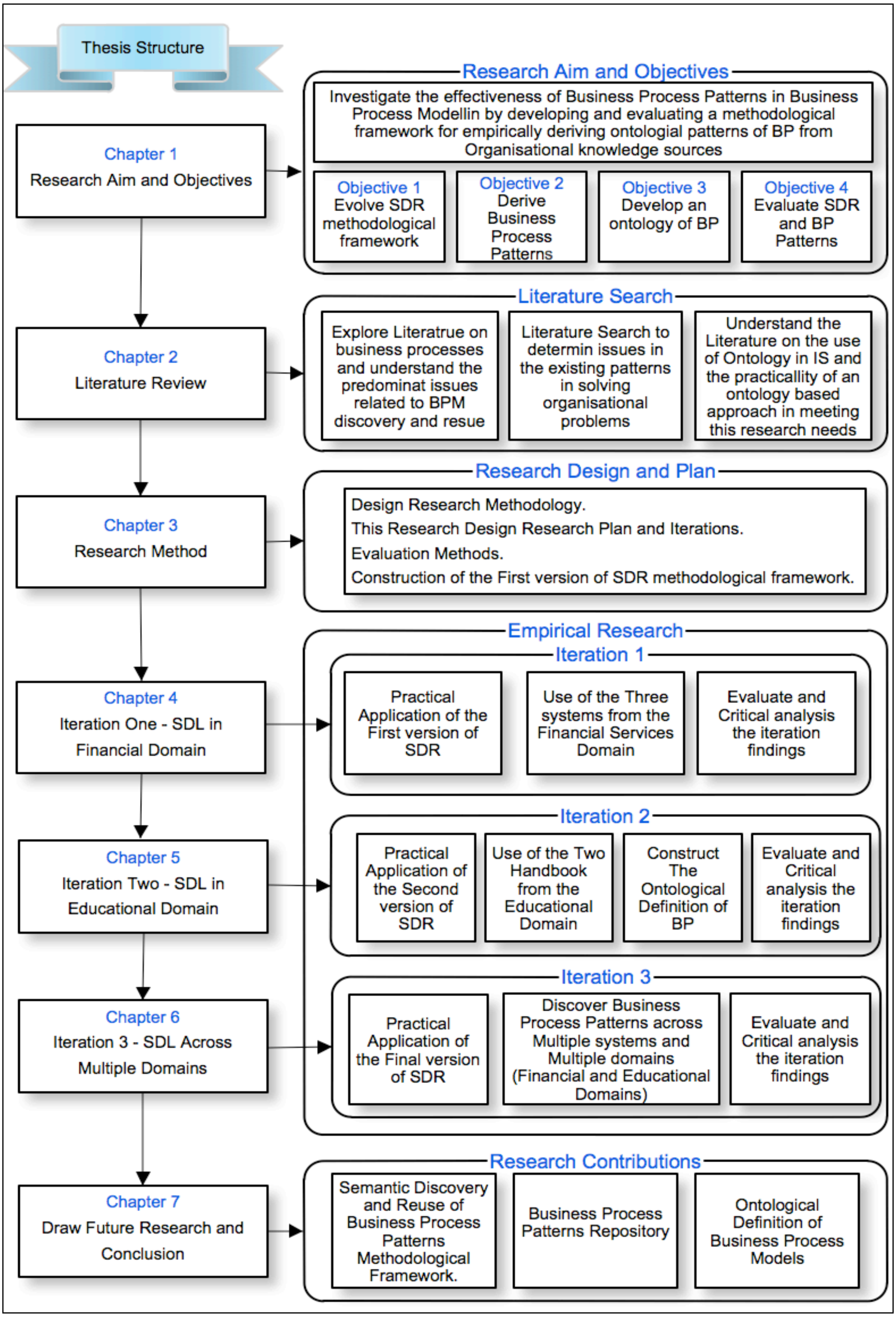

Thesis Structure
Chapter 1
Research Aim and Objectives
Chapter 2
Literature Review
Chapter 3
Research Method
Chapter 4
Iteration One - SDL in Financial Domain
Chapter 5
Iteration Two - SDL in Educational Domain
Chapter 6
Iteration 3 - SDL Across Multiple Domains
Chapter 7
Draw Future Research and Conclusion
Research Aim and Objectives
Investigate the effectiveness of Business Process Patterns in Business Process Modellin by developing and evaluating a methodological framework for empirically deriving ontologial patterns of BP from Organisational knowledge sources
Objective 1
Evolve SDR methodological framework
Objective 2
Derive Business Process Patterns
Objective 3
Develop an ontology of BP
Objective 4
Evaluate SDR and BP Patterns
Literature Search
Explore Literatrue on business processes and understand the predominat issues related to BPM discovery and resue
Literature Search to determin issues in the existing patterns in solving organisational problems
Understand the Literature on the use of Ontology in IS and the practicallity of an ontology based approach in meeting this research needs
Research Design and Plan
Design Research Methodology.
This Research Design Research Plan and Iterations.
Evaluation Methods.
Construction of the First version of SDR methodological framework.
Empirical Research
Iteration 1
Practical Application of the First version of SDR
Use of the Three systems from the Financial Services Domain
Evaluate and Critical analysis the iteration findings
Iteration 2
Practical Application of the Second version of SDR
Use of the Two Handbook from the Educational Domain
Construct The Ontological Definition of BP
Evaluate and Critical analysis the iteration findings
Iteration 3
Practical Application of the Final version of SDR
Discover Business Process Patterns across Multiple systems and Multiple domains (Financial and Educational Domains)
Evaluate and Critical analysis the iteration findings
Research Contributions
Semantic Discovery and Reuse of Business Process Patterns Methodological Framework.
Business Process Patterns Repository
Ontological Definition of Business Process Models

Figure 1-2: Overview of the Thesis

CHAPTER 2 - LITERATURE REVIEW

2.1 Introduction

The literature review of this chapter exposes the limitations in the semantic discovery and reuse of business process patterns in the business domain. The chapter critically reviews three intersecting fields of study that are necessary for this research: business processes, patterns and ontology.

The aim of this literature review is to: (1) provide a context to the research in so far as it provides an understanding of current research into business process patterns (including their relevance to business process model reuse and discovery) and (2) identify appropriate existing research related to the area of business processes, patterns and ontology. This literature review helps in uncovering constructs associated with the research domain and any pertinent gaps that exist and that are required to be addressed for later research activities. The identification of relevant and applicable constructs provides a high level vocabulary for the research, enabling a suitable research methodology to be selected and justified.

This chapter is organised as follows. While Section 2.2 provides an overview of BPM, Section 2.3 focuses on business process modelling and discovery techniques. Section 2.4 presents the concept of pattern, how this architectural mechanism has been applied in the related field of software engineering and the extant literature on patterns in the context of business modelling in IS development. Section 2.5 explores ontology in relation to BPM and specifically the different ways in which ontology has been used to evaluate BPM techniques as well as to conceptualise business processes themselves with the aim of enhancing reusability via semantically precise representations. The object paradigm is explained in Section 2.6. This paradigm is adopted in this research to model the business processes. The literature findings and research direction are presented in Section 2.7 and the literature summary is presented in Section 2.8.

2.2 Business Process Modelling: An Overview

2.2.1 Business Processes

Business processes express an organisation's behaviour and these processes exist independent of whether they are modelled or not. The traditional functional view of the organisation has now been superseded or integrated by a process view since processes can more clearly define and relate to the intended goals of the business while cutting across functional boundaries (Malone, Crowston & Herman, 2003).

As defined by Hammer and Champy (1993, p.85), a business process is "a collection of activities whose final aim is the production of a specific output that is of value to the customer. A business process has a goal and is affected by events occurring in the external world or in other processes". As with most terms many other definitions and classifications of business process exist. For example, Aguilar-Savén (2004) distinguishes between 'core' and 'supportive' business processes. A core (or primary) process is initiated from outside an organisation, e.g. the chain of activities that realises the delivery of a product to a customer. A supportive (or secondary) process creates the conditions for the primary process to be carried out. Table 2-1 summarises five definitions found in the literature and extracts the main concepts emphasised by the respective authors.

The concepts identified in Table 2-1 represent those elements that the business process community commonly and generally accepts as being fundamental in characterising business processes (Aldin and de Cesare, 2009 a). Figure 2-1 provides an example of a business process aimed at arranging a financial agreement with a customer. The example is represented in the Business Process Modelling Notation (BPMN) and merely serves the purpose of illustrating how the different elements of a business process contribute to representing the behaviour of an organisation.

Definitions	Concepts Identified
A business process is the set of internal activities performed to serve a customer (Jacobson et al., 1995).	Process Activities Serve Customer
A business process is a collection of activities that takes one or more kinds of input and creates an output that is of value to the customer. A business process has a goal and is affected by events occurring in the external world or in other processes (Hammer and Champy, 1994).	Process Activities Input Output Customer Goal Event
A business process is simply a structured set of activities designed to produce a specified output for a particular customer or market. It implies a strong emphasis on how work is done within an organization, in contrast to a product's focus on what. A process is thus a specific ordering of work activities across time and place, with a beginning, an end, and clearly identified inputs and outputs: a structure for action. (Davenport, 1992)	Process Activities Input Output Customer Product Time/place Rules
Business Process is a lateral or horizontal organisational form that encapsulates the interdependence of tasks, roles, people, departments and functions required to provide a customer with a product or a service. (Earl, 1994)	Process Tasks (i.e. activities) Roles Customer Product or Service
Business Process is a purposeful activity carried out collaboratively by a group, often crossing functional boundaries and invariably driven by outside agents or customers. (Ould, 1995)	Process Activities Customer Purposeful (i.e. having an aim)

Table 2-1: Definitions of business process drawn from the literature

Thus, the conceptual elements identified in Table 2-1 can be defined as follows:

- ***Process***: A set of activities, events, etc. that together and cohesively delivers a service and/or a product (e.g., Arrange Finance).
- ***Activity***: Specific behaviour carried out in an organisation (e.g., Collect and pass customer details to the Accounts Department).
- ***Service and Product***: The observable outcome of value of a process. The traditional distinction between service and product is that the former is intangible while the latter is tangible (e.g., providing the customer with the opportunity to apply for finance).
- ***Role***: The types of actors or agents that take part in processes (e.g., Front Office).
- ***Goal***: The aim of a process (e.g., obtaining finance).

- ***Event***: An occurrence that takes place at a specific point in time and that is capable of inducing some observable behaviour (activity or process) (e.g., customer request for finance).
- ***Rule***: A constraint defined for any part of the organisation and its processes (e.g., only customers with a clear credit check can be considered for a loan).

All of these elements are related to each other and some of these elements (process, activity, event, role) are more easily representable and explicit than the rest. In languages like BPMN explicit symbols are provided for representing a process (i.e., an entire diagram), an activity, an event and a role (e.g., partitions or swimlanes). The other elements of a business process are not as explicitly represented in most process modelling notations. Rules can be modelled via the use of a formal or informal 'rules' languages. An example of the former is the Object-Constraint Language (OCL) in UML activity diagrams, while an example of the latter is the use of natural language in textual annotations placed directly on the process model itself.

The concepts of service, product and goal are related to the expected outcome of a business process. While in textual representations of business processes (e.g., business use cases) these concepts tend to be explicitly represented (de Cesare et al., 2003), in diagrammatic models (such as in Figure 2-1), the outcomes of the process tend to derive from an interpretation of the model itself as well as from the original intention of the modeller.

To summarise these fundamental process elements: A business process is triggered by an initiating event which is followed by a sequence of activities and events that are carried out by (or are within the responsibility of) people or organisational units assuming specific roles. The activities and events are carried out in accordance with predefined rules. Altogether the business process is aimed at achieving a goal, which results in the delivery of a service or product.

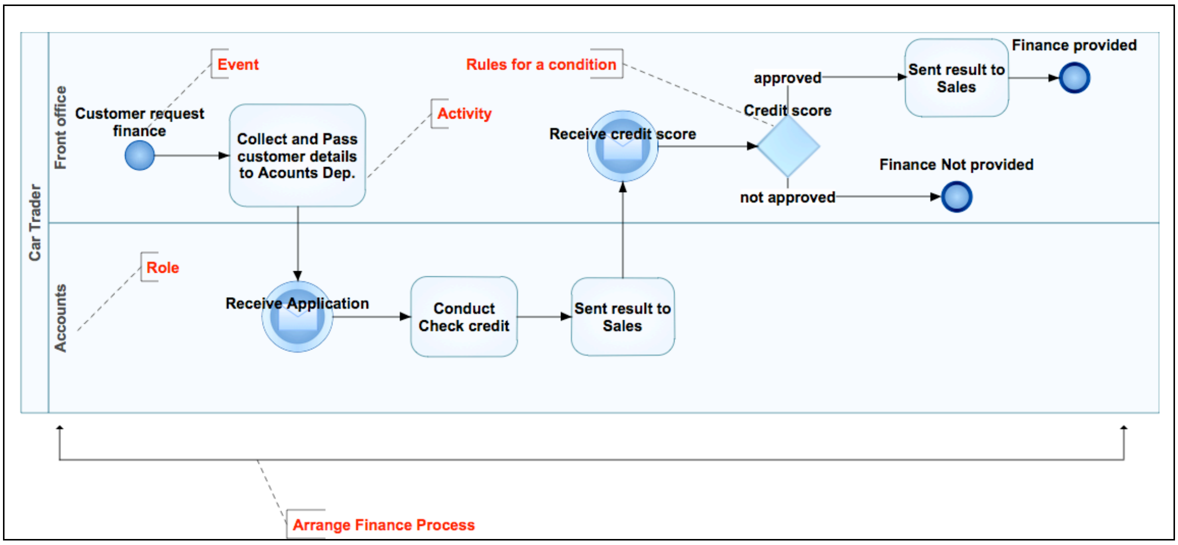


Figure 2-1: Example of a business process

Organisations and their underlying processes are continuously affected by changes occurring in the environment (e.g., market, society, etc.) of which they are part (Havey, 2000). Examples of large-scale factors affecting businesses in recent years include globalisation, deregulation, the growing pace of innovation, the increasing education and affluence of people, and new technologies. However, whether these changes are dramatic or subtle, BPM is employed to keep a business efficient and competitive (Morgan, 2007). It is worth mentioning that not only have organisations changed, but also that the way in which people view business processes has evolved. According to Ould (2006) business processes have gone through three waves of changes:

- In the *first wave*, processes are thought of as a 'way of doing things' and very rarely they would be mentioned in policy and procedure manuals. Those processes were usually presented using flowcharts in order to achieve a better understanding and to make small-scale improvements to them.
- In the *second wave*, the information perspective of processes prevailed. This was naturally determined by developers' tendency to focus on information requirements and use these as the basis for creating business processes, rather than designing the processes and subsequently making the information systems conform to them. In this wave, business processes were often preferred to be re-engineered to best practice and this costing a huge sum of money. Thus, processes were only regarded and used as a means to reinforce information.

- Currently there is *a third wave*, which is directed towards business process management's recent technology. In this wave, business processes are the deterministic factors about how information should be manipulated and what information needs to be stored. There is more emphasis now placed on business processes than on information. Organisations can therefore change the operational methods of their businesses. Today organisations think more distinctively about processes and how they can be executed, modified and monitored even in real-time via business management systems. These latest trends require suitable methods to identify, analyse, model, design, implement, evaluate and use processes. It can be concluded that the main aim of business process management systems is to enact models of business processes. This is primarily due to the need of organisations to be able to more readily and flexibly adapt their processes to change induced by both internal and external factors (Morgan, 2007) achieving better organisational performance.

Thus in the current third wave organisations have matured in their perception and use of business processes. Processes are increasingly becoming the primary design artefact adopted to both conceptualise and engineer the organisation. Consequently, business process design would require more systematic methods, tools and techniques with which process engineers can achieve the more demanding requirements expected by BPM in this third wave. As it will be argued, process reusability represents an approach that may cope with these increased demands.

2.2.2 Business Process Models

Models in essence are a means for explicating, agreeing and managing information in a well defined and structured manner (Ludewig, 2003). Without a model it would become very difficult to manage vast amounts of information in a coherent manner and come to any form of common understanding. Moreover the capability to store information means that models often provide a rich means of managing documentation concerns by preserving the information for future referencing and possible reuse for other purposes. According to Bandara et al. (2007, p. 21) "process modelling occurs before the background of an organisationally and information technology-supported setting in which process models are created to fulfil multiple purposes". Models of business processes can be understood as simplified, abstract representations of business processes. Curtis, Kellner & Over (1992, p.67) defined business process models as “an abstract description of an actual or proposed business process that represents selected business process elements that are considered important to the purpose of the model and which can be enacted by a human or machine”.

Workflow models are closely related to business process models. A workflow can be defined as “the automation of a business process in whole or in part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules” (Hollingsworth, 1995, p. 6). While a workflow model represents an automated process, a business process model may include both manual and automated activities.

One of the most important elements of a business model is the definition of the business processes that will operate within the organisations (Morgan, 2002). A business model is an abstraction of how a business functions, while a business process model is a type of business model that focuses on an organisation’s behaviour. What details to include in a business model differs according to the perspective of the model creator, which will lead to slightly different viewpoints of the goals and visions of the business, including its efficiency and the various elements that are acting in concert within the business. Business models provide “a simplified view of the business structure that will act as the basis for communication, improvements, or innovations, and define the information systems requirements that are necessary to support the business” (Ericksson and Penker, 2000, p. 10).

Business process models represent the main conceptual artefacts (e.g. processes, activities, events, roles, rules, etc.) underpinning the management of organisational processes and their continuous change (Mendling, 2008). It is through process models that it is possible to begin to systematically update and revise business processes periodically in order to achieve improved organisational performance and enable the organisation to deliver quality products and services as required by its customers (Jacobson, Ericsson & Jacobson, 1995). Business process modelling serves multiple purposes. Summarising purposes of BPM extracted from Ericksson and Penker (2000), Caetano, Silva and Tribolet (2005) and Luo and Tung (1999) and the perceived benefits of BPM as investigated by Indulska et al. (2009), it can be said that business process modelling helps to achieve the following:

- Supporting process improvement and re-engineering through business process analysis and simulation. BPM can be used for improving the current business by identifying possible ways to make the business more efficient. Normally, the current business is modelled and then re-engineered for enhancement or improvement opportunities;
- Facilitating a group to share their understanding of the process by using a common process representation, which helps human understanding and communication. This would be facilitated, for example, by adopting and agreeing a well-defined set of business process concepts among multiple stakeholders;
- Creating suitable information systems that support the business by providing a descriptive model for learning;
- Enabling decision support during process execution, and control;
- Providing the advantage of reuse. If the same business process model can act as the basis for several information systems, it can be reused as the basic input for defining the requirements of each system. It is worth stating that the final purpose (and benefit) of reuse is instrumental toward achieving the previous four purposes.

Process modelling could be used for the analysis of an existing business process (i.e., descriptive, prescriptive and explanatory) or to create new process models by modifying existing models rather than creating them from scratch (Lin, 2008). In practice, the modelling expert may encounter the similar modelling scenarios many times in their entire career and such experience would be of great value to the organisation if documented (Havey, 2005). Recording such experience is the initial step toward discovering common patterns of reusable behaviour. Another term used to denote such commonalities is process model fragment (Lin, 2008). In this context a fragment is a part of a business process model designed and managed to be reusable. Generally, reuse of pre-existing model fragments can facilitate and speed-up the construction of a new model. For their reuse advantages, process fragments and patterns are increasingly attracting the interest of both researchers and vendors. Since businesses change over the years there arises a need to keep existing business process models up-to-date and to synchronise or translate them into patterns. To facilitate these scenarios, techniques for discovering business process patterns from existing organisational assets (e.g., people, workflow systems, documentation, legacy systems, models, data, etc.) are required.

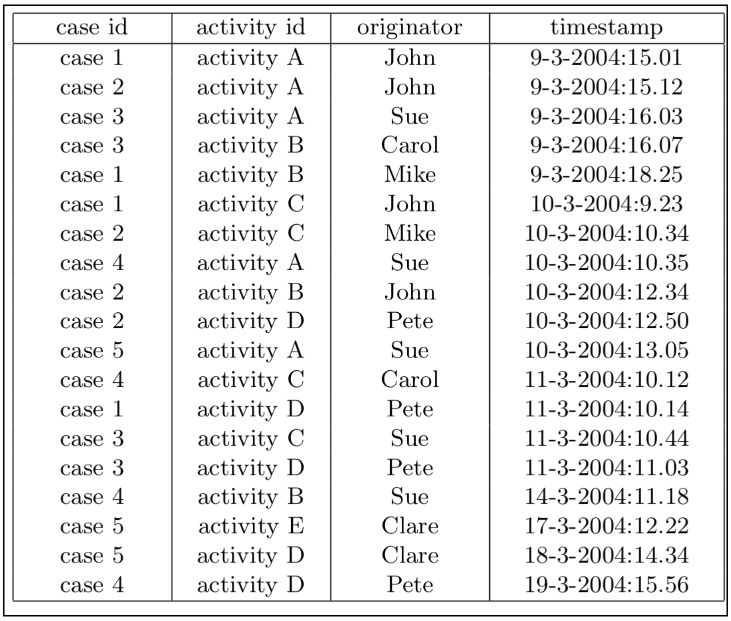
2.3 New Trends in Process Discovery and Representation

2.3.1 Process Mining

There are some research efforts by the BPM community, which have looked at process discovery by mining the control-flow of process models. Process discovery is aimed at correctly summarising an event log and describing how processes have actually taken place. The first papers on process mining appeared in 1996 when Cook and Wolf (1996) started mining process models from event logs in the context of software engineering. They called it 'process discovery'. Process mining in the business sense was first introduced in 1998 by Agrawal, Gunopulos & Leymann (1998) and they called it workflow mining. Since then many groups have focused on mining process models (van der Aalst et al., 2003; van der Aalst, Weijters & Maruster, 2004; van Dongen and van der Aalst, 2005; Greco, Guzzo & Pontieri, 2005; Schimm 2000). Process mining is " the automated acquisition of process models from the event logs of information systems such as Enterprise Resource Planning (ERP), Role Based Access Control (RBAC), and Workflow Management (WfM) systems" (van der Aalst and van Dongen, 2002; van der Aalst, de Medeiros & Weijters, 2004; van der Aalst et al. 2007, p. 240). Event logs contain information about the occurrence of business events and who performed a particular activity in the context of a particular business process involving some particular business information at a particular time, as presented in Table 2-2. However, van Dongen (2007) highlights that process mining may perform poorly for the following reasons:

- For event logs taken from computer systems used in an organisation, it is not clear which properties are satisfied. It is impossible to say whether such a log shows all possible interleaving of two activities, since that implies knowledge about the process, which the logs do not have, "the more dynamic a process is, the less knowledge [one] typically possess[es]" (van Dongen, 2007, p. 255).
- Event logs typically log information at a rather low level, whereas people talk about processes at a very high level and, according to van Dongen (2007), there is no given way to link the low-level activities to those high level activities, because interviews provide a subjective, qualitative source of information, whereas event logs are an objective and quantitative source of information.

- Goedertier et al. (2008) state that event logs rarely exhibit complete behaviour, because of the presence of noise. Thus, incomplete event logs can invalidate the ability to produce the correct underlying process model. van der Aalst et al. (2007) state that process mining is particularly useful in the context of human centric processes that are supported, but not fully controlled by computer systems.



case id	activity id	originator	timestamp
case 1	activity A	John	9-3-2004:15.01
case 2	activity A	John	9-3-2004:15.12
case 3	activity A	Sue	9-3-2004:16.03
case 3	activity B	Carol	9-3-2004:16.07
case 1	activity B	Mike	9-3-2004:18.25
case 1	activity C	John	10-3-2004:9.23
case 2	activity C	Mike	10-3-2004:10.34
case 4	activity A	Sue	10-3-2004:10.35
case 2	activity B	John	10-3-2004:12.34
case 2	activity D	Pete	10-3-2004:12.50
case 5	activity A	Sue	10-3-2004:13.05
case 4	activity C	Carol	11-3-2004:10.12
case 1	activity D	Pete	11-3-2004:10.14
case 3	activity C	Sue	11-3-2004:10.44
case 3	activity D	Pete	11-3-2004:11.03
case 4	activity B	Sue	14-3-2004:11.18
case 5	activity E	Clare	17-3-2004:12.22
case 5	activity D	Clare	18-3-2004:14.34
case 4	activity D	Pete	19-3-2004:15.56

Table 2-2: An Example of an Event Log File (from van der Aalst et al., 2007)

2.3.2 Imperative vs. Declarative Approaches

Currently business process modelling languages and models are procedural (or imperative). A business process model is procedural when it contains explicit prescriptive information about how processes should proceed, but only implicitly keeps track of why these design choices have been made (Goedertier and Vanthienen, 2007a). When modelling business processes procedurally, modellers inevitably make a number of modelling assumptions that are not present in the earlier specified requirements. Procedural process models are modelled with procedural languages such as Workflow Nets (van der Aalst, 1999), the Business Process Execution Language (BPEL) (Andrews et al., 2003), the Business Process Modelling Notation (BPMN) (Object Management Group, 2009) and the Unified Modelling Language's (UML) Activity Diagrams (Object Management Group, 2009). The procedural approach is dominant within the context of contemporary workflow technology (Pesic and van der Aalst, 2006), because it focuses on the control-flow perspective of business processes.

The procedural approach requires specifying all execution alternatives explicitly in the process model. It specifies exactly the ‘how’ of procedures by tracking ‘how things have to be done’ (van der Aalst and Pesic, 2006). The fact that the procedural approach requires all execution alternatives to be explicitly specified in the model causes some problems with respect to flexibility of workflow management systems. This has several consequences (Pesic, 2008):

- Procedural models with multiple execution alternatives tend to be large and complex, which makes it hard to understand and maintain these models.
- In a procedural approach all execution alternatives must be anticipated in advance.
- Explicitly specifying the procedure in the model can result in over-specifying the process.

Declarative modelling uses business rules at its core. The modeller makes assertions about the business organisation and such assertions must hold true and validated against during model execution. The declarative nature of business rules provides several benefits including unambiguous representation, explicit declaration, rapid adaptation and promotion of reuse (Morgan, 2002).

In order to deal with the problem of flexibility Pesic and van der Aalst (2006) propose a descriptive or declarative approach to enable system users to manoeuvre within the process model or even change the model while working as this is considered to be the most suitable for dynamic process management. In fact, according to Koehler et al. (2005), declarative approaches have a clear advantage over traditional modelling techniques as this type of approach produces models that “can be analysed, reused, and reversed”; adopting a declarative approach helps in defining rules that facilitate, for example, model transformation, specialisation and integration, all of which are essential in process reusability. Also Koehler et al. (2005) believe that the declarative approach can pave the way for future automatic consistency checking of transformation rules as well as bidirectional reconciliation of evolving models.

In Pesic’s thesis (2008) the declarative approach was proposed as a more suitable option for achieving a higher degree of flexibility because it does not require explicit specification of execution alternatives. Instead, a declarative approach allows for the implicit specification of execution alternatives. This approach is based on using activities and constraints for declarative specification of the control-flow perspective of process models (Pesic, 2008). Constraints are rules that should be followed during the execution. Applying a constraint in a business process model would implicitly specify the possible execution alternatives. Thus, everything that does not violate constraints is allowed. Declarative process models are represented with declarative languages, such as ConDec (van der Aalst and Pesic, 2006), that are used to build a wide range of representations: from very strict models, that define the process in detail, to very relaxed models that state only what should be done without specifying how it should be done. ConDec specifies the 'what' by starting from all the possibilities and using constraints to approximate the desired behaviour (Pesic, 2008). Further reading on constructing an on-line book purchasing business process model in a procedural language using Petri nets and a declarative language using ConDec can be found in Pesic & van der Aalst (2006).

The declarative approach requires specifying all execution alternatives implicitly in the process model. It specifies exactly 'what' should be done without specifying 'how' it should be done (Pesic and van der Aalst, 2006) because users are driven by the system to produce the required result, while the manner in which the results are produced depends on the preference of users. The fact that the declarative approach requires all execution alternatives to be implicitly specified in the model causes a problem in supervising the flexibility of workflow management systems. This assumption has several consequences:

- It is unclear whether end-users are really capable of adjusting a particular plan to execute a business process when using a declarative approach (Pesic, 2008).
- There is lack of empirical evidence on how declarative approaches perform in real-world settings (Weber et al., 2009).
- It is unclear how well users can cope with the gained flexibility provided by the declarative approach (Pesic and van der Aalst, 2006), especially when processes and their context become rather complex. While it might be expected that the declarative approach will be effective to deal with business processes when few constraints are applied, it is not clear whether end users are capable of translating a large number of constraints into effective updates of their initial plans. Since all constraints must be satisfied, one can argue that the sheer number of constraints will obscure from an end user's view of what proper actions are still available.

The adoption of the declarative approach need not entirely supersede the procedural approach. Notwithstanding the benefits stated above, Sadiq et al. (2005) believe that the combined use of the declarative and procedural approaches would be advantageous since procedural models are assumed to be easier to understand, while declarative modelling promises a higher degree of flexibility. In their opinion they represent two sides of the same coin.

Given the flexibility with which declarative modelling approaches are able to represent specific conditions, constraints or rules that classes of business processes must exhibit (or not), these types of models are particularly appropriate to the representation of more generalised models such as patterns. It can be argued that declarative representations can help modellers to both represent and use patterns of generalised behaviour in the sense that it would be possible to state a minimal set of necessary conditions that are required for the application of a certain pattern. The following section will focus on patterns in information systems development while Section 2.5 (and specifically 2.5.2) will discuss the major literature in the area of ontologies for business process modelling; ontologies which can add significant support to reuse.

2.4 Patterns: General Concepts

An organisation encounters many problems in its day-to-day operations and in its strategic positioning against other organisations. According to Nelson and Nelson (2003) these problems occur over and over in slightly different forms but with the same fundamental characteristics. A pattern is “a core of the solution” to these common problems (Alexander et al., 1977). The concept of patterns initially originated from the work of the architect Christopher Alexander (Alexander et al., 1977; Alexander 1979). Alexander defined pattern as follows: "Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice" (Alexander et al., 1977, p. 5). Therefore, while on one hand a pattern is used to solve problems, on the other hand these problems and their solutions are not unique but may be found and adopted in multiple contexts and situations.

2.4.1 Patterns in Information Systems Development

Many patterns have been developed and are based on the different phases of the software engineering lifecycle as well as different design paradigms and features (Gzara, Rieu & Tollenaere, 2000). These differences result in various levels of suitability and expressive power. In addition, the use of patterns has gained a lot of attention in several facets of information systems development (ranging from business modelling to implementation). Beck and Cunningham (1987) initially introduced patterns in software programming by adopting ideas and principles first described by Alexander et al. (1977) in the field of civil architecture. The pattern concept was developed further and introduced at a design level. Examples of initial design patterns modelled by Coad (1992) included ‘item description’, ‘time association’ and ‘event logging’. Coad, de Luca & Lefebvre (1999, p.3) later adopted the term archetype to indicate “a form from which all classes of the same kind more or less follow”. Design patterns finally became a mainstream architectural technique thanks to Gamma et al. (1995) who systematically compiled a catalogue of over 20 design patterns.

Subsequently patterns were introduced by Hay (1996) to represent generic data structures typically used to model the information requirements of business organisations. Similarly to Hay, Fowler (1997) defined a set of analysis patterns with the intention of reflecting “conceptual structures of business processes rather than actual software implementations” (p.xv). The works of both Hay and Fowler mainly focused on structural patterns (data/information). Some process patterns can be identified in Fowler, but these remain mainly underdeveloped. Furthermore Fowler’s work tends to be directed toward software designers. As a result his analysis patterns in many areas commit more to software artefacts rather than to generic business domain structures and behaviour. As suggested by Yang et al. (2005) such analysis patterns seem to be the solutions for reusable problems in software development processes.

More recently, the process communities have taken a similar approach by identifying and codifying its own set of common problems. In business modelling Ericksson and Penker (2000) developed a set of business patterns, which came closer to a generic representation of organisational structures and processes. These include resource and rule patterns, goal patterns and process patterns. Although these patterns like the previous (Fowler and Hay) are ultimately aimed toward the facilitation of realising software artefacts that will help to effectively and efficiently develop and ‘run’ information systems, Ericksson and Penker’s business patterns are modelled and described from a perspective that is closer to that of the enterprise rather than that of the software developer. Another form of business modelling patterns are those proposed by Malone et al. (1993; 2003). The MIT Process Handbook project started in 1991 with the aim to establish an online library for sharing knowledge about business processes. The business processes in the library are organised hierarchically to facilitate easy process design alternatives. The hierarchy builds on an inheritance relationship between verbs that refer to the represented business activity. There is a list of eight generic verbs including ‘create’, ‘modify’, ‘preserve’, ‘destroy’, ‘combine’, ‘separate’, ‘decide’, and ‘manage’. These business process patterns provide a systematic means of (re-) designing new processes by finding a richer structured repository of process knowledge through describing, analysing, and redesigning a wide variety of organisational processes.

Within the business domain there are examples of more specialised patterns like the Resources Events Agents (REA) framework (Geerts and McCarthy, 2002). REA is a domain ontology specialised for the enterprise context. Its main area of application was initially accounting information systems, but REA has now become more widely used for the conceptualisation of enterprises in the more general sense. REA has also been used as the foundation for more specialised and reusable pattern-based frameworks. One of these is represented by READY (Dynamic REA) (Batra and Sin, 2008). READY focuses on revealing typical interaction scenarios in accounting applications. The READY model (Batra and Sin, 2008) represents predictable activities such as ‘search’, ‘select’, ‘create transaction’, ‘add line items’, ‘review transaction’, and ‘commit transaction’. These generic patterns can be found in key revenue, expenditure, and conversion accounting cycles. By providing patterns of dynamic behaviour of accounting scenarios, the READY model has been shown to be a valid and useful extension of the REA model (Batra and Sin 2008).

There has also been an increased interest in business process patterns specifically in the form of workflows. This greater interest is primarily due to the emergence of the service-oriented paradigm in which workflows are composed by orchestrating or choreographing Web services. van der Aalst et al. (2000) produced a set of so called workflow patterns, advanced workflow patterns (van der Aalst et al., 2000), workflow data patterns (Russell et al., 2005), and workflow resource patterns (Russell et al., 2004). Workflow patterns proposed by van der Aalst are referred to as "Process Four" or P4lists and describe 20 patterns specific to processes. This P4 catalogue provides a comprehensive account of patterns for processes of 'control flow'. This initiative started by systematically evaluating features of workflow management systems and assessing the suitability of their underlying workflow languages. However, as Thom et al. (2007) justly point out, these workflow patterns are relevant toward the implementation of workflow management systems rather than identifying business activities that a modeller can consider repeatedly in different process models. In fact the workflow patterns (van der Aalst et al., 2003) are patterns of reusable control structures (for example, sequence, choice and parallelism) rather than patterns of reusable business processes subject to automation. As such these patterns do not resolve the problems of domain reuse in modelling organisational processes. Consequently, new types of business process patterns are required for reusing organisational process models (Aldin, de Cesare and Lycett, 2009 b, c).

2.4.2 Limitations in Existing Patterns

It can be seen from the background investigation that existing patterns provide limited support to resolving the problems of domain reuse in modelling organisational processes. Although, more and more researchers and practitioners recognise the importance of reusability in business process modelling (di Dio, 2007), little consensus has been reached as to what the essential ingredients of business process patterns should be. Therefore the need arises to provide patterns that support the reuse of BPM, as patterns offer the potential of providing a viable solution for promoting reusability of recurrent generalised models.

An additional limitation is that none of the previous work provides guidelines to modellers as to how these patterns can be discovered. Most of the patterns community mentioned earlier agrees that patterns are developed out of the practical experience of real projects by stating, for example that "patterns reflect lessons learned over a period of time" (Kaisler, 2005, p. 45). During that process someone creates and documents a solution for a certain problem. In similar situations this person refers to the solution that had been documented before and adds new experiences. This may lead to a standard way of approaching a certain problem and therefore constitutes the definition of a pattern.

Thus each pattern captures the experience of an individual in solving a particular type of problem.

In reality, with every new project, analysts create new models without referencing what has already been done in previous projects (Havey, 2000). So, providing systematic support toward the discovery and reusability of patterns in BPM can help to resolve this problem. In addition, writing effective patterns is very difficult, because patterns are not only providing facts (like a reference manual or user's guide), but are also telling a story which captures the experience patterns are trying to convey. A pattern should add value and help its users to comprehend existing models, customise models to fit user needs and help to construct new models.

2.4.3 The Value of Patterns

Due to the fact that patterns have attracted significant interest of researchers and vendors, this section names some advantages found in the literature on using patterns in general as well as on using the business process patterns offered by this research.

Design patterns have two major benefits. First, they provide a way to solve issues related to software development using a proven solution (Agerbo and Cornils, 1998). The solution facilitates the development of highly cohesive modules, which make the overall system easier to understand and maintain. Second, design patterns make communication between designers more efficient. Software professionals can immediately picture the high-level design in their heads when they refer the name of the pattern used to solve a particular issue when discussing system design (Agerbo and Cornils, 1998).

Veryard (2000, p. 209) proposed advantages of using business patterns as they are: "Instrumental: Tells you what to do. Structural: Can be depicted in schematic form. Not just a vague bit of advice. Reusable: Relevant to multiple contexts. Proven: based on experience, not just a bright idea. Business/Enterprise: Meaningful to business not just software people (defining the business benefit that can be derived from the pattern, i.e. why should we re-use this?)".

As patterns have become a widely accepted architectural technique in software engineering. A pattern generally includes a generic definition of the problem, a model solution and the known consequences of applying the pattern. In business process modelling the use of patterns is quite limited. Apart from a few sporadic attempts proposed by the van der Aalst et al. (2003) and Malone et al. (1993; 2003) pattern-based business process modelling is not commonplace. The benefits of adopting patterns are numerous. For example, as the academic literature and industry reports document, the adoption of design patterns in software engineering projects improves reuse of shared experiences, reduces redundant code, reduces design errors and accelerates the learning curve (Cline, 1996). As a consequence, it is conceivable that patterns of BPM can produce similar advantages, first reducing both time and cost of generating business process models and their subsequent transformation into software designs of enterprise applications. Second, improving modelling by replacing an ad hoc approach with a successful one. Third, promote reuse of business processes. Finally, reuse has the longer-term benefit of encouraging and reinforcing consistency and standardisation.

A conceptual technology that has gained popularity recently and that can play a useful role in the systematic discovery as well as the precise representation and management of business process patterns is ontology. Ontologies have the potential of improving the quality of the produced patterns and of the modelling process itself due to the fact that ontologies are aimed at providing semantically accurate representations of real world domains.

2.5 Ontology

Recently, research on ontology has become increasingly widespread in the information systems and computer science communities, (Wand & Weber, 1990) while ontology as a concept has been rather confined to the philosophical sphere in the past. Ontology describes the study of reality and objects independently of any knowledge about them (Gruber, 1993). It also provides a worldview of a particular domain consisting of a set of concepts (e.g., actor, process, resource, etc.), their definitions and inter-relationships. The following sections provide an overview on ontology and the way it is adopted in this research.

2.5.1 From Philosophy to Information Systems

In its original form, the term ontology originates from a branch of philosophy known as metaphysics and is there defined as “ the science of what is, of the kinds and structures of objects, properties, events, processes and relations in every area of reality” (Smith, 2003, p. 3). Ontology is about understanding what things actually exist in the world and how they are related. Although ontological theory has been a focal area of study in the philosophical disciplines for along time, it was explicitly established in the 16th century in philosophy, when ontology was referred to as the "systemic account of existence" (Gruber, 1995 p.3). Philosophical ontology provides a 'sphere of investigation' (Smith, 2003), because it focuses on the need to describe the objects that exist within a domain, by developing theories to provide a description of what 'objects exist' in the real world or in any domain along with the relationships that exist between those objects and their categorisation (i.e., what types of objects exist) (Smith, 2003). Therefore, the main focus is on describing the objects themselves in any domain, and the relations existing among them. In the sense that the focus is not explaining why certain objects exist but to provide a description of reality that is truly exhaustive, which can serve answers for describing any domain and relationships in that domain.

More recently, however, the concept of ontology has been expanded as well as specialised in the fields of computer science and artificial intelligence where ontology is either recognised as an "explicit specification of a conceptualisation" (Gruber, 1993, p.5) or as a "logical theory accounting for the intended meaning of a formal vocabulary" (Guarino, 1998, p.7). Gruber defines specification of conceptualisation as "the objects, concepts, and other entities that are assumed to exist in some area of interest and the relationship between them" (Gruber, 1995, p.1). As a result, a number of domain ontologies have been created over the last two decades, such as the Enterprise Ontology (Dietz, 2006) and Toronto Virtual Enterprise (TOVE) (Fox & Gruninger, 1994) for enterprise modelling, and the REA ontology in the field of accounting (Geerts & McCarthy, 2002). In addition to domain ontologies, there are foundational ontologies such as the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE), task ontologies such as the Web Services Modelling Ontology (WSMO) and application ontologies such as those that describe a specific application domain such as retail banking.

Lately, however, there has been a growing recognition that ontological principles and concepts need not be restricted to the computer science and artificial intelligence fields and that they can be fruitfully applied and developed further in various fields within the broader information systems area (McBride, 2003; Sharman, Kishore and Ramesh, 2007; Guarino, 1998; Jurisica et. al., 2004). In fact ontology can guide the development of new IS by helping analysts and designers choose appropriate processes, rules and software components. Consequently, the IS ontology is not only concerned with the description of the objects, but of those descriptions, what they are and how people perceive something within a domain and what languages are used to describe them. This led to a growing interest in the role that ontologies can play in improving the quality of business process models and of the modelling process itself, because ontology can contribute toward defining a precise and sufficient semantic description model for business processes that encompasses business process description on a higher level of abstraction capable of enabling business level BPM facilities as well as connection to the technical implementation level for supporting automated execution of processes. Ontologies can be used as the underlying data model of IS, in order to ensure semantic interoperability and advanced information processing, meaning that all element descriptions as well as all information interchanged are based on ontologies.

2.5.2 Ontology in Business Process Modelling

Recently it has been recognised that ontology can guide the development of new information systems by helping analysts and designers choose appropriate processes, rules and software components (McBride, 2003; Sharman, Kishore and Ramesh, 2007; Guarino, 1998; Jurisica et. al., 2004). This has led to a growing interest in the role that ontologies can play in improving the quality of business process models. In a way ontology can be thought of as a declarative approach for modelling entities in a domain of discourse since it provides a declarative basis that enables reasoning about what information is and is not available to business process occurrences at a given time. Hence, this level of abstraction enables business level BPM facilities to better connect to the technical implementation level for supporting the automated execution of business processes.

Among the literature that investigates ontologies in BPM at least two groups can be identified. The first group of research adopts ontology as a means to evaluate existing modelling languages and techniques. The second group is more interested in the benefits that ontologies can provide to improve the alignment between business process models and the implemented software technology primarily in the form of Web Services.

Group One: Evaluation of Modelling Techniques via Ontologies

Research in this area is aimed at assessing business modelling languages/techniques via an ontological analysis of the underlying notations. Normally the semantics of the notation is compared against a well-recognised ontology. In most of this work the Bunge-Wand-Weber (BWW) ontology is adopted. Table 2-3 represents core ontological constructs in the BWW representation model. The BWW ontology is an extension of an ontology presented by Bunge (1977) and proposed by Wand and Weber (1990) within the field of Information Systems.

Ontological Construct	Explanation
THING*	A thing is the elementary unit in the BWW ontological model. The real world is made up of things. Two or more things (composite or simple) can be **associated** into a **composite** thing.
PROPERTY*: IN GENERAL IN PARTICULAR HEREDITARY EMERGENT INTRINSIC NON-BINDING MUTUAL BINDING MUTUAL ATTRIBUTES	Things possess properties. A property is modelled via a function that maps the thing into some value. For example, the attribute "weight" represents a property that all humans possess. In this regard, weight is an attribute standing for a property **in general**. If we focus on the weight of a specific individual, however, we would be concerned with a property **in particular**. A property of a composite thing that belongs to a component thing is called an **hereditary** property. Otherwise it is called an **emergent** property. Some properties are inherent properties of individual things. Such properties are called **intrinsic**. Other properties are properties of pairs or many things. Such properties are called **mutual**. **Non-binding mutual properties** are those properties shared by two or more things that do not "make a difference" to the things involved; for example, order relations or equivalence relations. By contrast, **binding mutual properties** are those properties shared by two or more things that do "make a difference" to the things involved. **Attributes** are the names that we use to represent properties of things.
STATE*	The vector of values for all property functions of a thing is the state of the thing.
TRANSFORMATION*	A transformation is a mapping from one state to another state.
STABLE STATE*	A stable state is a state in which a thing, subsystem, or system will remain unless forced to change by virtue of the action of a thing in the environment (an external event).

Table 2-3: Core ontological constructs in the BWW representation model (from Wand & Weber, 1990)

This model articulates a set of high-level, abstract constructs that are intended to be a means of representing all real-world phenomena. BWW (Wand & Weber 1990; Weber, 1997; Wand et al., 1995a) adapted the ontology framework into a theory of representation that is closer to the demands and terminology of the Information Systems community (Recker et al., 2006). The BWW model has over the years reached a significant level of maturity, adoption and dissemination, and has been used in many research projects, especially in the context of the analysis and evaluation of various process modelling techniques

Keen and Lakos (1996) determined essential features for a process modelling scheme by evaluating six process modelling techniques in a historical sequence by using the BWW representation model. Among the modelling techniques evaluated were: Flowcharts, Data Flow Diagrams (DFD), Integrated DEFinition (IDEF3) and Petri Nets. The evaluation is restricted to the assessment of the ontological completeness of each technique. From the analysis the authors concluded that, in general, the BWW ontology facilitates the interpretation and comparison of process modelling techniques. The authors did not, however, empirically verify their findings on the features of process modelling schemes and in choice of appropriate modelling constructs.

Green and Rosemann (2005) also extended the use of BWW to the area of enterprise systems interoperability and specifically to represent business process modelling languages like Electronic Business using extensible Mark up Language (ebXML), Business Process Modelling Language (BPML), Business Process Execution Language for Web Services (BPEL4WS), and Web Service Choreography Interface (WSCI). All these standards, which proclaim to allow for specification of intra- and inter-organisational business processes, have been analysed in terms of their ontological completeness. However, this study found that some of these modelling languages provide a wider range of language constructs for specification requirements, while others overlap. At present, their analysis too, has not yet been empirically validated.

Recker and Indulska (2007) used the BWW ontology to evaluate the representational capability of Petri Nets. The findings of this study shows that Petri Nets suffer a number of deficiencies classified as construct deficit, construct redundancy and construct overload in modelling real world domains. Also, this study found that a large number of BWW ontology constructs do not appear to be represented in Petri Nets. This implies that Petri Net are not able to model various real world concepts that are covered by the BWW constructs. However, it is worth mentioning that according to Recker and Indulska (2007) this study needs to be empirically tested to confirm which of the proposed Petri Nets weaknesses have or do not have effects in the domain of process modelling as the outcome of this study represents just a significant first attempt at the evaluation of Petri Nets

A twofold criticism can be raised against these studies that adopt the BWW to validate modelling languages and techniques. On the one hand, concerns remain on a lack of understandability of the BWW constructs, the problematic application of these constructs to other loosely defined modelling grammars, and the limited empirical testing of the implications of the BWW models (Green and Rosemann, 2000). In addition, no examples have been given by BWW of ontologically completed modelling of business processes (Keen & Lakos, 1996). As an ontological analysis of a modelling technique requires not only detailed knowledge of the selected ontology and technique, but also a good understanding of the languages in which the ontology and the grammar are specified. On the other hand, the research mentioned above has mostly been of a purely theoretical nature. Most of the evaluations lack empirical verification of the theoretical findings (Recker et al., 2005). For a much more comprehensive treatment of the ontology interested readers are encouraged to refer to the use of BWW by Weber and Zhang (1996), Wand & Weber (1995), Parsons & Wand (1997), Green & Rosemann (2004).

While the line of research summarised above adopts the Bunge-Wand-Weber ontology as a means to assessing modelling tools (Green & Rosemann, 2000), methods (Green, 1996), and interoperability standards (Green et al., 2005) (as well as for the selection and introduction of standard software (see Soffer et al., 2001)), another stream of research has applied ontologies as a way of modelling business processes and aligning business systems with the underlying software applications.

Group Two: Ontologies for the Alignment of Business and Software Systems

The potential of adopting ontologies as the foundation to modelling process models has been recognised in the literature for quite a while now (Hepp et al., 2005; Lin & Strasunskas, 2005; Hepp & Roman, 2007). In most cases the authors try to ontologically create process representations aimed at facilitating the automated processing of procedural models. Most of these works are geared toward a certain language, i.e. they only deal with the semantic annotation of process models represented with the help of a certain language.

The Semantics Utilised for Process Management within and between Enterprises (SUPER) is a European Union funded project (http://www.ip-super.org/) in which the consortium's aim was to develop innovative approaches for business process management using Semantic Web standards (Hepp & Roman, 2007). With the aim of bridging the gap between businesses and the information technology (IT) world the project sought to enable at least semi-automation of the business process management lifecycle using Semantic Web and Semantic Web services technologies (Wetzstein et al., 2007). The SUPER ontology is aimed at providing an extensive conceptualisation of the BPM domain, ranging from process modelling to the definition of business strategies. This is done by providing a set of constructs to generate semantic business process models via the application of ontologies, which describe enterprise models and business processes in general. The project also demonstrates the potential benefits of the application of ontologies for companies. In the context of the SUPER project, in order to formally represent business process knowledge, several ontologies for different languages such as BPMN, BPEL, Event-driven Process Chain (EPC), Petri Nets, etc. are proposed to cover not only behavioural aspects, but also organisational, functional or data perspectives. Figure 2-2 represents the ontological framework (BPMO) that is used to identify the elements that BPMN consists of and to annotate them ontologically using SUPER framework.

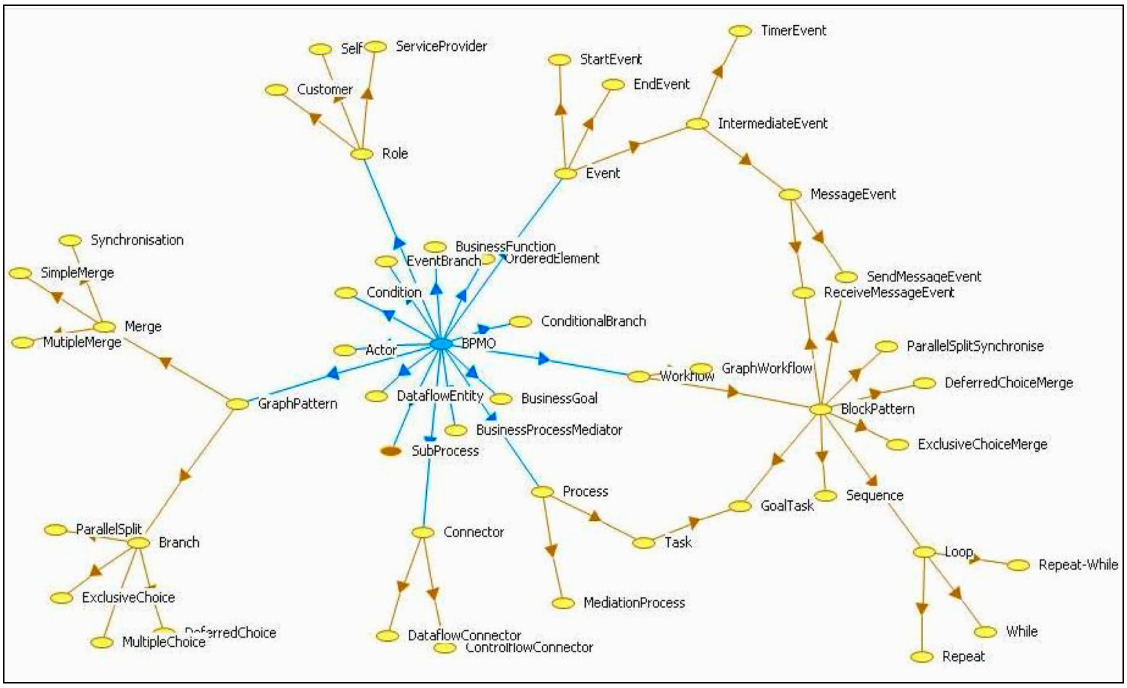


Figure 2-2: Ontological Framework of BPMO Class Hierarchies of SUPER (from Pendrinaci et al, 2008)

Brockmans et al. (2006) proposed an approach for semantically aligning business processes, by modelling processes using Petri nets and providing a representation of the Petri nets in OWL as a means of semantically enriching the business process models.

Petri Net notation has been well presented in ontology, although it only provides semantic annotation without understanding the underlying meaning of the process models.

In Thomas and Fellmann (2006) a proposal to annotate EPCs with semantics (sEPC) is presented. This proposal includes four instances of ontologies named ‘Business Ontology’, ‘Business Process Concepts’, ‘sEPC model’ and ‘the underlying EPC model’. From the analysis the authors concluded that, further research is needed regarding suitable ontologies and tools for the annotation of process models. Therefore, a prototype for a sEPC repository is currently under planning and it will provide interfaces or plug-ins for well-established modelling tools. The authors did not, however, empirically verify their findings on how well the mapping of EPC notation with ontology.

Lautenbacher & Bauer (2006) demonstrated how UML activity diagrams could be annotated with inputs, outputs, preconditions and effects (functional semantics) to start an automatic synthesis of business processes. Their finding is that UML2 activity diagrams and other diagrams currently could not be supported. Especially diagrams to cover organisational or resource aspects should be included and additional semantics and ontologies created (e.g., for the organisational structure of the company) to provide real mapping between business processes built using activity diagrams and their ontological representation.

The main limitation of this second line of research is that the ontologies produced provide semantic annotations for business process modelling languages without describing and presenting the knowledge of the domain presented (Hepp & Roman, 2007). Most of these works are designed toward presenting a certain language, i.e. they only deal with the semantic annotation of process models represented with the help of a certain modelling language (Thomas & Fellmann, 2009). An additional limitation of SUPER is that it considers the business process models as a given, which affects the requirements engineering methodologies to acquire correctly semantically annotated business process models (Decreus & Poels, 2009).

In reality, there are other ontologies aside from those mentioned earlier, but there is no generally accepted ontology (Wand et al., 1995a). A different ontology may employ different concepts and constructs, which can lead to different outcomes. Despite this, the vision is that ontologies can contribute toward improving the adaptability (flexibility and agility) of business change and consequently improve alignment between requirements of the organisation and its enterprise software systems (see Hepp & Roman, 2007). While this section shows some of the ontological approaches used with BPM, the next section will present the object paradigm that is adopted for this research.

2.6 Object Paradigm

Based on what has been said previously, an ontology of business processes that is firmly grounded in the real world nature of organisations needs to be produced. The paradigm that is adopted in this research is perduratism according to which the existence of a thing is determined by its extension in this universe, namely a spatio-temporal extension. For example, an individual having a temporal part consisting of all of him/her throughout 1990. That part of the individual starts to exist in on 1st January 1990 and then ceases to exist on 31st December 1990. And if that individual lived to 80 then s/he would be composed of 80 such one-year long temporal parts.

The Object Paradigm (Partridge, 1996) adopts the principles underlying perduratism (Sider, 2001). The object paradigm (not to be confused with the object-oriented paradigm) determines the existence of an object (or thing) through its spatio-temporal extension. Partridge (1996) argued that within the business domain these objects could be narrowed down to four different types as follows:

- **Particular things** are individual objects that exist, tangible and can be seen. For example, seeing a particular object like a table or chair.
- **General types of things** are collections of objects that portray similar properties. Also, a particular thing can be said to be an instance of it. For example, a table type that is a collection of all the tables in the world.
- **Relationships between things**: Objects can be linked to other objects and can extend to two or more objects. Relations can again be between particular and general. Particular relations exist between two particular objects, for example, Queen Elizabeth is the mother of prince Charles and general relations of which particular relations are instances like the relation mother-son.

- **Changes happening to things**: Change happens to objects on the occurrence of a certain event, and as a result, an object changes from one state to another. To provide a clear reflection of the objects that exist within this world it is important to capture and manage change (i.e., states and events). For example, a change such as a green tomato turning red. The change (turning red) happens to the tomato (a particular thing).

These four types of objects form the basis for constructing the base typology (i.e., top-level ontology) of the object paradigm, and the extension principle can be applied to different types of objects that were identified to exist within the business domain. The use of extension is important as the spatial aspect of the object is real and cannot be doubted. Moreover the use of extension complements reference as without extension there will not be anything tangible to refer to and also it gives a strong sense for reference. As extension allows pointing to what is being referred to, the specification is far more accurate and tangible. Along with recognising the spatial aspect of an object, the object paradigm also recognises that objects are often time bounded and they change with time. Change is an important aspect and given that everything within this world changes at some point in time. So managing change requires the ability to clearly model objects so that they can deal with change. In a coherent manner, the object paradigm recognises this need and treats the temporal dimension as another dimension along with the spatial dimensions. As a result, extensions of objects are not only just based on the spatial dimensions but also on the temporal dimension.

Daga et al. (2005a) proposed an ontology-based approach that focuses on the extraction of business content from existing system and improving it along several dimensions. This approach is based upon a work described in Partridge (1996), where it provides a way to re-engineer existing systems into objects. Thus, Daga’s ontology-based approach provides:

- Four Dimensional Extensions, the use of extension is important as the spatial and temporal aspect of the object is real and cannot be doubted. Moreover it complements reference, as without extension there will not be anything tangible to refer to which will give a strong sense for reference.
- Strong referencing, the principle ensures that the ontological model that is created is a direct and explicit reflection of the world. This is reflected on the limited ability of

the current approaches to clarify the real world semantics and suggests the need to use the reference principle as a primary means of clarifying the real world semantics.

- Clarifies the structure of information that the business consists of and making those concepts explicit, by describing and understanding the relevant objects that make up the domain and change these assumptions easily if the domain changes.
- Empirical studies in several previous researches showed the usefulness of this ontology.

This paradigm constitutes the foundation to the definition of Business Process Ontology (BPO) presented in Section 5.3. In the context of this research, the two main uses of ontology are identified: (a) For communication between implemented computational systems and its business process models, between humans, and vice-versa and (b) For reuse (and organisation) of knowledge, e.g. structuring or organising libraries or repositories of patterns.

2.7 Literature Findings and Research Direction

In this thesis the literature on business process modelling was reviewed with an emphasis on new lines of research in BPM that can contribute toward promoting the reusability of business process models as well as rendering BPM more systematic yet flexible enough to cope with continuous change. These new lines of research include patterns for BPM, process discovery, declarative modelling and ontologies (Aldin and de Cesare, 2010).

As BPM becomes more extensively adopted in organisations the need for more efficient and effective methods and techniques to process modelling becomes greater. While traditionally process models were mainly created from scratch where and when required, with the increase in the rate of change that modern organisations undergo, the focus now lies on producing quality process designs more rapidly and better aligned with an enterprise’s information system. Research has already started addressing this challenge and suggested some novel approaches to the problem. While most of the research to date does not provide a complete solution, as the limitations mentioned in the previous sections highlight, various elements of all approaches surveyed have together the potential to produce a viable solution to the problem.

The solution that is envisioned here is the proposal of a methodology aimed at discovering process patterns from diverse organisational assets and capable of modelling such patterns in a way that facilitates their reuse in the design or redesign of organisational processes. The literature surveyed in this thesis and the themes that can be drawn from it contribute toward the definition and development of a methodology for systematically discovering and using generalised patterns of business processes.

Business process patterns are essentially models representing general organisational behaviour. This means that when identifying patterns it is necessary to undergo a process of discovery not dissimilar to the process of scientific discovery. While the latter results in the formulisation of laws or theories that help us understand the physical universe, business process pattern discovery allows us to understand common ‘chunks’ of organisational behaviour that can be specialised and instantiated when required. As Section 2.3 highlighted, current research is limited to the discovery of processes from event logs. This is useful but insufficient for a systematic discovery of process patterns. In fact, as previously observed, event logs often provide very limited data as well as incomplete information (Section 2.3.1). Although process mining in itself may not be sufficient as it stands, it must be noted that the core idea and principles underlying this technique should be transposed so as to identify ways in which process patterns can be discovered from other organisational assets such as people, documentation, legacy systems and so on.

If process patterns are to be flexibly applied then it may be necessary that such patterns not be rigidly encoded, as procedural process models tend to be, i.e. with rigid execution paths and outcomes. Declarative models may provide the foundation for constraint-based patterns that provide support for foreseen and unforeseen process flows in which variants of a typical class of business process may emerge. Such models would be based on both logical as well as graphical types of representations of business processes. The former is typically the type of representation envisioned by the Semantic Web community in the representation of ontologies.

Since the business process patterns must be reused, their representations must be as clear and unequivocal as possible. This implies semantic precision or correspondence between a model and the real world system it represents. Ontologies are semantic models and tend to represent real world systems so that there is a one-to-one mapping between the symbols in the model and the real world things that those symbols represent. In order to discover process patterns, an ontology of business processes is required. This ontology would drive the interpretation of organisational data during process mining and discovery. Moreover the elements of the discovered patterns would be individually classified according to the classes defined in the business process ontology (Aldin and de Cesare, 2010).

In conclusion, the primary advantage of the proposed methodology in this research is that it allows people to explicitly represent the similarities among related processes in the form of patterns, and to easily find or generate sensible alternatives for how a given process could be performed and reused.

2.8 Summary

The literature review has uncovered relevant concepts and aspects in the areas of business processes, patterns and ontology. Consequently, the literature review provides the context and the grounding for the research presented in the remainder of the thesis. The limited research on patterns reuse for business process modelling, other than a few attempts, raises the need to better understand how patterns can be applied to business process model reuse. Patterns reuse and discovery are two interrelated problems that require specialised solutions. From the limitations noted above, methodologies neither exist for collecting and analysing process models of business organisations, nor for generating and utilising reusable patterns within organisations. The adoption of ontologies in BPM integration and distribution is also limited (Pedrinaci et al., 2008). For these reasons, a new empirical and ontologically based methodology will be introduced in subsequent chapters as a way to discover or identify reusable process patterns in an effective manner.

CHAPTER 3 - DESIGN RESEARCH METHODOLOGY

3.1 Introduction

This chapter presents the research methodology used for conducting the study in this research as well as an explanation of the work process and motivation behind the steps taken to achieve the result. According to Benbasat and Weber (1996, p. 392), “research methods shape the language we use to describe the world, and language shapes how we think about the world”. Therefore, suitable research methods are needed to guide the investigation. The research methodology adopted in this study is Design Research (DR) (March and Smith, 1995; Hevner et al., 2004). Design Research consists of the construction and the evaluation of the artefacts that resolve a significant recognised problem. Also, it provides a suitable and comprehensive framework for the design and the analysis of artificial phenomena such as organisations or information systems. It defines the research subjects and the methods applied to the subject in order to systematically enhance the body of knowledge (March and Smith, 1995).

This chapter is structured into two parts. The first part begins with Section 3.2, which explains the relationship between natural science and design science within the information systems (IS) discipline. It offers a comparison of design and natural science and argues for the importance of design science within information systems. Section 3.3 presents a short history of Design Research and positions design-led research within the IS community. It then describes the design artefacts of the research. Section 3.4 explains the research methodology. Section 3.5 explains details of the evaluation methods and metrics for Design Research. The second part of the chapter begins with Section 3.6, which describes the use of Design Research in this work. Section 3.7 discusses the awareness of the problem and types of solution the first stage of the research plan. Section 3.8 explains the individual iterations within the development stage. Section 3.9 represents the final evaluation stage of this research. Finally, a summary of the chapter is presented in Section 3.10.

3.2 The Logic of Design Research

A number of researchers, both inside and outside of the information systems discipline, have sought to provide some guidance to define Design Research. DR has been applied to engineering, computer science (Lee et al., 2008; Kuechler et al., 2005) and IS (Nunamaker, et al., 1990). Herbert Simon introduced design science in his book “Sciences of the Artificial” in 1969. His work inspired an entire body of research embracing goals to design and invent innovative artefacts such as constructs, frameworks, models, methods, processes and systems. Simon (1996) stated that, it is possible to create a science of the artificial (i.e., human-made) as an analogue to natural science; hence the term has been defined as ‘design science’.

According to Simon (1996), natural science is concerned with the state of natural things, how they are and how they work. On the other hand, design science is concerned with furthering our understanding of how things come about. Things in this context are artefacts, in view of the fact that “*an artefact progresses from an idea to things in the world*” (Purao, 2002, p.13). Design science seeks to extend the boundaries of human and organisational capabilities by creating new and innovative artefacts. Thus, design science is “*an attempt to create things that serve human purposes*” (March & Smith, 1995, p. 253).

There are two kinds of scientific interest in information technology (IT), descriptive and prescriptive. “Descriptive research aims at understanding the nature of IT… Prescriptive research aims at improving IT performance… Though not intrinsically harmful, this division of interests has created a dichotomy among IT researchers and disagreement over what constitutes legitimate scientific research in the field” (March & Smith, 1995, p.252). Descriptive research and prescriptive research correspond to natural science and design science respectively. Interestingly, Simon (1995, p. 96-8) points out a similar division of interests in the field of artificial intelligence (AI), which he refers to as the ‘social fragmentation of AI’.

Rather than argue over what constitutes legitimate scientific research, March and Smith (1995, p. 251) state that "*both design science and natural science activities are needed to ensure that IT research is both relevant and effective*". These alternative views motivate March and Smith (1995) to create the design and natural science framework for IT researchers. March and Smith (1995) recognise the importance of both types of scientific activities, which has been explained further in the following section. A unified approach, 'Design Research', that combines both design science and natural science perspectives that would bring greater relevance to IS research.

3.3 Design Research in Information Systems

Design Research is becoming firmly established as a research paradigm in several disciplines related to information sciences, information systems and technologies. A good starting point to design research in IS is provided by March and Smith (1995). Where they define design science as "*an attempt to create things that serve human purposes*" (March & Smith, 1995, p. 253). As opposed to natural science, which "*tries to understand reality*" (March & Smith, 1995, p. 253). However, design science and natural science are presented in parallel in March and Smith's framework for the research in IT.

March and Smith outline a design research framework with two dimensions, namely research activities and research outputs/artefacts (see Figure 3-1). In relation to the latter, it is based on Design Research outputs and it covers constructs, models, methods and instantiations. Those outputs represent the research artefacts and they are the result of the design research in IS discipline. As for the former, it is based on design and natural science and its activities comprise building, evaluating, theorising on and justifying artefacts. The artefacts of Design Research are classified into the four categories and take part or are produced in one of four suggested activities.

The first dimension of the framework is based on design science research outputs or artefacts. The research outputs are solution concepts and this solution is for addressing the research problem. March and Smith (1995) identified four essential types of artefacts that are important for the design research framework and which are recognised as valid research outputs of the design research process. These four types are *constructs*, *models*, *methods* and *instantiations*. In the following a detailed overview of these types are given as the work of March and Smith (1995) is widely recognised and built upon

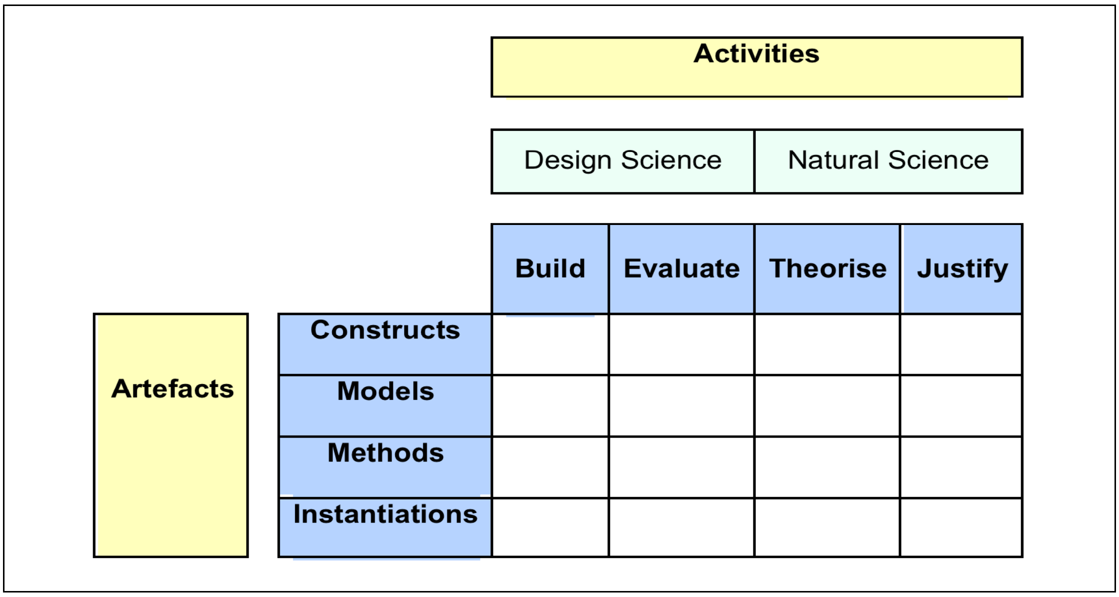


Figure 3-1: Design and Natural Science Framework (March & Smith, 1995)

- ***Constructs*** are the vocabulary and conceptualisation of a domain used to describe the problems and specify their solutions within the domain. These conceptualisations are important for both natural and design science as they define the terms used when describing and thinking about tasks. These constructs can be valuable aids, but also inhibitors that prevent researchers and practitioners to see crucial issues.
- ***Models*** in design science research, are a set of statements that express the relationships among constructs (March & Smith, 1995). Models are used to represent the problem and solution situations. The key criteria are that the model be useful in representing and communicating information system requirements and for developing artefacts to serve human purposes (March & Smith, 1995). In the notion of design science the model's function is utility not truth, in contrast to the model definitions in natural science, in which the model's function is to describe.
- ***Methods*** define processes that search the solution space to solve a problem. These methods can be formal, mathematical algorithms or informal descriptions that act as a guideline. Based on a set of underlying constructs and models of a solution space a method is a set of steps used to perform a task (March & Smith, 1995). Models are often inputs to methods, which can be used to translate from one model to another in the course of solving a problem. For example, system development methods

facilitate the construction of models of user needs and then translate that model into other models such as system requirements, system specifications and finally into an implementation (March & Smith, 1995). Also, methods are inputs to models, which can be used to produce models from the methods. For example, the SDR methodology (defined in this thesis) produces models that help to resolve a problem. The desire to use a particular method can influence the constructs and models that are developed

- ***Instantiations*** are the realisation of the artefacts in their environment. It shows that the constructs, models and methods can be implemented in the real world as a working system. Instantiations in IS research can be specific information systems or tools that address various aspects of designing information systems.

In Figure 3-2 the relation between these different artefacts in the design research framework are shown to illustrate the dependencies between them.

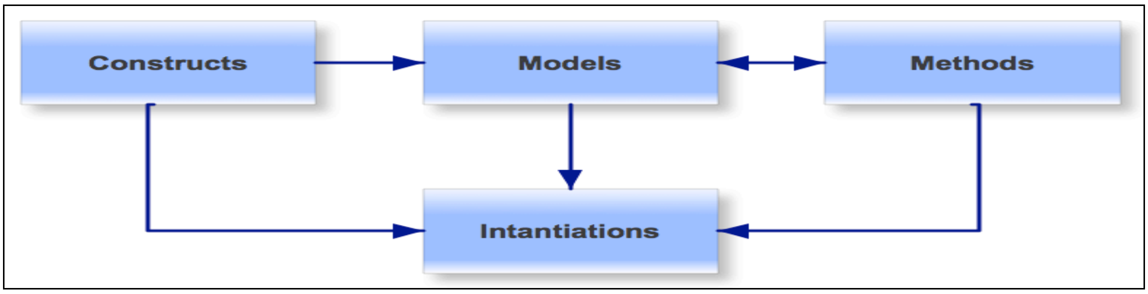


Figure 3-2: Relationship between Design Research Framework Artefacts

The second dimension of the framework concerns research activities. March and Smith (1995) identify *build* and *evaluate* as the two main activities in design science.

- ***Build:*** refers to the construction of constructs, models, methods and artefacts demonstrating that they can be constructed.
- ***Evaluate***: refers to the development of criteria and the assessment of the output's performance against those criteria.

Parallel to these two research activities in design science, March and Smith (1995) add the natural and social science couple, which are *theorise* and *justify*.

- ***Theorise***: refers to the construction of theories that explain how or why something happens. In the case of IT and IS research this is often an explanation of how or why an artefact works within its environment.
- ***Justify***: refers to theory proving and requires the gathering of scientific evidence that supports or refutes the theory.

Summarising, constructs, models, methods and instantiations are *built* to perform a specific task. These artefacts then become the object of study, which must be evaluated scientifically. They have to be *evaluated* in order to conclude if any progress has been made. In order to do this, it requires developing metrics and measuring the outputs according to those metrics. For instance, when an artefact has been applied in a specific environment, it is important to determine why and how the artefact worked or did not work (March & Smith, 1995). Such research applies natural science methods to artefacts in order to *theorise*. Then, given a generalisation or theory, which must *justify* that explanation, evidence has to be gathered to test the theory in question. Justification for artefacts generally follows the natural science methodologies governing data collection and analysis.

Hevner et al. (2004, p. 76) state that "there are mainly two paradigms which characterise the research in the information systems discipline, which are behavioural science paradigm and design science paradigm." According to Hevner et al. (2004, p. 76) the behavioural science paradigm is more focused on the "development and verification of theories that explain or predict human or organisational behaviour." On the other hand, the design science paradigm seeks to extend the boundaries of human and organisational capabilities by creating new and innovative artefacts (Hevner et al., 2004, p 75). Building on March and Smith (1995), Hevner et al. (2004) presented a design research framework and guidelines around building and evaluating IT artefacts.

Moreover Hevner et al. (2004) expressed their view on what constitutes good, rigorous and relevant design research in the form of seven guidelines. The authors contend that each of the guidelines should be addressed in some manner for design research to be complete. The guidelines are on *the product of design research*, *the relevance of the problem, the criteria for evaluating the design solutions*, *the research contributions*, *the* *research rigour*, *and design as a search* and *the communication* of the research (Hevner et al., 2004). In Table 3-1 Design Research guidelines are shown.

Guidelines	Description
Guideline 1: Design as an Artefact	Design science research must produce a viable artefact in the form of a construct, a model, a method, or an insanitation.
Guideline 2: Problem Relevance	The objective of design science research is to develop technology-based solutions to important and relevant business problems.
Guideline 3: Design Evaluation	The utility, quality, and efficacy of a design must be rigorously demonstrated via well-executed evaluation methods.
Guideline 4: Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artefacts, design foundations, and/or design methodologies.
Guideline 5: Research Rigour	Design science research relies upon the application of rigorous methods in both the construction and evaluation of the design artefacts.
Guideline 6: Design as a Search Process	The search for an effective artefact required utilising available means to reach desired ends while satisfying laws in the problem environment.
Guideline 7: Communication of Research	Design science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

Table 3-1: Design Research guidelines (from Hevner et al., 2004)

3.4 Design Methodology

Design Research methodology, as presented by Vaishnavi and Kuechler (2005), follows few phases. These phases are: *Awareness*, *Suggestion*, *Development*, *Evaluation* and *Conclusion*. Each of these phases has an output, which respectively are: a *Proposal*, *Tentative Design*, *Artefact*, *Performance Measures* and *Results*. For the sake of simplicity, each step of the design life cycle is introduced in Figure 3-3 (Vaishnavi & Kuechler, 2004) and explained as follows:

- ***Awareness of problem:*** This step is the beginning of the methodology of the Design Research. This stage is to identify and define the problem that will be addressed by the proposed research. The problem needs to be properly defined so that it is notable and worthy of further investigation. It also should be properly scoped so that the solution can be properly developed and effectively evaluated.

- ***Suggestion***: This step is a creative process in which designers envision a potential solution for the problem intended to be solved. This is an exploratory phase to gain further insight into the problem domain and form a basic solution through initial analysis and design.
- ***Development***: This step is where the artefact is implemented accordingly to the suggested solutions from the previous step.
- ***Evaluation***: This step is responsible for testing the newly implemented artefact against the specification in the proposal step. If the result is considered unsatisfactory, the design cycle goes back to the *awareness of problem* step for reviewing the initial problem and bringing together the new knowledge (i.e., circumscription) acquired throughout the steps. Otherwise, if the result is satisfactory, the cycle moves on to the conclusion step. This loop may happen many times until the evaluation of the artefact is considered satisfactory.
- ***Conclusion***: This step is the end of the design research cycle. Once the artefact has been built and the evaluation satisfactory, the designer will put together the knowledge acquired throughout the design cycle, providing guidelines (i.e., body of knowledge) for practitioners to use the artefact in their field. In both phases, guidelines (i.e., methodology) are generated explaining how to use the artefacts in similar situations.

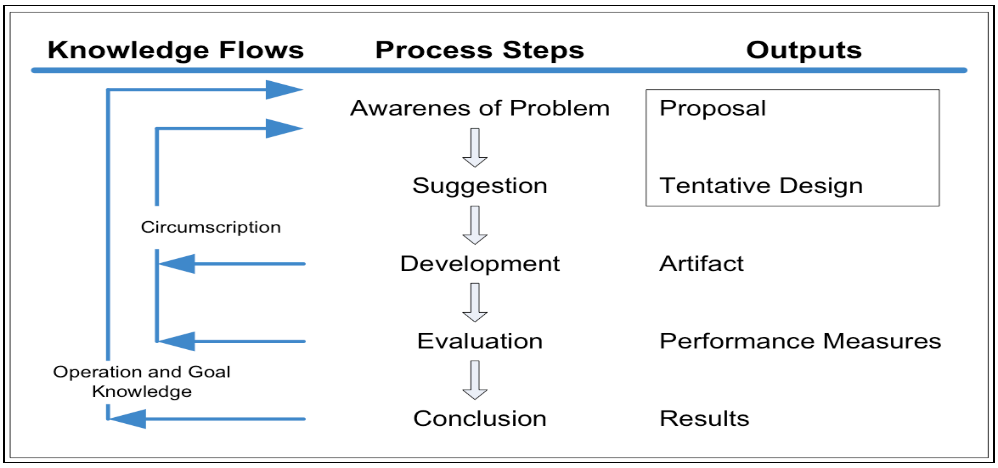


Figure 3-3: General Methodology of Design Research (from Viashnavi & Kuechler, 2004)

"Design Research involves the analysis of the use and performance of designed artefacts to understand, explain and very frequently to improve on the behaviour of aspects of Information Systems" (Vaishnavi & Kuechler, 2004, p. 493). Therefore Design Research, as a methodology provides empirical data gathered from which an innovative artefact can be designed. Hence, it is reasonable to identify outputs such as constructs, models, methods and instantiations form applying Design Research methodology and to realise how these artefacts can be determined.

3.5 Design Evaluation

Having outlined the artefacts and methodology of Design Research, attention is now paid to how artefacts can be evaluated. An important aspect of Design Research is the evaluation of the proposed artefacts; in other words the utility of the proposed artefacts must be demonstrated.

According to March and Smith (1995) building and evaluating IT artefacts have Design Research intent. Evaluation refers to the development of criteria and assessment of artefact performance against those criteria. Evaluation requires the development of metrics and the measurement of artefacts according to those metrics. Metrics measure the outcomes of the research.

March and Smith (1995) classified artefacts as follows: constructs, models, methods and instantiations and proposed some evaluation criteria for them. They strove to give as universal metrics as possible. See Table 3-2 for the evaluation criteria according to March and Smith (1995).

The choice of the criteria will depend on the reason for developing the artefacts in the first place, which will be related to the original research objectives. The evaluation can lead to conclusions about the design process as well as the design product, and may suggest further modification to either or both. However, Hevner et al. (2004) concluded that artefacts could be evaluated in terms of effectiveness, utility, functionality, completeness, consistency, accuracy, reliability, quality and usability. Table 3-3 presents categories of design research evaluation methods (Hevner et al., 2004) for evaluating the quality and the effectiveness of an artefact.

Artefacts ***Metric***	**Constructs**	**Models**	**Methods**	**Instantiations**
Completeness	✓	✓	✓	
Simplicity	✓			
Elegance	✓			
Understandability	✓			
Ease of use	✓		✓	
Validity	✓			
Fidelity with real world phenomena		✓		
Level of detail		✓		
Robustness		✓		
Consistency		✓	✓	
Operationality			✓	
Efficiency			✓	✓
Generality			✓	
Effectiveness			✓	✓
Quality of result			✓	

Table 3-2: Evaluation Criteria according to March and Smith (1995)

The selection of evaluation methods should match appropriately with the designed artefact and the selected evaluation metrics, with direction from the requirements of the business environment under investigation (Hevner et al., 2004).

Guideline	**Description**
1. Observational	**Case Study** – Study artefacts in depth in business environment
	Field Study – Monitor use of artefacts in multiple projects
2. Analytical	**Static Analysis** – Examine structure of artefacts for static qualities (e.g., complexity)
	Architecture Analysis – Study fit of artefacts into technical IS architecture
	Optimisation – Demonstrate inherent optimal properties of artefacts or provide optimality on artefacts behaviour
	Dynamic Analysis – Study artefact in use for dynamic qualities (e.g. performance)
3. Experimental	**Controlled Experiment** – Study artefacts in controlled environment for qualities (e.g. usability)
	Simulation – Execute artefacts with artificial data
4. Testing	**Functional (Black Box) Testing** - Execute artefacts interfaces to discover failure and identify defects
	Structural (White Box) Testing – Perform coverage testing of some metric (e.g. execution paths) in the artefact implementation
5. Descriptive	**Informed Argument** – Use information from the knowledge base (e.g. relevant research) to build a convincing argument for the artefact's utility
	Scenarios – Construct detailed scenarios around the artefacts to demonstrate its utility

Table 3-3: Design Research Evaluation Methods (from Hevner et al., 2004)

3.6 The Practical Application of DR

3.6.1 An Abstract Form of Design Research

This section describes the board outline of the developmental research phases as conducted in this research by translating the general methodology of DR (Vaishnavi & Kuechler 2004) into three main practical stages: (1) *Awareness of the Problem and Types of Solutions*, (2) *Development* and (3) *Final Evaluation*. The development stage consists of three iterations of (1) *Design*, (2) *Deployment*, and (3) *Evaluation*. Figure 3-4 illustrates the broad outline of the developmental research phases. These iterations are repeated in the *Development* stage as three iterations of research are planned to realise the proposal of evolving (generating) a methodology for identifying patterns of business processes from enterprise knowledge sources. Awareness of the problem space first forms the understanding of the problem to undertake the type of solutions that forms a proposal for the research in the *Awareness of the Problem and Types of Solutions* stage. Then in the *Development* stage, iterations deliver improvement in the artefact(s) as more knowledge is acquired that evolves awareness of the problem. A *Final Evaluation* stage leads to the conclusion about the design process as well as the design product and may suggest further modification to either or both.

Iterations in the *Development* stage begin with a *design* phase. For the first iteration this uses the first version of the Semantic Discovery and Reuse (SDR) methodological framework and draws requirements from the *Awareness of the Problem and Types of Solutions* stage. The iteration then cycles repeatedly through iterations of *Design*, *Deployment* and *Evaluation* to evolve an improved and more effective version of SDR. The iteration stops when either the process is interrupted, as inaccuracies in the artefacts are identified, or an effective solution is found. *Evaluation* of the artefact generates knowledge and learning that forms feedback to the next iteration. The feedback helps to inform the next version of the SDR methodological framework that then becomes the input into the next iteration. Analysis of the artefact and the research approach during the iteration occurs at this point, producing emergent assumptions regarding how the artefact ought to influence the phenomenon. The feedback is then passed to the next iteration to influence the research design. For the *Final Evaluation* scenarios are used in order to raise the discussion on the importance and limitation of the developed patterns in this research.

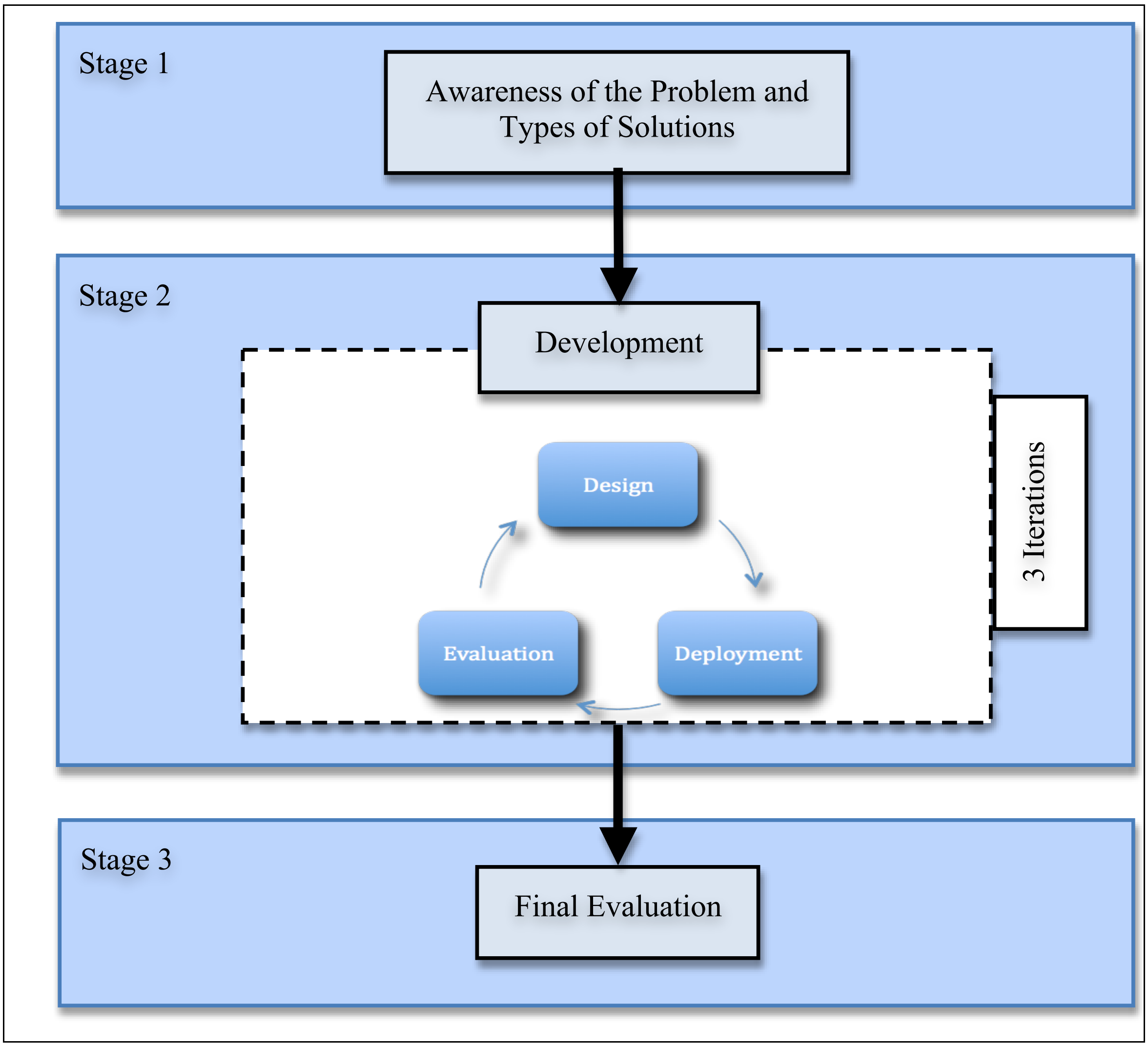


Figure 3-4: Broad Outline of the Overall Research

The following sections outline the problem space that the artefact is constructed to resolve with respect to the discussion in Chapter 2. The initial suggestion is provided within the *Awareness of the Problem and Types of Solutions* stage section, where the suggestion is abducted from the knowledge base and proposed as a solution to the problem space. In the *Development* section the iterations are clarified and form the research designs for Chapters 4, 5 and 6. Each chapter includes a research outline for the iteration consisting of the practical application of the methodological framework within a specific domain and outputs generated by the iteration and evaluation criteria to determine artefact utility. The emerging output is communicated within the evaluation section to describe how the artefact performs for the problem and how the solution may be improved in the next iteration. In the last section the *Final Evaluation* of the research is described.

3.6.2 Research Proposal

With regards to the hypothesis about existing patterns explained in Chapter 2, it has been found that existing patterns do not resolve the problems of domain reuse in modelling organisational processes (see subsection 2.4.2). Also, none of the previous work on patterns provides guidelines to modellers as to how these patterns can be discovered and reused. This encourages the proposal of this research, which is to develop and evaluate a methodology for empirically deriving ontological patterns of business processes from organisational knowledge sources (i.e., documentation, systems, domain experts, legacy application, etc.). A methodology that contributes to business modelling in IS development with a unique type of solution consisting of a systematic methodology for uncovering patterns of business behaviour from organisational knowledge assets in an empirical fashion. From this proposal the aim of this research is defined as follows:

To investigate the effectiveness of business process patterns in business process modelling, by developing and evaluating a methodology for empirically deriving ontological patterns of business processes from organisational knowledge sources.

This leads to drive the following research objectives:

Objective I: Evolve a methodological framework for identifying patterns of business processes from enterprise knowledge sources (i.e., documentation, systems, domain experts, etc.).

Objective II: Derive a set of generic business process patterns by applying the developed methodology and to organise the patterns within the repository.

Objective III: Investigate the development of an ontology of business process to be used for the derivation of the semantic process modelling.

Objective IV: Evaluate the methodological framework and derived patterns in terms of their significance to theory and practice through recognised evaluation techniques.

As a broad outline of the developmental research phases various stages are included. The first stage of *Awareness of the Problem and Types of Solutions* derives its initial requirements from the objectives stated in Section 3.6.2. Then the *Development* stage is defined by three iterations of *Design*, *Deployment* and *Evaluation*. Each of these iterations then derives its requirements from feedback of the iteration that precedes it. The *Final Evaluation* stage describes how the artefact performs for a specific domain problem, and how the solution has improved the organisational process.

3.6.3 Data Collection Sources

In order to provide an effective solution to the business problem, suitable data sources are required. Thus, data was acquired from: (1) multiple systems (retail banking, insurance and mortgages) from the financial services domain and are summarised in Section 4.2, and (2) two handbooks (staff and student) of Brunel University from the educational domain which are summarised in Section 5.4. The sourcing of such domains, and the context from which they were exhumed, are of particular importance. Both in directing the research and serving the research aim. Similarities are found in the way processes are performed in the domains under investigation, e.g. define product type in different systems within the same domain. Variation in the number of organisations and systems utilised provides the required level of external validity across multiple domains, enabling general conclusions to be drawn from both the SDR methodological framework and the subsequently discovered business process patterns.

3.7 Awareness of The Problem and Types of Solutions

This stage is aimed at setting up the necessary groundwork for this research by driving the initial requirements from the research objectives. The primary activity at this stage is the design of an initial version of the SDR methodological framework that feeds into the *Development* stage, which is described in Section 3.8.

3.7.1 SDR Cross-fertilisation of Disparate Disciplines

The issues identified in the literature review (Chapter 2) are investigated in the context of the overall discovery and reuse objectives. The lack of guidelines to modellers as to how business process patterns can be discovered must first be resolved as it forms the basis for attempting to resolve further issues. Evolving a methodology to support the finding of business process patterns represents an important area of work. Such a methodology guides the application process and acts as a reference document for situations where the methodology is applied. Therefore, in this stage the design of the initial version of the SDR methodological framework, for empirically deriving ontological patterns of business processes from organisational knowledge sources (i.e. documentation, legacy systems, domain experts, etc.), is essential.

In this research, the cross-fertilisation of disparate disciplines or research fields tackles the design of the SDR methodological framework of business process patterns. More specifically, three main domains (i.e., domain engineering, ontologies and patterns) are deemed relevant and helpful in addressing the research problem. Hence, as illustrated in Figure 3-5, the intersections amongst these research domains symbolises the context of the current research. The construct of the Semantic Discovery and Reuse methodological framework is based on the following foundations.

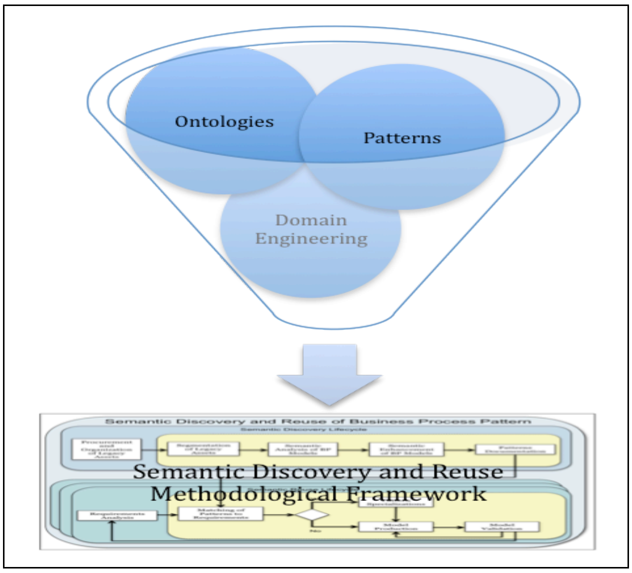

Ontologies
Patterns
Domain Engineering
Semantic Discovery and Reuse of Business Process Pattern
Semantic Discovery Lifecycle
Procurement and Organisation of Legacy Assets
Segmentation of Legacy Assets
Semantic Analysis of BP Models
Semantic Enhancement of BP Models
Patterns Documentation
Semantic Discovery and Reuse Methodological Framework
Requirements Analysis
Matching of Patterns to Requirements
Model Specialisations
Model Production
Model Validation
No

Figure 3-5: Relevant Domains to Develop the SDR Framework

First, Domain Engineering (DE) is an engineering discipline concerned with building reusable assets, such as specification sets, patterns, and components, in specific domains (Arango, 1988). A domain in this context can be defined as an area of knowledge that uses common concepts for describing phenomena, requirements, problems, capabilities, and solutions. The purpose of domain engineering is to identify, model, construct, catalogue, and disseminate artefacts that represent the commonalities and differences within a domain. Nowadays, domain engineering methods receive special attention from the information systems and software engineering communities. The reasons for the increased level of interest include: the need to manage increasing requirements for variability of information and software systems (intrinsic complexity relating to variability in customer requirements), the need to minimise accidental complexity when modelling the variability of a domain and the need to obtain, formalise, and share expertise in different, evolving domains.

Domain engineering deals with two main layers: the domain layer, which deals with the representation of domain elements, and the application layer, which deals with software applications and information systems artefacts (Foreman, 1996). In other words, programs, applications, or systems are included in the application layer, whereas their common and variable characteristics, as can be described, for example, by patterns, ontology, or emerging standards, are generalised and presented in the domain layer.

Domain Engineering is the process of defining the scope (i.e., domain definition), analysing the domain (i.e., domain analysis), specifying the structure (i.e., domain architecture development) and building the components (e.g., requirements, designs and documentations) for a class of subsystems that will support reuse (Nwosu and Seacode, 1999).

Domain engineering as a discipline has practical significance as it can provide methods and techniques that may help reduce time-to-market, product cost, and projects risks on one hand, and help improve product quality and performance on a consistent basis on the other hand. Thus, the main reason of bringing domain engineering into this research is that information used in developing systems in a domain is identified, captured and organised with the purpose of making it reusable when creating or improving other systems. Also, the use of domain engineering has four basic benefits (Prieto-Diaz, 1990), as follows:

- Identification of reusable entities.
- Abstraction of entities
- Generalisation of solution.
- Classification and cataloguing for future reuse.

Therefore, the SDR methodology is based on a dual lifecycle model as proposed by the domain engineering literature (Foreman, 1996). This model defines two interrelated lifecycles: (1) a lifecycle aimed at generating business process patterns called Semantic Discovery Lifecycle (SDL), and (2) a lifecycle aimed at producing business process models called Semantic Reuse Lifecycle (SRL). Figure 3-6 illustrates the SDR methodological framework.

Second, the phases of the first lifecycle (Semantic Discovery Lifecycle) have been classified according to the Content Sophistication (CS) methodology (Daga et al., 2005). CS is an ontology-based approach that focuses on the extraction of business content from existing systems and improving such content along several dimensions. According to Daga et al. (2005a) "CS provides a model of the business that more closely adheres to the semantics and relationships of objects existing in the real world". This approach is based on the REV-ENG methodology, described in Partridge (1996), which provides a way to re-engineer existing systems into objects. CS largely consists of four work disciplines based upon the type of work involved. These disciplines are summarised below:

- **Preparation -** The aim of this discipline is to establish the necessary groundwork for applying the CS approach;
- **Segmentation -** The aim of this discipline is to divide the concerned system into a hierarchy of work unit(s) that can facilitate its analysis and harmonisation;
- **Sophistication -** The aim of this discipline is to interpret and sophisticate the work units identified through the segmentation process; and
- **Harmonisation -** The aim of this discipline is to harmonise the sophisticated work units into a single coherent whole.

The disciplines and the associated activities in CS are organised in a way to reflect the underlying work that each of these activities carry out. CS was followed as it allows the organisation to understand and document knowledge in terms of its business semantics providing scope for future refinements and reuse. Therefore, the Semantic Discovery Lifecycle is based on the four disciplines of the Content Sophistication methodology. This lifecycle defines five phases, four of them based on CS as follows: (1) a phase aimed at acquiring legacy assets and organising them in a repository called *Procurement and Organisation of Legacy Assets (POLA)*, (2) a phase aimed at chunking the assets into workable segments called *Segmentation of Legacy Assets (SLA)*, (3) a phase aimed at interpreting elements of the process diagrams from an object paradigm perspective called *Semantic Analysis of BP Models (SA)* and (4) a phase aimed at generalising models to patterns called *Semantic Enhancement of BP Models* (SE). Figure 3-6 illustrates the SDR methodological framework.

Third, the last phase of SDL uses patterns. Patterns have become a widely accepted architectural technique in software engineering. A pattern generally includes a generic definition of the problem, a model solution and the known consequences of applying the pattern. Patterns can produce many advantages: (1) Reducing both time and cost of generating business process models and their subsequent transformation into software designs of enterprise applications. (2) Improving modelling by replacing an ad hoc approach with a successful one. (3) Promote reuse of business processes. (4) Reuse has the longer-term benefit of encouraging and reinforcing consistency and standardisation. Also, a pattern generally includes a generic definition of the problem, a model solution and the known consequences of applying the pattern. Thus, the fifth phase of the SDL, called *Pattern Documentation*, provides a way of documenting the discoverable patterns. Figure 3-6 illustrates the SDR methodological framework.

3.7.2 First Version of SDR

The first version of the Semantic Discovery and Reuse framework has been developed and driven from the literature as explained earlier in the domains and context section. Thus, this section now explains the first version of SDR.

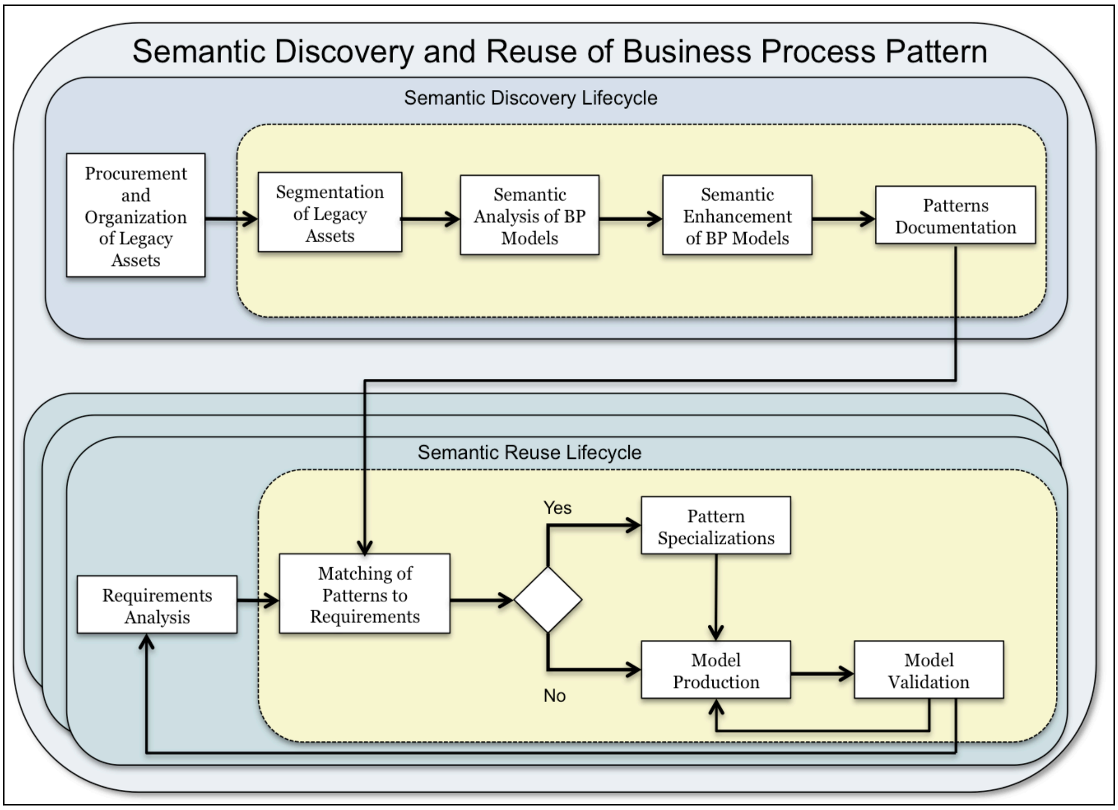

Semantic Discovery and Reuse of Business Process Pattern
Semantic Discovery Lifecycle
Procurement and Organization of Legacy Assets
Segmentation of Legacy Assets
Semantic Analysis of BP Models
Semantic Enhancement of BP Models
Patterns Documentation
Semantic Reuse Lifecycle
Requirements Analysis
Matching of Patterns to Requirements
Yes
No
Pattern Specializations
Model Production
Model Validation

Figure 3-6: First Version of SDR Methodological Framework

The first lifecycle SDL initiates with the procurement and organisation of legacy sources and finishes with the production of business process patterns, which then become part of the pattern repository. The repository feeds into the Semantic Reuse Lifecycle. The phases of the SDL as illustrated in Figure 3-6 are as follows.

<u>Phase 1: Procurement and Organisation of Legacy Assets (POLA)</u>

SDL is a process of discovery; therefore it is necessary to derive the business process patterns from legacy assets that demonstrate the existence of certain types of models as well as their generalised recurrence across multiple organisations. SDL, in this sense, is similar to the way scientific theories are discovered from data. Only model types which have been previously and demonstrably adopted by organisations and/or workflow systems can be modelled and become part of the patterns repository. Therefore, acquiring legacy assets and organising them in a repository is an essential initial step.

Phase 2: Segmentation of Legacy Assets (SLA)

Before any type of semantic analysis of the legacy assets can take place, the assets need to be 'chunked' into workable fragments. For example, all documentation and models related to financial transactions of retail bank accounts can be collected together and fed into the next phase.

The aim of segmentation is to divide legacy sources into a hierarchy of segments that facilitates the analysis of different legacy sources in order to bring them together; this ensures that irrelevant legacy assets (assets that do no provide enough information about specific data, for example, within the financial domain, the policy administration segment does not provide enough information on the type of behaviour carried out) are not included in the analysis. Therefore, segmentation provides manageable and semantically coherent work units, which are fed into the next phase.

Phase 3: Semantic Analysis of BP Models (SA)

This phase along with the following represents the core of SDL. In *SA* business process models are extracted from the legacy asset segments. These models are typical process flow diagrams such as BPMN diagrams. The elements of the process diagrams are then semantically interpreted in order to derive more precise ontological models of the processes themselves. The following steps are attached to this phase (see Figure 3-7):

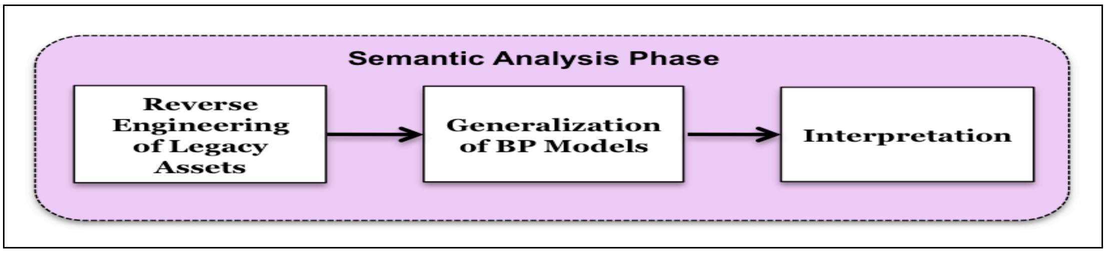

Semantic Analysis Phase
Reverse Engineering of Legacy Assets
Generalization of BP Models
Interpretation

Figure 3-7: Steps of the Semantic Analysis Phase

Step 1: Reverse Engineering of Legacy Assets

In this step a reverse engineering technique analyses the organisational legacy assets segments, identifies business processes and creates the representational models of the abstracted processes. These models are typical process flow diagrams such as UML activity diagrams or BPMN diagrams. The use of reverse engineering is to focus on more abstracted representations of the assets since, during this step, questions are asked concerning: What business processes take place? What types of activities take place? and What conditions are required for defining business processes? It is during this step that processes are identified, understood and visualised to support the legacy assets segments. The elements that are used in defining the organisational processes are adopted from Ericksson & Penker (2000) as shown in Table 3-4.

Elements	Description
Process	An abstraction showing a number of activities that form an input and create an output that are of a value or deliver a service to the customer. The activity specifies the behaviour carried out in an organisation. The input and output of the processes are resources in the business. A process has a goal and is affected by events that initiate and terminate the process.
Events	A change in the process state that notifies something has happened in the business. It is generated by one process and received by one or more other processes. Event can be external or internal. There is two events affect the process state. First, Preconditions: event that initiates the process to take place. Second, Postconditions: event that terminates the process
Resources	Concepts or things used in the business, such as physical things (e.g., a machine), abstract things (e.g., an agreement), people, or information resources (e.g., information about other resources such as data about an employee stored in a information system). The input and output of the processes are resources in the business. Input is the resources before a process is carried out, and Output is the resources resulting from a process
Goals	The desired state to be reached by a process. Goals are attached to the entire business and to processes within the business

Table 3-4: Ericksson & Penker (2000) organisational process element definitions

Those elements justify the previous questions of what business processes take place? What type of activities takes place? and What conditions are required for defining business processes? Those elements provide a clear definition for the organisational processes, which help to construct an effective model that reflects organisational elements using business process modelling notations.

Step 2: Generalisation of BP Models

In this step generalisation provides an abstraction principle for defining a process model as a refinement of another process model. It sees a relationship between a general and specific business processes as the specific process contains all the activities of the general process and more. The essence of generalisation is to focus the efforts on providing a clear understanding of when one process is a general case of another.

Step 3: Interpretation

Interpretation is the first distinct phase of Content Sophistication (Daga et al., 2005a). It identifies the business objects that the process commits to existing (i.e., using ontic commitments). *Interpretation* explicitly makes the business processes as much as possible close to the real world objects (its existence), which ensures the grounding of the patterns to real world behaviour. For this phase a modelling paradigm with a sound ontological foundation is required. This paradigm is the object paradigm (Partridge, 1996) (see Section 2.6). For this step the elements of the process models are semantically interpreted in order to derive more precise ontological models of the processes themselves.

Phase 4: Semantic Enhancement of Ontological BP Models (SE)

This phase takes the ontological models created in *SA* and aims at generalising them to existing patterns or to newly developed patterns.

Phase 5: Pattern Documentation (PD)

The pattern(s) derived from a cycle of SDL are finally documented and catalogued in the patterns repository. One of the most important uses of *PD* is to help people organise and share innovative patterns. This research adopts Ericksson and Penker's (2000) method of documenting patterns according to the template described in Table 3-5.

Template Term	Description
Name	Every business pattern has a short, distinct Name, which is a metaphor for the pattern. The name should be one that can be easily associated with the structure of the pattern.
Intent	The Intent section describes and summarizes the general purpose of the business pattern, answering the questions: What does the Business Pattern do? What problems does it solve?
Motivation	The Motivation section exemplifies the use of the pattern by describing the recurring problem in a concrete problem situation (a context) and shows how the pattern can be used to solve that problem.
Applicability	The Applicability section defines in which problem situations the pattern can be applied, and which problem it solves.
Structure	The Structure section contains a generic visual representation of the pattern in any modelling technique.
Participants	The Participants section defines and describes the model elements that participate in the pattern, as visualised in the Structure section.
Consequences	The Consequences section describes how the pattern supports the earlier defined goal, and eliminates the perceived problems within the problem situation.

Table 3-5: Pattern Documentation Template (from Ericksson and Penker, 2000)

The second lifecycle, *Semantic Reuse Lifecycle* (SRL), is aimed at producing business process models with the support of the patterns discovered during the SDL. The phases of the lifecycle, as illustrated in Figure 3-6, are purely indicative. An organisation can adopt its preferred business modelling process, but such a process should then be tailored in order to include essential reuse activities such as matching the business requirements specifications with existing business process patterns and adapting such patterns (e.g., through specialisation) to the specific requirement. The SRL is dependent on the SDL only in terms of the patterns that are produced by the SDL. The two lifecycles are, for all other purposes, autonomous and can be performed by different organisations. In this case the organisation performing the SDL would be specialised in the management and supply of process patterns, while its clients would consume the discovered patterns. The typical phases of the SRL as illustrated in Figure 3-6 are as follows:

Requirements Analysis:

A given business problem is studied producing a set of business requirements specifications.

Matching of Patterns to Requirements:

Given the requirements produced in the previous phase, the requirements specifications are matched against existing business process patterns in order to identify patterns that can help to model and provide proven solutions to the requirements.

Pattern Specialisation:

The patterns selected as possible template solutions to the specified requirements are then adapted to meet specific aspects of the problem space represented by the given requirements.

Model Production:

Models are produced as a solution to the business requirements.

Model Validation:

The models are validated (tested) against the business requirements until the solution provided is considered to be sufficiently adequate. At this stage it may be necessary to revisit the initial requirements if any omissions or amendments are identified. In this case the cycle is repeated.

To fully grasp the nature of each SDR phase, it is necessary to seek the fundamental meaning behind each of them, because for a phase to be valid it has to be reasonable to assume that the phase exists in isolation. It may be that new phases become emergent from aggregation or amalgamation of other phases.

3.8 Development Stage

This stage is aimed at the iterative *Design*, *Deployment* and *Evaluation* of SDR. These iterations deliver improvements in the artefact(s) because the individual iterations evolve the SDR methodology incrementally. Each increment can be considered as producing an improved version of the methodology (see Figure 3-8 for the structure of research iterations). Successive designs of SDR are based on the evaluation conducted at the end of the previous iteration.

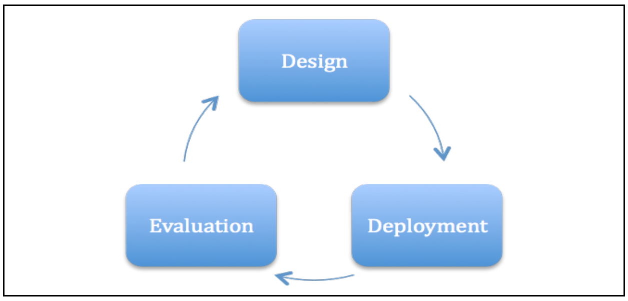


Figure 3-8: Structure of the Iterations in the Development Stage

The iterations are divided into *three* major phases: *Design, Deployment and Evaluation*. Iterations begin with a design phase where a suggestion for the research within the iteration is explicitly stated, for the first iteration this draws requirements from the stage of *Awareness of the Problem and Types of Solutions*. The *Deployment* phase aims to deploy the methodology in real life environments and produce improved versions of SDR. Evaluation criteria are also selected at the evaluation phase. The iteration then cycles repeatedly through iterations of *Design, Deployment and Evaluation*. The iteration stops when either the process is interrupted as inaccuracies in the artefacts are identified, or the finding of a good solution. *Evaluation* of the artefact generates better knowledge that forms feedback to the next iteration. The rest of this section provides an overview of each of three major phases in the *Development* stage.

The first iteration derives its initial requirements from the *Awareness of the Problem and Types of Solutions* to address the objectives stated in Section 3.6.2. Each of the following iterations then derives its requirements from feedback of the previous iteration. Table 3-6 represents the planned research iterations.

3.9 Final Evaluation Stage

The *Final Evaluation* is aimed at assessing SDR's effectiveness in discovering quality patterns that offer a solution for an organisational problem and its contribution toward improving the work of business process modellers and the quality of the artefacts they produce. Thus, the *Final Evaluation* stage consists of a final reflection in which results are critically analysed by utilising an academic domain scenario for designing a new university course explained in Section 6.11, and reflecting on how the use of one of the discovered business process patterns helped to provide a proven and clear model for designing a new process for introducing a new university course.

Iter.	Purpose	Source used	Why
Iteration 1	It aims to uncover the form and structure of business process patterns by generating a methodology specific to transform organisation knowledge assets to BP Patterns, a requirement implied by the research objectives and gaps in current discovery and reuse of BP patterns This phase is aimed at defining the SDR methodology in terms of its process, artefacts, and researcher steps. The initial design is used from the preparation stage as mentioned above	These three software systems are Retail Bank, Insurance and Mortgages respectively of financial industry domain. Beside each of these solutions has been developed over time as solutions to address the market needs. These are good systems to work with because they were developed to be generic, therefore they should provide a good service for the extraction of patterns.	First, the effectiveness of building SDR is analysed. Second, the effectiveness of using the knowledge base from the financial services is assessed. Third, the effectiveness of discovering the BP Patterns is evaluated, specifically adherence to DR principles.
Iteration 2	The Second iteration aims to discover business process patterns and refine existing ones by using improved SDR methodological framework from Iteration One. A requirement implied by the research objectives and gaps found in the Iteration One in current discovery of BP patterns	In this iteration, Brunel University handbooks inventories are utilised, as a form of organisational knowledge source. Two handbooks form the basis of the current work. These two handbooks are for staff and students of the Department of Information Systems and Computing (DISC) at Brunel University. Each of these sources has been developed over time as guidance to address the educational institution needs.	• Introducing the ontological definition of business process to provide more accurate representation of the process models. • Altering the process models to reflect new presentation for the process models using the ontological definition of BPs. • Defining generalisation rules to enable better discovery for processes commonality. • Applying the method across new legacy functionality in order to demonstrate generality of process.
Iteration 3	The motivation of doing Iteration Three is to evolve the Third/Final version of SDR methodological framework by applying the Semantic Discovery Lifecycle to different domain that is the Financial Services domains. Also, to check the ability of SDL to produce Patterns of Business Processes a cross multiple domains	Lifecycle across multiple domains (Educational and Financial) with multiple systems (Students, Staff, Retail Banking, Insurance and Mortgages).	• A hierarchy of the discovered patterns. The primary motivation behind this rational is to formally describe the different BPO elements that the discovered patterns generalised or extracted from so that unwanted ambiguities related to the application and use of the pattern can be avoided. • Continue discovering business process patterns from empirical data (e.g., legacy systems). This iteration is going back to discover BP patterns from the financial domains mentioned in Iteration One. • Continually test the existing BP patterns from the educational domains against legacy models and data of the financial domains.

Table 3-6: Research Plan of Iterations

3.10 Summary

The aim when conducting research is to make a contribution to academia as well as to practice. This implies that the research conducted should add to the existing base of theory in order to make a contribution and assist in solving current or expected practical problems (Hevner et al., 2004).

The research methodology adopted in this study is design research (March & Smith, 1995; Hevner et al., 2004). This methodology consists of the construction and evaluation of the artefacts that resolve a significant and recognised problem. In this research the use of Design Research is directly associated with the expected result, which is to create a methodology aimed at the Semantic Discovery and Reuse of Business Process Patterns. The result of the study has to be evaluated to ensure both the validity and the reliability of the methodology, and to determine to which extent the outcome can be used as a basis for further research.

In addition, the Design Research method is described and sets of frameworks are chosen to direct the research. The nature of Design Research dictates that this initial framework will be refined – together with the constructs, models, methods and instantiations that represent an effective solution to the clearly identifiable problem. The problem is made explicit – namely the discovery of business process patterns in organisations (in order to support their reuse). Issues identified in the literature review are used to produce a solution that evolves over three iterations in the *Development* stage. With the SDR methodology in place the research moves into the first practical iteration for identifying and discovering business process patterns.

CHAPTER 4 - ITERATION ONE: SDL IN THE FINANCIAL DOMAIN

4.1 Introduction

The core contribution in this iteration is to evolve the SDR methodological framework. SDR supports the semantic discovery and reuse of business process patterns from the empirical analysis of organisational knowledge sources, such as the legacy assets of the financial services domain. Also, SDR guides the application process and acts as a reference document for situations where the methodology is applied. The focus in iteration one is on evolving the first version of the Semantic Discovery Lifecycle (SDL) of SDR through the empirical analysis of organisational processes.

This chapter is organised as follows. Section 4.2 represents the purpose of performing iteration one. Section 4.3 explains the domain – financial services – under analysis in this iteration and its source data. Section 4.4 lays down the necessary groundwork for the practical application of the Semantic Discovery Lifecycle in the financial domain. Section 4.5 presents the research outputs for the iteration. Section 4.6 evaluates the research output artefacts. Section 4.7 discusses the implications of the outputs. Section 4.8 describes feedback to influence the design of the next iteration. Finally, the chapter is summarised in Section 4.9.

4.2 Purpose of Iteration One

This iteration aims to focus on facets 1 and 2 of the research objectives. It aims to uncover the form and structure of business process patterns by utilising the methodological framework specific to transform organisational knowledge assets to business process patterns, a requirement implied by the research objectives and gaps in the discovery of business process patterns. In order to undertake the task in this research, the application of the first version of the SDR methodology for the discovery of business process patterns is derived from the knowledge assets of the three financial services systems. These three software systems manage retail banking, insurance and mortgages respectively. Each of these solutions has been developed over time as solutions to address the market needs. In addition, data from these systems is employed to test the first version of the SDR methodological framework. Both SDR and the discovered patterns are simultaneously evaluated. The learning and the outcome of this iteration feed into the following development iterations.

4.3 Iteration One Data Sources

Iteration one utilises data sources derived from the financial services domain. The data sources from an organisation can represent any body of knowledge (i.e., system application data, documentation, models, expert knowledge, observations, etc.) and provides confirmation of the existence of certain behaviour and types of behaviour in an organisation. The range of financial system solutions offered by the legacy systems mentioned above covers the areas of banking and mortgage, insurance, payments, life and pensions, and investment management. O'Callaghan (1999, p. 4) defined a legacy system as "a large system delivering significant business value today from a substantial pre-investment in hardware and software that may be many years old. Characteristically, it will have a long maintenance tail. It is, therefore, by definition a successful system and is likely to be one that is, in its own terms, well engineered. It is a business-critical system with an architecture that makes it insufficiently flexible to meet the challenges of anticipated future change requirements". Legacy systems evolve over a number of years, and carry a rich amount of business knowledge in the form of business processes, rules and business content, which may only be documented in the enterprises' legacy sources. Therefore, documentation of three financial services legacy systems form the basis of the current iteration. These three software systems refer to retail banking, insurance and mortgages, respectively, of the financial industry domain. Moreover each of these solutions has been developed over time as solutions to address the market needs. These are good systems to work with because they were developed to be generic, therefore they should provide a good source for the extraction of patterns. The rest of this section provides an overview of each of these systems.

4.3.1 Retail Banking

The retail banking system consists of a core group of a related banking products offered under a single umbrella. It consists of a suite of functional modules. Two aspects of the system are notable. Firstly, the modular design of the system means that an organisation can choose from a variety of financial products and do not have to take the entire system. This allows the organisation to tailor the system according to the needs of the client. Secondly, the broad nature of customisation means that the system can offer a set of products and services according to its needs, together with the ability to react promptly to changes in the market. In addition being a fully modular system, retail banking can be readily tailored to areas of wealth management, private banking, institutional investment management, global bank branch, retail, and wholesale banking target markets, as well as providing support for risk management, portfolio management and statutory reporting.

4.3.2 Insurance

The insurance system delivers a package of related processes and products for any insurance company. It consists of a suite of functional modules that are common to most insurance companies, but leaves areas that are company specific for the organisation to complete. This facilitates a financial organisation to build its own specific products and processing rules. Two aspects of the system are noteworthy. Firstly, the functionality design of the system means that an organisation can choose from a variety of insurance processes and do not have to take the entire system. This allows the organisation to tailor the system according to the needs of the client. Secondly, a separate rules area where processing rules are defined. The customer can add and change rules and processes without changing the screens with which they are associated, together with the ability to react promptly to changes in the market. Each section provides a high-level understanding of the modular design of the insurance system.

4.3.3 Mortgages

The mortgages system defines and sets up the various accounting controls within any company especially for mortgages and enables to interpret the control information provided by mortgages and carry out diagnostic and remedial maintenance. This facilitates the organisation to build and maintain their own systems that match their own requirements. There are a few aspects of the system worthy to be mentioned. Firstly, the flexibility in the design of the system means that an organisation can choose from a variety of systems available and do not have to take the entire system. This allows the corporation to tailor the system according to the needs of the client. Secondly, there is a monitoring facility to check the financial integrity of the system describing the diagnostic and corrective measures that are available in the event of bookkeeping problems. Finally, the separate rules area where processing rules are defined. So the organisation can add and change rules and processes without changing the processes with which they are associated, together with the ability to react promptly to changes in the market.

4.4 Practical Application of SDR

The focus now turns to the *Development* stage to demonstrate the actual application of the first version of the Semantic Discovery Lifecycle (SDL, i.e., first lifecycle of SDR). See Figure 4-1 for a representation of SDL. SDL has been applied for the financial services domain as follows:

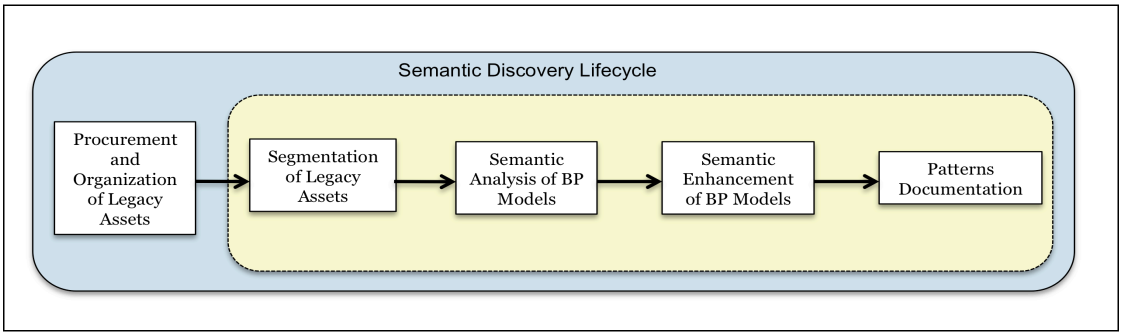

Semantic Discovery Lifecycle
Procurement and Organization of Legacy Assets
Segmentation of Legacy Assets
Semantic Analysis of BP Models
Semantic Enhancement of BP Models
Patterns Documentation

Figure 4-1: Version 1 of SDL (Section 3.7.2, Figure 3-6)

<u>Phase 1: Procurement and Organisation of Legacy Assets (POLA)</u>

Data from the financial services domain forms the entry for the first phase of *Procurement and Organisation of Legacy Assets* (POLA) of SDL. The financial legacy systems are retail banking, insurance and mortgages respectively. These systems were produced by a large provider of business solutions and adopted by various important financial institutions worldwide. The legacy data used for SDL are: (1) design and user manuals with system screen shots of the financial legacy systems and (2) knowledge acquired from literature on the financial domain as well as the legacy systems themselves.

Phase 2: Segmentation of Legacy Assets (SLA)

Given that each of the legacy systems' manuals incorporates over 9 sections (comprising tens of functions), *Segmentation of Legacy Assets* provides a way to approach and manage those systems, by dividing the financial services assets into a hierarchy of segments that facilitates the analysis of the three sub domains, and ensures that irrelevant assets are not included in the analysis. Figure 4-2 illustrates the financial legacy assets segmentation thinking demonstrated in this research.

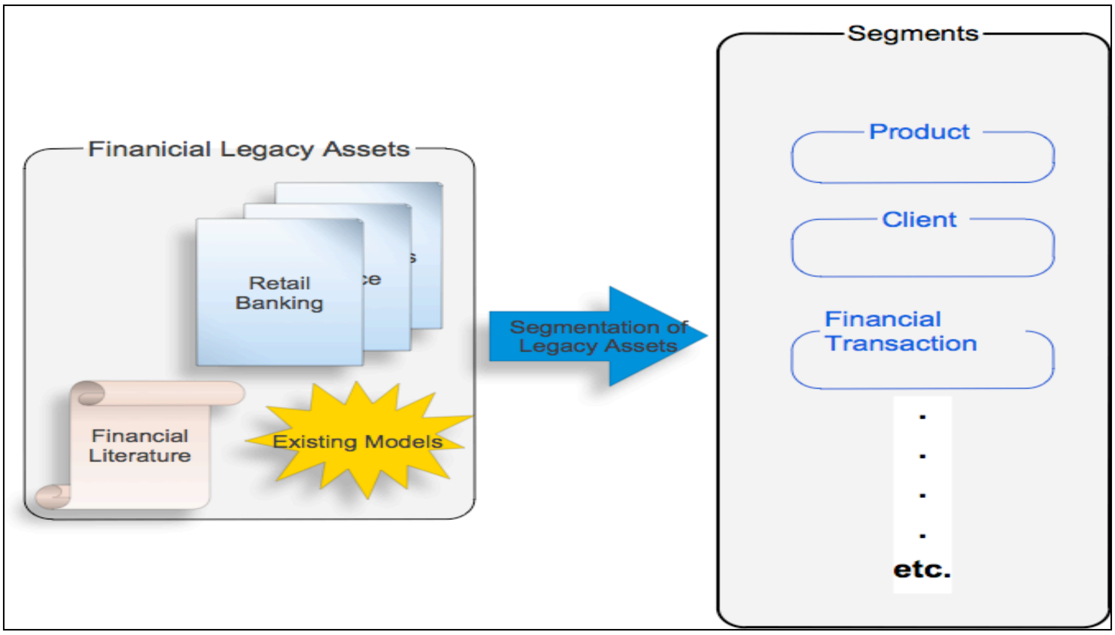


Figure 4-2: SLA Phase applied to Financial Services

This phase involves reformatting the assets in terms of data types that retail banking, insurance and mortgages consist of, such as product, client, financial transaction, etc. Table 4-1 It provides the list of segments related to retail banking, insurance and mortgages that are extracted from the financial legacy systems' manuals. Some segments are of a similar type among the sub domains, such as client, product, etc. Thus, the use of segmentation provides manageable and semantically coherent work units.

Financial Domain Systems	Segments	Processes
Retail banking	Client	Create Client Account
		Define Client Information
		Open Client Account
	Product	Define Product Template
	Payment	Define Automatic Payment Methods
	Account	Identify workflow Client Account
		Open Retail Accounts
	Transaction	Issue Cards
		Define Cheque Book
Insurance	Client	Define Client Details
	Product	Create Product Type
		Define Product
	New Business	Set up New Business
	Policy Administration	Process Policy
		Set up Policy Type
	Financial Transaction	Process Financial Transaction
	Marketing	
	Account	Open Account
		Manage Account Details
Mortgages	Transaction	Create Cash System Branch Transaction
		Define Mortgage Transaction Processing
	Product	Create Product Type
		Set up Mortgage Product
	Nominal	Set up Nominal Account
		Set up Nominal Transaction
	Account	Set up Bank Account
		Set up Account Control System
		Create Account Types
	Policy	
	Investment	Define Investment Account Process

Table 4-1: Segmentation of Legacy Assets

In addition, the segment can be divided into any number of processes, deemed necessary when starting *Semantic Analysis*. Each segment is further decomposed into processes that the segment corresponds to. Table 4-1 illustrates the list of processes related to each segment in the sub domains. Each of these processes are identified and allocated to the concerned segment. An important point that should be clarified here is that even if some process(es) do not clearly show enough information to reflect on its segment, it should not be taken that the segmentation process is ineffective as in *Semantic Analysis* the *Reverse Engineering* step will help in electing and constructing the definition of business process elements.

Phase 3: Semantic Analysis of BP Models

Moving to the *Semantic Analysis of BP Models* (*SA*), the core phase of SDL, one example will be explained in detail. This example is related to the 'Client' segment and specifically the 'Define' process. More processes followed the same approach of this example and can be found in Appendix A. The 'Define' process of the 'Client' segment derives from all parts (chapters and/or sections) of the manuals providing information related to the generation or amendment of client types. This phase is composed of three steps as presented in Figure 4-3. The next phase will be *Semantic Enhancement*.

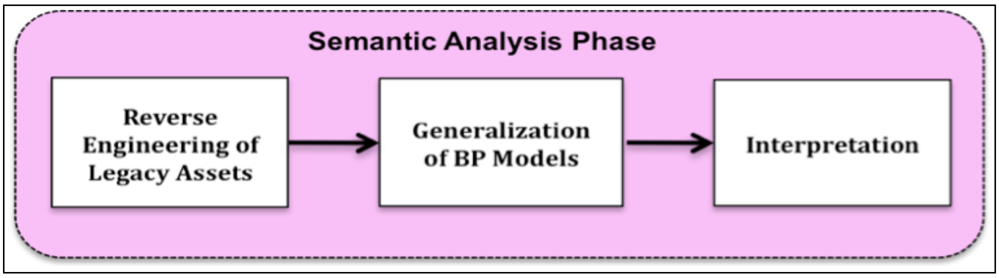

Semantic Analysis Phase
Reverse Engineering of Legacy Assets
Generalization of BP Models
Interpretation

Figure 4-3: Steps of The Semantic Analysis of BP Models Phase

Step 1: Reverse Engineering of Legacy Assets

This step is used to identify processes and create business process models. These models are typical process flow diagrams such as Business Process Modelling Notation (BPMN) diagrams. The use of reverse engineering is to focus on more abstracted representations of the financial service systems. During this step questions are asked concerning process definitions:

- What business processes take place?
- What types of activities take place?
- What conditions are required for the business processes?

It is during this step that the processes are identified, understood and visualised. The elements that are used in defining the organisational processes are adapted from Ericksson & Penker (2000) as stated earlier in Table 3-4. For the 'Client' segment, ‘Define Client Information’ is a fundamental business process. Table 4-2 demonstrates the ‘Define Client Information’ business process for retail banking.

Define Client Information	
Terms	**BP Elements**
Client Information Created or Maintained	Event
Set up Client Type	Activity
Add Client Details Form	Activity
Complete Client Employment Details Form	Activity
Complete Client Financial Details	Activity
Set up Maintainable Client Details	Activity
Attach Product Template to Client	Activity
Produce Client Information Reports	Activity
New client	Precondition
Client Information to be used with other systems	Postcondition
Client	Resources
Retail Banking Assistant	Resources
Client Details Form	Resources
Client Maintenance Form	Resources
Client Employment Form	Resources
Client Financial Form	Resources
Client Information Report	Resources

Table 4-2: Define Client Information for Retail Banking

Answering the above questions helped to produce the ‘Define Client Information’ business process table. This table summarises the activities used in generating the process. The number of activities reflects the number of steps needed to define this process. In Table 4-2 to create the ‘Define Client Information’ process required the representation of various activities starting from the first activity of ‘Set up Client Types’ to the final activity of ‘Produce Client Information Reports’. Also, additional information is defined for this specific process such as resources used and produced (input and output), goal, precondition and postcondition. The resources represent the type of information required as an entry for the process (e.g., different types of forms that need to be used such as client name and address details form, client financial details form, and client employment details forms), while the output represents the type of information that can be obtained from processing ‘Define Client Information’.

The process ends with the ‘production of Client Reports’. The precondition is the trigger for this process and it must hold true for the process to start. In this case the precondition is represented by the existence of a new client. The postcondition holds true upon completion of the process. Further details of the other business process models of the three legacy workflow systems can be found in Appendix A.

After *Reverse Engineering* is completed, the conceptualisation of the ‘Define Client Information’ process can take place by creating process models using BPMN modelling elements. Figure 4-4 illustrates a BPMN model of Table 4-2.

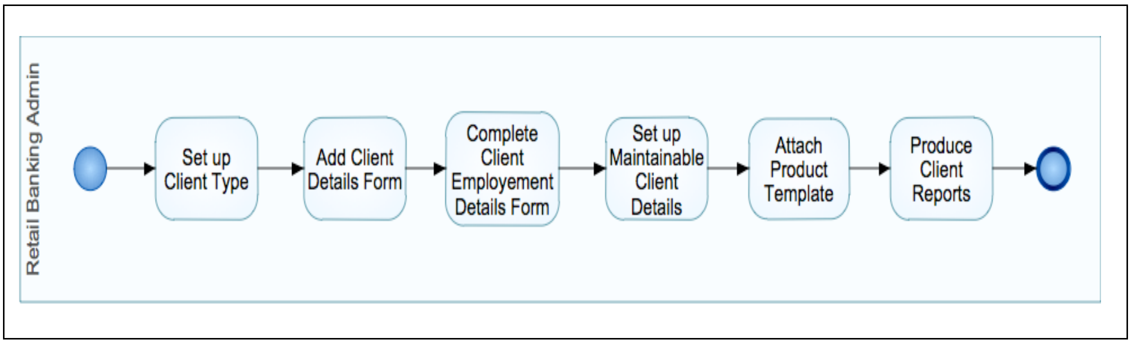


Figure 4-4: BPMN Model of 'Define Client Information'

Step 2: Generalisation of BP Models

This step is carried out in an informal way in iteration one by bringing together similar types of business processes from different sub domains. Similar process type means that two or more processes are doing similar types of activities or that they can use similar input types and create similar output types. Similar types of business processes are brought together in one table. In Table 4-3 three process models for ‘Define Client Information’ are represented. The three models represent the workflow behaviour that the legacy applications automate within the business processes of the respective financial organisations that have adopted this technology. Using this table helped to bring together similar types of processes and their associated activities. Therefore this step helped to visualise and combine similar business process types of different domains.

Retail Banking	Insurance	Mortgages
Define client information	Define client Details	Set up individual details
Set up client type	Set up client name details	Record basic customer details
Add Client Details	Set up client address details	Record correspondence details
Complete Client Employment Details	Set up client financial details	Record further customer details
Complete Client Financial Details	Set up client occupations	Record customer financial details
Set up Maintainable Details	Define maintainable details	Set any joint applicant
Attach Product Template to Client	Define client suitable product	Set up amendable details
Produce Client Information Reports	Set client type as agent	Produce Reports
	Define client correspondence details	
	Produce Client Report	

Table 4-3: Similar Business Processes for 'Define Client Information' in Retail Banking, Insurance and Mortgages

Step 3: Interpretation

Once the processes are extracted and modelled, the next step is interpretation. This step consists of ontologically analysing each element of the process models (e.g., events, activities, etc.). This analysis enables the analyst to ‘unbundle’ or ‘interpret’ the semantic content of the process elements. This step is essentially a transformation from a traditional process paradigm to the object paradigm. The reason for undertaking such a transformation lies in the greater expressivity of the object paradigm. Expressing a model in a semantically richer and more precise paradigm facilitates the next phase of *Semantic Enhancement*. However, this step was not performed in iteration one because there was not enough knowledge to map business process elements when using Ericksson and Penker’s (2000) process definition Moreover Ericksson and Penker’s (2000) business process elements provide limited information in relation to the meaning of such elements. For example, such a definition of a business process does not distinguish between the triggering and dissolution events. Similarly for concepts like inputs, outputs and roles. Thus, the need for a better definition of business process elements is required and this will be accomplished by proposing an ontological definition of business process in iteration two.

Phase 4: Semantic Enhancement of Ontological BP Models (SE)

This phase takes the ontological models created in *Semantic Analysis* and aims at generalising them to existing patterns or to newly developed patterns, but as these ontological models of 'Define Client Information' are not defined in this iteration, this phase could not be addressed. Also, the way in which these ontological models are going to be generalised has not been defined. Therefore, the principles underlying generalisation of business process models will be also defined in iteration two.

Phase 5: Pattern Documentation (PD)

The researcher could not derive ontological patterns of business processes in iteration one because of the need for an ontological definition of business processes. However with a few similar types of processes defined in *Semantic Analysis*, these processes would have the potential to be generalised to a pattern in the next iterations. In iteration two, this research adopted Ericksson and Penker's (2000) template to represent the document of the patterns.

4.5 Research Output Artefacts

This main research output for iteration one was the first version of the SDR Methodological Framework. SDR incorporates two lifecycles, for iteration one the first lifecycle, SDL, is defined with five phases and performed on three financial services legacy systems in order to develop the discovery process lifecycle. The following constitute the main output artefacts of the SDL phases during iteration one.

- **Legacy Segments**. These represent a hierarchy of work units that facilitates the analysis of the financial services legacy assets and deemed necessary when producing the business process models.
- **BP Models**. Business process models are constructed through the analysis documents of three financial services legacy systems using reverse engineering and modelling techniques to transform the legacy asset segments to business process models.

- **Generalised BP Models**. These models are constructed by using previously discovered business process models to bring similar types of processes into one generalised business model.

The utilised and produced artefacts provide a summary of the iteration. Iteration one evaluates the research following the approach detailed in part one of Chapter 3 from Hevner (2004) and March and Smith (1995). The choice of criteria will depend on the reasons for developing the artefacts in the first place, which is related to the original research objectives. The evaluation can lead to a conclusion about the design process as well as the design product and may suggest further modifications to either or both.

4.6 Evaluation of Iteration One Artefacts

In iteration one, the utilised and produced artefacts provide a summary of the iteration that is subsequently evaluated. The aim now is to generate a complete set of criteria that can help in evaluating this iteration's artefacts. Thus, this section evaluates the first version of SDL. SDL essentially provides a series of phases and steps to be performed in order to develop ontological patterns of business processes. In this way, the SDL also provides a mechanism for improving the organisational assets of the knowledge sources. Given that the SDL provides a novel way for organisational knowledge transformation, two aspects primarily make up the evaluation criteria here (March & Smith, 1995):

- **Methodological Study** - The first set of criteria solely focus on evaluating the methodological framework away from the application of the method. Factors relevant here are the aspects of which the SDL methodology is composed (i.e., the basis of the approach, its structure and the concepts that make up SDL).
- **Operational Study** - The second aspect looks at the operational aspect of the method, where the focus is on evaluating the ability of SDL to perform the intended task (i.e., the use of the method). Factors such as completeness, efficiency, validation and precision (March and Smith, 1995) are factors that are used to reflect on the model. See Table 4-4 representing iteration one artefacts and the evaluation criteria used with the operational study.

Artefacts	Criteria	Description
Semantic Discovery Lifecycle (SDL)	Completeness	Measures the percentage of applying SDL to the financial services scenarios.
	Efficiency	Measures performance of the lifecycle from Phase One through to interpretation to business process patterns documentation in the last phase.
	Validation and documentation	Measures SDL's way of documenting the process models to enable future reuse
	Precision	Precision is the degree to which an experiment can be repeated and achieve the same result.

Table 4-4: Iteration One Artefacts and Evaluation Criteria

The fact that the discovery lifecycle is partially run in iteration one has produced a weak evaluation of the artefacts. Despite this a few chosen evaluation criteria have been defined and selected on the basis that: (a) each criterion reflects an area of theoretical knowledge that can be used as a means to evaluate the iteration; (b) an inter-dependency exists between the various criteria, which means the value and importance of each criterion is derived in part from the manner in which others are applied (e.g., making SDL simpler has a direct effect on the discovered patterns making reusability more effective); and (c) criteria allow to frame the right sort of questions to ask, which then provide the supportive metrics as the basis for improvement.

4.6.1 Methodological Study

The first aspect of the evaluation criteria reflects entirely on SDL of the SDR methodology, away from the application of the method. The SDL is broadly based on the CS methodology and patterns, as explained earlier in Section 3.7.1, as the primary concern was to provide a responsive process that would be easily understood and adopted by organisations. In noting that, it should be clarified, that there are many advantages of using the CS methodology, such as catering for iterative and incremental development, and a simple and consistent structure. There is however a practical issue associated with the adoption of the CS methodology in that, the SDL is largely an emerging topic and the need to gain practical relevance means that in certain situations SDL has to tailor the CS methodology, as CS is mainly used in for static content of Legacy Information Systems (LIS) unlike SDL which is developed for organisational processes.

During iteration one the focus was on understanding the feasibility of the first version of the SDL lifecycle. As a result the SDR methodology essentially reflected a mere skeleton, primarily made up of the necessary phases and steps relevant from a simple applicability perspective for the Semantic Discovery Lifecycle. Iteration two provides the necessary reflection on the actual application of the whole discovery lifecycle. Care was taken when implementing any changes and each change was analysed on the basis that:

- There was enough knowledge to make the necessary change.
- It was strategically important to bring in the change.
- The impact of the changes on the rest of the methodology.

As a result, some phases and steps within the Semantic Discovery Lifecycle will need to be modified. Artefacts were also analysed as part of the evaluation process.

4.6.2 Operational Study

The other evaluation aspect looks at the application of the SDR methodological framework within the context of study to reflect on the ability of its lifecycle to perform the intended task. The novel nature of the SDR means the focus is on how the lifecycle (SDL) fares in terms of operational efficiency (i.e., relevance, effectiveness and validation). The relevance of the SDL within the ambit of the SDR methodological framework is substantial. The SDR methodology has been provided with a mechanism for harvesting and integrating organisational business knowledge from multiple sources. Moreover, by clearly identifying and stating the discovery lifecycle through which to deliver business process patterns, the SDL has clearly impacted upon the practical relevance of the methodology.

The effectiveness and ease of use of the SDL in many respects is directly proportional to broad issue of ‘scalability’. The key question here is how to ensure consistency and robustness within the SDL, if it were to be applied within a project environment with several people working on it. The practical need of a method in such a situation is significant to reflect on the fact that the models that emerge from the SDL are not illusionary, but are consistent with the desired outcome. The understanding and consistency levels may vary at this stage as the SDL is emerging and the intention is to improve things in iteration two. The SDL reflects a more complete process with clearly identified phases, steps, activities, artefacts and roles than it did in iteration one. The benefits are however directly proportional to the practical application of the SDL lifecycle (i.e., further improvements need training and continuous development) and thus involves a substantial amount of time and effort.

SDL Lifecycle Operational Evaluation:

Completeness of the lifecycle is evaluated by measuring the percentage coverage of SDL. This, however, is not proved since the lifecycle was only applied partially to the financial services scenarios. To improve completeness of the lifecycle, the scope of iteration two must include all the phases and the steps in the lifecycle. The SDL lifecycle will be evolved to incorporate additional steps and reduction in the number of phases.

Validation and documentation, one important consideration in SDL is related to activities for validation and documentation of the developed patterns, mainly, in the context of domains, where asset complexity and volume are enormous. Although documentation is very important for any methodology it is often neglected. Reusing patterns is affected by the way patterns are documented within SDL.

Efficiency is expressed both by the capability of SDL to model a variety of legacy sources and by the characteristics which can save time needed for modelling and improving the quality of modelling such as model reusability, reliability and time scale for model building. Therefore, SDL is evaluated by using three financial services legacy systems (retail banking, insurance and mortgages). Applying SDL to different legacy system documentation enabled the researcher to produce business process models from the semantic analysis phase. In addition, from those three services domain, the researcher was able to measure process performance from the *Procurements and Organisation of Legacy Assets* phase to transformation towards patterns in *Pattern Documentation*. Many process models and similar types of processes have been constructed and documented, but no business process patterns were constructed during the first iteration due to the limitation in the definition of the *Semantic Analysis* phase. So the average time taken to construct patterns cannot be accurately recorded since the design process was a mixture of *Design*, *Deploy* and *Evaluate*, with half completed phases as new requirements were discovered and needed to be defined. Time measurement is imprecise since exact timings of phases were not recorded - only man-days were noted in a diary. A timeframe for the research can be used to estimate performance of the process however. The iteration took 95 man-days and produced 32 business process models and 5 similar types of business processes discovered at the end of that period, averaging to ~3 man-days for the construction of a single model as presented in Table 4-5.

Activity	
Iteration One	95 days
Number of Business process models	32 Models
Number of similar type processes models	5 Models
Estimated man day per model	~ 3 per days

Table 4-5: Estimated man-days of Business Process Models in SDL

In iteration one, *precision* was not measured since the SDL lifecycle is highly manual. Starting from collecting the assets, section segmentation and the semantic analysis for constructing models by hand using excel sheets and BPMN was time consuming and labour intensive. Comparison of documentation reduced performance of the process. The method would clearly benefit from process automation or tools that aid capture of models. Measurement of SDL performance also includes the performance of the researcher in learning and applying Design Research techniques as a factor. ‘Understanding by doing’ which represents the learning acquired during the iteration is a necessary activity within Design Research. Performance can expect to increase, as the researcher becomes more adept and the process more stable. The feedback suggestion for the second iteration is that improved performance by integrating semi-automated techniques in parts of SDL phases in Section 5.5 to increase efficiency in business process patterns discovery.

4.7 Discussion

Understanding SDL is necessary in order to describe and clarify the structure of the underlying phases, steps and techniques used to semantically discover business process patterns and its consumable resources required for the process. Upon reflection, it can be stated that the understanding of the phases and their associated steps is gradual as it involves a lot of learning alongside the project management issue previously noticed. The reason for this issue is due to SDL consisting of 5 phases and 3 steps, and some of the techniques used within those phases require better definition and understanding (such as *Interpretation*) and understanding the need in defining a Business Process Ontology to focus on aspects such as 'objects' and 'real world semantics' involves a substantial amount of effort and time.

In addition, the SDL process is highly manual. Extracting processes using Ericksson and Penker's (2000) business process definition, conceptualising process models using BPMN and bringing similar typed processes together by hand were time consuming. The method would clearly benefit from process automation or tools that aid capture of models.

Apart from what has been said earlier, two issues remain as an area of further exploration: Firstly, the SDL is largely dependent on the *Semantic Analysis* phase. While it is the case that interpretation is the primary aspect of the current work; the fact that a substantial amount of intellectual work is done in *Semantic Analysis*, points to the situation where *Semantic Analysis* is packed with too much information. Going forward, it points to a case for either re-distribution of work elsewhere or the need to create more phases to better manage the *Semantic Analysis* phase. Secondly, it should also be stated that SDL has not been applied to the fullest. The *Semantic Enhancement* phase and all its related concepts have yet to be applied and tested within a practical context. As a result, it certainly means that the true extent of the ability of the SDL will only emerge, once the whole lifecycle is applied in a practical context as achieved in iteration two.

Finally, the reliance on financial services documents as the sole source of data clearly limits the opportunity to go beyond the printed word and to probe motivations and reasons for courses of action more closely. These financial documentations typically record decisions about the solution, not about the factors, which have influenced that solution. Consequently, the important events of the system (i.e., what triggers a process) are rarely documented. Also, these documents do contain extensive process information, but they do not state how these processes were used, or whether alternative processes were considered. However, the documents do provide useful evidence of process variety and best practice. Further research would benefit from further exploring via interviews with domain experts or key stakeholders in order to develop a deeper understanding of these processes.

Therefore, these issues provide feedback to the design of the next iteration in order to increase the emergent benefits of SDL.

4.8 Feedback

The feedback for iteration two is summarised as follows:

1. SDL phases should be grouped or disjoint as:

   (a) *Procurement and Organisation of Legacy Assets (POLA)* and *Segmentation of Legacy Assets (SLA)* could be classified respectively as step 1 and 2 for one new phase called *Preparation of Legacy Assets (PLA)*.

   (b) Add *Reverse Engineering of BP Models* as step 3 in PLA.

   (c) *Semantic Analysis of BP Models* has one step called *Interpretation*

   (d) *Semantic Enhancement* will include generalisations of ontology process models.

   (e) *Pattern Documentation* is kept.
2. Discovered business process terms need to be mapped to BPMN elements, as a way of more the representation of process elements.
3. Generalisation of the ontology process models should be improved by replacing the informal procedure used in iteration one with more appropriate techniques and bring the generalisation step as part of *Semantic Enhancement* of the Ontology BP Modelling phase. This phase might be accomplished manually, with automated assistance or semi-automatically.
4. Business process ontology rules governing construction and interpretation of elements are required to ensure commitments are precise, because in iteration one the accuracy of the ontology business process models was constrained by the semantic gap between the process paradigm and the object paradigm. This was due to process element definitions that were either implicit or missing in the object paradigm.

Table 4-6 represents the feedback derived from the previous discussion and which will be used as the basis of iteration two.

Phases	Technique name or Step Name	Reason	How
Preparation of Legacy Assets (PLA)		New phase to group all the phases and the steps of preparing the Legacy Assets under this phase (PLA)	1. POLA phase 1 to step1. 2. SLA phase 2 to Step 2 3. Reverse Engineering step 1 phase 2 to step 3.
	Procurement and Organisational legacy Assets (POLA)	No Changes in its functionality just became as step 1 in PLA	No Changes
	Segmentation of Legacy Assets (SLA)	No Change in its functionality just became as step 2 in PLA	No Changes
	Reverse Engineering of Legacy Assets	Use BPMN Elements for defining BP extracted terms	Using BPMN modelling elements.
Semantic Analysis of BP Models (SA)		Kept as Phase 2 in SDL, only for semantic analysis	
	Interpretation	Business process ontology need to be defined	Using the object paradigm
Semantic Enhancement of BP Models			
	Apply Generalisation	Not used in iteration one, to produce a special model of others	Using an automated OWL, semi-automated OWL FACT++ Reasoner.
Patterns Documentation			
	Ericksson and Penker template	No changes, to document the discovered patterns and provide an easy accessibility.	No changes

Table 4-6: Feedback from Iteration One

4.9 Summary

Iteration one draws requirements from the initial suggestion of the preparation stage. The iteration then cycles repeatedly through mini iterations of *Design*, *Deployment* and *Evaluation* on real world scenarios. This stage is aimed at the iterative design, deployment and evaluation of the SDR methodological framework. These iterations deliver improvement in the artefact(s) because the individual iterations evolve the SDR methodology and environment incrementally. Initial construction of the SDL process began as an idea without a clear understanding of the form that each phase should take. Assuming that patterns could be constructed from legacy systems was simplistic and naive, however it was a necessary first step from which the research could progress.

It can broadly be stated that using SDL is coupled with a necessary learning curve, which cannot be avoided, though prior knowledge of business process modelling skills can expedite the understanding process. Also, there is a need to develop a Business Process Ontology (BPO). This iteration is however necessary as pattern discovery requires a new way of thinking rather than relying on past experience. Therefore, iteration two should be based on the feedback from the current iteration, so its steep learning process can be minimised and effectively managed to address iteration two.

CHAPTER 5 – ITERATION TWO: SDL IN THE EDUCATIONAL DOMAIN

5.1 Introduction

To facilitate the possibility and practicality of pursuing an ontology-based process patterns discovery approach, the focus now moves to provide a complete semantic discovery lifecycle. SDL is: (a) evolved to include an ontological business process model by interpreting organisational business processes, (b) restructured to conform to conventional discovery of business process patterns, and (c) defined so as to discover more general ontological process models that cater for more instance processes. Continual improvement of designed artefacts is one characteristic of Design Research and arises as knowledge and learning generated through build and evaluation cycles improves the designed artefact. The SDL is constructed and evaluated against Brunel University students and staff handbooks in the higher education domain, because the discovery of real world business process patterns should be discovered from the empirical analysis of organisational processes.

This chapter is organised as follows: Section 5.2 presents the evolvement of the SDL. Section 5.3 provides the ontological definition of business processes. Section 5.4 explains the chosen scenarios of Brunel University. Section 5.5 lays down the necessary groundwork for the practical application of the second version of the SDL. Section 5.6 presents the research outputs for iteration two. Section 5.7 evaluates the research outputs. Section 5.8 discusses the implications of the outputs. Section 5.9 describes feedback to influence the design of the next iteration. Finally, the chapter is summarised in Section 5.10.

5.2 Refinement of SDR Methodological Framework

Outcomes from Chapter 4 are centred on the evolvement of the Semantic Discovery Lifecycle's five phases and steps, the development of business process models and the generalised process models of the financial services domain. Figure 5-1 illustrates the first version of SDL phases, steps and its associated artefacts in iteration one.

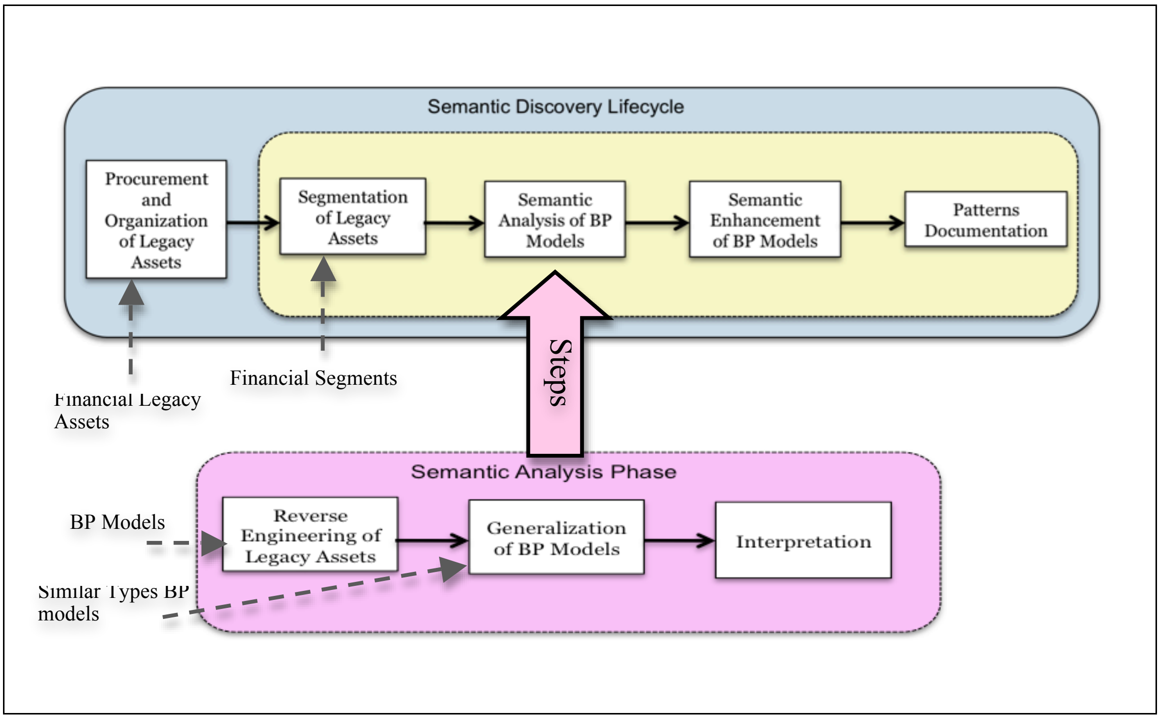

Semantic Discovery Lifecycle
Procurement and Organization of Legacy Assets
Segmentation of Legacy Assets
Semantic Analysis of BP Models
Semantic Enhancement of BP Models
Patterns Documentation
Financial Legacy Assets
Financial Segments
Steps
Semantic Analysis Phase
BP Models
Reverse Engineering of Legacy Assets
Generalization of BP Models
Interpretation
Similar Types BP models

Figure 5-1: First Version of SDL Phases, Steps and Associated Artefacts

In this chapter the researcher was able to move the research methodology forward focusing on: (a) evolving the classification of SDL phases and steps based on the previous knowledge and learning acquired from iteration one, (b) defining automated generalisation of business process ontology models based on running a reasoner for inferencing, and (c) developing the ontological definition of business processes based on the object paradigm. This iteration was motivated by the thinking that ontologies, formalised semantic models of real world systems or domains, are well suited to uncover and represent the underlying meaning of business processes. Additional knowledge appears more opportunistic when observing the discovery lifecycle and ontology together, with greater opportunity for effectiveness gains as knowledge increments are made. Reasons behind undertaking these changes in the second version of SDL are summarised as follows:

- The second version of SDL will clearly benefit from automation, as the first version of SDL process is highly manual and time-consuming.
- Improve the flow of the first version of SDL through re-distributing the steps among the phases and focus on specialisation of process types. Thus, the second version of SDL has the following phases: *Extract and Model processes* (phase 1), *Interpret* and

Formalise processes (Phase 2), *Generalise processes* (Phase 3) and *Document the Patterns* (Phase 4).

- *Extracted BP terms* are mapped to BPMN elements as a way of facilitating visualisation of the processes. In iteration one Ericksson and Penker (2000) non-standard process terms were adopted.
- For the *Semantic Analysis* phase the definition of the business process ontology model for *Interpreting* business processes and its *formalisation* in Protégé, enables consistency checking and automated classification of processes. Consistency amongst Business Process Ontology (BPO) model elements is highly important since incompatibility in their definition would most likely have negative effects on the whole process definition and overlap among the terms, i.e. verify whether there are any contradictions in the business process ontology.
- The *Semantic Enhancement* phase uses the reasoner to classify the business process models and identify inconsistencies, inferred subclass relations, and inferred equivalencies. Thus, for generalising business processes, reasoners can be used to derive inferences from the asserted processes, e.g. infer whether a particular process in an ontology is a sub-process of another, or whether particular activities, event or individuals in an ontology belongs to a specific class.

Figure 5-2 shows how SDL was refined in this iteration. From Figure 5-2 it can be noticed that in iteration two the steps of *Procurement and Organisation of Legacy Assets (POLA)*, *Segmentation of Legacy Assets (SLA)*, and *Reverse Engineering of BP Models* have been grouped and classified respectively as steps 1, 2 and 3 for a new phase called *Preparation of Legacy Assets (PLA)*, which represents the first phase of the refined Semantic Discovery Lifecycle. It is worth mentioning that the practicalities of each step have not been changed.

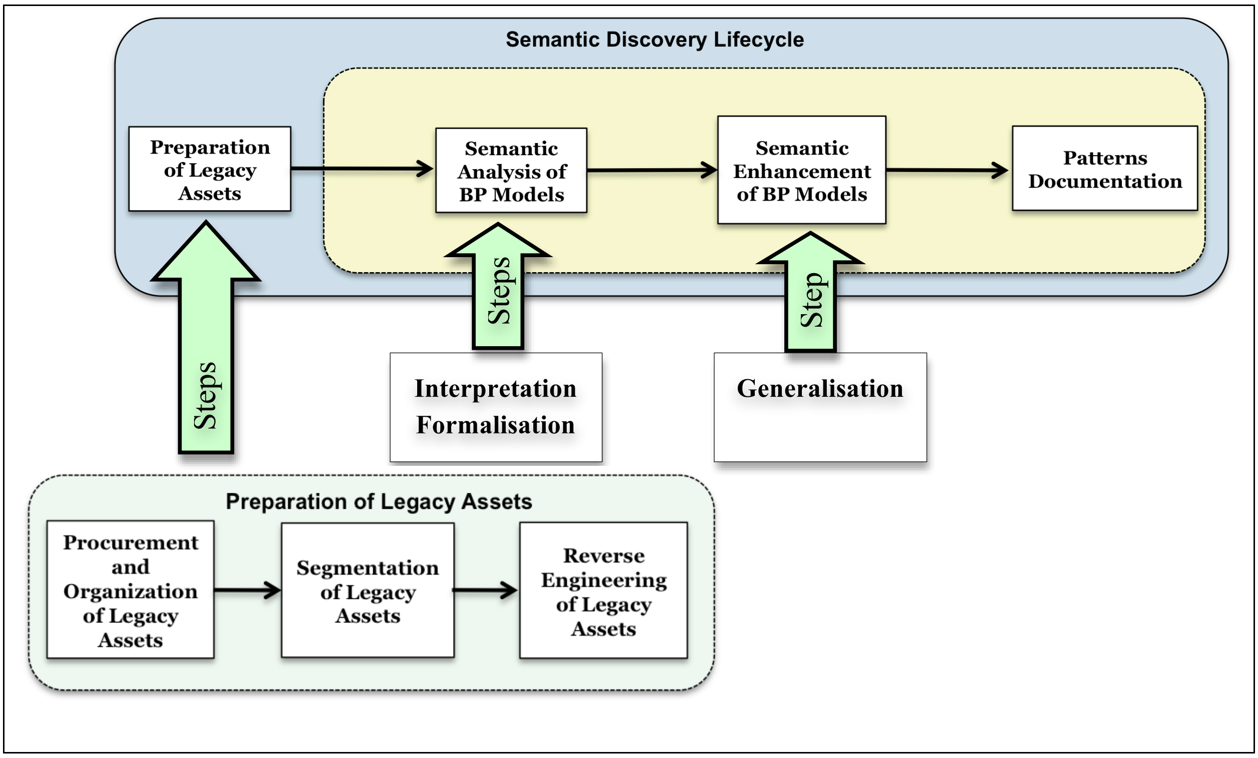

Semantic Discovery Lifecycle
Preparation of Legacy Assets
Semantic Analysis of BP Models
Semantic Enhancement of BP Models
Patterns Documentation
Steps
Steps
Step
Interpretation Formalisation
Generalisation
Preparation of Legacy Assets
Procurement and Organization of Legacy Assets
Segmentation of Legacy Assets
Reverse Engineering of Legacy Assets

Figure 5-2: Second Version of SDL Phases and Steps for this Iteration

The refined phases of the second version of SDL are now defined as follows:

Phase 1: Preparation of Legacy Assets

This provides SDL with organisational legacy assets that demonstrate the existence of certain types of models as well as their generalised recurrence across multiple organisations. Also during this phase business process models are extracted from the legacy assets. These models are typical process flow diagrams such as BPMN diagrams.

In Figure 5-2 only *Interpretation* takes place in the *Semantic Analysis of BP Models,* which represents the second phase of SDL.

Phase 2: Semantic Analysis of BP Models.

This phase along with the following represents the core of SDL. The elements of the process diagrams generated in phase one are semantically interpreted in order to derive more precise ontological models of the processes themselves and semantically richer than its predecessors. *Interpretation* identifies the business objects that the process commits to existing. *Interpretation* explicitly makes the business processes as much as possible close to real world objects, which ensures the grounding of the patterns to real world behaviour. For this phase the object paradigm (Partridge, 1996) provides a sound ontological foundation. Explanation for the ontological definition of business processes can be found in Section 5.3.

In Figure 5-2 only *Generalisation of Ontology BP* models takes part in the *Semantic Enhancement of Ontology BP Models*, which represents the third phase of SDL.

Phase 3: Semantic Enhancement of BP Models (SE).

This phase takes the ontological models created in *SA* and aims at generalising them to existing patterns or to newly developed patterns. Generalisation is an abstraction principle that allows defining an ontological model as a refinement of other ontological models. It sees a relationship between a general and specific model where the specific ontology model contains all the activities of the general model and more.

Phase 4: Pattern Documentation

Patterns Documentation is the fourth and last phase of SDL. Documentation plays an important role, bringing people from different groups together to negotiate and coordinate common practice as it plays a central role for global communication.

5.3 Business Process Ontology

As previously mentioned it was necessary to model a business process ontologically for making the processes as much as possible close to their real world counterparts, which ensures the grounding of the patterns to real world behaviour. Thus this definition is necessary for this phase here and this section will present the BPO that has been derived and used in this phase. During the *Semantic Analysis of BP Models* the focus is on the ontological definition of a business process model. This phase elicits the semantics underlying a business process in its fundamental parts (states, events, inputs, outputs). The process of elicitation, called *Interpretation*, identifies those real world objects whose individual process parts ontologically commit to (or refer to), as the modeller's aim is to provide a representation of all possible scenarios when presenting business process models. These scenarios correspond to possible future states in which the organisation can be. A business process model must cater for all foreseeable types of states (or possible worlds).

The paradigm that the proposed BPO is based on is the object paradigm (Partridge 1996) as explained in Section 2.6. The object paradigm (not to be confused with the object-oriented paradigm) determines the existence of an object (or thing) through its spatio-temporal extension. Hence, in the object paradigm a thing exists because it has a spatio-temporal extension in our universe. The Object Paradigm adopts the principles underlying perduratism (Sider, 2001) in which the identity of an object or thing is determined by its spatiotemporal extension. Figure 5-3 represents the foundation ontology that the object paradigm acknowledges the existence of.

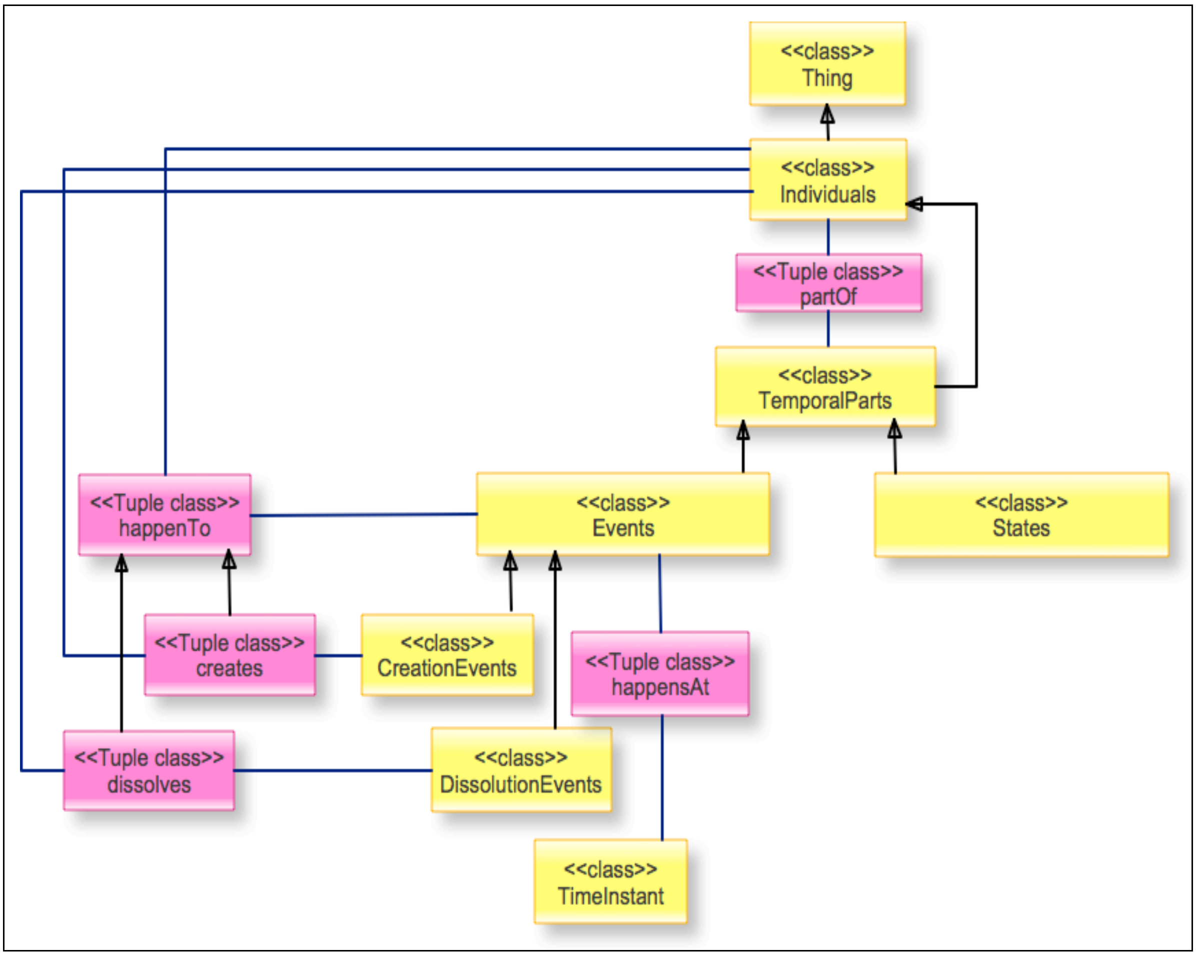


Figure 5-3: Foundational Ontology of the Object Paradigm (Partridge 1996)

Explanation of the foundational terms of Figure 5-3 can be found in Table 5-1, as follows:

Object Paradigm Terms	Description
Things (or Objects)	Anything that exist or can exist in any possible universe
Individuals	Things with a spatio-temporal extension.
Classes or (Types)	Sets of things. Types or sets of similar objects. The extension of a class is given by the extensions of all its instances;
Tuples	Relationships between things
Tuple Classes	Classes of similar tuples
Individuals	Things with a four-dimensional (4D) extension and, unlike classes, do not have instances (hence the name individual);
Temporal Parts	Temporal parts of 4D objects. A temporal part is an individual that is part of the overall temporal extension of an individual. There are two types of temporal parts, states and events.
States	A state is a temporal part of an individual over a period of time (i.e. with a time length greater than zero). Temporal parts with duration. A special predecessor relationship can exist between states whereby one state temporally precedes another.
Events	An instantaneous temporal part of an individual with no time length. An event has only a spatial extension. Temporal parts that occur instantaneously thus have no duration. Particular classes of events are Creations Event and Dissolutions Event. The former represent events from which objects are generated, while the latter represent events that dissolve or terminate an object. Events happen at specific time instants and happen to one or more objects.
Creation Events	An event that triggers the creation (coming into existence) of an individual.
Dissolution Events	An event that dissolves an individual, i.e. a dissolution event coincides with the last instant of an individual's spatio-temporal extension.
temporalPartOf	A temporalPartOf tuple represents the relationship between a temporal part and its individual whole.
happensTo	A happensTo tuple represents the relationship between an event and an individual affected by that event.
happensAt	A happensAt tuple represents the relationship between an event and the time instant it occurs at.
Time Instants	A time instant is an individual moment in time.

Table 5-1: Object Paradigm: Definition of Terms

Having explained the foundational ontology of the object paradigm, the researcher in the next section introduces the business process ontology and discusses its relevance and usefulness in *Interpretation* of the *Semantic Analysis of BP Models*. Figure 5-4 represents Business Process Ontology and provides a broad overview of the types of objects that the Business Process Ontology acknowledges the existence of.

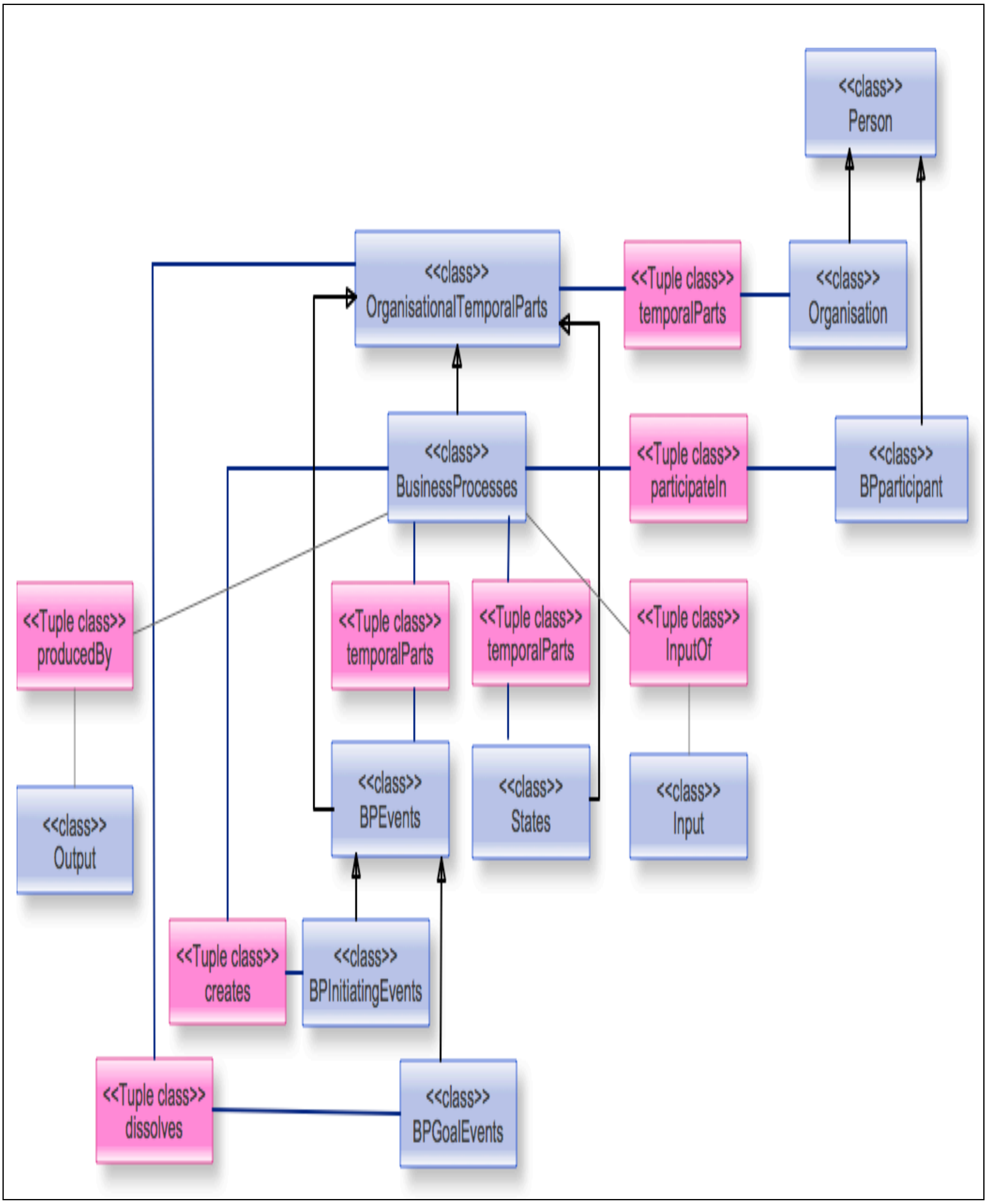

<<class>> Person
<<class>> OrganisationalTemporalParts
<<Tuple class>> temporalParts
<<class>> Organisation
<<class>> BusinessProcesses
<<Tuple class>> participateIn
<<class>> BPparticipant
<<Tuple class>> producedBy
<<Tuple class>> temporalParts
<<Tuple class>> temporalParts
<<Tuple class>> InputOf
<<class>> Output
<<class>> BPEvents
<<class>> States
<<class>> Input
<<Tuple class>> creates
<<class>> BPInitiatingEvents
<<Tuple class>> dissolves
<<class>> BPGoalEvents

Figure 5-4: Business Process Ontology

Explanation of the Business Process Ontology terms of the previous Figure 5-4 can be found in Table 5-2 as follows:

Business Process Ontological Term	Descriptions
Classes or (Types)	Sets of things. Types or sets of similar objects. The extension of a class is given by the extensions of all its instances;
Tuple Classes	Classes of similar tuples
Tuples	Relationships between things
Organisational Temporal Parts	Are temporal parts of an organisation. These temporal parts can either have a duration (i.e. states) or be instantaneous (i.e. events) thus having no duration. Some temporal parts of an organisation can be composed of other temporal parts (i.e. activities and business process events) to form a business process.
Business Processes	Are states of organisation that are normally instances of predefined (i.e. previously modelled) classes. Business processes have as a minimum the following temporal parts: one initiating event, one activity and one goal. The events and activities that form a business process are temporally sequenced.
Business Process Events	Are temporal parts of a business process with no duration. Such events can start/trigger or end a business process or an activity. Its subclasses are Business Process Initiating Events and Business Process Goals
Business Process Initiating Events	The initiating events of a business process are those events that trigger a business process by effectively creating it. Such events are normally predefined (i.e. previously classified in a business process model).
Business Process Goals	Are events that terminate a business process and which are normally predefined (i.e. previously classified in a business process model)
Inputs	Are individual things that take part in (or participate in) a business process (or its activities).
Outputs	Are individual things that are produced by a business process (or its activities).
Persons	Are human individuals or organisations
Organisations	An organisation is a type of person that is legally recognised by a State and which by law has rights and obligations
BPParticipants	Are persons who take part in a business process at one stage or another and assume roles and responsibilities in relation to that process

Table 5-2: Business Process Ontology: Definition of Terms.

When an organisation carries out what is normally known as a business process, the organisation and/or one of its parts (i.e., organisational units) enters a particular state (*Business Process)*. This state is initiated by an event (*BP Initiating Event)*, which is triggered by *BP Participants* that take part in the business process. The state then terminates as a consequence of a final (dissolution) event (*BP Goal Event)*. The rationale behind the importance of Business Process Ontology definition is that it affects human and organisation's understanding of the represented phenomenon (Shanks et al., 2002). The researcher particularly highlights the value of this definition when examining its use during *Semantic Analysis of BP Models*. The business process model concept is criticised for being confused mainly with different communication and understanding of its terms. Hence, the resulting ontology contributes to clarifying the essence of a business process by shaping the boundaries and identifying the elements of a process, as well as resolving any conflict it has with other concepts. The relationships among the different types of objects that the object paradigm and the BPO acknowledge the existence of are presented in Figure 5-5.

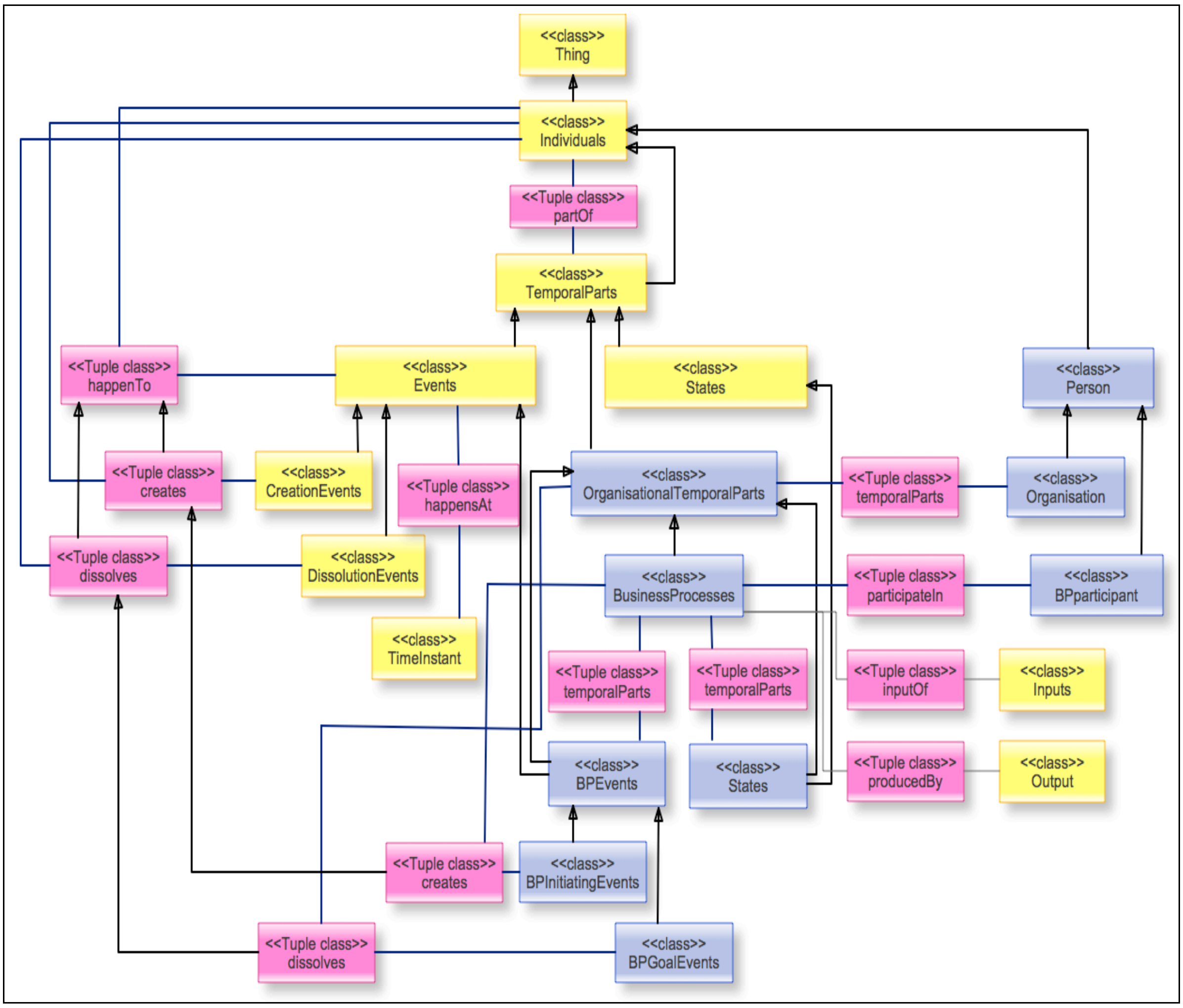
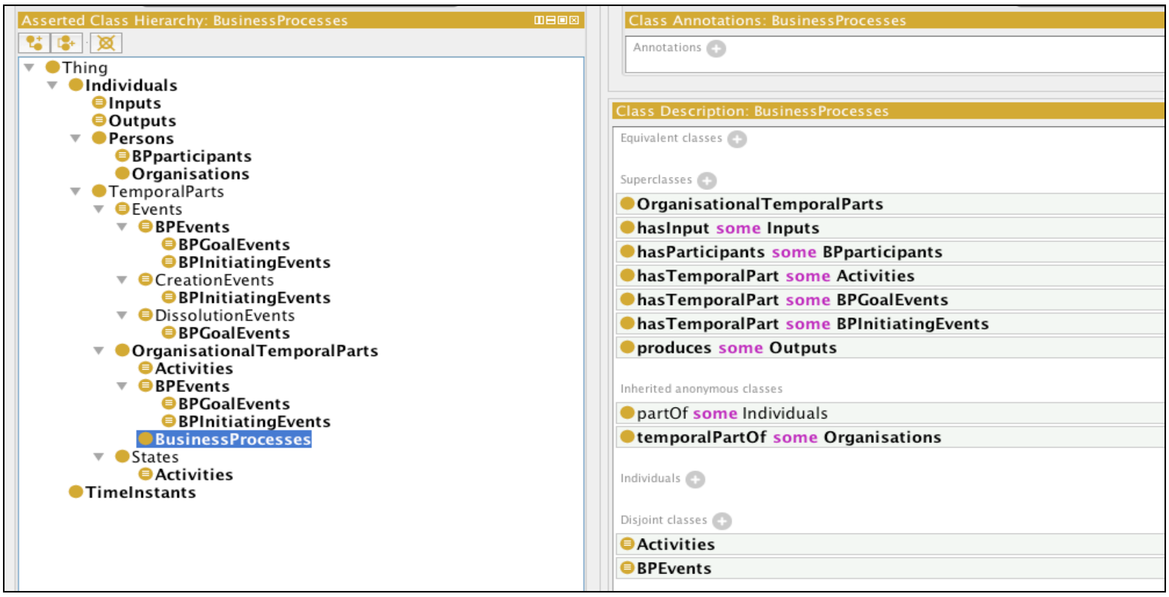


Figure 5-5: Relationships between the Business Process Ontology and the Foundational Ontology

Protégé-OWL is used as an ontology development platform for constructing the Business Process Ontology (Figure 5-6). Protégé - developed at Stanford University- is authorised by the World Wide Web Consortium (W3C), it is an open source integrated environment, a standalone application, and enjoys an extendible architecture with several plug-ins (Corcho et al. 2003). However, after implementing the ontology in Protégé-OWL, the research represents it using the OWL language given that it is a general-purpose language for representing information on the Web (Appendix E). OWL is the proposed standard for Web ontologies. It allows describing the semantics of knowledge in a machine-accessible way (Antoniou and van Harmelen, 2010). Representing the developed ontology in OWL makes it formal and gives flexibility to different organisations and other beneficiaries to use or reuse the ontology in different existing and future applications. OWL reasoning engines, such as FaCT++, are deployed to check for the inconsistencies automatically.



Asserted Class Hierarchy: BusinessProcesses

```
Thing
  Individuals
    Inputs
    Outputs
    Persons
      BPparticipants
      Organisations
    TemporalParts
      Events
        BPEvents
          BPGoalEvents
          BPInitiatingEvents
        CreationEvents
          BPInitiatingEvents
        DissolutionEvents
          BPGoalEvents
      OrganisationalTemporalParts
        Activities
        BPEvents
          BPGoalEvents
          BPInitiatingEvents
        BusinessProcesses
      States
        Activities
  TimeInstants
```

Class Annotations: BusinessProcesses

Annotations

Class Description: BusinessProcesses

Equivalent classes

Superclasses

OrganisationalTemporalParts
hasInput some Inputs
hasParticipants some BPparticipants
hasTemporalPart some Activities
hasTemporalPart some BPGoalEvents
hasTemporalPart some BPInitiatingEvents
produces some Outputs

Inherited anonymous classes

partOf some Individuals
temporalPartOf some Organisations

Individuals

Disjoint classes

Activities
BPEvents

Figure 5-6: Screen shot of the Developed Business Process Ontology in Protégé

Developing an ontology for organisational processes, primarily on the basis of business process modelling thinking, would be of value to academics and practitioners alike, particularly those interested in process reuse and business developments. In this research, it is hoped that developing this ontology will enable the precise identification and categorisation of the key concepts and relationships and produce unambiguous semantics of the organisational processes.

5.4 Iteration Two Data Sources

In this iteration, Brunel University handbooks are utilised, as a form of organisational knowledge source. Brunel University is a higher education institution and one of a number of British universities created in the 1960s and situated in Uxbridge, West London in England.

The range of services offered by higher education institutions includes teaching, research, exacting applied work and social services activities of universities. Within the realm of teaching, it also includes specialised services like enrolments, inductions, leaves, recruitments, finances, graduation, policies and more. However, the education systems around the globe have undergone dramatic changes. In large part this is the willingness to tie education more closely to the economy in order to drive economic growth, and on the other to develop the education sector in such a way that it directly generates income for institutions, national economies and for profit firms who are moving into particular sectors. As a result, it has been thought of as another domain for iteration two of this research.

Two handbooks form the basis of the current work. These two handbooks are for staff and students of the Department of Information Systems and Computing (DISC) at Brunel University. Each of these sources has been developed over time as guidance to address the educational institution needs. The rest of this section provides an overview of each of these handbooks.

5.4.1 Brunel DISC Staff

The DISC staff handbook serves the purpose of communicating the staff policies, procedures and rules to all employees, and equally important, their responsibilities. It accommodates a suite of processes of interest to DISC staff during their employment:

- Set out and clarify the terms and conditions of employment i.e. grievances.
- Set out and clarify the rules and regulations that staff must abide by i.e. discipline.

- Summarise the benefits you may be entitled to i.e. leave payment.
- Set out some of the main procedures that take part during staff's employment i.e. supervisions and terminations.
- The Staff Handbook forms part of every employee's contract of employment and should be read in conjunction with that contract.

5.4.2 Brunel DISC Students

The DISC students handbook contains essential information for new and continuing students, including details of academic and support services, activities, facilities and administrative procedures. It consists of a suite of student related processes such as induction and enrolment procedures and policy guidelines.

5.5 Practical Application of SDL

In light of the above, the focus now turns to the practical application of the Semantic Discovery Lifecycle (SDL) in order to (a) identify ontological models of business processes from the handbooks, and (b) develop business process patterns for this domain. The phases carried out were as follows:

Phase 1: Preparation of Legacy Assets (PLA)

The phase started with bringing a new knowledge source from Brunel University staff and student handbooks and ended with the production of BPMN models as follows:

Step 1: Procurement and Organisation of Legacy Assets (POLA)

The knowledge sources used in iteration two are Brunel University students and staff Handbooks in the higher education domain. These handbooks were produced by Brunel University for staff and students at the university. The legacy assets used for SDL are: (1) Handbook documents for staff and students and (2) Knowledge acquired from the literature on the educational domain.

Step 2: Segmentation of Legacy Assets (SLA)

Given that each of the handbooks incorporates around 6 sections (comprising tens of procedures), *Segmentation of Legacy Assets* provides a way to approach and manage those systems, by dividing each handbook into a hierarchy of segments that facilitates the analysis of this domain and ensures that irrelevant assets are not included in the analysis. Figure 5-7 illustrates Brunel University legacy assets segmentation.

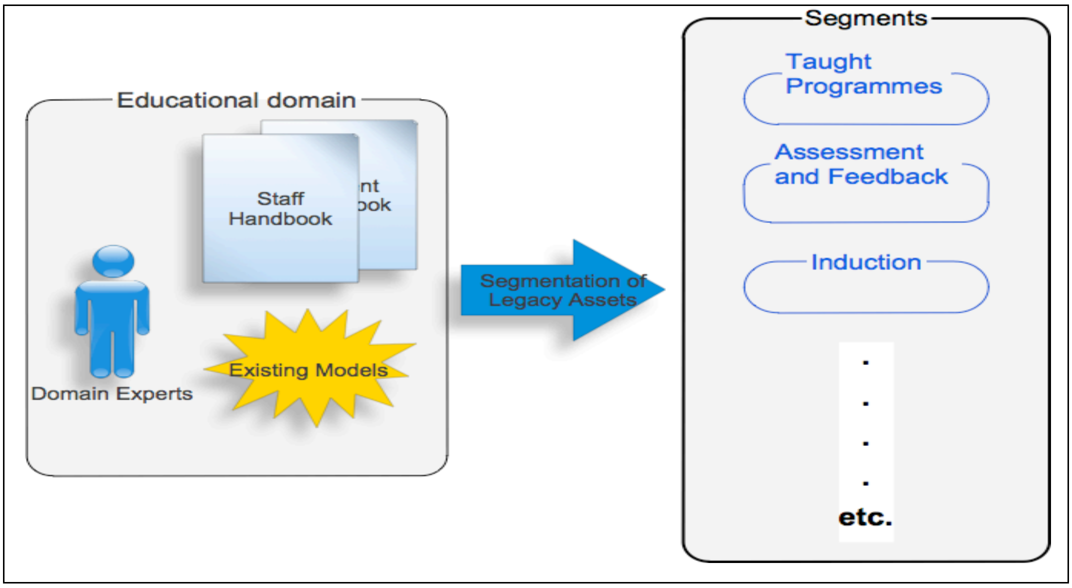


Figure 5-7: Brunel University Segmentation

This phase involved reformatting the assets in terms of types that the handbooks consist of ’Staff’ and ‘Student’ segments Table 5-3 provide a list of all segments related to Brunel University handbooks.

In addition, the segments were divided into processes, deemed necessary when reverse engineering the handbook segments to build BPMN diagrams. Table 5-3 illustrates a list of all the processes related to Brunel procedures. Each of these processes are identified and allocated to the concerned segments. The process consists of all parts (chapters and/or sections) of the Brunel handbooks.

Educational Domain	Segments	Processes
DISC Handbook	Staff	Arrange Staff Induction
		Terminate of Fix Term Contract
		Apply For Leave
		Claim Expenses
		Prepare Assessment Process
		Prepare Examination Paper
		Hold Mitigating Circumstance Panel
		Appoint of External Examiner
		Process of PGR Application
		Arrange PhD Induction
		Apply for Ethical Approval
	Student	Define Examination Process
		Identify Cheating Rules
		Review Modules
		Submit Coursework
		Apply for Ethical Approval
		Apply for Mitigating Circumstance

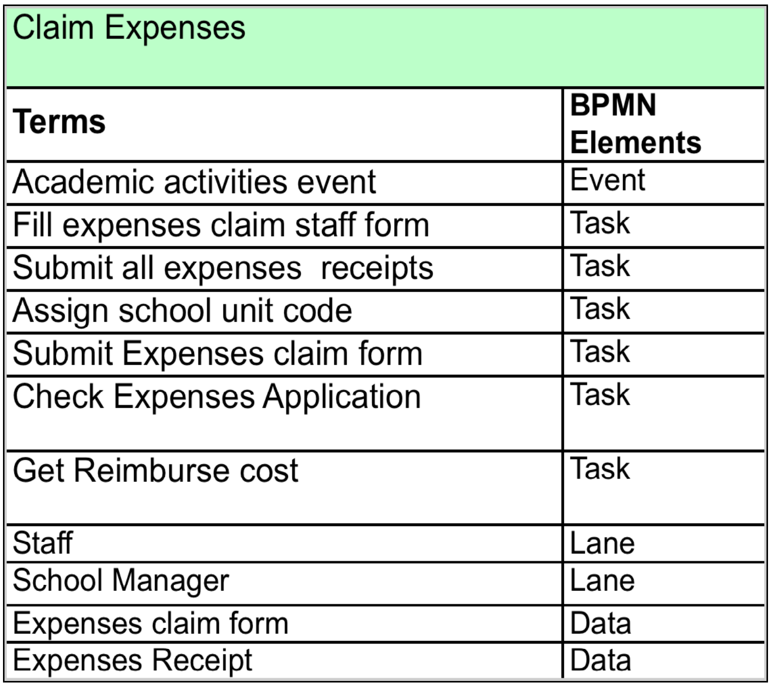
Table 5-3: Segmentation of Legacy Assets

Step 3: Reverse Engineering of Legacy Assets

In this step the reverse engineering technique uses the list of processes identified in Step 2 (*Segmentation of Legacy Assets*) to focus on more abstracted representations of the assets, since during this step questions are asked and required to be answered concerning more details to define processes:

- What business processes take place?
- What type of activities takes place?
- What conditions are required for business processes?

It is during this step that the processes are explicitly defined, understood and visualised to support those business processes. An example representation is demonstrated below. For the ‘Staff’ segment, ‘Claim Expenses’ is a process. The reverse engineering technique is used to extract all the elements related to the ‘Claim Expenses’ process. Table 5-4 shows all the extracted elements related to ‘Claim Expenses’.



Claim Expenses	
Terms	**BPMN Elements**
Academic activities event	Event
Fill expenses claim staff form	Task
Submit all expenses receipts	Task
Assign school unit code	Task
Submit Expenses claim form	Task
Check Expenses Application	Task
Get Reimburse cost	Task
Staff	Lane
School Manager	Lane
Expenses claim form	Data
Expenses Receipt	Data

Table 5-4: Claim Expenses Business Process for Brunel University Staff

Answering the above questions helped to define ‘Claim Expenses’ according to BPMN elements. The table summarises the *Tasks* used in defining ‘Claim Expenses’. In Table 5-4 the representation of the process required to define various *tasks* starting from the first *task* of ‘Fill expenses claim staff form’ to the final *task* of ‘Get reimburse cost’. Also, additional information is defined for this business process such as *Data*, *Lanes* and *Events*. The *Data* can be either inputs or outputs. Input represents the type of information required as an entry for ‘Claim Expenses’, while the output represents the type of information that can be produced by processing ‘Claim Expenses’. *Lanes* ‘Staff’ and ‘School manager’ are the participants in ‘Claim Expenses’ business process. *Events* of this process include ‘Academic Activities Event’ which can be initiated by the academic staff to claim expenses. Other business processes can be found in Appendix A.

After the *Reverse Engineering* step, the conceptualisation of the ‘Claim Expenses’ process can take place by creating BPMN diagrams for Table 5-4. Figure 5-8 illustrates the BPMN model of ‘Claim Expenses’ business process presented in Table 5-4.

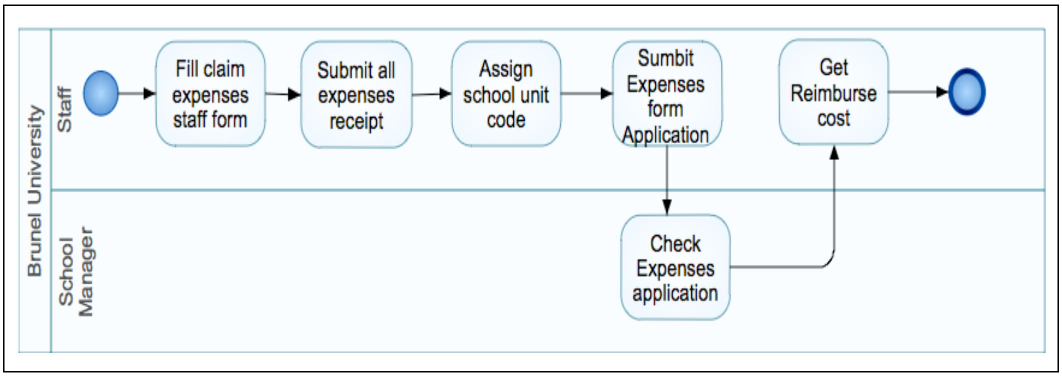

Brunel University
Staff
School Manager
Fill claim expenses staff form
Submit all expenses receipt
Assign school unit code
Sumbit Expenses form Application
Get Reimburse cost
Check Expenses application

Figure 5-8: BPMN Model for 'Claim Expenses'

Phase 2: Semantic Analysis of BP Models (SA)

The second phase of SDL is *Semantic Analysis of BP Models*. Elements of the 'Claim Expenses' process previously developed in phase one are semantically interpreted using the ontological definition of business process in order to derive more precise ontological models of the processes themselves and far more sophisticated and semantically richer than its predecessors through ***Interpretation*** Following the rules of transformation explained earlier. The interpretation starts by analysing the first row in Table 5-5 which signifies that the sign 'Fill expenses claim staff form' BPMN task element commits to the existence of an object called 'Fill in Expenses Form For Cost Spent on Academic Activities', which is an *Activity* in the BPO. Also the *BP Initiating Event* has been recognised as 'Money spent on academic activities', which triggers 'Claim Expenses' as illustrated in Table 5-5.

ClaimExpenses			
Terms	**BPMN Elements**	**Interpretation**	**BPO Element**
Academic activities event	Event	MoneySpentOnAcademicActivities	BP Initiating Event
Fill expenses claim staff form	Task	FillinExpensesFormForCostspent OnAcademicActivities	Activities
Submit all expenses receipts	Task	AttachProofOfExpensesReceipts	Activities
Assign school unit code	Task	AssignSchoolUnitCode	Activities
Submit Expenses claim form	Task	SubmitExpensesForm	Activities
Check Expenses Application	Task	CheckSubmittedFormandReceipt	Activities
		ProvideExpensesDecisionLetter	Activities
Get Reimburse cost	Task	CalcuateExpenses	Activities
		ReimburseCostAccepted	BP Goal Event
Staff	Lane	Staff	BP Participant
School Manager	Lane	SchoolManagers	BP Participant
Expenses claim form	Data	ExpensesClaimForm	Input
Expenses Receipt	Data	ExpensesReceipt	Input
		ExpensesDecisionLetters	Output

Table 5-5: Ontological Interpretation of Claim Expenses

In Table 5-5 to ontologically represent 'Claim Expenses', the process requires various activities starting from the *first activity* of 'Fill in Expenses Form For Cost Spent on Academic Activities' to the *final activity* of 'Calculate Expenses'. The other individual terms follow a similar pattern. Also, additional information is defined for this business process such as *Initiating Event*, *Dissolution Event*, *Input* and *Output*. Using the Business Process Ontology helped to add more understanding to BPMN such as 'Get Reimburse Cost' (originally a BPMN task) is further interpreted in BPO as 'Calculate Expenses' and 'Reimburse Cost Accepted' (activities and goal event respectively). For this business process it was not clear what the goal event was. While using the BPO definition, the researcher was able to state that the goal event occurs when the expenses are accepted thus terminating the process. Also to have a clearer model other activities have been added to improve the flow of the business process. Other ontological business processes can be found in Appendix B.

After creating Table 5-5 and to show an improved model of business process, the conceptualisation of ‘Claim Expenses’ can take place by creating a BPMN model for Table 5-5. Figure 5-9 illustrates the BPMN model of the interpreted ‘Claim Expenses’ ontology process model.

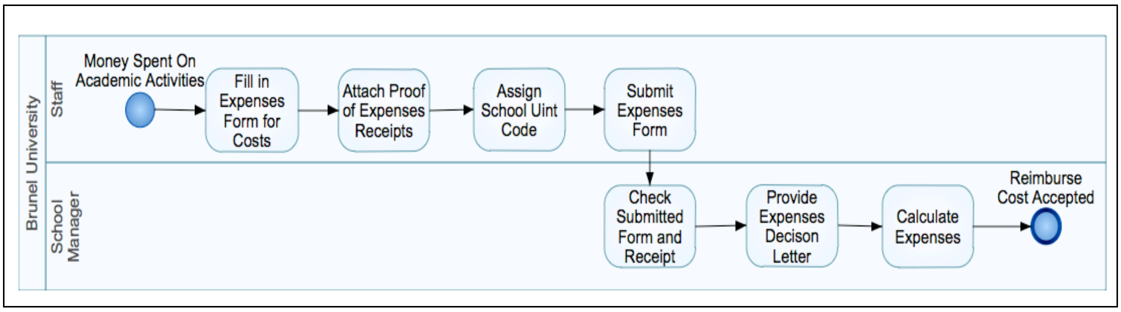
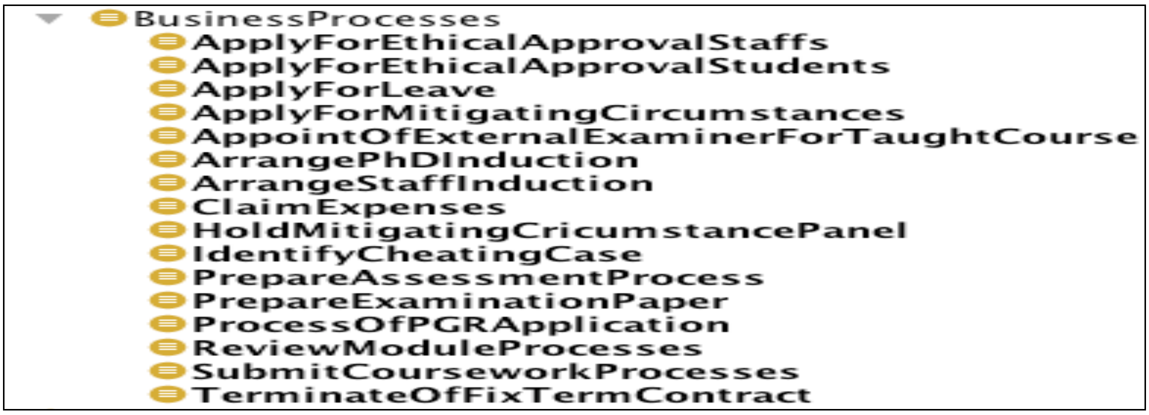

Brunel University
Staff
School Manager
Money Spent On Academic Activities
Fill in Expenses Form for Costs
Attach Proof of Expenses Receipts
Assign School Uint Code
Submit Expenses Form
Check Submitted Form and Receipt
Provide Expenses Decison Letter
Calculate Expenses
Reimburse Cost Accepted

Figure 5-9: BPMN model for 'Claim Expenses' based on the BPO

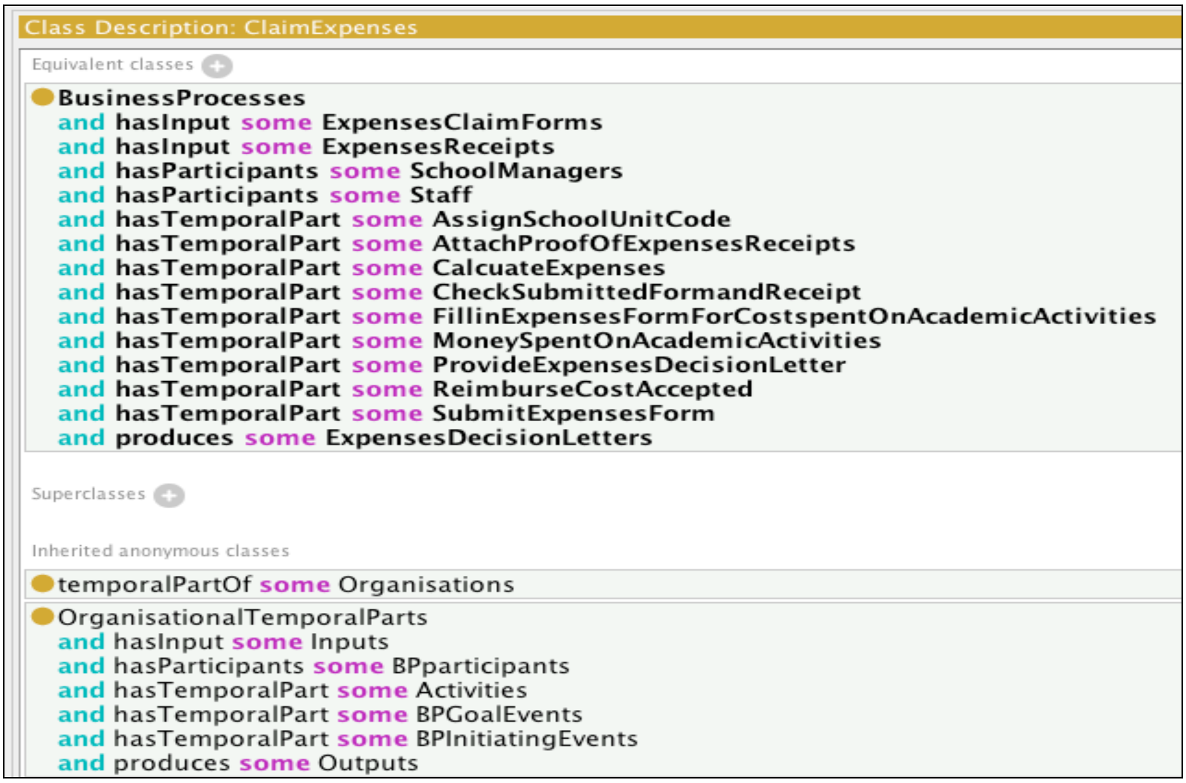
During *Semantic Analysis* the creation of the Business Process Ontology in Protégé-OWL matured to enable the analyst to create a new ontology, link it to external ontologies and carry out consistency checks on the working or completed ontology. All Brunel staff and students interpreted business processes have been uploaded in a semi-automated tool - Protégé, using the Business Process Ontology definition constructed in OWL earlier. See Figure 5-10 for all the Brunel staff and students Ontology Business Processes.



BusinessProcesses
ApplyForEthicalApprovalStaffs
ApplyForEthicalApprovalStudents
ApplyForLeave
ApplyForMitigatingCircumstances
AppointOfExternalExaminerForTaughtCourse
ArrangePhDInduction
ArrangeStaffInduction
ClaimExpenses
HoldMitigatingCricumstancePanel
IdentifyCheatingCase
PrepareAssessmentProcess
PrepareExaminationPaper
ProcessOfPGRApplication
ReviewModuleProcesses
SubmitCourseworkProcesses
TerminateOfFixTermContract

Figure 5-10: Brunel University Staff and Student Business Process Ontology in Protégé

The ‘Claim Expenses’ business process for staff is a subclass of business processes, where it inherits the definition of business process. This is only one example of many other Brunel processes presented in Protégé. See Figure 5-11.



```
BusinessProcesses
 and hasInput some ExpensesClaimForms
 and hasInput some ExpensesReceipts
 and hasParticipants some SchoolManagers
 and hasParticipants some Staff
 and hasTemporalPart some AssignSchoolUnitCode
 and hasTemporalPart some AttachProofOfExpensesReceipts
 and hasTemporalPart some CalcuateExpenses
 and hasTemporalPart some CheckSubmittedFormandReceipt
 and hasTemporalPart some FillinExpensesFormForCostspentOnAcademicActivities
 and hasTemporalPart some MoneySpentOnAcademicActivities
 and hasTemporalPart some ProvideExpensesDecisionLetter
 and hasTemporalPart some ReimburseCostAccepted
 and hasTemporalPart some SubmitExpensesForm
 and produces some ExpensesDecisionLetters

temporalPartOf some Organisations

OrganisationalTemporalParts
 and hasInput some Inputs
 and hasParticipants some BPparticipants
 and hasTemporalPart some Activities
 and hasTemporalPart some BPGoalEvents
 and hasTemporalPart some BPInitiatingEvents
 and produces some Outputs
```

Figure 5-11: Illustrates 'Claim Expenses' Development in Protégé OWL

OWL reasoning engines such as FaCT++ are deployed to check for the consistency of the developed business process ontologies.

Phase 3: Semantic Enhancement of BP Models (SE):

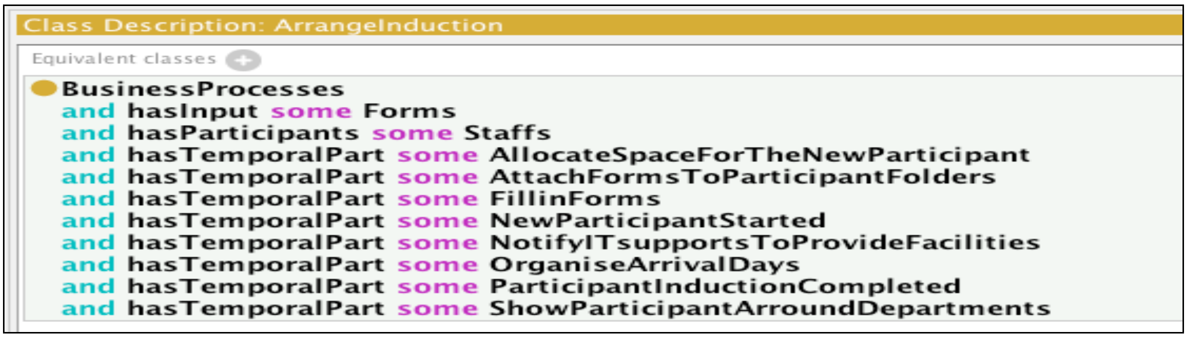
This phase starts by taking the ontological business process model of Brunel staff and students created in the *Semantic Analysis* phase and aims at generalising them to existing patterns or to newly developed patterns. Generalisation works by collecting process elements (i.e., activities, event, etc.) and analyses them for commonalities before generalising them. The approach is different to abstraction, as rather than hiding details, generalisation aims to make the original collection redundant, without any loss of information together with a smaller and simpler collection of processes. The generalisation of ontological process models has been accomplished in a semi-automated way, using the OWL reasoning engine FaCT++, to bring similar processes under general processes. Many generalisation rules for business processes have been defined using different BPO elements (see Appendix C for all the types of general processes). Each generalisation achieved is described as follows:

First: Generalise Similar Business Processes Elements Types

1. Arrange Induction has been defined as a general business process since it has been generalised according to BPO elements, i.e. Input, Participant, BP Initiating Event, BP Goal Event and Activities, see Table 5-6 and Figure 5-12.

BPO	Arrange Induction
BP Participant	Staffs
Input	Forms
BP Initiating Event	New Participant Started
BP Goal Event	Participant Induction Completed
Activities	Allocate Space For The New Participant
	Attach Forms To Participant Folders
	Fill in Forms
	Notify IT supports To Provide Facilities
	Organise Arrival Days
	Show Participant Around Departments

Table 5-6: Arrange Induction Generalised BP ontology Model



```
BusinessProcesses
  and hasInput some Forms
  and hasParticipants some Staffs
  and hasTemporalPart some AllocateSpaceForTheNewParticipant
  and hasTemporalPart some AttachFormsToParticipantFolders
  and hasTemporalPart some FillinForms
  and hasTemporalPart some NewParticipantStarted
  and hasTemporalPart some NotifyITsupportsToProvideFacilities
  and hasTemporalPart some OrganiseArrivalDays
  and hasTemporalPart some ParticipantInductionCompleted
  and hasTemporalPart some ShowParticipantArroundDepartments
```

Figure 5-12: Introduces 'Arrange Induction' General BPO and Restriction in Protégé

From running the FaCT++ Reasoner, both ‘Arrange PhD Induction’ and ‘Arrange Staff Induction’ have been classified as subclasses of *Arrange Induction* (General BP). This was achieved through defining a few necessary and sufficient conditions.

Thus, both ‘ Arrange PhD Induction’ and ‘ Arrange Staff Induction’ satisfy ‘Arrange Induction’ restrictions, and became a subclass of it. See Figure 5-13.

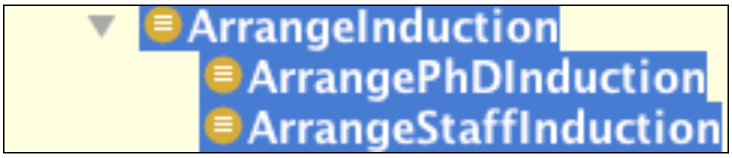


Figure 5-13: Sub-Processes of 'Arrange Induction'

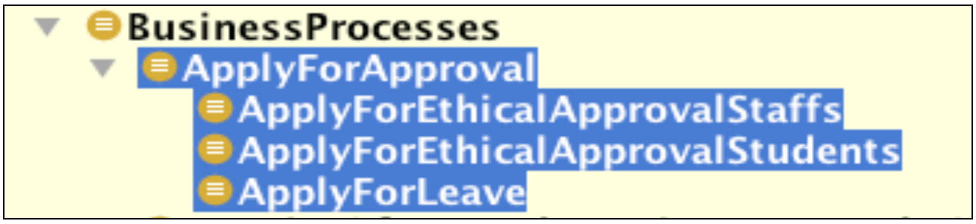
This can be considered a good example of generalisation since both ‘Arrange Staff Induction’ and ‘Arrange PhD Induction’ differ in the type of form that is used as an input, and the initiating event that triggers the process. Activities are of a similar type like organise arrival, fill in forms, etc., and the dissolution events are also of a similar type as illustrated in Table 5-7.

More examples of generalisation follow the same route of the previously discussed ‘Arrange Induction’ general process. Below are two more examples of generalised processes.

BPO	***Arrange Induction***	**Arrange PhD Induction**	**Arrange Staff Induction**
BP Participant	*Staff*	Research Admin	Line Manager
Input	*Forms*	Research Starting Form	Induction Check List Form
BP Initiating Event	*New Participant Started*	New PhD Candidate Started	New Staff Starts Work
BP Goal Event	*Participant Induction Completed*	PhD Induction Completed	Staff Induction Completed
Activities	*Allocate Space For The New Participant*	Allocate Space For PhD Candidate	Allocate Space For New Staff
	Attach Forms To Participant Folders	Submit Research Starting Forms	Attach Induction List Form To Employee Folder
	Fill in Forms	Fill in Research Starting Form	Fill in Induction Check List Form To Ensure Induction Completion
	Notify IT supports To Provide Facilities	Notify IT to Provide Facilities	Notify IT Support To Provide Facilities
	Organise Arrival Days	Organise PhD Arrival	Organise Staff Arrival Day
	Show Participant Around Departments	Show PhD Around Department	Show New Staff Around

Table 5-7: Arrange Induction Generalised BPO Models with Similar Type Processes

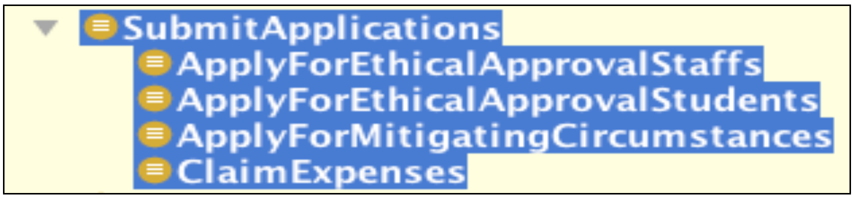
2. Apply For Approval has been generalised by having similar types (Input, Output, Participant, BP Goal Event and Activities). Both 'Apply For Ethical Approval Staff', 'Apply for Ethical Approval Students' and 'Apply For Leave' have been classified as subclasses of *Apply For Approval* (General BP). See Figure 5-14 below.



- BusinessProcesses
  - ApplyForApproval
    - ApplyForEthicalApprovalStaffs
    - ApplyForEthicalApprovalStudents
    - ApplyForLeave

Figure 5-14: 'Apply For Approval' and Sub-processes

3. Submit Applications has been generalised by having similar types (Input, Participant and Activities). All business processes 'Apply For Ethical Approval Staff', 'Apply for Ethical Approval Students', 'Apply For Mitigating Circumstance' and 'Claim Expenses' have been classified as subclasses of *Submit Applications* (General BP). See Figure 5-15.



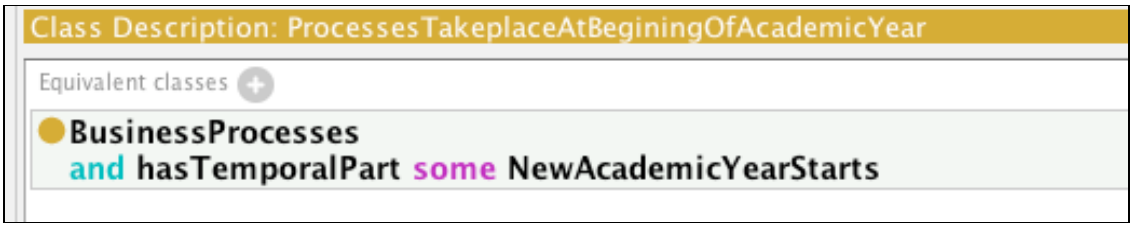
- SubmitApplications
  - ApplyForEthicalApprovalStaffs
  - ApplyForEthicalApprovalStudents
  - ApplyForMitigatingCircumstances
  - ClaimExpenses

Figure 5-15: 'Submit Application' and Sub-Processes

Second: Generalisation of Similar Business Processes Initiating Event

This type of generalisation depends on the Initiating Event of a business process. Thus, it brings all the business processes that have a similar type of initiating event to trigger their business processes; below are the generalised business processes achieved from applying this generalisation rule:

1. Processes Take Place At The Beginning Of The Academic Year: is a General Business Process by defining a restriction that all processes should have a similar type of initiating event 'New Academic Year Starts' that triggers different business processes when new academic year starts, as presented in Figure 5-16.



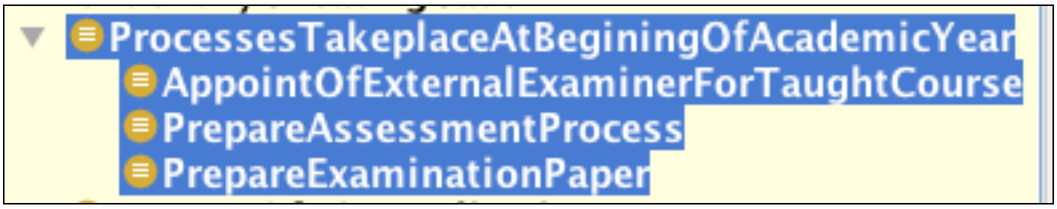
Class Description: ProcessesTakeplaceAtBeginingOfAcademicYear

Equivalent classes

BusinessProcesses
and hasTemporalPart some NewAcademicYearStarts

Figure 5-16: Defines 'Processes Take Place At The Beginning of The Academic Year' Restriction

When using FaCT++ it classifies together all the processes that have similar initiating events as in Figure 5-17.



ProcessesTakeplaceAtBeginingOfAcademicYear
AppointOfExternalExaminerForTaughtCourse
PrepareAssessmentProcess
PrepareExaminationPaper

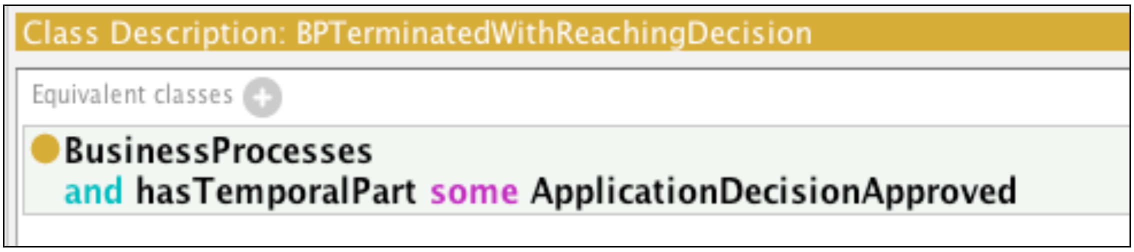
Figure 5-17: 'Processes Take Place At The Beginning Of The Academic Year' and Sub-Processes

Another example of generalised business process is provided by those having a similar type of Initiating Event as in ‘BP Triggered When Deadline Reached’ and ‘BP Trigger When New Participant Started’ available in Appendix C.

Third: Generalisation of Similar Business Processes Goal Event:

This type of generalisation depends on the goal event of a business process. Thus, it classifies all the business processes that have a similar type of goal event to dissolve their business processes. Below are the generalised processes obtained from this generalisation rule:

1. **BP Terminated With Reaching a Decision** is a general process defined by one restriction that all processes should have a similar type of goal event, i.e. ‘Application Decision Approved’ (see Figure 5-18).



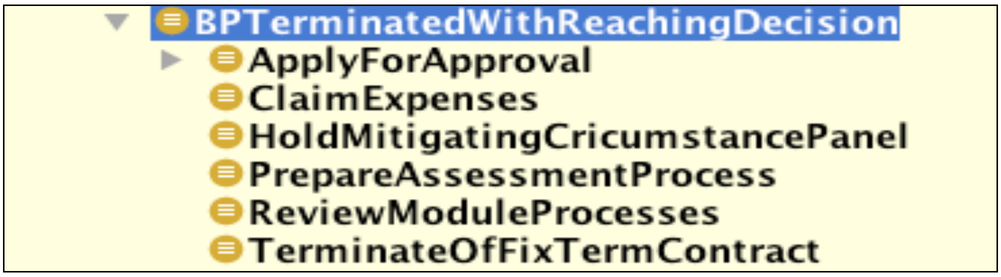
Class Description: BPTerminatedWithReachingDecision

Equivalent classes

BusinessProcesses
and hasTemporalPart some ApplicationDecisionApproved

Figure 5-18: 'BP Terminated When Reaching Decision' with Its Restriction in Protégé

When using FaCT++ it brings all the processes that have similar Goal event as in Figure 5-19.



- BPTerminatedWithReachingDecision
  - ApplyForApproval
  - ClaimExpenses
  - HoldMitigatingCricumstancePanel
  - PrepareAssessmentProcess
  - ReviewModuleProcesses
  - TerminateOfFixTermContract

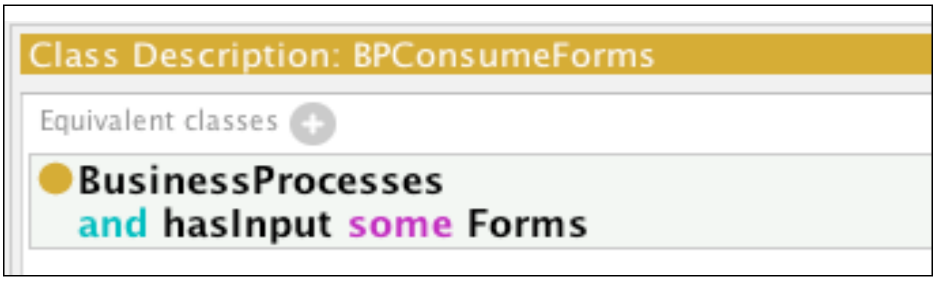
Figure 5-19: General BPO 'BP Terminated When Reaching Decision' with Its Sub-Processes

Other examples are available in Appendix C.

Fourth: Generalisation of Similar Business Processes Inputs:

This type of generalisation depends on the input that a business process has. Thus, it brings all the business processes that have similar input types. Below are the generalised processes achieved from this generalisation rule:

1. **BP Consumes Forms** is a general Business Process Ontology by defining one restriction that all processes should have a similar type of input 'Forms', as presented in Figure 5-20.



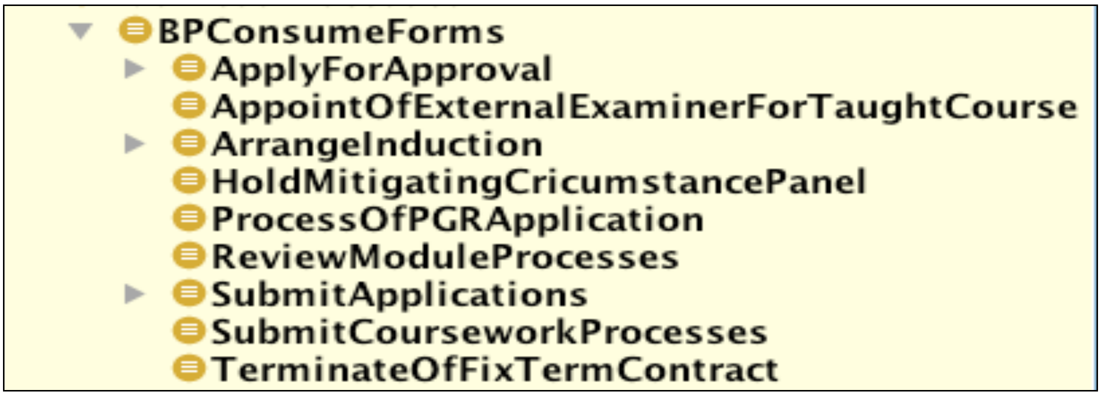
Class Description: BPConsumeForms

Equivalent classes

BusinessProcesses
and hasInput some Forms

Figure 5-20: 'BP Consume Forms' with Its Restriction

When using FaCT++ it classifies all the processes that have similar inputs as presented in Figure 5-21.



BPConsumeForms
- ApplyForApproval
- AppointOfExternalExaminerForTaughtCourse
- ArrangeInduction
- HoldMitigatingCricumstancePanel
- ProcessOfPGRApplication
- ReviewModuleProcesses
- SubmitApplications
- SubmitCourseworkProcesses
- TerminateOfFixTermContract

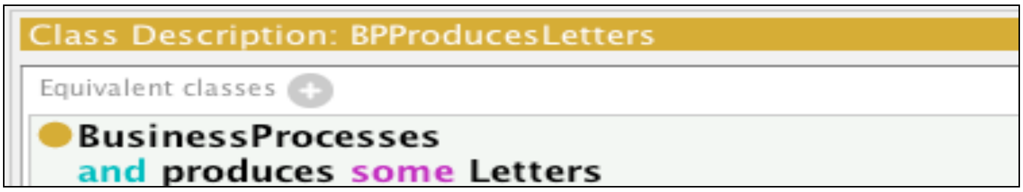
Figure 5-21: General BPO 'BP Consume Forms' with its Sub-Processes

More BP can be generalised based to on input types in the educational domains such as ‘BP Consume Sheets’, ‘BP Consume Letters’ and ‘BP Consume Guides’ which are available in Appendix C.

Fifth: Generalisation of Similar Business Processes Outputs:

This type of generalisation depends on the output that a business process produces. Thus, it classifies all the business processes that have similar output types. Below are the generalised processes achieved from this generalisation rule:

1. Creating a general process called ‘**BP with Similar Output Letters’** by defining a restriction that all processes should have similar types of Output ‘Letters’. See Figure 5-22.



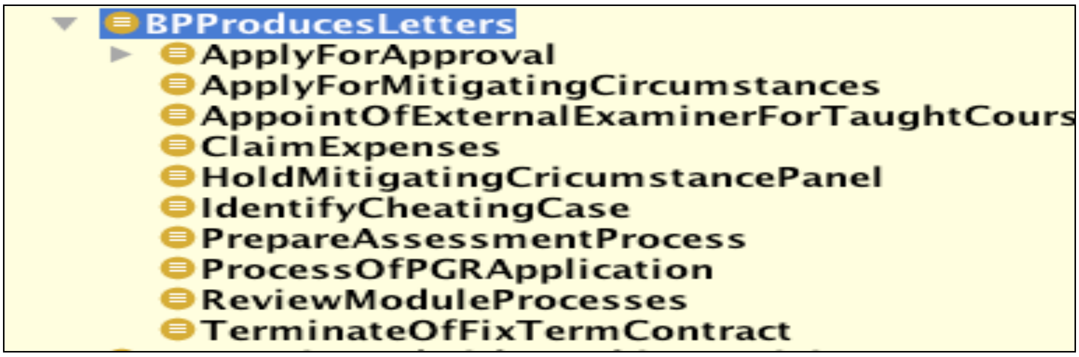
Class Description: BPProducesLetters

Equivalent classes

BusinessProcesses
and produces some Letters

Figure 5-22: 'BP Produces Letter' General BPO With Its Restriction

When running the Reasoner FaCT++ it classifies all the processes that have similar outputs, as presented in Figure 5-23.



BPProducesLetters
ApplyForApproval
ApplyForMitigatingCircumstances
AppointOfExternalExaminerForTaughtCours
ClaimExpenses
HoldMitigatingCricumstancePanel
IdentifyCheatingCase
PrepareAssessmentProcess
ProcessOfPGRApplication
ReviewModuleProcesses
TerminateOfFixTermContract

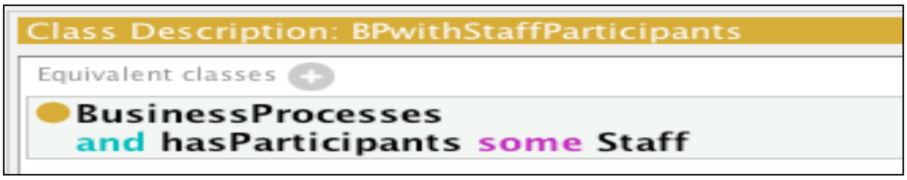
Figure 5-23: General BPO 'BP Produces Letters' with Its Sub-Processes

Further examples for generalising processes according to their similar output types have not been obtained, as no other processes have similar output types.

Sixth: Generalisation of Similar Business Processes Participants

This type of generalisation depends on the participant that takes part in a business process. Thus, it classifies all the business processes that have similar participant types. According to the handbook there were only two groups: staff and students. Below are the generalised processes achieved from this generalisation rule:

1. **BP with Staff Participants** is a general business process obtained by defining a restriction that all processes should have ‘Staff’ as a participant (Figure 5-24).



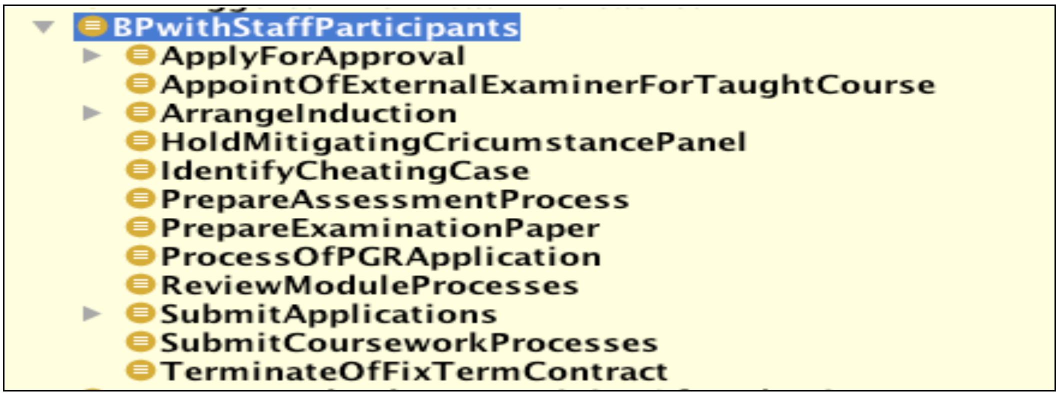
Class Description: BPwithStaffParticipants

Equivalent classes

BusinessProcesses
and hasParticipants some Staff

Figure 5-24: 'BP With Staff Participants' General BPO with its Restriction

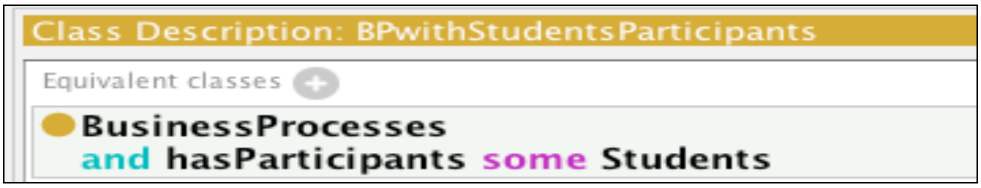
When using FaCT++ it classifies all the processes that have similar output. See Figure 5-25.



- BPwithStaffParticipants
  - ApplyForApproval
  - AppointOfExternalExaminerForTaughtCourse
  - ArrangeInduction
  - HoldMitigatingCricumstancePanel
  - IdentifyCheatingCase
  - PrepareAssessmentProcess
  - PrepareExaminationPaper
  - ProcessOfPGRApplication
  - ReviewModuleProcesses
  - SubmitApplications
  - SubmitCourseworkProcesses
  - TerminateOfFixTermContract

Figure 5-25: General BPO 'BP with Staff Participant' with its Sub-Processes

2. Similarly another example is **BP with Students Participants** as presented in Figure 5-26 and 5-27.

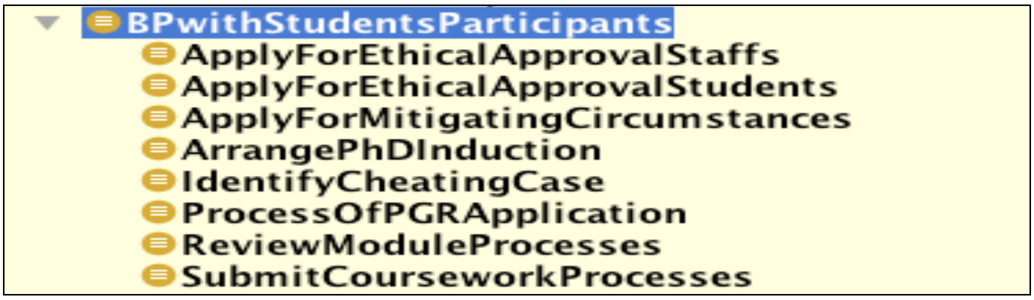


Class Description: BPwithStudentsParticipants

Equivalent classes

BusinessProcesses
and hasParticipants some Students

Figure 5-26: 'BP With Students Participants' with its Restriction



- BPwithStudentsParticipants
  - ApplyForEthicalApprovalStaffs
  - ApplyForEthicalApprovalStudents
  - ApplyForMitigatingCircumstances
  - ArrangePhDInduction
  - IdentifyCheatingCase
  - ProcessOfPGRApplication
  - ReviewModuleProcesses
  - SubmitCourseworkProcesses

Figure 5-27: 'BP with Students Participant' and Sub-Processes

Phase 4: Pattern Documentation (PD):

The researcher believes that this phase is pertinent within the SDL. Thus, this is the last phase of SDL although the design process is iterative. At this stage, documentation is the main design activity included within this phase. Business process patterns documentation is important since poor documentation of pattern is one of the main barriers to effective knowledge sharing and dissemination. All the various types of patterns (general models) have been documented using Ericksson and Penker's (2000) template, an example of the 'Arrange Induction' patterns is provided in Table 5-8.

Documentation plays a key role in facilitating pattern maintenance, use, and reuse, because it specifically describes the purpose of the pattern. For example, the 'Arrange Induction' pattern provides the basic structure for making advance arrangements for welcoming and inducting new candidates into an organisation. Thus, it tells what the modeller should be expecting from the reuse of this pattern. Documentation clarity and simplicity are also considered major issues affecting business process pattern usefulness and value. To offer clarity and simplicity, the research adopted Ericksson and Penker's pattern documentation template as each part of this template supports the definition of the pattern, i.e. intent (the purpose of this pattern), applicability (where and how it can be used), structure (how this pattern looks like), etc. Hence, the way these patterns are documented is designed to address the needs of various audiences. More discovered patterns as the one provided earlier from Brunel University handbooks have been documented in Appendix D.

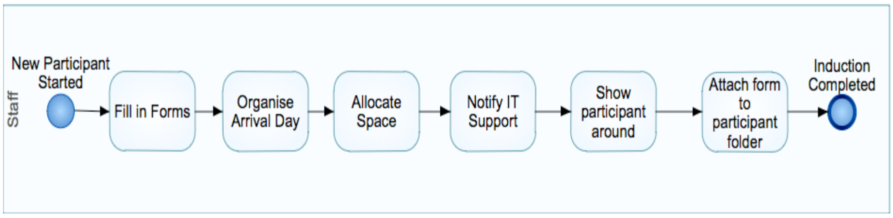
Business Process Pattern	
Name	Arrange Induction
Intent	Arrange Induction falls under the Generalise Similar Type Business Processes pattern category. It provides the basic structure for making advance arrangement for the welcome and induction of new person in an organisation.
Motivation	This pattern is a business process pattern that suggests a standard format for new participant Induction in an organisation. It should be used as a guide only. The induction arranged should be suitable for the individual's particular role and is dependant on the nature of the member's role e.g. such as arrange new academic staff induction, arrange PhD student induction.
Applicability	These guidelines are designed to assist the manager with the induction of a new member of staff. The aim of induction is to help the newcomer to adjust as quickly as possible to the new working environment, in order to achieve maximum working efficiency in the shortest possible time. It is important to remember that induction is a process that should take place over a number of weeks; it is not a one-day event. The checklists that follow are designed to suggest general issues/ topics that may need to be included in the induction programme in order to assist the new employee settle in as easily and effectively as possible. So this pattern starts when a newcomer starting and end up when full induction provided.
Structure	 Staff New Participant Started Fill in Forms Organise Arrival Day Allocate Space Notify IT Support Show participant around Attach form to participant folder Induction Completed
Participants	Input: Forms Output: Reports Participant: Staff BP Initiating Event: New Participant Started BP Goal Event: Induction Completed Activities: Fill in Forms, Organise Arrival Day, Notify IT Support, Show Participant Around and Attach forms to participant folder.
Consequences	This pattern provides a powerful model to guide any organisation on what is required when a new participant or candidate starting their first day in the organisation. By identifying possible actions, and necessary prerequisites, problems can be eliminated and the goals of successful induction of the new candidate can be achieved.

Table 5-8: 'Arrange Induction' Pattern Documentation

5.6 Research Output Artefacts

This section describes research outputs for this iteration.

1. **Second version of the SDR Methodological Framework**. This is one of the main outputs. The SDR incorporates two lifecycles, for iteration two the former lifecycle SDL is preceded with four phases. Also, a set of encapsulated tasks within a single SDL is performed and a scenario is used to develop the process lifecycle. The following is the output of SDL.

2. **Business process pattern(s).** This is the second main output. The process of producing business process pattern(s) from Brunel University staff and students handbooks involves using several phases and ends with documenting them in the repository.

The other research outputs for iteration two are instrumental outputs of SDL:

- **Legacy Segment(s)**. It is a hierarchy of work unit(s) that facilitates the analysis of Brunel University staff and student handbooks assets and deemed necessary when producing the BP Models.
- **BP Model(s)**. Business process models are constructed through the ground analysis of two handbooks for Brunel University staff and student documents using reverse engineering and modelling techniques to transform the handbook asset segments to BP Models.
- **Business Process Ontology model(s)**. The ontological definition of business process forms the input for the semantic discovery rules that directly transform the model(s) to ontology model(s). Multiple mini-iterations have compared the process models to the ontology models for Brunel University staff and student handbooks.

5.7 Evaluation of Research Artefacts and Evolutions

Iteration two artefacts are evaluated using the same criteria as described in iteration one, despite this section evaluates the whole SDL lifecycle of the SDR methodology. The second version of SDL provides a series of phases and steps to be performed in order to develop ontological patterns of business processes. In this way, the SDL also provides a mechanism for improving the organisational assets of the knowledge sources. Given that the SDL provides a novel way for organisational knowledge sources transformation, two aspects primarily make up the evaluation criteria here (March & Smith 1995) (methodological study and operational study), as explained below.

5.7.1 Methodological Study

During iteration one, the focus was heavily based on understanding the feasibility of the SDL lifecycle and as a result the SDL essentially reflected a mere skeleton, primarily made up of the necessary phases and steps relevant from a simple applicability perspective for the Semantic Discovery Lifecycle. Iteration two provided the necessary reflection on the actual application of the whole discovery lifecycle. This can be noticed because project management activities, necessary for predicting things like time-scales and cost-benefits have been catered for when implementing any changes and each change was analysed on that basis. As a result, going forward with this iteration, it was proposed that SDL runs well and accomplished its task with the production of business process patterns (Table 5-8).

5.7.2 Operational Study

The other evaluation aspect looks at the application of the SDL lifecycle within the context of study to reflect on the ability of its lifecycle to perform the intended task of the production of business process patterns. The SDL reflects a more complete process with clearly identified phases, steps, activities, artefacts and roles than it did in iteration one. The benefits are however directly proportional to the practical application of the SDL lifecycle (i.e., further improvements need training and continuous development) and thus involves substantial amount of time and effort.

SDL Lifecycle Operational Evaluation

Completeness of the lifecycle is evaluated by measuring the percentage coverage of SDL. In the previous iteration, the research could not prove it since the lifecycle was only applied partially to the financial services domain. In iteration two the completeness of this iteration has been proved as the whole Semantic Discovery Lifecycle is run. The SDL was applied to new data sources from the higher education domain of Brunel University to demonstrate improved completeness of SDL, by increasing coverage to new types of organisational knowledge sources like handbooks for Brunel University staff and students. The possible coverage of SDL shows improvement over iteration one by being capable of covering a new domain. This however is only a measure of maximum coverage of SDL to different domain and delivers the same aim of producing patterns of business processes. Also, the research can arguably defend the completeness of the second version of SDL in the sense that the lifecycle has preformed without any obstacles as well it delivers its duty of producing patterns of business processes.

Validation and documentation, one important consideration in SDL lifecycle is related to activities for validation and documentation of the developed patterns. Business process pattern documentation was organised in a structured document and the details of the solution were made available in a human-readable format. Business process patterns documentation is highly structured. The patterns are documented by using Ericksson and Penker's (2000) template that identifies the information needed to understand the organisational problem and the solution in terms of the relationships between the processes and activities necessary to implement the solution. There is no uniform agreement within the process pattern community on how to describe a pattern template. Different authors prefer to use different styles for their pattern templates. Some prefer to be more expressive and less structured, while other authors prefer their pattern templates to be more precise and high grain in structure. For this research using Ericksson and Penker's (2000) documentation template provides a clear definition to support the reuse of the discovered patterns. However, constructing an additional business process pattern hierarchy would ease human readability and structural classification; this represents feedback for iteration three.

The SDL demonstrates an improvement in *efficiency* over iteration one. The performance of the lifecycle is measured from *Preparation Of Legacy Assets* using Brunel University staff and students handbooks through to *Pattern Documentation*. A breakdown of the time taken to construct patterns of business processes is given in Table 5-9. Accuracy of the recorded times is improved over the last iteration since the researcher had a clearer perception of the stages of construction. Timings were recorded for all the phases and the steps of SDL and precision of the measurements is given to the nearest day (see Table 5-9). There was an improvement in the precision of days in iteration two as better project management has been calculated with each phase needs. The improved performance is attributed to the researcher having a clearer understanding of the SDL. Performance improvement can also be attributed to increased efficiency in the researcher at managing workload and the application of the SDL.

Activity	
Iteration Two	45 days
Number of Business process models	16 Models
Number of Business Process Patterns	14 Patterns
Estimated man day per pattern	~ 1 per day

Table 5-9: Performance of Business Process Patterns

The process is still semi-automated however. Models constructed by using BPMN modelling tools, patterns were constructed using Protégé – OWL and FaCT++. Thus, SDL had benefitted from automation through tools that aid the capture and expression of patterns. Finally, SDL would, however, benefit from application to systems from a separate organisation or domain, so for iteration three it is necessary to apply SDL in its entirety to another domain in order to improve generalisation. It would benefit from going back to applying the newly developed SDL on the financial services legacy assets.

BP Patterns Operational Evaluation:

Since business process patterns are semantically discovered, using ontology evaluation criteria can help to evaluate the quality of the discovered patterns, as presented in Table 5-10. The evaluation criteria adopted are from Gruber (1995) and Staab and Studeer (2010).

Artefacts	Criteria	Description
BP Patterns	Clarity and Conciseness	Means to state exact and unambiguous definitions for all discovered patterns in order to effectively communicate the intended discovery.
	Customisability	Checks discovered and designed patterns abilities to expand the existing shared elements without altering the existing ones. This supports the reusability and pattern extendibility.
	Coherence	Also known as soundness or consistency, stipulates that the ontological definitions of the patterns (i) are individually sound and (ii) do not contradict each other.

Table 5-10: Evaluation Criteria

Clarity and Conciseness, to use this criterion a few questions need to be asked (Staab and Studeer, 2010) Do the discovered patterns communicate effectively the intended meaning of the defined purpose? Have the definitions of the patterns met their aim and are they independent of context? Are the definitions of the patterns documented? Are the patterns understandable? In this iteration each discovered business process pattern from the educational domain was discovered and developed for a reason such as 'Arrange Induction pattern'. This pattern suggests a standard format for new participants being inducted into an organisation. This pattern communicates its defined purpose through the steps needed to take place for the induction arrangement, and what is suitable for the individual's particular role and is dependant on the nature of the member's role, e.g. such as arrange new academic staff induction, arrange PhD student induction. By using Ericksson and Penker's (2000) documentation template enables easy documentation and clarity of purpose of the discovered educational patterns offering a way of tracking the different types of patterns. In addition, the template enables the creation of a list of elements used within each pattern; this is an important point for pattern conciseness (Noy and McGuinness, 2001) by preventing unnecessary definitions and explicit or implicit redundancies that may lead to the production of irrelevant patterns. Thus, in iteration three human understanding should be considered in order to judge the discovered patterns and to decide whether or not a pattern is of a value to the domain.

Customisability is minimal ontological commitment and extendibility (Gruber, 1995). In other words, for sake of reusability, the pattern elements should be kept to a minimum in order to allow for different extensions and thus fit a large number of contexts. The principle of minimal ontological commitment states that "an ontology should make as few claims as possible about the world being modelled, allowing the parties committed to the ontology freedom to specialise and instantiate the ontology as needed" (Gruber, 1995, p.909). As those patterns have been developed ontologically using the business process ontology. Patterns are discovered and designed in a way that gives its different users the ability to expand the existing shared elements without altering the existing ones. These discovered business process patterns are a general solution, which can be specialised to meet domain processes specific needs.

5.8 Discussion

Understanding SDL is necessary in order to describe and clarify the structure of the underlying phases, steps and techniques used to semantically discover business process patterns and the consumable resources required for the process. Upon reflection, it can be stated that both the learning and the knowledge gained from doing iteration one, helped to progress and evolve the second version of the Semantic Discovery Lifecycle phases and steps in iteration two. Also, the project management issue mentioned previously in iteration one has been addressed, as most of the project management issues that influence a performance of the lifecycle stated previously arise from uncertainty in the definition of its phases. Thus, iteration two benefited from the previous feedback and evolved the second version of SDR specifically the Semantic Discovery Lifecycle by reducing the number of phases and bringing related phases under one phase. The reason for undertaking such a step was to improve the flow of the lifecycle through re-distributing the step's intellectual work among the phases and adding more specialisation to each phase. Thus, the second version of SDL has phases to *Extract and Model processes* (Phase 1), *Interpret and Formalise processes* (Phase 2), *Generalise processes* (Phase 3), and *Document the Patterns* (Phase 4). Thus, iteration three will benefit from both the learning and the knowledge gained from doing the previous iterations and to apply the lifecycle to multiple domains and across multiple systems respectively.

The iteration is also motivated by one of the main objectives of this thesis, which emphasises on providing ontological foundations for business process models. As providing the ontological definition of business process addresses the problem of inconsistency and provide patterns that are systematically connected to real world problems to offer reusability.

Interestingly, little consensus has been reached as to what the essential ingredients of business process patterns should be. Nonetheless, the researcher believes that most of the discovered and developed patterns here can be of a benefit to an organisation even if their benefit may vary, such as the pattern that classifies processes using the same input, i.e. "BP consumes Forms". This pattern provides a practical way to approach the issues on which type of document should be used with various types of business processes.

In addition, iteration two benefited from transforming some of the SDL phases to be automated using BPMN elements to extract and conceptualise domain business processes, Protégé-OWL for *Interpreting* business process elements to BPO elements, and an ontology reasoner for generalising BPO models to produce patterns, unlike the first version of SDL process which was highly manual.

Therefore, these issues provide feedback to the design of the next iteration in order to increase the emerge benefit of SDL.

5.9 Feedback

Feedback from the previous discussion suggests that iteration three:

1. Should test SDL against multiple systems across multiple domains, such as educational and financial domains (retail banking, insurance and mortgages). As in the second iteration the second version of SDL was able to produce patterns of business processes from the educational domain specifically Brunel University staff and students handbooks, to further demonstrate the generality of SDL, the lifecycle will be applied again to the financial services domain on its sub domains (retail banking, insurance and mortgages) to produce business process patterns and to check the ability of SDL to produce patterns of business processes across multiple domains and across multiple systems as will be demonstrated in iteration three. Therefore the first and second iterations feedback is to apply the process to multiple domains and across multiple systems respectively.
2. Create and define a maturity model of business process patterns based on the type of different discovered patterns from the two domains.
3. In *Pattern Documentation* phase, add a step of build a hierarchy for the discovered business process patterns, as patterns are detailed documentations of best practices. They were intended to be a reference point for practitioners wishing to understand a problem or its solution in greater detail. Thus beside

documentation, organising business process patterns in a structural way that is human-readable.

5.10 Summary

Iteration two draws requirements from the feedback of iteration one. The iteration delivers improvement to the artefact(s) because the individual iteration evolves the SDR methodology incrementally.

This iteration also, produces well accepted and defined business process patterns as those discovered patterns offer to solve a particular problem (not just in principle), as well have a significant human component describing how and when it is useful. An ontological definition of BP was developed and applied to alter and extend rules governing transformation of BP models into BPO models.

It can broadly state that using the second version of SDL has been evolved from iteration one, and the researcher was able to run the whole lifecycle and produce the desired artefacts. However, iteration three should be based on the feedback from the current iteration, so its steep learning process can be minimised and effectively managed to address iteration three.

A summary of conclusion points from this iteration follows:

1. The Semantic Discovery Lifecycle is at least proved to be more efficient than iteration one for the sequence of the phases and steps to produce business process patterns.
2. Definition of Business Process Ontology using the object paradigm approach helped to lower the risk of discovering irrelevant patterns. Likewise a powerful guide to achieve an optimal level of formalisation in terms of process modelling.
3. Identifying the generalised BPO model using the FaCT++ reasoner improved the consistency of the discovered models.
4. *Business process pattern documentation* was organised using Ericksson and Penker's (2000) template that identifies the information needed to understand the

organisation problem and the solution in terms of the relationships between the processes and activities necessary to implement the solution.

CHAPTER 6 – ITERATION THREE: SDL ACROSS DOMAINS

6.1 Introduction

This chapter presents the research carried out in order to further investigate the use of the Semantic Discovery Lifecycle in different domains. As well it describes a new empirical experiment, which emphasises on the applicability of the final version of the Semantic Discovery lifecycle to produce business process patterns across various domains and among multiple systems. Pattern-based business process modelling discovered from multiple systems and domains improves the productivity of modellers as well as help achieve improved levels of traceability between business requirements and software systems.

This chapter is organised as follows: Section 6.2 presents the purpose of starting this iteration. Section 6.3 presents the evolvement of the final version of the Semantic Discovery Lifecycle. Section 6.4 lays down the necessary groundwork for the practical application of the final version of the Semantic Discovery Lifecycle. Section 6.5 presents further discovery for business process patterns across multiple domains. Section 6.6 summarises the research outputs for iteration two. Section 6.7 evaluates the research outputs. Section 6.8 discusses the implications of the outputs. Research feedback and further work are explained in Section 6.9. Design Research evaluation for the three iterations is discussed in Section 6.10. Section 6.11 presents the *Final Evaluation* of the whole research. Finally, the chapter is summarised in Section 6.12.

6.2 Purpose of Iteration Three

Previous iterations designed and constructed the Semantic Discovery Lifecycle and produced business process patterns from the academic domain of Brunel University. The motivation of doing iteration three is to evolve the third/final version of SDR methodological framework by applying the Semantic Discovery Lifecycle to different domains and to verify SDL's capability to produce business process patterns across multiple domains (financial and educational). Thus this iteration unfolds with the following points in mind:

- Continue discovering business process patterns from empirical data (e.g., legacy systems). This iteration is going back to discover business process patterns from the financial domains mentioned in iteration one (Chapter 4).
- Continually test the existing business process patterns from the educational domain against legacy models and data of the financial domains.
- Construct a hierarchy of the discovered business process patterns. The primary motivation behind this rationale is to formally describe the different BPO elements that the discovered patterns were generalised or extracted from so that unwanted ambiguities related to the application and use of the pattern can be avoided. Defining this taxonomy of business process patterns is based on the type of testing that the patterns have undergone, as follows:
    1. Tested against one type of data source, such as Brunel University staff and students handbook.
    2. Tested against multiple systems of one domain, such as the financial services systems for retail banking, insurance and mortgages.
    3. Tested against multiple systems across multiple domains, such as educational (staff and students) and financial domains (retail banking, insurance and mortgages). This requires keeping in mind the different types of patterns discovered from the educational domain of Brunel University when discovering patterns from the financial domain to further help in generalising patterns across domains.

6.3 Refinement of SDR Methodological Framework

Outcomes from Chapter 4 and Chapter 5 are centred on the evolvement of SDL and the development of business process patterns from one domain either the financial or educational domain. Discovering patterns from one domain has the drawback of not being able to identify more generalised patterns. Therefore, Chapter 6 has moved the research forward somewhat, focusing on the final version of SDL and its ability to discover patterns across multiple domains and systems. In this chapter the aim is to apply the framework to both domains (educational and financial). Motivated by finding more generalised business process patterns and to address the limitations in the literature on patterns related to specific domains as stated in Section 2.4.2. See Figure 6-1 the final version of the Semantic Discovery Lifecycle gains as knowledge increments are made from the previous two iterations.

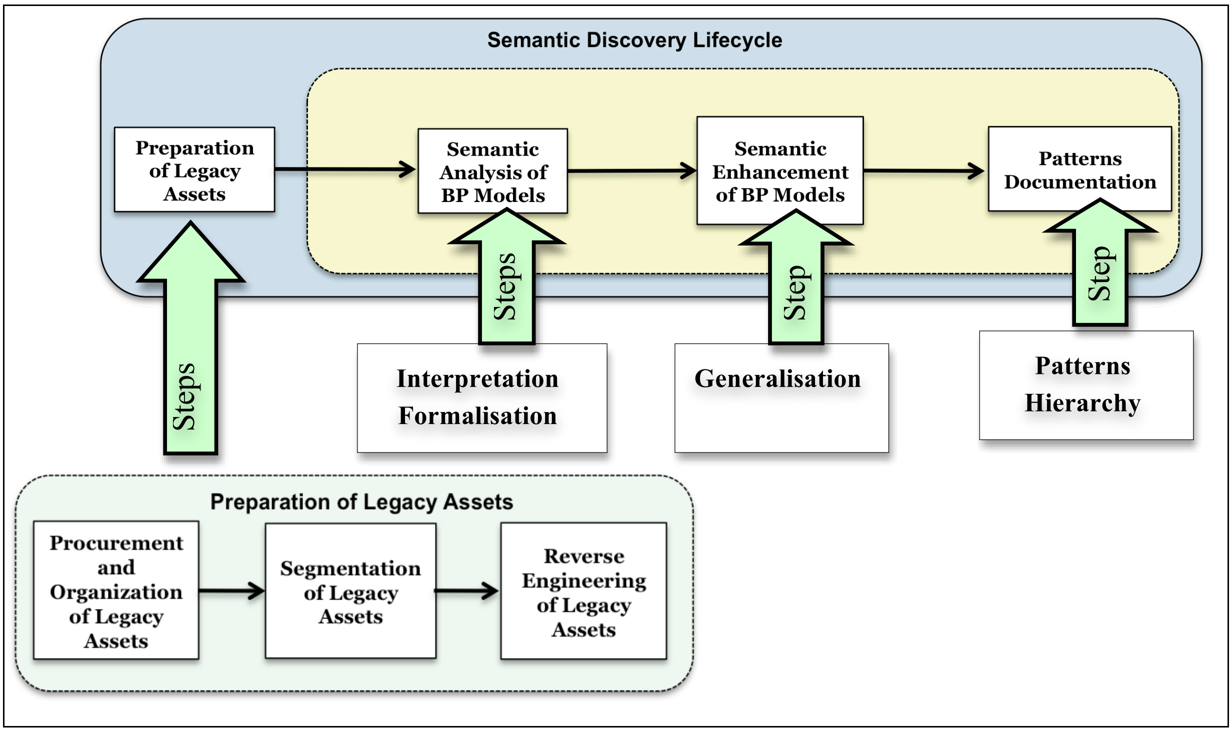

Semantic Discovery Lifecycle
Preparation of Legacy Assets
Semantic Analysis of BP Models
Semantic Enhancement of BP Models
Patterns Documentation
Steps
Steps
Step
Step
Interpretation Formalisation
Generalisation
Patterns Hierarchy
Preparation of Legacy Assets
Procurement and Organization of Legacy Assets
Segmentation of Legacy Assets
Reverse Engineering of Legacy Assets

Figure 6-1: Final Version of The Semantic Discovery Lifecycle

The final refinement has been applied for the final version of SDL by only adding a step to the *Patterns Documentation* (the Fourth Phase of the Lifecycle) as it can be noticed in Figure 6-1. Thus, all SDL phases and steps kept its functionality defined and used in iteration two, and only the evolvement in the *Pattern Documentation* phase is explained, as follows:

<u>Phase 4: Pattern Documentation with Patterns Hierarchy Step</u>

In this research business process patterns used a template proposed by Ericksson and Penker (2000) to represent the different (e.g., intent, motivation, etc.) aspects of a process pattern. Additional, thinking added to structure a hierarchy of the discovered patterns. The primary motivation behind this rationale is to describe the different BPO elements that the discovered patterns generalised or extracted from so that unwanted ambiguities related to the application and use of the pattern can be avoided.

6.4 The Practical Application of SDL

The focus now moves towards the practical application of the Semantic Discovery Lifecycle in order to continue to (a) identify ontological model of business processes from the financial services domains, and (b) develop business process patterns by using the financial services process models developed earlier in iteration one. However, this iteration's actual work starts with the second phase (*Semantic Analysis* phase) of the Semantic Discovery Lifecycle as follows.

Phase 1: Preparation of Legacy Assets (PLA)

This phase is already accomplished in iteration one and ended by producing business process models of the financial services domains using BPMN. An example representation is demonstrated for the 'Product' segment. 'Define Retail Banking Product Templates' is a business process for defining a product template in the Retail Banking of the financial services domain. Using the reverse engineering technique to extract all the elements related to 'Define Retail Banking Product Templates' business process. Table 6-1 shows all the extracted elements related to 'Define Retail Banking Product Templates' business process of 'Product' segment within the retail banking of the financial services domain.

'Define Retail Banking Product Templates' terms have been mapped to BPMN elements unlike iteration one where discovered business processes have been defined according to Ericksson and Penker's (2000) business process definition. BPMN provides a notation that is more readily understandable by all business users, starting from the business analysts who create the initial drafts, to the technical developers who are responsible for implementing those processes, and finally, to the business people who will manage and monitor the processes (White, 2004). BPMN is targeted at users, vendors and service providers that need to communicate business processes in a standard manner. Also, later in this step conceptualisation of those discovered processes takes place by using a BPMN modelling tool. Table 6-1 summarises business process elements used in constructing 'Define Retail Banking Product Templates', other business processes can be found in Appendix A.

Define Retail Banking Product Template	
Terms	**BPMN Elements**
New Retail Banking Product Definition	Event
Select Product Category	Task
Create Product Template	Task
Set up Fixed Rules	Task
Record Product Amendable Details	Task
Set up Product Interest Condition	Task
Complete Product Interest Type Form	Task
Set up Product Status	Task
Produce Product Report	Task
Retail Banking Admin	Lane
Product Interest Type Form	Data
Product Interest Condition Form	Data
Product Template Form	Data
Fix Rule Guide	Data

Table 6-1: 'Define Retail Banking Product Template' Terms mapped to BPMN Elements

After the reverse engineering step, the conceptualisation of 'Define Retail Banking Product Templates' is taking place by creating BPM using BPMN form Table 6-1, as in Figure 6-2.

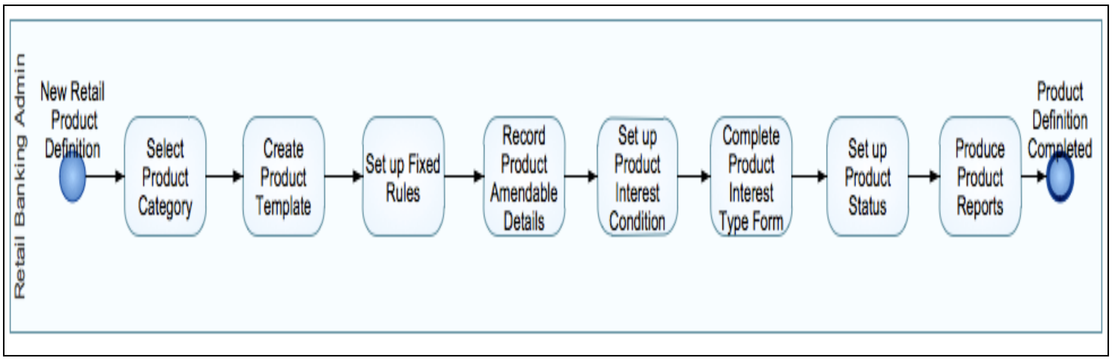

Retail Banking Admin
New Retail Product Definition
Select Product Category
Create Product Template
Set up Fixed Rules
Record Product Amendable Details
Set up Product Interest Condition
Complete Product Interest Type Form
Set up Product Status
Produce Product Reports
Product Definition Completed

Figure 6-2: BPMN Model of 'Define Retail Banking Product Template'

Phase 2: Semantic Analysis of BP Models (SA)

Moving to the second phase of SDL, the *Semantic Analysis of BP Models* phase, the elements of the 'Define Retail Banking Product Templates' process previously developed in phase one (*Preparation of Legacy Assets*) are semantically interpreted using Business Process Ontology definition, in order to derive more precise ontological models of the processes themselves and far more sophisticated and semantically richer than its predecessors through ***Interpretation*** by following the rules of transformation explained in Section 5.3. The interpretation starts by analysing the first row in Table 6-2, which signifies that the sign 'Select Product Category' BPMN task commits to the existence of an object called 'Assign Category For The New Retail Products', which commits to an *Activity* in BPO elements according to Business Process Ontology definition explained in Section 5.3. Also *BP Initiating Event* has been recognised that 'New Retail Banking Product Introductory' trigger the 'Define Retail Banking Product Templates' business process as illustrated in Table 6-2.

In Table 6-2 to define 'Define Retail Banking Product Templates' for Product, required a few activities starting from the *first activity* of 'Assign Category for The New Retail Product' to the *final activity* of 'Produce Retail Product Reports'. The other individual terms follow a similar pattern, as it is also a type of activity that is offered by Retail Banking services. Also, additional information is defined for this business process such as *Initiating Event*, *Dissolution Event*, *Input* and *Output*. Also, to have a clearer model other activities have been added to improve the flow of the business process. Using BPO definition provide more interpretation and model that are related to real world processes, it helps to justify they type of event that trigger and terminate 'Define Retail Banking Product Templates' BP such as 'New Retail Banking Product Introductory' is the initiating event that trigger this BP and 'New Retail Product Launched To Market' is when the process terminated. This however adds more clear definition to the BP unlike what used to be represented in BPMN. Also, some BPMN tasks have been further interpreted. Other ontological business processes can be found in Appendix B.

DefineRetailBankingProductTemplates			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Retail Banking Account Opened	Event	NewRetailBankingProductIntroductory	BP Initiating Event
Select Product Category	Task	AssignCategoryForTheNewRetailProducts	Activities
Create Product Template	Task	FillinRetailProductTemplateForms	Activities
Set up Fixed Rules	Task	SelectFixedRulesForTheRetailProductFromRulesGuides	Activities
Record Product Amendable Details	Task	DefineWhichRetailProductDetailCanBeAmended	Activities
Set up Product Interest Condition	Task	AssignInterestConditionsFormsToRetailProducts	Activities
Complete Product Interest Type Form	Task	FillinRetailProductInterestTypeForms	Activities
Set up Product Status	Task	ChooseLifeStatusToLaunchRetailProducts	Activities
Produce Product Report	Task	ProduceRetailProductReports	Activities
		NewRetailBankingProductLaunchedToMarket	BP Goal Event
Retail Banking Admin	Lane	RetailBankingAdmins	BP Participants
Product Interest Type Form	Data	RetailProductInterestTypeForms	Input
Product Interest Condition Form	Data	RetailProductInterestConditionForms	Input
Product Template Form	Data	RetailProductTemplateForms	Input
Fix Rule Guide	Data	RetailProductFixRuleGuides	Input
Product Report	Data	RetailProductReports	Output

Table 6-2: 'Define Retail Banking Product Template'

After creating Table 6-2 and to show an improved model of BPMN, the conceptualisation of ‘Define Retail Banking Product Templates’ business process can take place by creating a BPMN diagram for Table 6-2. Figure 6-3 illustrates the BPMN model of the interpreted ‘Define Retail Banking Product Templates’ ontology process model.

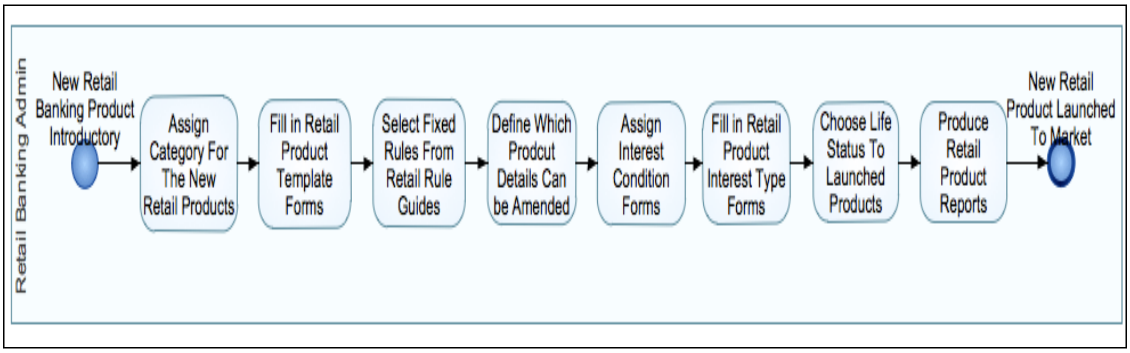
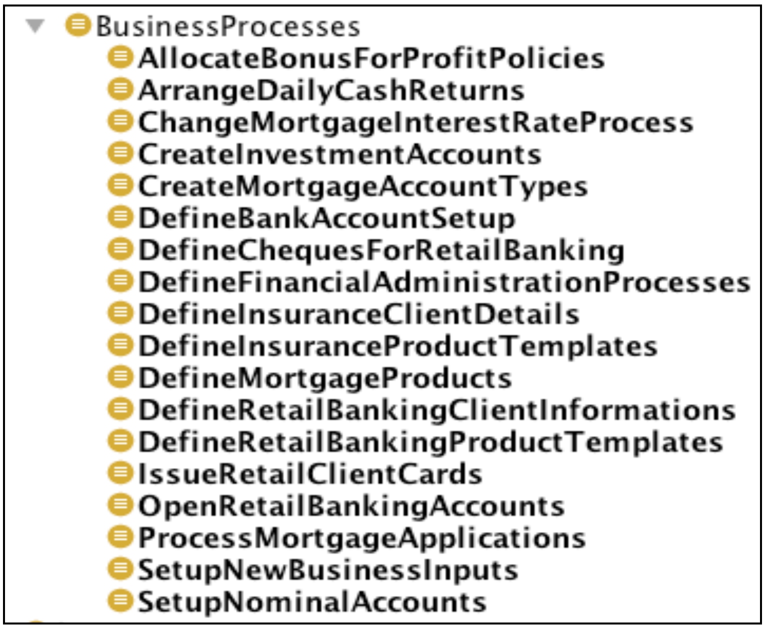

Retail Banking Admin
New Retail Banking Product Introductory
Assign Category For The New Retail Products
Fill in Retail Product Template Forms
Select Fixed Rules From Retail Rule Guides
Define Which Prodcut Details Can be Amended
Assign Interest Condition Forms
Fill in Retail Product Interest Type Forms
Choose Life Status To Launched Products
Produce Retail Product Reports
New Retail Product Launched To Market

Figure 6-3: 'Define Retail Banking Product Template' using BPMN

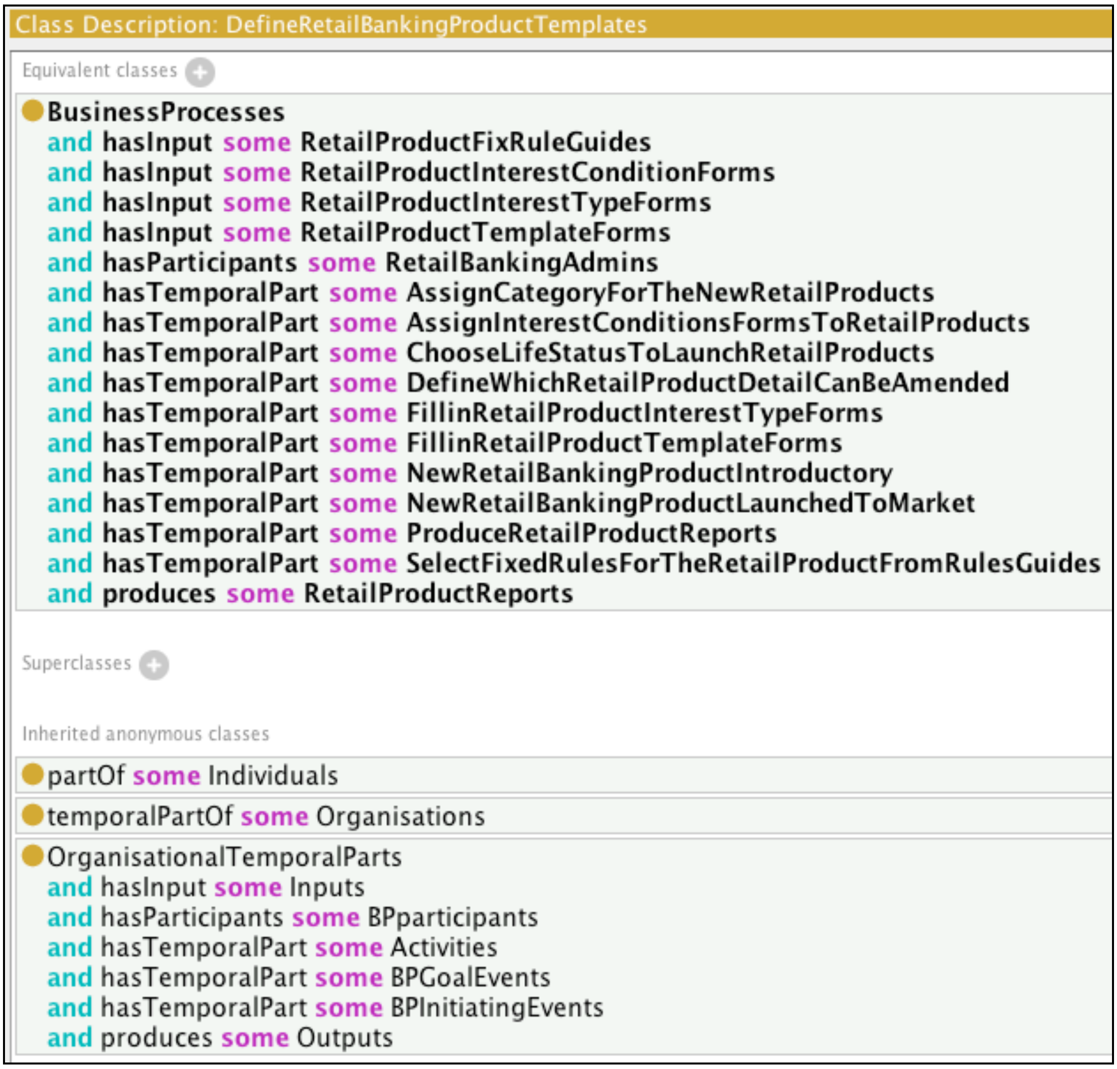
During the *Semantic Analysis phase* the creation of the business process ontology in Protégé has matured to enable the analyst to create a new ontology, linked to the external ontology and carry out consistency checks on the working or completed ontology. All the business processes interpreted from the domains of retail banking, insurance and mortgages have been uploaded in Protégé (Appendix D) using the Business Process Ontology constructed in OWL earlier in Section 5.3. See Figure 6-4 for all the financial services business processes.



- BusinessProcesses
  - AllocateBonusForProfitPolicies
  - ArrangeDailyCashReturns
  - ChangeMortgageInterestRateProcess
  - CreateInvestmentAccounts
  - CreateMortgageAccountTypes
  - DefineBankAccountSetup
  - DefineChequesForRetailBanking
  - DefineFinancialAdministrationProcesses
  - DefineInsuranceClientDetails
  - DefineInsuranceProductTemplates
  - DefineMortgageProducts
  - DefineRetailBankingClientInformations
  - DefineRetailBankingProductTemplates
  - IssueRetailClientCards
  - OpenRetailBankingAccounts
  - ProcessMortgageApplications
  - SetupNewBusinessInputs
  - SetupNominalAccounts

Figure 6-4: Financial Services Business Process Ontology

‘Define Retail Banking Product Templates’ business process ontology for retail banking is a subclass of business processes, where it inherits the definition of business process. This is only one example of many other financial services processes presented in Protégé as presented Figure 6-5, further examples are available in Appendix D.



Class Description: DefineRetailBankingProductTemplates

Equivalent classes

```
BusinessProcesses
 and hasInput some RetailProductFixRuleGuides
 and hasInput some RetailProductInterestConditionForms
 and hasInput some RetailProductInterestTypeForms
 and hasInput some RetailProductTemplateForms
 and hasParticipants some RetailBankingAdmins
 and hasTemporalPart some AssignCategoryForTheNewRetailProducts
 and hasTemporalPart some AssignInterestConditionsFormsToRetailProducts
 and hasTemporalPart some ChooseLifeStatusToLaunchRetailProducts
 and hasTemporalPart some DefineWhichRetailProductDetailCanBeAmended
 and hasTemporalPart some FillinRetailProductInterestTypeForms
 and hasTemporalPart some FillinRetailProductTemplateForms
 and hasTemporalPart some NewRetailBankingProductIntroductory
 and hasTemporalPart some NewRetailBankingProductLaunchedToMarket
 and hasTemporalPart some ProduceRetailProductReports
 and hasTemporalPart some SelectFixedRulesForTheRetailProductFromRulesGuides
 and produces some RetailProductReports
```

Superclasses

Inherited anonymous classes

```
partOf some Individuals
```

```
temporalPartOf some Organisations
```

```
OrganisationalTemporalParts
 and hasInput some Inputs
 and hasParticipants some BPparticipants
 and hasTemporalPart some Activities
 and hasTemporalPart some BPGoalEvents
 and hasTemporalPart some BPInitiatingEvents
 and produces some Outputs
```

Figure 6-5: 'Define Retail Banking Product Template Developed in Protégé OWL

OWL reasoning engines, such as FaCT++, are deployed to check for the consistency for the developed business process ontologies of the financial services domains before moving to the following phase.

Phase 3: Semantic Enhancement of BP Models (SE):

This phase starts by taking the Business Process Ontology model of the financial services domain created in *Semantic Analysis* and aims at generalising them to existing patterns or to newly developed patterns. The generalisation of BPO models has been accomplished in a semi-automated way, using OWL reasoning engines such as FaCT++, to bring similar types of processes under general processes. Many generalisation rules for business processes have been defined using different BPO elements, as presented in Table 6-3, and all presented in Appendix C.

Generalisation Types	Description	Financial Domain Patterns
Generalise Similar Business Processes elements Type	This Type of generalisation depends on all or most of the element of BPO. It brings business processes which have most or all business process elements are of a similar types	• Define Product Type • Create Client Information • Create Product Type
Generalisation of Similar Business Processes Initiating Event	This type of generalisation depends on the Initiating Event of a business process. It brings all the business processes that have similar type initiating event to trigger their business processes	• BP Triggered by Opening Accounts • BP Triggered By New Product Introductory
Generalisation of Similar Business Processes Goal Event	This type of generalisation depends on the Goal Event of a business process. It brings all the business processes that have similar type goal event to dissolve their business processes	• BP Terminated By Product Launching • BP Terminated By Applying Changes
Generalisation of Similar Business Processes Inputs	This type of generalisation depends on the Input that a business process needs to resource its processes. It brings all the business processes that have similar Input type	BP Uses Forms
Generalisation of Similar Business Processes Outputs	This type of generalisation depends on the Output that a business process needs to resource its processes. It brings all the business processes that have similar Output type	• BP Produces Reports • BP Produces Letters • BP Produces Receipts
Generalisation of Similar Business Processes Participants	This type of generalisation depends on the Participant that takes part in a business process. It brings all the business processes that have similar Participant type	• BP Needs Staff Participants • BP Needs Client Participants

Table 6-3: Different Generalisation Types of Business Process Patterns

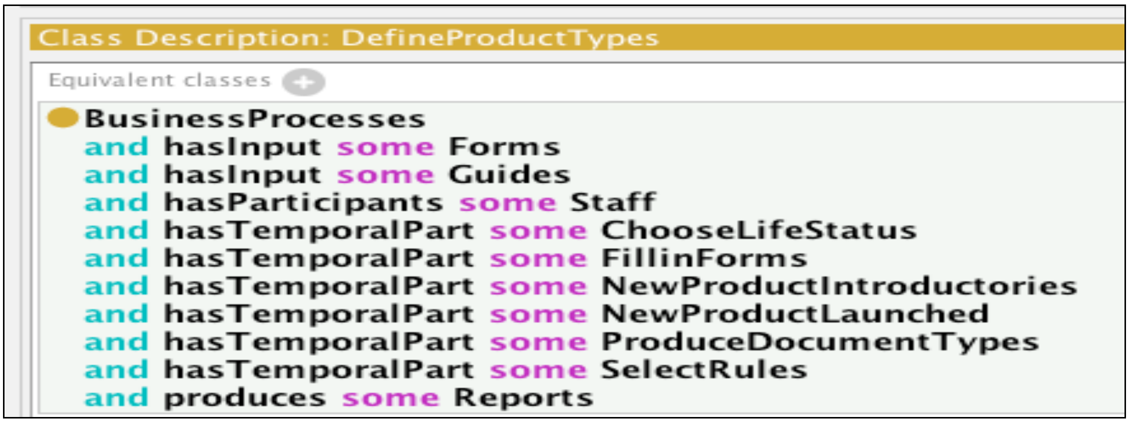
The explanation on each generalisation achieved in Table 6-3 is described as follows:

First: Generalise Similar Business Processes Elements Type

1. Define Product Types has been defined as a general business process ontology model, which has been generalised according to all the elements of the BP ontology (Input, Output, Participant, BP Initiating Event, BP Goal Event and Activities), see Table 6-4 below and the Figure below.

BPO	Define Product Type
BP Participant	Staffs
Input	Forms
Input	Guides
Output	Reports
BP Initiating Event	New Product Introductory
BP Goal Event	New Product Launched To Markets
Activities	Fill in Forms
	Select Rules
	Choose Life Status
	Produce Document Types

Table 6-4: 'Define Product Type'



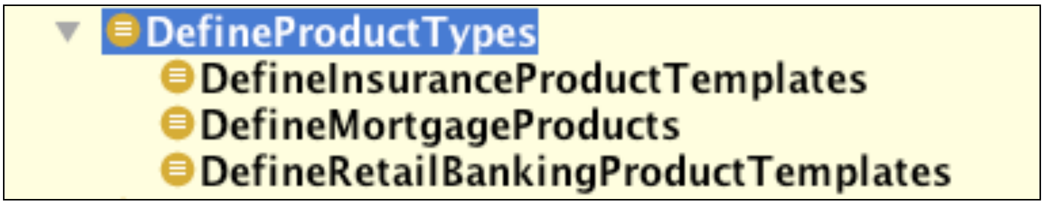
Class Description: DefineProductTypes

Equivalent classes

```
BusinessProcesses
 and hasInput some Forms
 and hasInput some Guides
 and hasParticipants some Staff
 and hasTemporalPart some ChooseLifeStatus
 and hasTemporalPart some FillinForms
 and hasTemporalPart some NewProductIntroductories
 and hasTemporalPart some NewProductLaunched
 and hasTemporalPart some ProduceDocumentTypes
 and hasTemporalPart some SelectRules
 and produces some Reports
```

Figure 6-6: 'Define Product Type' Model in Protégé

From running FaCT++ ‘Define Insurance Product Templates’, ‘Define Mortgage Products’ and ‘Define Retail Banking Product Template’ are classified as subclasses of the general BPO model ‘Define Product Types’. See Figure 6-7 below.



- DefineProductTypes
  - DefineInsuranceProductTemplates
  - DefineMortgageProducts
  - DefineRetailBankingProductTemplates

Figure 6-7: 'Define Product Type' and Subclasses

‘Define Insurance Product Templates’, ‘Define Mortgage Products’ and ‘Define Retail Banking Product Templates’ differ in the type of form that is used as an input, the type of output reports, BP Initiating event that triggers the business process is differ type of introductory; Activities are of a similar type like fill in forms, select rules, etc., and the dissolution events are of a similar types, as illustrated in Table 6-5 below.

BPO Elements	***Define Product Types***	**Define Insurance Product Templates**	**Define Mortgage Products**	**Define Retail Banking Product Templates**
BP Participant	*Staff*	Insurance Admin	Mortgages Admin	Retail Banking Admin
Input	*Forms*	Insurance Product Template Forms	Research Starting Form	Retail Product Template Forms
Input	*Guides*	Insurance Rule Guides	Mortgage Interest Rate Guides	Retail Product Fixed Rule Guides
Output	*Reports*	Insurance Product Reports	Mortgage Product Reports	Retail Product Reports
BP Initiating Event	*New Product Introductory*	New Insurance Product Introductory	New Mortgage Product Introductory	New Retail Banking Product Introductory
BP Goal Event	*New Product Launched To Markets*	New Insurance Product Launched To Market	New Mortgage Product Launched To Markets	New Retail Banking Product Launched To Market
Activities	*Fill in Forms*	Fill in Insurance Product Template Forms	Fill in Mortgage Product Template Forms	Fill in Retail Product Template Forms
	Select Rules	Select Insurance Rules From Rule Guide	Select Mortgage Specific Rules	Select Fixed Rules For The Retail Product From Rules Guides
	Choose Life Status	Choose Life Status To Release Product To Market	Choose Life Date To Release Products	Choose Life Status To Launch Retail Products
	Produce Document Types	Produce Insurance Product Reports	Produce Mortgage Product Reports	Produce Retail Product Reports

Table 6-5: 'Define Product Type' with Similar Processes

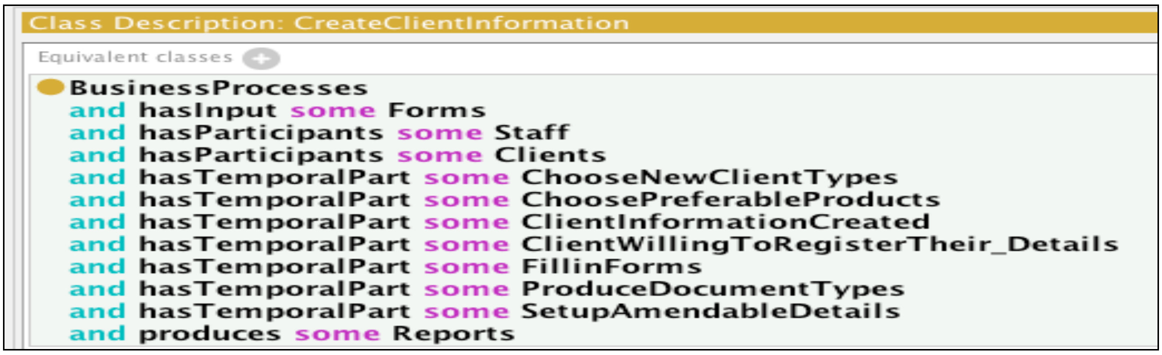
In Table 6-5, only similar elements have been generalised and presented, other elements have been eliminated because they will not be generalised. The non-generalised elements are activities for the ‘Define Product Types’ example as presented in Table 6-6.

BPO Elements	Non Generalised Elements of Define Insurance Product Templates	Non Generalised Elements of Define Mortgage Products	Non Generalised Elements of Define Retail Banking Product Templates
Activities	Choose Category For The New Insurance Product	Assign Mortgages Processing Rules	Assign Retail Product Shell Code
	Decide on The Type of Investment	Assign Interest Rate Using Available Interest Rate	Define Which Client Can be Assigned for the product
	Assign Transaction Code of the Process	Define Amendable Details in Mortgages Product	Define Interest Condition that Match Product Definition

Table 6-6: Non Generalised BPO Elements in 'Define Product Types'

This can be considered a beneficial process generalisation example for two reasons: (1) All BPO elements are generalised apart from a few activities as mentioned earlier in Table 6-6, this offers a distinct and a solid foundation for defining a product type in an organisation, and (2) using ‘Define Product Types’ provides a proven and a clear steps for process modelling that facilitates the definition, improvement and amendment of existing or new product in an organisation. Thus, a new product would be ready to be launched by any organisation. More examples of generalisation follow the same route of the previously discuss ‘Define Product Types’ general business process ontology model. Below are two more examples of generalised Business Process Ontology models.

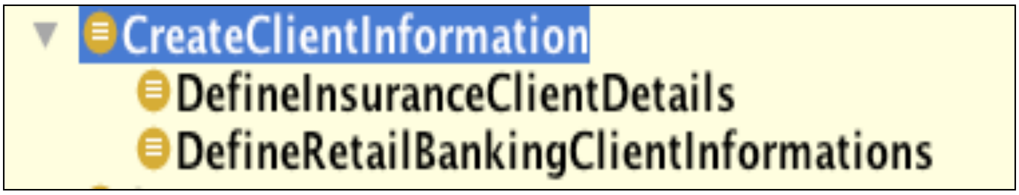
2. Create Client Information has been generalised by having similar types of Input, Output, Participant, BP Initiating Events, BP Goal Event and Activities, as presented in Figure 6-8.



```
BusinessProcesses
 and hasInput some Forms
 and hasParticipants some Staff
 and hasParticipants some Clients
 and hasTemporalPart some ChooseNewClientTypes
 and hasTemporalPart some ChoosePreferableProducts
 and hasTemporalPart some ClientInformationCreated
 and hasTemporalPart some ClientWillingToRegisterTheir_Details
 and hasTemporalPart some FillinForms
 and hasTemporalPart some ProduceDocumentTypes
 and hasTemporalPart some SetupAmendableDetails
 and produces some Reports
```

Figure 6-8: 'Create client Information' General BPO Definition

This generalised BPO model also has some non-generalised elements, which have been eliminated. In Figure 6-9, both ‘Define Insurance Client Details’ and ‘Define Retail Banking Client Information’ have been classified as subclasses of *Create Client Information* (General BP).

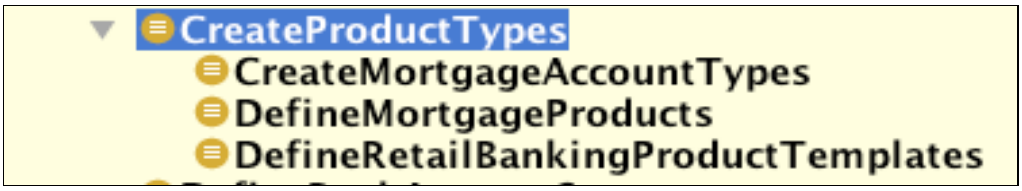


▼ CreateClientInformation
- DefineInsuranceClientDetails
- DefineRetailBankingClientInformations

Figure 6-9: 'Create Client Information' and Subclasses

‘Create Client Information’ is another beneficial process generalisation example, because this generalised model can be used to model organisations that require a client definition within its organisation, by offering a simple way to model client information. Interestingly this general business process interacts with other business processes, typically via the exchange of resources or information between the processes. For example, for opening bank account a client definition is necessary and can be attached to a bank product, i.e. saving premium account.

3. Create Product Types has been generalised by having similar types of Input, Participant and Activities. The business processes ‘Create Mortgage Account Types, ‘Define Mortgage Products’, and ‘Define Retail Banking Product Templates’ have been classified as Subclasses of *Create Product Types* (General BP) Figure 6-10.



▼ CreateProductTypes
- CreateMortgageAccountTypes
- DefineMortgageProducts
- DefineRetailBankingProductTemplates

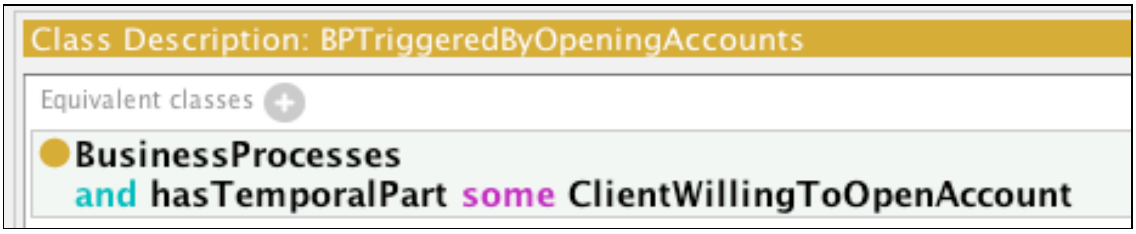
Figure 6-10: 'Create Product Types' and Subclasses

‘Create Product Types’ pattern discovered from the financial services domain offers well-defined steps for process modelling that facilitates the creation of a new product and makes it ready to be launched by the organisation. Hurby (2006) developed a business pattern called creating a new product using REA-based application models that model economic exchanges that actually occurred and stated that the resource output of this type of pattern is of an economic importance, as almost every company needs a process in which it creates a new service or product.

Second: Generalisation of Similar Business Processes Initiating Event:

This type of generalisation depends on the Initiating Event of a business process. Thus, it brings all the business processes that have similar types of initiating events to trigger their business processes. The reason for developing this type of generalisation is because it enables organisations to keep a record of all the important initiating events, which typically help to keep a record of when process should take place. It can be easily argued that this type of generalisation makes it possible to record business-initiating events and, at a later point in time, analyse these events and draw conclusions for its business processes. These conclusions typically lead to activities or decisions in the organisation, such as to discontinue a product definition. Below are the generalised Business Processes Ontology models achieved from this generalisation rule:

1. **BP Triggered by Opening Accounts** is a general Business Process Ontology models by defining a restriction that all processes should have a similar type initiating event 'Client Willing To Open Accounts' when a client requests to open an account, as presented in Figure 6-11.



Class Description: BPTriggeredByOpeningAccounts

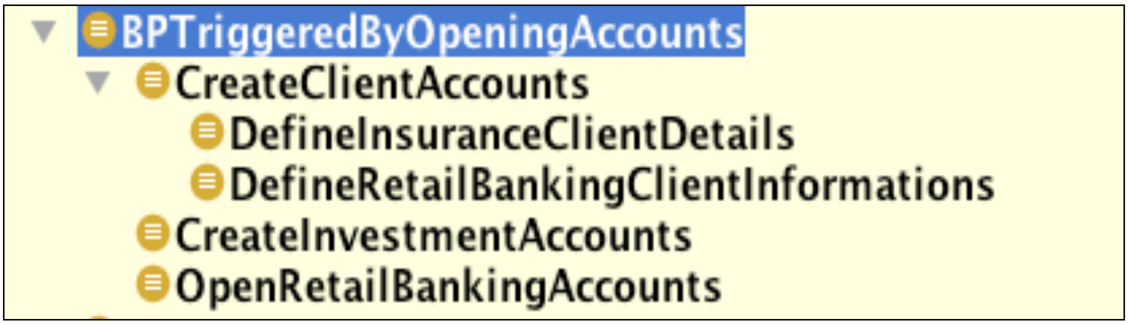
Equivalent classes

● BusinessProcesses
and hasTemporalPart some ClientWillingToOpenAccount

Figure 6-11: General BP Ontology Model with its defined Restriction

When using a reasoner it classifies all the processes that have similar initiating event, as presented in Figure 6-12.

Another example of generalised BPO model by having similar type Initiating Events is 'BP Triggered By New Product Introductories' which was presented in Appendix C.



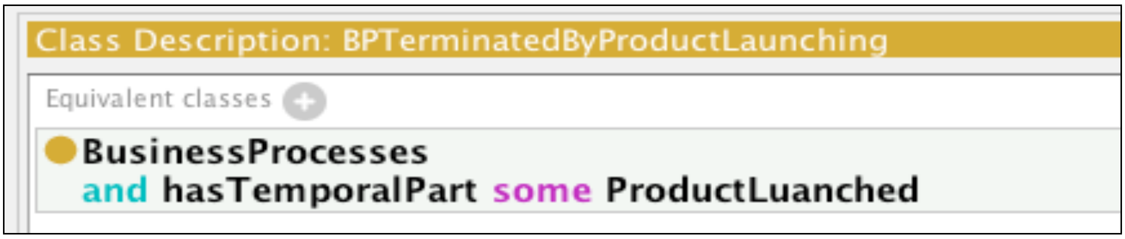
BPTriggeredByOpeningAccounts
- CreateClientAccounts
  - DefineInsuranceClientDetails
  - DefineRetailBankingClientInformations
- CreateInvestmentAccounts
- OpenRetailBankingAccounts

Figure 6-12: 'BP Triggered By Opening Accounts' and Subclasses

Third: Generalisation of Similar Business Processes Goal Event:

This type of generalisation depends on the Goal Event of a business process. Thus, it brings all the business processes that have a similar type of goal event to dissolve their business processes. The reason to define this type of generalisation is that a business process exists for a reason, which strives to achieve a goal. Thus, any business process without a corresponding goal should be eliminated. The more clearly a business goal is stated, the easier it is to define and design the corresponding activities and events so that the goal can be achieved. So to design a business process, the modeller must first describe the goal that motivates that process, then connect it to the process described, because a goal event expresses the desired state for or result of a business process. Having stated the importance of this type of generalisation, this enables the research to defend its validation. Below are the generalised Business Process Ontology models achieved from defining this generalisation rule:

1. **BP Terminated By Product Launching** is a general BPO model by defining a restriction that all processes should have similar type Goal event ‘Product Launched’, as presented in Figure 6-13.



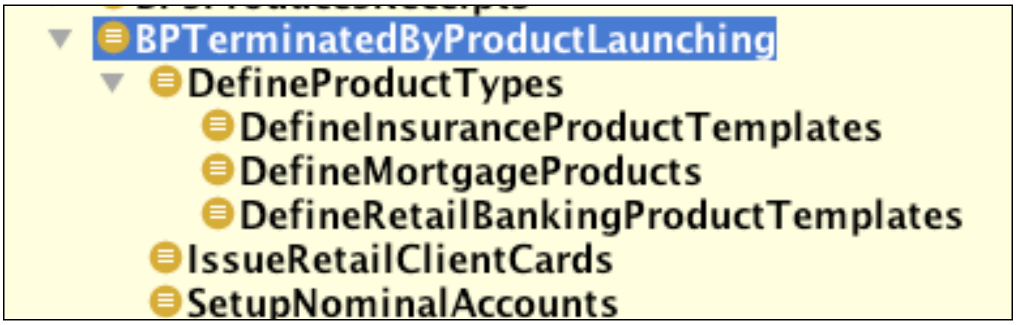
Class Description: BPTerminatedByProductLaunching

Equivalent classes

BusinessProcesses
and hasTemporalPart some ProductLuanched

Figure 6-13: 'BP Terminated By Product Launching' and its Restriction

When running FaCT++ it brings together all the processes that have similar Goal events, as presented in Figure 6-14.



```
▼ BPTerminatedByProductLaunching
    ▼ DefineProductTypes
        DefineInsuranceProductTemplates
        DefineMortgageProducts
        DefineRetailBankingProductTemplates
      IssueRetailClientCards
      SetupNominalAccounts
```

Figure 6-14: ' BP Terminated By Product Launching' and Subclasses

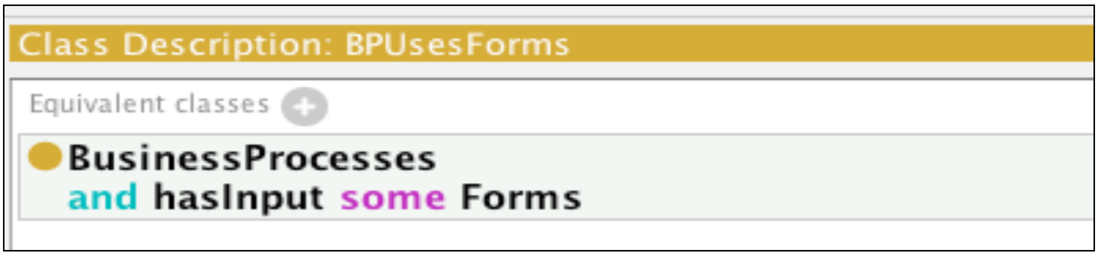
Another example of generalised Business Process Ontology models by having similar Goal Events is ‘BP Terminated By Applying Changes’ and “BP Terminated by Client Information Created” which have been presented in Appendix C.

Fourth: Generalisation of Similar Business Processes Inputs:

This type of generalisation depends on the input that a business process needs to resource its processes. Thus, it generalises all the business processes that have similar input types. The reason to develop this type of general process model is to provide a practical way to approach the issues on which type of document should be used as a resource within different business processes, including its different versions and copies. It might be argued that this general model does not offer much for an organisation, but according to Ericksson and Penker (2000) who have developed a ‘Resources Use’ general model argued that this type of model is important to understand that resources can be used in one way for one process, and in a totally different way in another process. Thus, neglecting the fact that an input can be used in different processes in different ways will in many cases lead to processes that do not make optimal use of its resources. Below are the generalised Business Process Ontology models achieved from this generalisation rule:

1. **BP Uses Forms** is a general BPO model by defining a restriction that all processes should have similar type Input ‘Forms’, as presented in Figure 6-15.

]



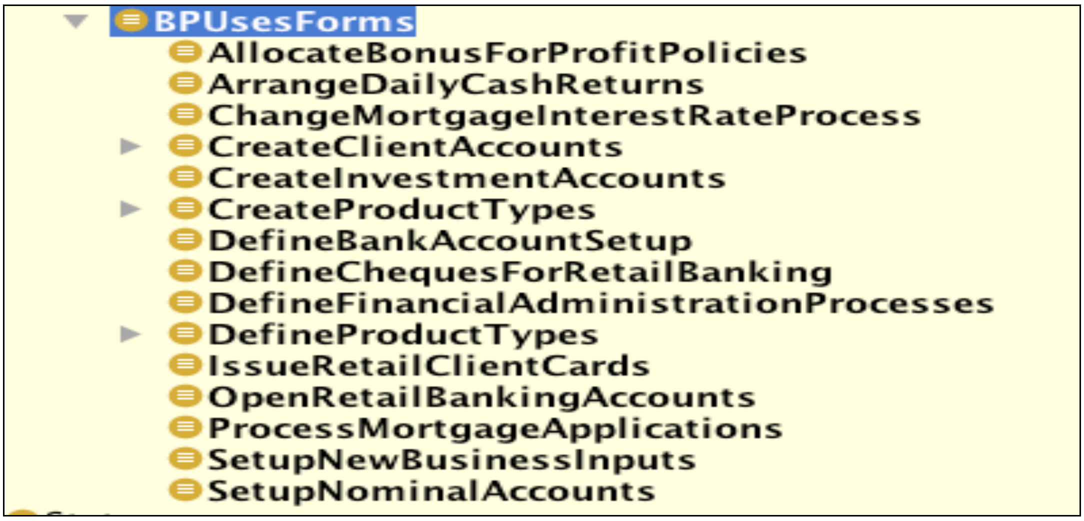
Class Description: BPUsesForms

Equivalent classes

BusinessProcesses
and hasInput some Forms

Figure 6-15: 'BP Uses Forms' General BPO Model with its Restriction

When using Reasoner FaCT++ it classifies all the processes that have similar Inputs, as presented in Figure 6-16.



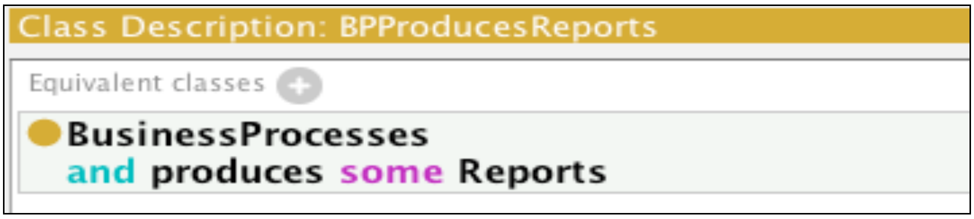
BPUsesForms
AllocateBonusForProfitPolicies
ArrangeDailyCashReturns
ChangeMortgageInterestRateProcess
CreateClientAccounts
CreateInvestmentAccounts
CreateProductTypes
DefineBankAccountSetup
DefineChequesForRetailBanking
DefineFinancialAdministrationProcesses
DefineProductTypes
IssueRetailClientCards
OpenRetailBankingAccounts
ProcessMortgageApplications
SetupNewBusinessInputs
SetupNominalAccounts

Figure 6-16: 'BP Uses Forms' and Subclasses

Fifth: Generalisation of Similar Business Processes Outputs:

This type of generalisation depends on the Output that a business process produces. Thus, it brings together all the business processes that have similar Output type. This type of generalisation provides a practical way to approach the issues on which type of document should be delivered from different business processes. Below are the generalised Business Process Ontology models achieved from this generalisation rule:

1. **BP Produces Reports** is a general BPO model produces by defining a restriction that all processes should have similar type Output ‘Reports’, as in Figure 6-17.



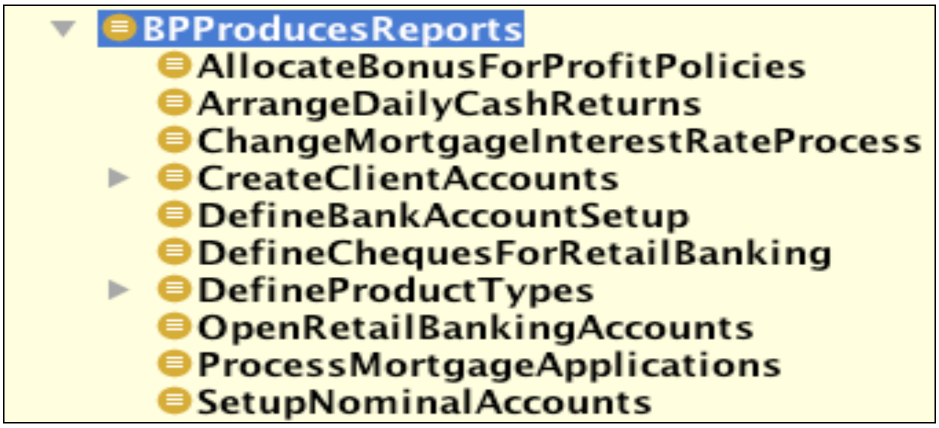
Class Description: BPProducesReports

Equivalent classes

BusinessProcesses
and produces some Reports

Figure 6-17: ‘BP Produces Reports’ with its Restriction

When running Reasoner it brings all the processes that have similar Output, as presented Figure 6-18.



```
▼ BPProducesReports
      AllocateBonusForProfitPolicies
      ArrangeDailyCashReturns
      ChangeMortgageInterestRateProcess
    ▶ CreateClientAccounts
      DefineBankAccountSetup
      DefineChequesForRetailBanking
    ▶ DefineProductTypes
      OpenRetailBankingAccounts
      ProcessMortgageApplications
      SetupNominalAccounts
```

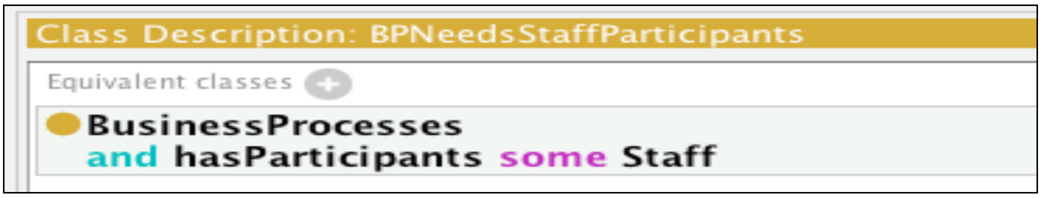
Figure 6-18: 'BP Produces Reports' and Subclasses

More BPO can be generalised according to having similar output type, such as ‘BP Produces Letters’, ‘BP Produces Letters’ and ‘BP Produces Receipts’, available in Appendix C.

Sixth: Generalisation of Similar Business Processes Participants:

This type of generalisation depends on the Participant that takes part in a business process. Thus, it brings all the business processes that have similar Participant types, and according to the financial participant, generalisation was for two groups only Staff or Clients. The aim of this type of generalisation is that, it enables the easy connect of roles of different participants in the business process. The roles are defined for a certain context, usually by a specific organisation. Using this pattern also makes it possible to locate and define certain connections, such as that a certain organisational process can only take place with one type of participant. Ericksson and Penker (2000) business patterns defined actor-role pattern to be used in all problem situations in which there is a need to separate actors from roles. Below are the generalised Business Process Ontology models achieved from this research generalisation rule:

1. **BP Needs Staff Participants** is a general BPO by defining a restriction that all processes should have a similar participant 'Staff', see Figure 6-19.

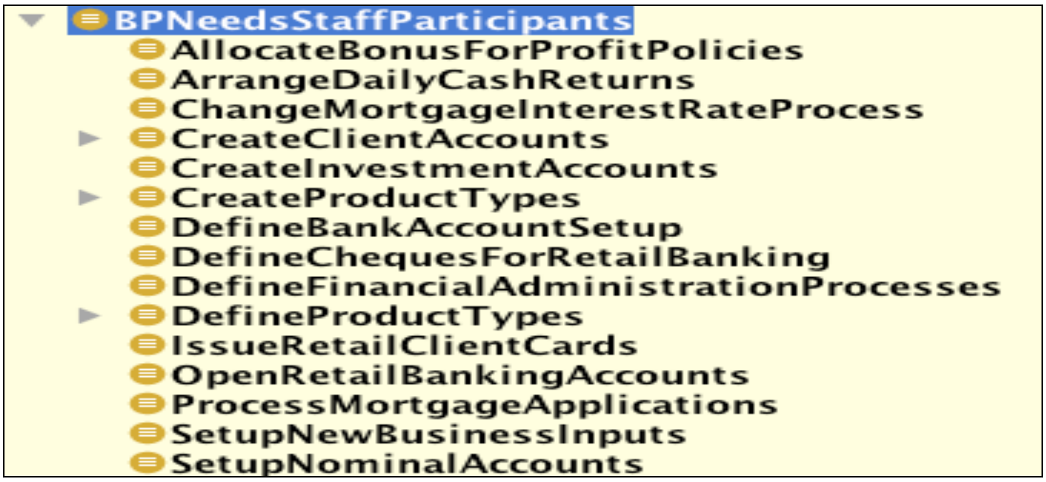


Class Description: BPNeedsStaffParticipants

BusinessProcesses
and hasParticipants some Staff

Figure 6-19: 'BP Needs Staff Participants' BPO Model Restriction

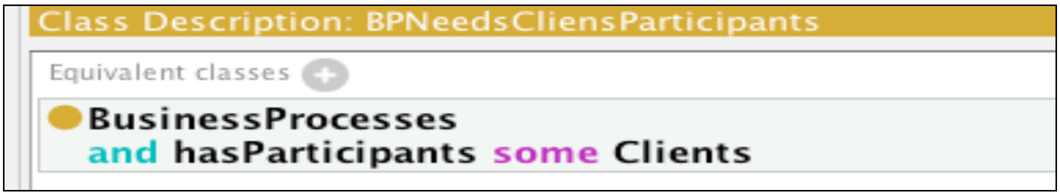
When running Reasoner it classifies all the processes that have similar Output, as presented in Figure 6-20.



BPNeedsStaffParticipants
AllocateBonusForProfitPolicies
ArrangeDailyCashReturns
ChangeMortgageInterestRateProcess
CreateClientAccounts
CreateInvestmentAccounts
CreateProductTypes
DefineBankAccountSetup
DefineChequesForRetailBanking
DefineFinancialAdministrationProcesses
DefineProductTypes
IssueRetailClientCards
OpenRetailBankingAccounts
ProcessMortgageApplications
SetupNewBusinessInputs
SetupNominalAccounts

Figure 6-20: 'BP Needs Staff Participants' and Subclasses

2. **BP Needs Client Participants** is a general BPO model by defining a restriction that all processes should have similar type participant 'Clients', see Figure 6-21.

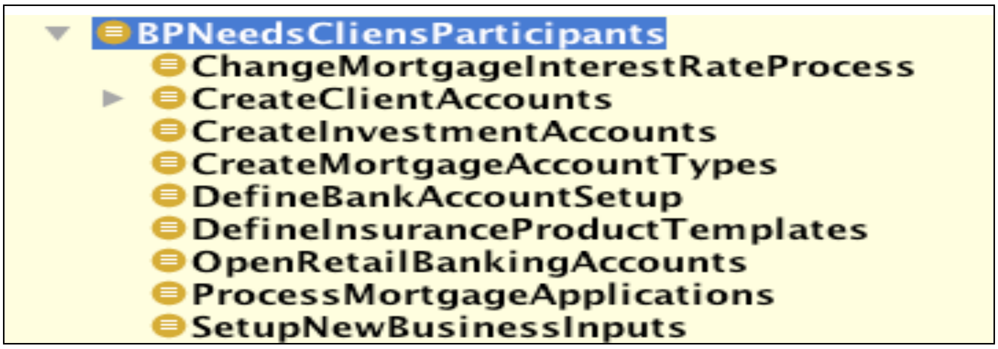


Class Description: BPNeedsCliensParticipants

BusinessProcesses
and hasParticipants some Clients

Figure 6-21: 'BP Needs Clients Participants with its Restriction

When using Reasoner FaCT++ it classifies all the processes that have similar Output, as in Figure 6-22.



BPNeedsCliensParticipants
ChangeMortgageInterestRateProcess
CreateClientAccounts
CreateInvestmentAccounts
CreateMortgageAccountTypes
DefineBankAccountSetup
DefineInsuranceProductTemplates
OpenRetailBankingAccounts
ProcessMortgageApplications
SetupNewBusinessInputs

Figure 6-22: 'Need Client Participant' and Subclasses

Phase 4: Pattern Documentation (PD):

The researcher believes that this phase is pertinent within the SDL. Thus, this is the last phase of SDL although the design process is iterative. At this stage, documentation is the main design activity included within this phase. Business process patterns documentation is important since poor documentation of a pattern is one of the main barriers to effective knowledge sharing and dissemination (Havey, 2000). In addition, documentation plays a key role in facilitating pattern maintenance, use, and reuse. Therefore, documentation clarity and simplicity are considered major issues affecting business process patterns usefulness and value. Hence, the way these patterns are documented is designed to address the needs of various audiences. One example of patterns documentation presented in Table 6-7 'Define Product Type Pattern' business process pattern using Ericksson and Penker (2000). More examples for the various discovered types of patterns are available in Appendix D.

Additional thinking added to structure a hierarchy of the discovered financial domain patterns in Figure 6-23. The primary motivation behind this rationale is to formally describe the different BPO elements that the discovered patterns generalised or extracted from so that unwanted ambiguities related to the application and use of the pattern can be avoided.

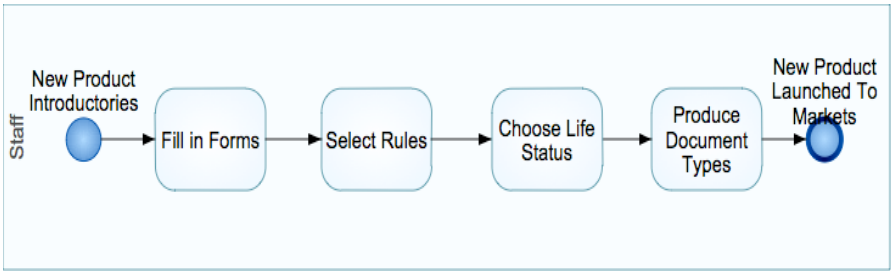
Pattern One	
Name	Define Product Type
Intent	This pattern is a business process pattern that describes the way in which products can be defined, such as new mortgage product or new university course.
Motivation	Suppose that ABC product is required to be defined as a new product at XYZ organisation to keep XYZ organisation up to date with markets need. This ABC product definition has start and end event on when the need for a new product introductory to XYZ organisation take place, and when launching the ABC product to XYZ organisation can be consider fully defined. You define these products to the organisation in the form of sequence of steps. This ensures the accurate representation for the product. Certain rules for the ABC product can be selected. These rules that you set for a product that define its nature. ABC product is set to life status to release the product to the market and make it available for customers use, as well produce document like report, guides, forms that support its definition, this pattern can be used with any type of product definition.
Applicability	Define Product Types pattern lays the foundation for the definition of any product within an organisation in a flexible and high-quality model. The Define Product Types pattern can be implemented to clarify the product structures within an organisation, or to build an information system that define information about product and its structure.
Structure	 Staff New Product Introductories Fill in Forms Select Rules Choose Life Status Produce Document Types New Product Launched To Markets
Participants	Input: Forms Output: Reports Participant: Staff Initiating Event: New product Introductory Goal Event: New product launched to markets Activities: Fill in Forms, Select Rules, Choose Life Status and Produce Document Types.
Consequences	Using Define Product Types pattern provides a proven and a clear architecture for process modelling that facilitates the definition, improvement and amendment of existing or new product in an organisation. Thus, A new product is ready to be launched by the company.

Table 6-7: 'Define Product Type' Business Process Pattern Documentation

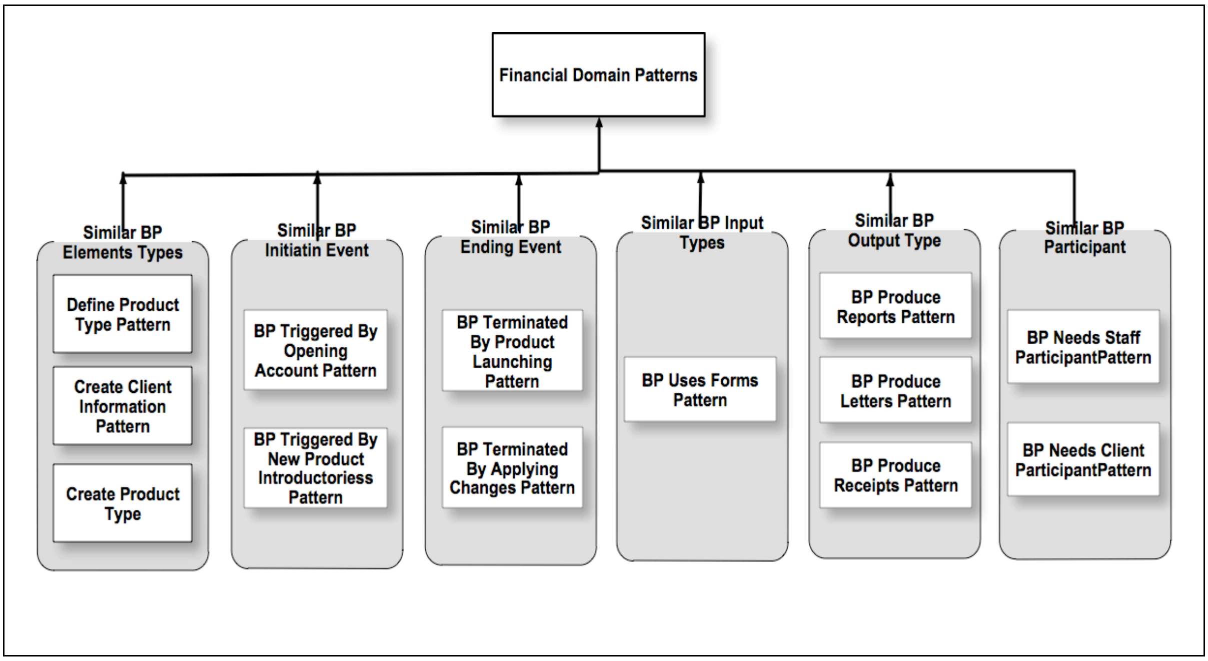

Financial Domain Patterns
Similar BP Elements Types
Define Product Type Pattern
Create Client Information Pattern
Create Product Type
Similar BP Initiatin Event
BP Triggered By Opening Account Pattern
BP Triggered By New Product Introductoriess Pattern
Similar BP Ending Event
BP Terminated By Product Launching Pattern
BP Terminated By Applying Changes Pattern
Similar BP Input Types
BP Uses Forms Pattern
Similar BP Output Type
BP Produce Reports Pattern
BP Produce Letters Pattern
BP Produce Receipts Pattern
Similar BP Participant
BP Needs Staff ParticipantPattern
BP Needs Client ParticipantPattern

Figure 6-23: Financial Domain Business Process Pattern Hierarchy

6.5 Patterns Across Domains

To enrich the business process patterns discovered in this iteration from the financial services domain and the previous iteration from the educational domain, further empirical investigation are preformed by utilising the *Semantic Enhancement* phase of the Semantic Discovery Lifecycle across multiple domains (educational and financial) with multiple systems (students, staff, retail banking, insurance and mortgages). The reason behind undertaking such a step is directly aimed at extending the reusability of patterns and to strongly argue on the commonality of the discovered business process patterns across multiple domains even though those patterns are discovered or extracted from different domains. Generally the generalisation is based on summarising the same processes and neglecting processes with different specificities. Figure 6-24 represents Business Process Patterns hierarchy of the financial and education domains.

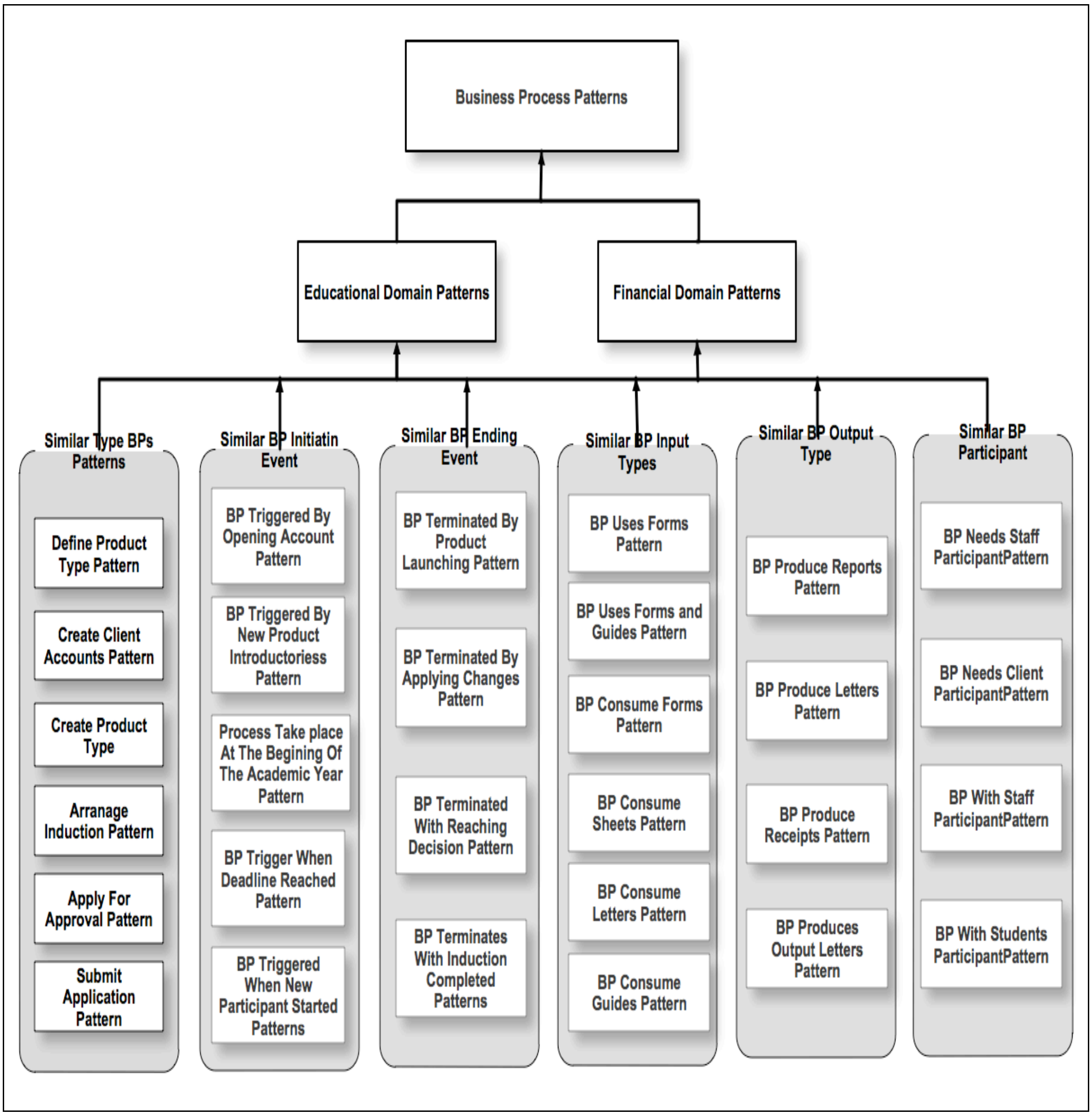
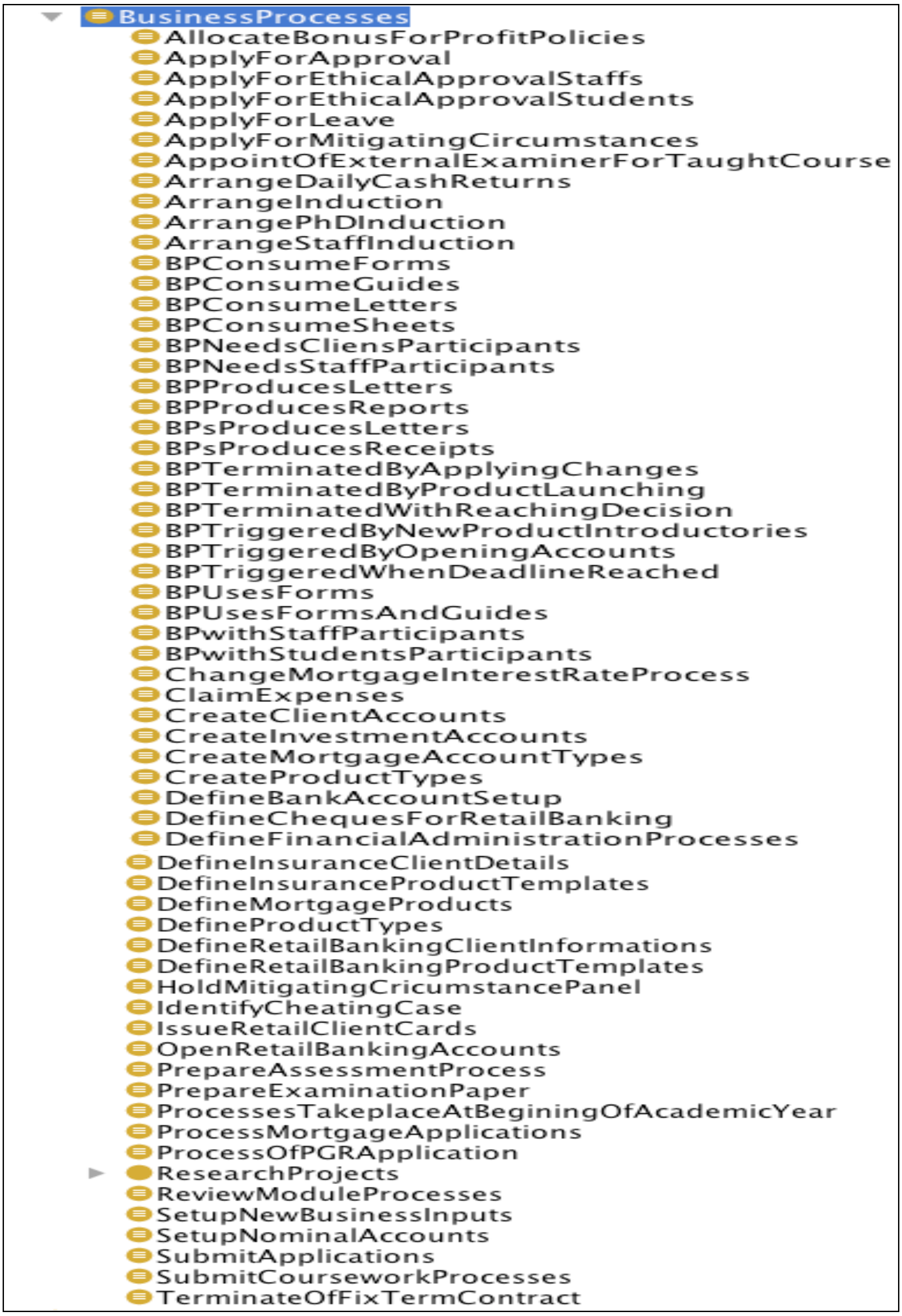

Business Process Patterns
Educational Domain Patterns
Financial Domain Patterns
Similar Type BPs Patterns
Define Product Type Pattern
Create Client Accounts Pattern
Create Product Type
Arrange Induction Pattern
Apply For Approval Pattern
Submit Application Pattern
Similar BP Initiatin Event
BP Triggered By Opening Account Pattern
BP Triggered By New Product Introductoriess Pattern
Process Take place At The Begining Of The Academic Year Pattern
BP Trigger When Deadline Reached Pattern
BP Triggered When New Participant Started Patterns
Similar BP Ending Event
BP Terminated By Product Launching Pattern
BP Terminated By Applying Changes Pattern
BP Terminated With Reaching Decision Pattern
BP Terminates With Induction Completed Patterns
Similar BP Input Types
BP Uses Forms Pattern
BP Uses Forms and Guides Pattern
BP Consume Forms Pattern
BP Consume Sheets Pattern
BP Consume Letters Pattern
BP Consume Guides Pattern
Similar BP Output Type
BP Produce Reports Pattern
BP Produce Letters Pattern
BP Produce Receipts Pattern
BP Produces Output Letters Pattern
Similar BP Participant
BP Needs Staff ParticipantPattern
BP Needs Client ParticipantPattern
BP With Staff ParticipantPattern
BP With Students ParticipantPattern

Figure 6-24: Business Process Patterns Hierarchy Across Domains

To this aim, a Protégé – OWL document has been constructed to bring the Business Process Ontology models of the two domains with their associated discovered patterns (Appendix E) as presented in Figure 6-25.



- BusinessProcesses
  - AllocateBonusForProfitPolicies
  - ApplyForApproval
  - ApplyForEthicalApprovalStaffs
  - ApplyForEthicalApprovalStudents
  - ApplyForLeave
  - ApplyForMitigatingCircumstances
  - AppointOfExternalExaminerForTaughtCourse
  - ArrangeDailyCashReturns
  - ArrangeInduction
  - ArrangePhDInduction
  - ArrangeStaffInduction
  - BPConsumeForms
  - BPConsumeGuides
  - BPConsumeLetters
  - BPConsumeSheets
  - BPNeedsCliensParticipants
  - BPNeedsStaffParticipants
  - BPProducesLetters
  - BPProducesReports
  - BPsProducesLetters
  - BPsProducesReceipts
  - BPTerminatedByApplyingChanges
  - BPTerminatedByProductLaunching
  - BPTerminatedWithReachingDecision
  - BPTriggeredByNewProductIntroductories
  - BPTriggeredByOpeningAccounts
  - BPTriggeredWhenDeadlineReached
  - BPUsesForms
  - BPUsesFormsAndGuides
  - BPwithStaffParticipants
  - BPwithStudentsParticipants
  - ChangeMortgageInterestRateProcess
  - ClaimExpenses
  - CreateClientAccounts
  - CreateInvestmentAccounts
  - CreateMortgageAccountTypes
  - CreateProductTypes
  - DefineBankAccountSetup
  - DefineChequesForRetailBanking
  - DefineFinancialAdministrationProcesses
  - DefineInsuranceClientDetails
  - DefineInsuranceProductTemplates
  - DefineMortgageProducts
  - DefineProductTypes
  - DefineRetailBankingClientInformations
  - DefineRetailBankingProductTemplates
  - HoldMitigatingCricumstancePanel
  - IdentifyCheatingCase
  - IssueRetailClientCards
  - OpenRetailBankingAccounts
  - PrepareAssessmentProcess
  - PrepareExaminationPaper
  - ProcessesTakeplaceAtBeginingOfAcademicYear
  - ProcessMortgageApplications
  - ProcessOfPGRApplication
  - ResearchProjects
  - ReviewModuleProcesses
  - SetupNewBusinessInputs
  - SetupNominalAccounts
  - SubmitApplications
  - SubmitCourseworkProcesses
  - TerminateOfFixTermContract

Figure 6-25: BPO Models and Patterns From the Financial and Educational Domains

After constructing the ontology of business processes and patterns across domains spme points can be recognised and discussed below:

First: It has been found that some ontological models generalised previously on the Input of a process, such as Forms, can be further generalised across the two domains and classified as 'Generalisation of Similar Patterns Inputs Types'.

- **Pattern Utilises Document Types** is a general Business Process Ontology model by defining a restriction that all patterns discovered earlier require an Input for its process pattern of a document type, as presented in Figure 6-26 and documented in Appendix D.

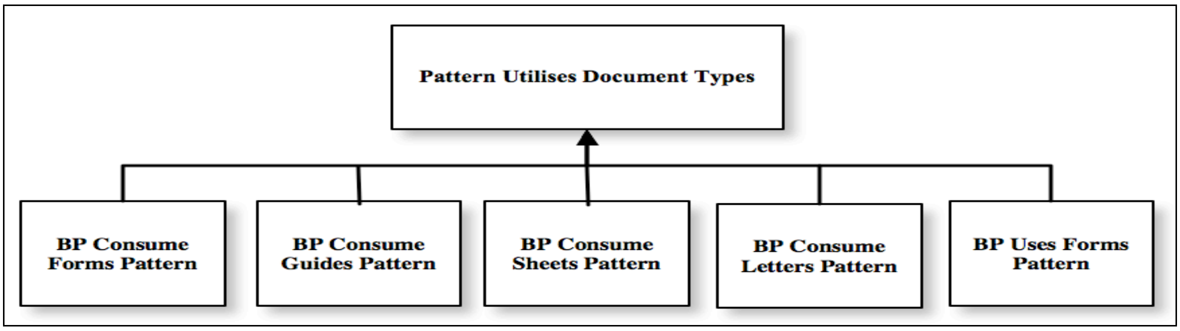

Pattern Utilises Document Types
BP Consume Forms Pattern
BP Consume Guides Pattern
BP Consume Sheets Pattern
BP Consume Letters Pattern
BP Uses Forms Pattern

Figure 6-26: General Pattern of 'Utilises Document Types'

Second: similarly, further generalised across the two domains are classified as 'Generalisation of Similar Patterns Output Types'. This type of generalisation brings all the business process patterns that have similar Output type, below are the generalised patterns achieved from this generalisation rule.

- **Pattern Provides Document Types** is a general Business Process Ontology model by defining a restriction that all patterns require an Output for its process pattern of a document type, as presented in Figure 6-27 and documented in Appendix D.

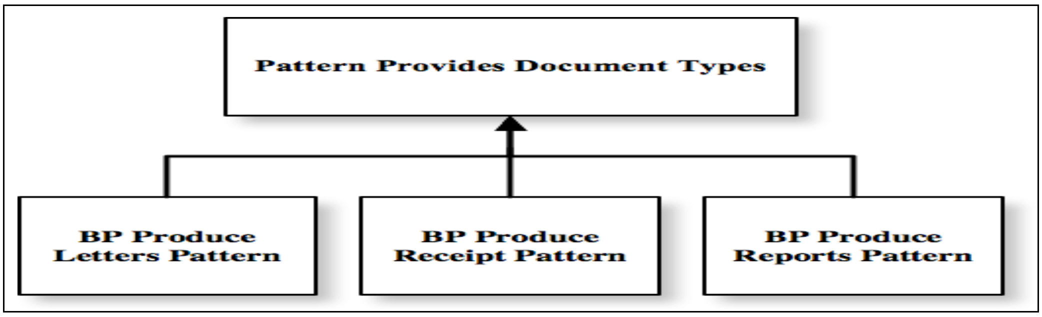
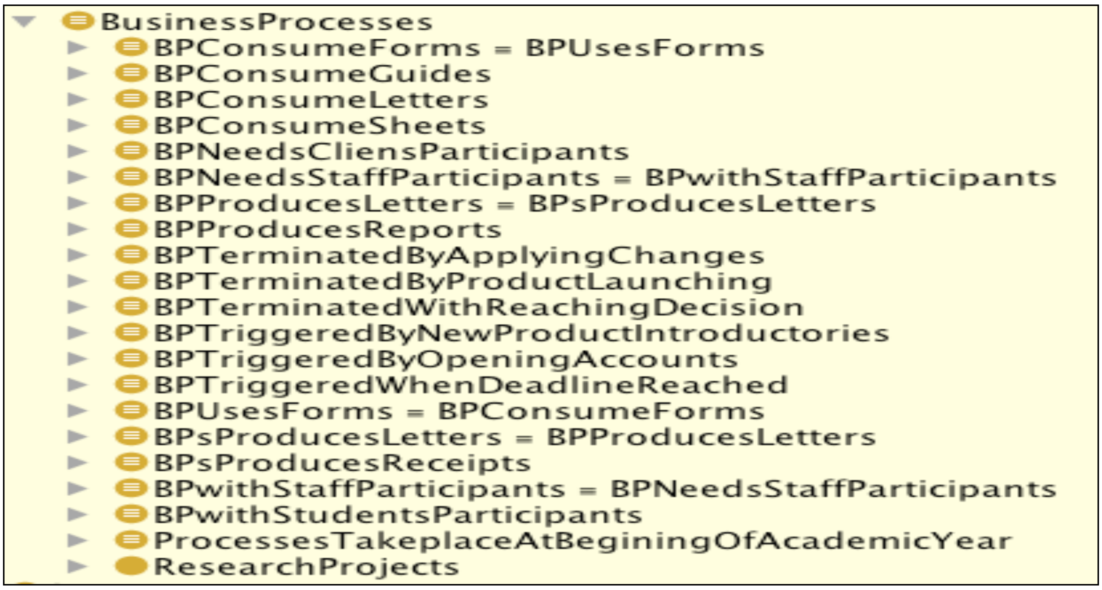

Pattern Provides Document Types
BP Produce Letters Pattern
BP Produce Receipt Pattern
BP Produce Reports Pattern

Figure 6-27: General Pattern of 'Provides Document Types'

Third: A reasoner is used to check the consistency of the general BPO across domain i.e., to verify whether there are any logical contradictions in the different ontological processes. Figure 6-28 represents the reasoner classification for business process patterns across the two domains.



BusinessProcesses
- BPConsumeForms ≡ BPUsesForms
- BPConsumeGuides
- BPConsumeLetters
- BPConsumeSheets
- BPNeedsCliensParticipants
- BPNeedsStaffParticipants ≡ BPwithStaffParticipants
- BPProducesLetters ≡ BPsProducesLetters
- BPProducesReports
- BPTerminatedByApplyingChanges
- BPTerminatedByProductLaunching
- BPTerminatedWithReachingDecision
- BPTriggeredByNewProductIntroductories
- BPTriggeredByOpeningAccounts
- BPTriggeredWhenDeadlineReached
- BPUsesForms ≡ BPConsumeForms
- BPsProducesLetters ≡ BPProducesLetters
- BPsProducesReceipts
- BPwithStaffParticipants ≡ BPNeedsStaffParticipants
- BPwithStudentsParticipants
- ProcessesTakeplaceAtBeginingOfAcademicYear
- ResearchProjects

Figure 6-28: Reasoner Classification Across Domains

A closer look at the ontological process patterns in Figure 6-28 highlights that running the reasoner led to recognise that some business process patterns are of a similar type across the two domains as presented in Table 6-8.

Educational Domain Patterns	Financial Domain Patterns	Type of Similarity
BP Consume Forms	BP Uses Forms	Input Type – Forms
BP Produce Letters	BP Produce Letters	Output Type – Letters
BP with Staff Participant	BP Needs Staff Participants	BP Participants Type – Staff

Table 6-8: Type of Similarity in BP Patterns Across Domains

Fourth: Due to the limited data source access used from Brunel University, further generalisation across the two domains would be achieved, an example of ‘Create Product Type’ pattern of the financial services domain. Despite this the researcher could not find similar processes in the DISC handbooks, thinking of universities creating new course types enable to further generalised the two domains patterns. Another example is ‘Create Client Information’ pattern of the financial services domain, we can think of a client, as a student at the university and creating student information is a must within the universities. Those two examples and more would offer further generalisations across the financial and educational domains, but due to the limited access the researcher used what data sources and information that could be obtained.

Figure 6-29 represents the generalisation diagram achieved from bringing the financial and educational domains patterns. Interestingly, from what has been discovered and explained previously, it can be argued that discovering patterns on multiple domains is advantageous since it (1) facilitates capturing greater details among the different domains and (2) improves the validity of the findings through the ability of undertaking comparisons and further generalisation across domains. Reflecting on the work done previously, this step is deemed appropriate in this iteration to support the researcher's argument on the SDR capability to discover patterns across multiple systems and domains.

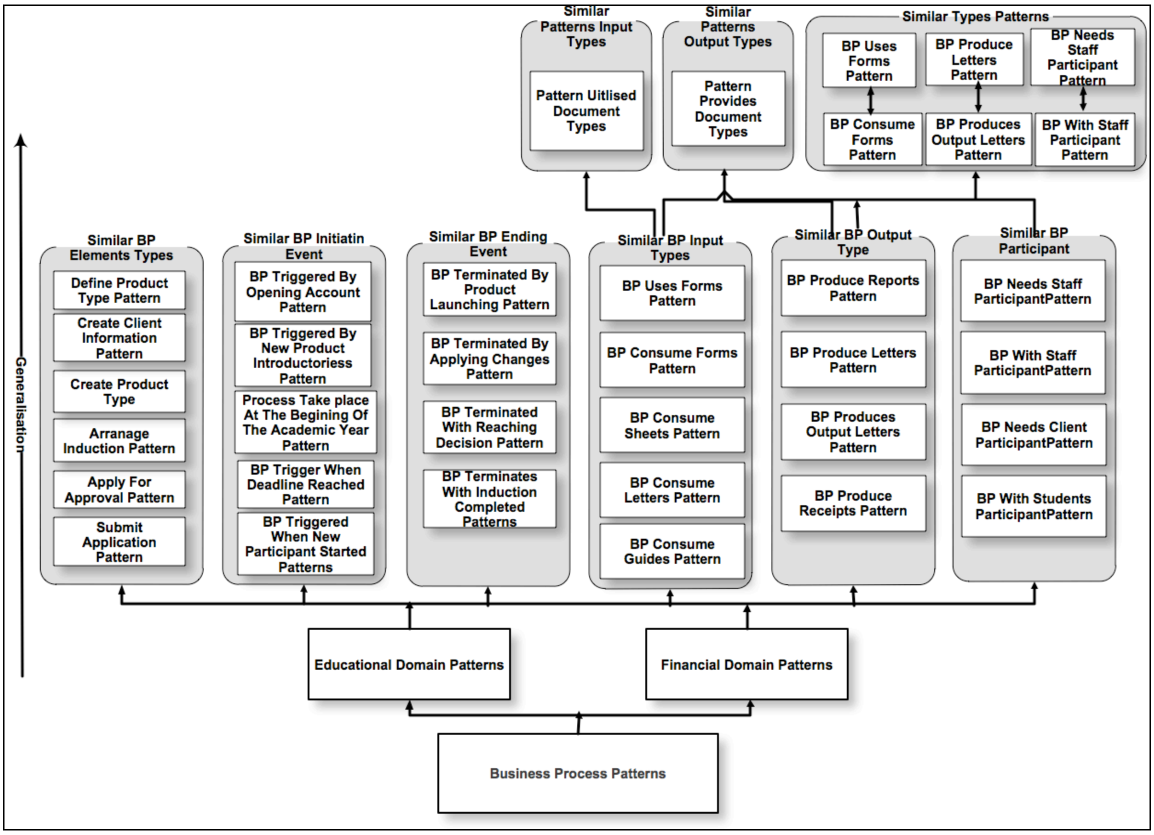

Similar Patterns Input Types
Pattern Uitlised Document Types
Similar Patterns Output Types
Pattern Provides Document Types
Similar Types Patterns
BP Uses Forms Pattern
BP Produce Letters Pattern
BP Needs Staff Participant Pattern
BP Consume Forms Pattern
BP Produces Output Letters Pattern
BP With Staff Participant Pattern
Generalisation
Similar BP Elements Types
Define Product Type Pattern
Create Client Information Pattern
Create Product Type
Arranage Induction Pattern
Apply For Approval Pattern
Submit Application Pattern
Similar BP Initiatin Event
BP Triggered By Opening Account Pattern
BP Triggered By New Product Introductoriess Pattern
Process Take place At The Begining Of The Academic Year Pattern
BP Trigger When Deadline Reached Pattern
BP Triggered When New Participant Started Patterns
Similar BP Ending Event
BP Terminated By Product Launching Pattern
BP Terminated By Applying Changes Pattern
BP Terminated With Reaching Decision Pattern
BP Terminates With Induction Completed Patterns
Similar BP Input Types
BP Uses Forms Pattern
BP Consume Forms Pattern
BP Consume Sheets Pattern
BP Consume Letters Pattern
BP Consume Guides Pattern
Similar BP Output Type
BP Produce Reports Pattern
BP Produce Letters Pattern
BP Produces Output Letters Pattern
BP Produce Receipts Pattern
Similar BP Participant
BP Needs Staff ParticipantPattern
BP With Staff ParticipantPattern
BP Needs Client ParticipantPattern
BP With Students ParticipantPattern
Educational Domain Patterns
Financial Domain Patterns
Business Process Patterns

Figure 6-29: Generalisation Diagram across the Financial and Educational Domains

6.6 Research Output Artefacts

This section describes research output for this iteration.

1. **Final version of SDR Methodological Framework**. This is one of the main outputs. The SDR incorporates two lifecycles. For iteration three the former lifecycle SDL is preceded with four phases and it has been matured over the past two iterations by using SDR in different domains and proved its ability to discover patterns across multiple domains and systems. Also, a set of encapsulated tasks within a single SDL is performed and a scenario used to develop the process lifecycle. The following is the output of SDL Phases.
2. **Business process pattern(s)**. This is the second main output. The process of producing business process pattern(s) involves using several techniques and ends with documenting the financial services domain patterns in the repository. Also, patterns across domains (financial and educational domains) have been discovered and documented in this iteration.

The rest of the research outputs for iteration three are instrumental outputs of SDL:

- **BP Model(s)**. Business process models are constructed through the ground analysis of three financial services system using reverse engineering and modelling techniques to transform the asset segments to BP Models.
- **Ontology BP model(s)**. The ontological definition of BP forms the input for the semantic discovery rules that directly transform BP Model(s) to BPO model(s). Multiple mini-iterations have compared the process models to the Business Process Ontology models for the financial services domain.

6.7 Evaluation of Research Artefacts and Evolutions

This iteration continues with the previous criteria used to evaluate the two previous iterations. In fact the research presented in this chapter primarily targets the applicability of the SDL lifecycle of SDR methodological framework to produce business process patterns using different domains since in iteration two SDL proved its ability to produce business process patterns from the educational domain, and to further support its ability to discover patterns across multiple domains and systems. It also became apparent that SDL possesses a well-structured lifecycle and phases evident by the project management activities, necessary for predicting things like time-scales and cost-benefits. Thus only operational evaluation to check SDL ability to achieve its aim proposed in this iteration.

SDL Lifecycle Operational Evaluation:

Completeness of the lifecycle is evaluated by measuring the percentage coverage of SDL lifecycle. In the previous iteration, the research was able to run a full iteration to discover patterns from the educational domain. In iteration three the completeness of this iteration has been proved as the whole Semantic Discovery lifecycle ran smoothly from the *Preparation of Legacy Assets* Phase to the *Pattern Documentation* Phase. A further refinement has been added to the documentation phase of the SDL, a new data source from the financial domain used to demonstrate improved completeness of SDL and discover patterns from the financial domain such as ‘Define Product Types’ pattern, ‘BP consumes Forms’ pattern, etc. This however is not only a measure of maximum coverage of SDL to different domains but also further support for SDL completeness added, when it proved its ability to discover patterns across multiple domains (financial and education domains) and across multiple systems, such as ‘BP Utilises Document Types’ pattern and more. Thus, the research can arguably defend the completeness of the final version of SDL in the sense of the lifecycle has preformed without any obstacles as well it delivers its duty of producing patterns of business processes from different and across domains.

Validation and documentation, as stated in the previous evaluation that validation and documentation of the developed patterns is an important consideration in SDL lifecycle. *Pattern Documentation* phase of SDL are intended to be a reference point for practitioners wishing to understand a problem or its solution in greater detail. As the researcher discovery patterns form the educational domain in the previous iteration and continued with a new domain (financial domain) in this iteration, then moved over to discover patterns across the two domains led to an enormous number of discovery patterns. In fact, it can easily happen to lose record of the number and the types of discovered patterns. Also, it is worth mentioning that patterns in this research have been developed from and across only the financial and educational domains, which means further problems can occur when more patterns across different domains are discovered or added to the similar exist types. Thus, to keep a record of the different discovered patterns, not only the pattern template of Ericksson and Penker has been used but also taxonomy of the different discovered patterns is constructed so that the details of the patterns solution are organised and made available in a human-readable format. Constructing a pattern hierarchy adds advantages as (1) Adding and updating patterns is easy, (2) make patterns retrieval more efficient and (3) a hierarchical model is simple to construct and operate on.

Efficiency, the Semantic Discovery Lifecycle for discovering business process patterns has been increased in this iteration Table 6-9. A breakdown of the time taken to construct patterns of business processes from the financial services domain is given in Table 6-9. Accuracy of the recorded times is improved over the last iterations since the researcher had a clearer perception of the stages of construction. Timings were recorded for all the phases and the steps of SDL and precision of the measurements is given to the nearest half a day. The improved performance is attributed to the researcher having a clearer understanding of the SDL over the past iterations. Performance improvement can also be attributed to increased efficiency in the researcher at managing workload and the application of the SDL.

Activity	
Iteration Three	15 Days
Number of Business process models	18 Models
Number of Business Process Patterns	15 Patterns
Estimated man day per pattern	~ 0.5 per day

Table 6-9: Represent Performance of Iteration Three

Precision is the degree to which an experiment can be repeated and achieve the same result. Precision of pattern discovery by the Semantic Discovery lifecycle is an improvement over the manual activities of iterations one and providing semi-automated phases in iteration two since human error is removed, and consistency check was preformed using a reasoner. Also, in this iteration as one becomes more experienced in the methodology, it become easier to identify in the data sources used certain patterns previously defined. For example when using documentation in natural language (e.g. handbooks or user guides) the modeller more readily maps certain sentence structure with business process patterns.

BP Patterns Operational Evaluation:

Clarity and Conciseness, similar to the previous iteration patterns are discovered from the financial domain for a purpose such as 'Create Client Information Pattern', this pattern helps to describe the way on which client details information can be created within an organisation, such as staff, clients and students. Client information is attached to any product in an organisation. Defining the purpose of developing this pattern supports its clarity criterion. As well, the Ericksson and Penker (2000) template enables ease of documentation and clarity of purpose for the discovered financial patterns, as it offers way of tracking the different types of patterns, patterns applicability, structure and consequence. Also, Ericksson and Penker's template enables the creation of a list of the elements used within each pattern, and this is an important point for patterns conciseness (Noy and McGuinness, 2001). By increasing the conciseness of the patterns, the researcher prevents unnecessary definitions, and explicit or implicit redundancies in the definitions, which may add no value to the understanding of the phenomenon under investigation and lead to the production of irrelevant patterns.

Coherence, generalising business process models using ontology editors like Protégé provide functionality for syntax checking and axioms testing, through which most of the inadvertent inconsistencies can be detected and resolved. A number of such tests have been run on the financial services patterns and across the domains. Also, reasoners like FaCT++, Hermit or Pellet allow for more sophisticated consistency testing. They do not only detect inconsistencies between the stated elements (activities, input, output, event), but also check for contradictory conclusions that can be inferred from these elements. Reasoning has been used in this iteration to discover patterns across the financial and educational domains. This helped not only to discover similar types of patterns across the two domains such as 'PB Consumes Forms' pattern of Brunel University handbook and 'BP Uses Forms' pattern from the financial services domain, but also enables further generalisation for similar types patterns such as 'Pattern Utilises Document Types' business process pattern. This pattern brought all the process patterns of the financial and educational domains that uses document types i.e. forms, sheets, etc. to one general pattern as explained earlier in Section 6.5.

Customisability is minimal ontological commitment and extendibility (Gruber, 1995). In other words: For sake of reusability, the patterns elements should be kept to a minimum and more general in order to allow for different extensions and thus fit a large number of processes contexts. The principle of minimal ontological commitment states that "an ontology should make as few claims as possible about the world being modelled, allowing the parties committed to the ontology freedom to specialise and instantiate the ontology as needed" (Gruber, 1995, p.14). As those patterns have been developed ontologically using Business Process Ontology definition. Patterns discovered and designed in a way that gives its different users the ability to expand the existing shared vocabulary without altering existing ones, such as 'Create Product Type Pattern'. These discovered business process patterns are a general solution for defining products in an organisation, which can be specialised to meet domain process specific needs either for creating bank account products in the financial domain or create new course in the university for the academic domain.

6.8 Discussion

Whilst the previous iterations designed and evolved the final version of the Semantic Discovery Lifecycle, the main focus of this iteration was to discover business process patterns from the new domain (financial Services domain) and demonstrate SDL's ability to produce patterns across multiple domains and systems. A conclusion from this iteration follows:

- Precision of discovered business process patterns produced by the reasoner is high. Researcher error can be removed through automation of manual processes. Most likely because Business Process Ontology definition provides a viable semantic based-approach to support the ontological interpretation for existing organisation process and adds concept richness through a desire to uncover links to real-world concept in the domain.
- Financial and educational domains data sources were integrated to provide patterns that support reuse between domains. The empirical work proved successful in finding commonality and ability to discover patterns across multiple systems and

domains. Discovered patterns are able to provide an effective solution to the process reuse problem.

- For this researcher 6 categories of generalisation have been used to classify the discovery of patterns, the researcher can easily argue the benefit of this type of classification: (1) discovered patterns are of a business process type, and business processes consist of elements that construct its process. Those elements affect the behaviour of those business processes. Thus, discovering business process patterns using business process elements provide powerful finding as each element have a distinct role in accomplishing the whole organisation processes, (2) Adding constraint for each type of discovered pattern help modellers to both represent and use patterns of generalised behaviour in the sense that it would be possible to state a minimal set of necessary conditions that are required for the application of a certain pattern, to enable ease of reuse, and (3) This categorisation plays a key role in facilitating pattern documentation, maintenance, use, and reuse. Clarity and simplicity are considered major issues affecting business process patterns usefulness and value. Hence, the way these patterns are categories is designed to address the needs of various audiences and can be tracked easily.

6.9 Feedback and Further Work

The following limitations of the SDR methodological framework drive recommendations for further work that form feedback from this final design iteration. General application of the SDL should be further improved through continuous application of the SDL to additional systems with differing architectures and domains.

1. Apply SDL to additional systems and domains in order to improve generality of the approach.

As the SDL phase's currently only semi-automated using a BPMN modelling tool to produce BPMN models and Protégé – OWL to construct BPO models to generalise BP models and produce semantically accurate business process patterns. Thus, automating the SDL phases could increase the efficiency and the quality of discovered patterns.

2. Extend SDL phases to automate discovering patterns and support designer intervention when transforming organisational data sources to business process patterns in order to improve SDL process performance, efficiency and precision.

The Second Lifecycle - the Semantic Reuse Lifecycle can be constructed as a plug-in to ease and efficient reuse of the different solutions documented in the repository.

3. Evolve the Semantic Reuse Lifecycle to offer an easy access for the different solution offered with the Discovery Lifecycle repository (Section 7.4).

6.10 DR Evaluation For The Research Iterations

Practical designs and their evaluation activities are covered in Chapter 4, 5 and 6. This section explains how DR has helped in reaching the aim of this research. In this research as Design Research execution followed three iterations in which each represented a specific "search process to discover an effective solution to a problem" (Hevner et al. 2004 p. 88). Existing research frameworks proposed by Hevner et al. (2004) and March and Smith (1995) were integrated within the SDR construction and provided procedures and language for the research. The evaluation of SDR comprises informed argument, scenarios and static analysis testing (taken from Hevner et al. (2004) Design Research evaluation methods), as follows:

- Informed argument because the researcher presents an argument in favour of the use of the methodological framework lifecycles and support the methodology ability to discover business process patterns, which have qualities to offer reusability, but also have some limitations. Those arguments have been constructing because data from certain organisation (prototype organisation) e.g. financial services. These financial services are not an organisation, they are a prototypical organisation because these systems are generalised for any organisation e.g. mortgages company so that it is a prototypical mortgage company. Whereas, for the case of Brunel University handbooks. It is a specific organisation. Thus, it is not really a scenarios but it is a BP.
- Static analysis, this research is not static on the contrary it is dynamic as the researcher to an active part on discovering patterns across multiple domains and

systems over the three iterations. Static analysis addresses how the methodology is structured and what phases and steps to include in its lifecycles. Consequently, the structural analysis does not really tell much, in contrast to the descriptive analysis then later to be finished with the need for the informed argument to provide the critical reflection on the applicability of SDR.

At the same time, design artefacts in this research are classified by March and Smith (1995), and Hevner et al. (2004), into constructs, models, methods, and instantiations. The developed SDR methodological framework in this iteration represents a method artefact that is based on developing an ontological patterns *model,* which includes *constructs*. Moreover, the implementation of the developed ontology and patterns in Protégé-OWL represents an *instantiation*.

- Constructs: The individual classes of the developed Business Process Ontology represent a design constructs artefacts. The ontology presented contains 17 design constructs.
- Model: One of the current research outputs is the model in the form of Business Process Ontology Models and patterns. In fact, the BP models of the developed ontology is an example where the research shows how different design dimensions and concepts interrelate and interlink with each other. The research also expresses the hypothesised relationships between the design constructs graphically through the model as discussed in Chapter 3.
- Methods: SDR methodological framework and its key value drivers, the research identify the major function and provide guidelines to discover and reuse business process patterns effectively.
- Instantiations: developing Business Process Ontology and patterns through Protégé development platform and representing it through OWL is an example of instantiations artefacts that are developed in this research.

6.11 Final Evaluation Stage

This represents the final stage in the broad outline of the development research stages (see Figure 3-4), which aimed at providing a *Final Evaluating* to the discovered process patterns by using one of the business process patterns i.e. Define Product Type pattern within detailed scenarios. This motivation leads to an understanding of how the pattern refers to as the forces that lead to the solution.

Designing a brand new course within an academic environment can be a daunting task for new, as well as experienced, academic staff. An effective course definition begins with asking questions in order to understand who the students are, deciding what they would like to learn, determining how to measure whether students are learning, and planning when it is most appropriate to start the new course also what activities, assignments and materials that are favourable to student learning to be attached.

Although courses may vary in size, subject matter and level, a business process pattern that offers a systematic process will help academic staff to plan and structure the suitable activities for the new course so as to effectively reach desired instructional goals for designing the course and add further specialisation to meet the needs of the new environment imposed in it.

Designing an academic course has not been part of the discovered business processes in this research. Now the question can the research use a relevant pattern to help in address the issue on what should be involved in designing a new academic course. The researcher suggests the use of ‘Define Product Type’ business process pattern discovered and documented form the financial domain to address the issue of designing a new academic course. The use of this pattern is due to the similarity in the processes running across various domains and systems.

Let’s invoke the design of new university course scenarios again by applying the ‘Define Product Type’ pattern. First, the initiating event that triggers the design of new university course is set when new course introductory is required. This event triggers a set of activities to proceed in designing a new course. Second, it is the role of the academic staff member to handle the process of designing new course. Third, using the four main activities used in ‘Define Product Type’ pattern and add further specialised activities that meet the need of designing an academic course. Expanding on how each of the general activities has helped in designing the new academic course, the first activity is ‘Fill in form.’ This activity enables the staff to fill the require information related to defining a course i.e., student numbers, subject to be taught, etc. ‘Select rules’ activities enable the staff to decide on the type of assessment and what restriction should be combined with the course definition. ‘Choose life status’ is related to when the new product is most appropriate to be life and student can undertake the course. The last activity is related to ‘Produce Document Type’ where a course specification should be fully designed and approved. Input resources such as course template form and output resources, such as course definition report, are part of the definition of new course process. More specialised activities related to designing new course in an academic environment such as hold departmental meeting to discuss the criteria of the new course, review other university courses and search market need for the new course.

Using ‘Define Product Types’ pattern provides a proven and a clear model for designing a new process for introducing a new course that facilitates the definition, improvement and amendment of existing or new courses in an academic environment. Thus, A new course is ready to be launched by the university.

6.12 Summary

Chapter 6 has reflected on the practical application of the final version of the Semantic Discovery Lifecycle within the context of multiple systems and domains. The chapter began by describing the final refinement underpinning for SDL phases and the processes, clarified the necessary work to form the basis for developing patterns across multiple domain. The SDL practical application was discussed next, that clearly focused on how the SDL should be applied. This laid the ground for the actual application of the final version of SDL and resulted in the identification of a number of patterns commonality across domains and the development of many business process patterns. By reflecting on the potential for use and implications of the SDL, the chapter continue with previous iterations criteria. The evaluation of these artefacts in the work done to date broadly reflected that:

- The approach is practically feasible;
- The SDL has resulted in significant improvement in terms of improving the knowledge of the organisation business data and the theory thereof; and
- Further work is vital to improve on the current understanding.

CHAPTER 7 – CONTRIBUTIONS AND PROPOSAL FOR FUTURE RESEARCH WORK

This chapter summarises the contributions made by the research presented in this thesis. It revisits some of the key contributions made by the work presented here and discusses the extent to which the initial aim has been achieved. It also discusses the potential for further research in the area of discovery and reuse of pattern-based business process modelling. The remainder of this chapter focuses mainly on four topics, namely, the overview of the thesis and what has been accomplished in each chapter in Section 7.1, the most important research values and contributions of this thesis in Section 7.2, the objectives of the research and how they have been addressed in Section 7.3, and limitation and plans for future work in Section 7.4.

7.1 Research Overview

The research presented in this thesis is aimed at providing a methodological framework characterised as a manageable and effective solution for empirically deriving ontological patterns of business processes from organisational knowledge sources. The thesis as a whole was presented in three parts and organised in seven chapters. This chapter is a summary of the previous six chapters and the findings of this study. This research utilises empirical data as its main source, which makes it a distinctive piece of research, as both the methodology and the patterns are developed and discovered from empirical data.

Part One: The Introduction and Literature Review (Chapters 1 and 2) outline the research; presenting context, reasons for undertaking such research and literature that reflects the current state of the research in the area. The research presents a review of the existing literature on business process modelling and emphasises the potential of both ontologies and patterns to positively impact BPM. Consequently, the literature begins by introducing business processes, opening a discussion that it is essential that business process changes are understood and represented systematically and their impact clearly defined. It then covers the limited reuse of previously developed models and reviews the scarce pattern literature in BPM. The literature then brings the recent increased interest on how ontologies have been suggested as a means to develop more systematic, yet flexible, approaches to business process design (Hepp et al., 2005; Lin and Strasunskas, 2005; Hepp and Roman, 2007). Also, further claimed benefits of ontology-based business process modelling also include improved model distribution, integration and interoperability (Thomas & Fellmann, 2009).

Part Two: Design Research was presented in Chapter 3 and covered the pragmatic application of Design Research in the IS field and tailored its application to this research project. DR is the chosen method used to undertake the research and is covered in the middle section of the thesis (Chapter 3). Design Research execution followed three iterations of *Design, Deployment* and *Evaluation* in which each represented a specific "search process to discover an effective solution to a problem" (Hevner et al., 2004 p. 88). The problem statement at the outset of the research concerned the SDR methodological framework. The assumption was that subsequent iterations would refine the methodology and helped to produce two versions before reaching the mature and the final version of the SDR methodological framework. Interestingly, the learning that took place in each iteration provided solutions to identified problems that together form the overall design of SDR. Also, the *Awareness of the research problem and proposed solution* led to the construction of the first version of the SDR methodological framework, which evolved over the three iterations to reach the final version of SDR.

Part Three: Practical designs and their evaluation activities formed the third part of the thesis and are covered in Chapters 4, 5 and 6. The design method evolved in support of each of the design iterations using the broad outline of the developmental research phases that were constructed by translating the general methodology of DR (Vaishnavi and Kuechler, 2004). Chapter 4 started with the application of SDR to support the semantic discovery of business process patterns from the empirical analysis of organisational knowledge sources, such as the legacy assets of the financial services domain. This chapter presented iteration one and drew its requirements from the initial suggestion. Subsequently the work cycled through mini iterations of *Design, Deploy* and *Evaluation* on real world scenarios, even though the initial construction of the SDL process began as an idea without a clear understanding of the form that each phase would take. Assuming that patterns could be constructed from legacy systems was simplistic and naive, however it was a necessary first step from which the research could progress. Chapter 5 presented iteration two, which drew requirements from the feedback of iteration one. The iteration delivered improvements in the artefact(s) because the individual iteration evolved the second version of SDR methodology incrementally. Well-defined business process patterns were discovered from the educational domain of Brunel University. An ontological definition of business process was also developed and applied to alter and extend rules governing transformation of BPM into ontological business process models. Chapter 6 represents iteration three; it described a new empirical experiment, which emphasises the applicability of the final version of the Semantic Discovery Lifecycle to produce business process patterns across various domains and among multiple systems (financial and educational domains). Pattern-based business process modelling discovered from multiple systems and domains improve the productivity of modellers as well as helped achieving improved levels of traceability between business requirements and software systems.

7.2 Research Contributions and Values

The contributions of the current research are multi-folds. The key contributions can be summarised as:

First: the evolvement of the Semantic Discovery and Reuse of Business Process Patterns (SDR) methodological framework specifically the Semantic Discovery Lifecycle SDL (Chapter 4-6). The efficacy of this methodology was verified through its empirical discovery from real world data sources to develop business process patterns; also its ability to discover patterns across multiple domains and systems was demonstrated. The final version of SDL phases is presented in Figure 7-1, this lifecycle has provided some important contributions:

(1) Discovered patterns are of a business process type and business processes consist of elements that construct its process. Those elements affect the behaviour of those business processes. Thus, discovering business process patterns using business process elements provides a powerful mechanism as each element has a distinct role in accomplishing the organisation processes, (2) Adding constraints to each type of

discovered pattern helps modellers to both represent and use patterns of generalised behaviour in the sense that it would be possible to state a minimal set of necessary conditions that are required for the application of a certain pattern, to enable ease of reuse. (3) This categorisation plays a key role in facilitating pattern documentation, maintenance, use, and reuse, and (4) Running a reasoner is used within the *Semantic Analysis* and *Semantic Enhancement* phases to check the consistency of the general BPO across the domain and carry out consistency checks on the working or completed ontology, this has help to verify whether there are any logical contradictions in the different ontological processes.

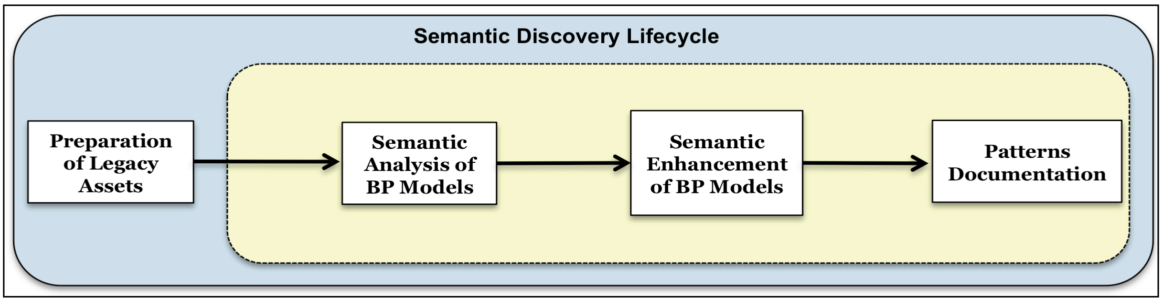

Semantic Discovery Lifecycle
Preparation of Legacy Assets
Semantic Analysis of BP Models
Semantic Enhancement of BP Models
Patterns Documentation

Figure 7-1: The Final Version of The Semantic Discovery Lifecycle

Second: develop and document business process patterns from the empirical analysis of the educational (Chapter 5) and financial (Chapter 6) domains, as well as discover business process patterns across multiple domains and systems (Chapter 6).

Third: definition of a powerful guide for business process model interpretation to business process ontology models in order to achieve an effective level of formalisation in terms of process modelling which lowers the risk of discovering irrelevant patterns.

The primary value and relevant benefits of the research are as follows:

- **Industry**, which will be provided with a semi automated methodological framework for an easy and manageable way to semantically discover and reuse business process patterns from their legacy knowledge assets across multiple systems and domains. The current research provides practitioners in industry with valuable, systematic, and customisable means to design, implement, analyse, evaluate, discover and change new and existing models to make reuse more manageable, effective, and also efficient. The researcher strongly believes that this methodology improves the current practices and functions of patterns in regards to organisational process behaviour. This improvement would be reflected on: (1) Business analyst and modeller, by allowing them to achieve their strategic goals and objectives through better utilisation of reuse potentials; and (2) Business owners and user community in large, by providing them with useful and up to date organisation that meets their needs and enhances the quality of their processes.

- **Academia**, which will benefit from the cross-fertilisation of the disparate disciplines of business process management, software engineering, the Semantic Web and Model-Driven Development. It will also benefit from the opening up of new areas of research into approaches. Constructing SDR contributes toward more precise representations of business process semantics essential in the work that the W3C is carrying out in the area of Semantic Web services and the choreography/orchestration of processes.

In addition, a reusable business process patterns-based methodological framework produces a reduction in the effort required to produce process models and for their future subsequent translation into software designs of enterprise applications. This benefit would be coupled with the acquired advantage of facilitating staff training in BPM as well as staff familiarisation with the organisation's more general procedures. In fact adequate knowledge of generalised patterns of business processes would provide BPM staff the opportunity to be more readily capable of identifying common behavioural themes even in the presence of newly designed processes (Zlatkin & Kaschek, 2005). In addition, reusability of business process models is likely to help reduce the development time and increase process quality of the models themselves as well as their mapping into information system requirements. Business process reuse could be significantly simplified if an easy to use and extensible reuse environment were available (Kaschek, Wiltsche and Rinderer, 1998; Calabrese et al., 2006). As an antecedent to the use of generalised models it is necessary to discover recurring patterns in the first place and represent them appropriately.

7.3 Meeting Research Objectives

Objectives were formulated at the start of the research and presented in Chapter 1. The research aim expands into the objectives of this research and forms the basis of the design iterations. With the major contributions stated earlier, each of the objectives are revisited and discussed in relation to the completed research activities. The following objectives were formulated:

Objective I: *Evolve a methodological framework for identifying patterns of business processes from enterprise knowledge sources (i.e., documentation, systems, domain experts, etc.)*. Objective I was achieved through the cross-fertilisation of the disparate disciplines. More specifically, three main domains (i.e. domain engineering, ontologies and patterns) were deemed relevant and helpful in addressing this objective. This objective was raised because of the lack of guidelines to modellers as to how business process patterns can be discovered must be resolved as they form the basis for further reuse. Thus, evolving the SDR methodological framework to support the finding of business process patterns represents an important area of work. SDR has been evolved through iterations of *Design*, *Deployment* and *Evaluation* presented in Chapters 4, 5 and 6. Each iteration contributed to produce an incrementally refined version of SDR through its applicability to different environments and to reach its final and mature methodological framework in iteration three (Chapter 6). This methodology guides the application process and acts as a reference document for situations where the methodology is applied.

Objective II: *Derive a set of generic business process patterns by applying the developed methodology and to organise the patterns within the repository*. This objective was achieved by deriving business process patterns from the educational domain in Chapter 5 and from the financial domain in Chapter 6. Furthermore the research proved SDR's ability to discover patterns across domains and systems as discussed in Chapter 6, when a set of generic business process patterns across the financial and education domains and across multiple systems was derived. In this research, a well-documented template adapted from (Ericksson and Penker, 2000) was used since poor documentation of patterns is one of the main barriers to effective knowledge sharing and dissemination. In addition, using a repository for the documented patterns plays a key role in facilitating pattern maintenance, use, and reuse. Therefore, organised repository offers clarity and simplicity and considers major issues affecting business process patterns' usefulness and value.

Objective III: *Investigate the development of an ontology of business process to be used for the derivation of the semantic process modelling*. This objective is met in Chapter 5 to model a business process ontologically for making the business processes as much as possible close to the real world objects they represent since the business process patterns must be reused, their representations must be as clear and unequivocal as possible, which ensures the grounding of the patterns to real world behaviour. The paradigm that the proposed business process ontology was based on is the object paradigm (Partridge, 1996). Reusing the ontological definition of business process has many benefits: It lowers the time and cost of development, avoids duplicate efforts, ensures interoperability, etc. The developed ontological definition is significant not only because it is novel, but also because it is relevant and purposeful. The relevance of the ontological definition comes from the fact that this definition helped to interpret empirical data sources across domains as constructed in Chapters 5 and 6. On the other hand, the developed ontological definition is purposeful as it improved the current definition of business processes because of its accurate representation of real world organisational behaviour within this research. The developed ontological definition, in particular, is also deemed intuitive and significant. This is because it provides a common language and terminology amongst information systems and business modellers to enhance their interoperability.

Objective IV: *Evaluate the methodological framework and derived patterns in terms of their significance to theory and practice through recognised evaluation techniques*. Evaluation is decisive since it assesses the extent of success of the constructed methodological framework and process patterns. Evaluation of the developed artefacts in this research is a continuous process from Chapter 4 to Chapter 6. For this research a complete set of criteria that helped in evaluating its artefacts was developed by adopting the two aspects which primarily made up the evaluation criteria here (March & Smith 1995); these aspects were, *Methodological Study* and *Operational Study* and the criteria adopted included completeness, efficiency, effectiveness and precision. These criteria were used to reflect on the developed artefacts.

7.4 Research Limitations and Future Work

In addition to the significant contributions made in this research and briefly highlighted in the previous section, the current research also provides some important directions for future research in order to continue developing this vital research and overcome any limitations. The thesis presents a comprehensive lifecycle of the Semantic Discovery and Reuse methodological framework to the problem of discovering business process patterns from organisational knowledge sources. The data sources used whilst undertaking this research have all been derived from two domains (financial and educational). A useful addition would be to apply the framework to further domains in order to determine their generality (and utility of the same semantic discovery lifecycle phases and steps). Although the discovery lifecycle provided an effective pattern discovery across multiple domains and systems, its utility within this research clearly depends on knowledge sources (in the form of documents – handbooks and user guides), in order to undertake the discovery of business process patterns. This was due to the difficulty in obtaining data sources and information from organisations given that most of this information is confidential; it represents a company's knowledge and expertise so the company will be reluctant to share their data. It especially became clear for the researcher during this research that such data would also provide additions to the semantic model and the discovered patterns. Privacy concerns limited the researcher's ability to acquire such data.

This research developed the ontological definition of business processes as a novel approach for providing semantic accurate representations of real world organisation processes. This approach has been tested empirically within the current research as it was applied to construct the educational and financial ontology process models. However, it is would advantageous to conduct further testing and validation for this definition. Utilising the ontological definition of business process to interpret varied organisational processes within a wide range of domains is important for future research. This would refine the definition if needed and reinforce its efficacy and value. The researcher hopes that this step will help in providing additional validation for this definition as an ontology alignment for organisational processes in the field of information systems, as the potential of adopting ontologies as the foundation to modelling process models has been recognised and discussed in Chapter 2.

The Semantic Discovery and Reuse methodological framework has two distinct lifecycles for the business process patterns being discovered and reused. The discovery lifecycle provides a precise description for the phases and steps used in discovering and developing existing patterns. In contrast, for the reuse lifecycle, despite its complete phases definition, an empirical experiment for its validity has not been accomplished, due to the difficulty in finding an organisation that has a long-term strategy to manage the change, reuse and consistency checking of the processes changes over time. Thus, further work can be done to evaluate the reuse lifecycle within an organisation and change its phases within the SDR methodological framework to make the investment using discovered and evolved patterns worthwhile.

Reference

Agerbo, E. & Cornils, A. 1998, "How to preserve the benefits of design patterns", *Proceedings of the 13th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, Vancouver, British Columbia, Canada, October 18-22, p.134-143.

Agrawal, R., Gunopulos, D. & Leymann, F. 1998, "Mining process models from workflow logs", *Proceedings of the Sixth International Conference on Extending Database Technology (EDBT)*, Valencia, Spain, Lecture Notes in Computer science, Springer Berlin / Heidelberg, Volume 1377, pp. 469–483.

Aguilar-Savén, R.S. 2004, "Business process modelling: Review and framework", *International Journal of Production Economics,* vol. 90, no. 2, pp. 129-149.

Aldin, L. & and de Cesare, S. 2009 a, "A Comparative Analysis of Business Process Modelling Techniques", *Proceedings of the 14th Annual UK Association of Information Systems Conference (UKAIS)*, Oxford University, Oxford, U.K, 31st March - 01st April, pp. 8-28.

Aldin, L. & and de Cesare, S. 2010, "A Literature Review on Business Process Modelling: New Frontiers of Reusability", *Enterprise Information System Journal,* (under review).

Aldin, L. & de Cesare, S. and Lycett, M. 2009 b, "Semantic Discovery and Reuse of Business Process Patterns", *4th Annual Mediterranean Conference on Information Systems,* Athens University of Economics and Business (AUEB), Greece, September 25-27, pp. 1-6.

Aldin, L., de Cesare, S. & and Lycett, M. 2009 c, "A Semantic-Based Framework for Discovering Business Process Patterns", *6th International Workshop on Ontology-Driven Software Engineering (ODiSE): ACM Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA),* Orlando, Florida, USA, Oct. 25-29, pp. 26-40.

Alexander, C. 1979, *The timeless way of building,* New York: Oxford University Press.

Alexander, C., Ishikawa, S., Silverstein, M. & Centre for Environmental Structure 1977, *A pattern language: towns, buildings, construction,* New York: Oxford University Press.

Andrews, T., F. Cubera, H. Dholakia, Y. Goland, J. Klein, F. Leymann, K. Liu, D. Roller, D. Smith, S. Thatte, I. Trickovic, and S. Weerawarana: 2003, *Business Process Execution Language for Web Services Version 1.1* (Technical report): BEA Systems, International Business Machines Corporation, Microsoft Corporation, SAP AG, Siebel Systems, http://ifr.sap.com/bpel4ws.

Antoniou and van Harmelen, 2010, "Web Ontology Language: OWL", In: S. Staab, R. Studer (eds.), *The Handbook on Ontologies in Information Systems*, Springer, Berlin, pp. 91-111

Arango, G., 1988, "Domain Engineering for Software Reuse", PhD Thesis, Department of Information Systems and Computer Science, University of California, Irvine.

Azoff, A., Kellett, A., Roy, S. & and Thompson, M. 2007, *Business Process Management: Building End-to-end Process Solutions for the Agile Business*: Butler Direct Ltd., Technology Evaluation and Comparison Report.

Batra, D. and Sin, T (2008) The READY Model: Patterns of Dynamic Behaviour in REA-Based Accounting Applications, *Information Systems Management*, vol. 25, pp. 200-210

Beck, K. & Cunningham, W. 1987, "Using Pattern Languages for Object-Oriented Program", Technical report, Tektronix, Inc., presented at *Workshop on Specification and Design for Object-Oriented Programming (OOPSLA)*, Orlando, Florida, USA.

Benbasat, I. & Weber, R. 1996, "Research Commentary: Rethinking "Diversity" in Information Systems Research", *Information Systems Research,* vol. 7, no. 4, pp. 389-399.

Bandara, W., Tan, H.M., Recker, J., Indulska, M. & Rosemann, M. 2007, "Bibliography of process modelling: An Emerging research field", Journal Article, Available from: http://eprints.qut.edu.au/8754/.

Brockmans, S., Ehrig, M., Koschmider, K., Oberweis, A. & Studer, R. 2006, "Semantic alignment of business processes", *Proceedings of the Eighth International Conference on Enterprise Information Systems (ICEIS)*, Paphos, Cyprus, pp. 191-196.

Bunge, M. 1977, *Treatise on Basic Philosophy, Ontology I: The Furniture of the World,* 3rd edn. Boston: Reidel Publisher Press.

Caetano, A., Silva, A.R. & Tribolet, J. 2005, "Using roles and business objects to model and understand business processes", *Proceedings of the 2005 ACM symposium on Applied computing*, Santa Fe, New Mexico, March 13-17, pp. 1308.

Calabrese, F., Di Dio, G., Fasolino, A.R. & Tramontana, P. 2006, "A Methodology for searching reusable business processes", *Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT-ICIW)*, Guadeloupe, French Caribbean, February19-25, pp. 153.

Cline, M.P. 1996, "The pros and cons of adopting and applying design patterns in the real world", *Communications of the ACM,* vol. 39, no. 10, pp. 47-49.

Coad, P. 1992, "Object-oriented patterns", *Communications of the ACM,* vol. 35, no. 9, pp. 152-159.

Coad, P., de Luca, J. & Lefebvre, E. 1999, *Java Modelling Colour with UML: Enterprise Components and Process,* Upper Saddle River, NJ, USA: Prentice Hall PTR Press.

Cook, J.E. & Wolf, A.L. 1996, *Process discovery and validation through event-data analysis*, University of Colorado at Boulder, Boulder, CO, Technical Report. CU-CS-817-96.

Corcho, O., Fernandez-Lopez, M., Gomez-Perez, A. (2003) 'Methodologies, Tools and Languages for Building Ontologies: Where is their Meeting Point?', *Data & Knowledge Engineering*, vol. 46, no. 1, pp. 41-64.

Crowe, M.K. 1996, "Information and business process", *Systemic Practice and Action Research,* vol. 9, no. 3, pp. 263-272.

Curtis, B., Kellner, M.I. & Over, J. 1992, "Process modelling", *Communications of the ACM,* vol. 35, no. 9, pp. 75-90.

Daga, A., de Cesare, S., Lycett, M. & and Partridge, C. 2005, "An Ontological Approach for Recovering Legacy Business Content", *In Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS)*, January 3-6, Los Alamitos, California, IEEE Computer Society Press, pp. 224.

Davenport, T.H. 1993, *Process Innovation: Reengineering Work Through Information Technology,* Boston, MA, USA: Harvard Business School Press.

Decreus, K. & Poels, G. 2009, "Mapping semantically enriched Formal Tropos to business process models", *Proceedings of the 24th Annual ACM Symposium on Applied Computing,* March 9-12, New York, NY, USA, pp. 371-376.

de Cesare, S., Lycett, M. & Paul, R. 2003, "Actor Perception in Business Use Case Modelling", *Communications of the AIS*, vol. 12, pp. 223-241.

di Dio, G. 2007, "ARWOPS: A Framework for Searching Workflow Patterns Candidate to be Reused", *Second International Conference on Internet and Web Applications and Services (ICIW)*: IEEE CNF, May 13-19, Mauritius, pp. 33.

Dietz, J.L.G. 2006, *Enterprise ontology: theory and methodology,* Secaucus, NJ: Springer-Verlag, New York, Inc.

Earl, M. 1994, "The New and Old of Business Process Redesign", *Journal of Strategic Information Systems,* vol. 3, no. 1, pp. 5-22.

Eriksson, H. & Penker, M. 2000, *Business modelling with UML: business patterns at work*, New York, Chichester: John Wiley & Sons.

Foreman, J. 1996, "Product Line Based Software Development- Significant Results, Future Challenges", *Software Technology Conference*, Salt Lake City, UT, United State, April 23.

Fowler, M. 1997, *Analysis patterns: reusable object models*, Menlo Park, Calif: Addison Wesley.

Fox, M.S. and Gruninger, M., 1994, "Ontologies for Enterprise Integration", *Proceedings of the 2nd Conference on Cooperative Information Systems*, June 24-27, Toronto, Ontario, Canada.

Gamma, E., Helm, R., Johnson, R. & Vlissides, J. 1995, *Design patterns: elements of reusable object-* oriented software, Reading, Mass: Addison-Wesley.

Geerts, G.L. & McCarthy, W.E. 2002, "An ontological analysis of the economic primitives of the extended-REA enterprise information architecture", International Journal of Accounting Information Systems, vol. 3, no. 1, pp. 1-16.

Goedertier, S., Martens, D., Baesens, B., Haesen, R. & Vanthienen, J. 2008, "Process mining as first-order classification learning on logs with negative events", *Business Process Management workshop*, Lecture Notes in Computer Science, Springer, Heidelberg, vol. 4928, pp. 42-5.

Goedertier, S. & Vanthienen, J. 2007a, "Declarative process modelling with business vocabulary and business rules", *On the Move to Meaningful Internet Systems: OTM Workshops*, Lecture Notes in Computer Science, Springer, Heidelberg, vol. 4805, pp. 603-612.

Greco, G., Guzzo, A., Pontieri, L., 2005, " Mining Hierarchies of Models: From Abstract Views to Concrete Specifications". In: van der Aalst, W.M.P., Benatallah, B., Casati, F., Curbera, F. (eds.) *Business Process Management workshop*, Lecture Notes in Computer Science, Springer, Heidelberg, pp. 32–47.

Green, M. 1996, "A framework for real-time rendering in virtual reality", *ACM Symposium on Virtual Reality Software and Technology*, (VRST), July 1–4, Hong Kong, pp. 3-10.

Green, P. & Rosemann, M. 2004, "Applying ontologies to business and systems modelling techniques and perspectives: Lessons learned", *Journal of Database Management,* vol. 15, no. 2, pp. 105-117.

Green, P. & Rosemann, M. 2000, "Integrated process modelling: An ontological evaluation", *Information Systems,* vol. 25, no. 2, pp. 73-87.

Green, P., Rosemann, M., and Indulska, M. 2005, "Ontological Evaluation of Enterprise Systems Interoperability Using ebXML," *IEEE Transactions on Knowledge and Data Engineering,* vol. 17, no. 5, pp. 713-725.

Gruber, T.R. 1995, "Toward principles for the design of ontologies used for knowledge sharing", *International Journal of Human Computer Studies,* vol. 43, pp. 907-928.

Gruber, T.R. 1993, "A translation approach to portable ontology specifications", *Knowledge Acquisition,* vol. 5, no. 2, pp. 199-220.

Guarino, N. 1998, "Formal Ontology and Information Systems", *Proceedings of FOIS'98,* Trento, Italy, June 6-8: Amsterdam, IOS Press, pp. 3-15.

Gzara, L., Rieu, D. & Tollenaere, M. 2000, "Patterns Approach to Product Information Systems Engineering", *Requirements Engineering,* vol. 5, no. 3, pp. 157-179.

Hammer, M. & Champy, J. 1993, *Reengineering the corporation: a manifesto for business revolution,* London, UK: Brealey Publishing Press.

Havey, M. 2005, *Essential business process modelling,* Cambridge, UK: O'Reilly Press.

Hay, D.C. 1996, *Data model patterns: conventions of thought,* New York, USA: Dorset House Publisher Press.

Hepp, M., Leymann, F., Domingue, J., Wahler, A. & Fensel, D. 2005, "Semantic business process management: A vision towards using semantic web services for business process management", *IEEE International Conference on e-Business Engineering (ICEBE)*, October 18-20, Beijing, China, pp. 535.

Hepp, M. & Roman, D. 2007, "An ontology framework for semantic business process management", *Proceedings of Wirtschaftsinformatik* Citeseer, February 28 - March 2, Karlsruhe, pp. 1-18.

Hevner, A.R., March, S.T., Park, J. & Ram, S. 2004, "Design Science in Information Systems Research", *MIS Quarterly,* vol. 28, no. 1, pp. 75.

Hruby, P., Kiehn, J. and Vibe Scheller, C., 2006, *Model-Driven Design Using Business Patterns*. Springer-Verlag Berlin Heidelberg: Springer Verlag Press.

Indulska, M., Green, P., Recker, J., & Rosemann, M., 2009, "Business Process Modelling: Perceived Benefits", *Proceeding in Conceptual Modelling – ER*, Lecture Notes in Computer Science, Springer, Heidelberg, vol. 5829, pp. 458-471.

Jacobson, I., Ericsson, M. & Jacobson, A. 1995, *The object advantage: business process reengineering with object technology,* Wokingham: Addison-Wesley.

Jurisica I, Mylopoulos J, Yu E., 2004, "Ontologies for knowledge management: an information systems perspective", *Journal of Knowledge and Information Systems*, vol. 6, pp. 380–401.

Kaisler, S.H. 2005, *Software paradigms*, Hoboken, N.J., USA: John Wiley & Sons.

Kaschek, R.; Rinderer, T.; Wiltsche, M., 1998, "Towards Reuse Support at the Business Process Meta Level". *Proceeding of the International Conference on the Development and Reengineering of Information Systems*, Ljubljana, October 1998, pp. 5-9.

Keen, C. & Lakos, C. 1996, "Analysis of the Design Constructs Required in Process Modelling", *Proceedings of the International Conference on Software Engineering: Education and Practice*, pp. 434.

Koehler, J., Hauser, R., Sendall, S., and Wahler, M., 2005, “Declarative Techniques for Model-Driven Business Process Integration”, *IBM Systems Journal,* vol. 44, no. 1, pp. 47-65.

Kuechler W, Vaishnavi V and Petter S, 2005, “The aggregate general design cycle as a perspective on the evolution of computing communities of interest”, *Computing Letters,* vol. 1, no. 3, pp. 123–128.

Lee, C., Jung, S., Lee, G.G., 2008, “Robust dialog management with n-best hypotheses using dialog example and agenda”, *In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Ohio, United State, June 23-30, pp. 630-637.

Lin, Y. 2008, *Semantic Annotation for Process Models*, PhD Thesis, Norwegian University of Science and Technology, Trondheim.

Lin, Y. and Straunskas, D., 2005, “Ontology-based semantic annotation of process templates for reuse”. In Proceeding of 10th CAiSE/IFIP8.1/EUNO, International Workshop on Evaluation of Modeling Methods in System Analysis and Design (EMMSAD), Porta, Portugal, pp. 31–37.

Ludewig, J. 2003, "Models in software engineering–an introduction", *Journal of Software and Systems Modelling,* vol. 2, no. 1, pp. 5-14.

Luo, W. & Tung, Y.A. 1999, "A framework for selecting business process modelling methods", *Industrial Management and Data Systems,* vol. 99, no. 7, pp. 312-319.

Lautenbacher, F. and Bauer, B., 2006, “Semantic Reference and Business Process Modelling Enables an Automatic Synthesis”, Hinkelmann K et al (Hrsg) Proceedings of Workshop SBPM at ESWC06, Budva, pp. 89–100.

McBride, B. 2003, “The resource description framework (RDF) and its vocabulary description language RDFS”. In: S. Staab, R. Studer (eds.), *The Handbook on Ontologies in Information Systems*, Springer, Berlin, pp. 51-66.

Malone, T., Crowston, K., Lee, J. & Pentland, B. 1993, "Tools for inventing organizations: toward a handbook of organizational processes", In: Malone et al., *Toward a handbook of organizational processes*, MIT Press, pp. 72.

Malone, T., Crowston, K. & Herman, G. 2003, *Organizing Business Knowledge: The MIT Process Handbook,* Cambridge, Mass, London: MIT Press.

March, S.T. & Smith, G.F. 1995, "Design and natural science research on information technology", *Decision Support Systems,* vol. 15, no. 4, pp. 251-266.

Mendling, J., 2008, “Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness”, Lecture Notes in Business Information Processing, Vol. 6, pp. 103-133.

Morgan, T. 2007, "Business Process Modelling and ORM", Proceedings of the OTM confederated international conference on On the move to meaningful internet systems, Vilamoura, Portugal *Lecture Notes in Computer Science,* Berlin, Heidelberg, Springer-Verlag, vol. 4805, pp. 581.

Morgan, T. 2002, *Business rules and information systems: aligning IT with business goals,* Prentice Hall*:* Addison-Wesley Professional.

Nelson, K. and Nelson, H., 2003, "The Need for a Strategic Ontology," in *Proceedings of the MIS (Management Information Systems) Quarterly Special Issue Workshop on Standard Making: A Critical Research Frontier for Information Systems*, Seattle, December 12--14, pp. 361--371.

Noy, N. F., McGuinness, D., 2001, *Ontology Development 101: A Guide to creating your first Ontology*, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880.

Nunamaker Jr, J., Chen, M. & Purdin 1990/91, "Systems Development in Information Systems Research", *Journal of Management Information Systems (JMIS),* vol. 7, no. 3, pp. 89-106.

O'Callaghan, A. 1999, "Migrating large-scale legacy systems to component-based and object technology: The evolution of a pattern language", *Communications of the Association for Information Systems:* Vol. 2, Article 3.

OMG., 2009, "Business Process Modelling Notation. *V1.2", OMG Available Specification*, *formal/2009-01-03*, 316.

Ould, M.A. 1995, *Business processes: modelling and analysis for re-engineering and improvement,* West Sussex, England: John Wiley and Sons.

Ould, M.A. 2006, *Business process management: a rigorous approach,* Tampa, FL USA: Meghan-Kiffer Press, British Computer Society.

Parsons, J. & Wand, Y. 1997, "Choosing classes in conceptual modelling", *Communications of the ACM,* vol. 40, no. 6, pp. 63-69.

Partridge, C. 1996, *Business Objects Re-Engineering For Re-Use,* 1st edn, England, UK: Butterworth-Heinemann College.

Pedrinaci, C., Domingue, J. & de Medeiros, A.K.A. 2008, "A Core Ontology for Business Process Analysis", Proceeding in European Conference on the Semantic Web, *Lecture Notes in Computer Science,* Berlin, Heidelberg, Springer-Verlag, vol. 5021, pp. 49.

Pesic, M. & van der, A. 2006, "A declarative approach for flexible business processes management", *Business Process Management workshop, Lecture Notes in Computer Science,* Berlin, Heidelberg, Springer-Verlag, vol. 4103, pp. 169-180.

Pesic, M.M. 2008, *Constraint-based workflow management systems: shifting control to users*, Published PhD, Technische Universiteit, Eindhoven.

Pesic, M., Schonenberg, M.H., Sidorova, N., and van der Aalst. W.M.P., 2007, "Constraint-Based Workflow Models: Change Made Easy". In R. Meersman and Z. Tari, editors, Proceedings of the 15th International Conference on Cooperative Information Systems (CoopIS), Lecture Notes in Computer Science, Heidelberg, Springer-Verlag, vol. 4803, pp. 77–94.

Petri, C. 1962, *Kommunikation mit Automaten*, PhD thesis, University Bonn, Germany.

Prieto-Daiz, R. 1990, "Domain analysis: an introduction", *SIGSOFT Software Engineering Notes, Association for Computing Machinery (ACM),* vol. 15, no. 2, pp. 47-54.

Purao, S. 2002, "Design Research in the Technology of Information Systems: Truth or Dare", *Information systems*, Georgia State University Department of CIS Working Paper, Atlanta.

Recker, J., & Indulska, M., 2007, "An Ontology-Based Evaluation of Process Modelling with Petri Nets", *IBIS - International Journal of Interoperability in Business Information Systems*, vol. 1, no.2, pp. 45-64.

Recker, J., Rosemann, M., Indulska, M. & Green, P. 2006, *Business process modelling: A maturing discipline*, BPM Centre Report BPM-06-20, BPMcenter.org.

Russell, N., ter Hofstede, A.H.M., Edmond, D. & van der, A. 2005, "Workflow Data Patterns", *Proceeding of 24th International Conference on Conceptual Modelling (ER),* Klagenfurt, Austria, October 24-28*,* vol. 3716, pp. 353–368.

Russell, N., ter Hofstede, A.H.M., Edmond, D. & van der, A. 2004, "Workflow Resource Patterns", *BETA Working Paper Series*, Technical Report WP 127, Eindhoven University of Technology*,* vol. 127, pp. 1-66.

Sadiq, S. W., Orlowska, M. E., and Sadiq, W., 2005, "Specification and validation of process constraints for flexible workflows", *Information Systems*, vol. 30, no. 5, pp. 349– 378.

Schimm, G. 2000, "Generic linear business process modelling", Conceptual Modelling for E-Business and the Web, vol. 120, pp. 31-39.

Sharman, R., Kishore, R. & Ramesh, R. 2007, *Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems,* 1st edn, Integrated Series in Information Systems, Springer, US, Vol. 14.

Soffer, P., Golany, B., Dori B. and Wand, Y., 2001, "Modelling Off-the-Shelf Information Systems Requirements: An Ontological Approach", *Requirements Engineering,* vol. 6, no. 3, pp. 183-198.

Sider, T. 2003, *Four-Dimensionalism: An Ontology of Persistence and Time,* Oxford, UK: Clarendon Press.

Simon, H.A. 1996 *The Sciences of the Artificial,* 3rd edn, Cambridge (MA): the MIT Press (Original edition 1969).

Smith, B. 2003, "Ontology ", In Floridi L. (ed.), *Blackwell guide to the philosophy of computing and information,* Blackwell, Oxford, pp.155-166.

Staab, S. and Studer, R., 2010, *The Handbook on Ontologies in Information Systems*, Springer Dordrecht Heidelberg, London: Springer Press.

Thom, L., Lau, J., Iochpe, C., Reichert, M. 2007, "Workflow Patterns for Business Process Modelling", n: *8th Workshop on Business Process Modeling, Development, and Support in conjunction with CAISE*, Trondheim, Norway, June 12-14, international Conference on Advanced Information Systems Engineering.

Thomas, O. & Fellmann, M. 2006, "Semantic event-driven process chains", *In Proceeding Workshop on Semantic Business Process and Product Lifecycle Management (SBPM),* June, pp. 64–75.

Thomas, O.; *Fellmann, M.*, 2009, "Semantic Process Modelling - Design and Implementation of an Ontology-Based Representation of Business Processes". In: *Business & Information Systems Engineering,* vol. 1, no. 6, pp. 438-451.

Vaishnavi, V. & Kuechler, W. 2004, "Design research in information systems", *MIS Quarterly,* vol. 28, no. 1, pp. 75-105.

van der Aalst, W. & van Dongen, B. 2002, "Discovering Workflow Performance Models from Timed Logs", in: Y. Han, S. Tai, D. Wikarski (Eds.), *International Conference on Engineering and Deployment of Cooperative Information Systems* (EDCIS), *Lecture Notes in Computer Science*, vol. 2480, Springer-Verlag, Berlin, 2002, pp. 45–63

van der Aalst, W., Weijters, T. & Maruster, L. 2004, "Workflow mining: Discovering process models from event logs", *IEEE Transactions on Knowledge and Data Engineering,* Vol. 16, no. 9, pp. 1128-1142.

van der Aalst, W.M.P. 2007, "Exploring the CSCW spectrum using process mining", *Advanced Engineering Informatics,* vol. 21, no. 2, pp. 191-199.

van der Aalst, W.M.P. & Pesic, M. 2006, "DecSerFlow: Towards a truly declarative service flow language", *Lecture Notes in Computer Science*, Springer-Verlag, Heidelberg, vol. 4184, pp. 1-23.

van der Aalst, W. 1999, "Formalization and verification of event-driven process chains", *Information and Software Technology,* vol. 41, no. 10, pp. 639-650.

van der, A., A.H.M.Ter, H. & Kiepuszewski, B. 2000, "Advanced Workflow Patterns", *7th International Conference on Cooperative Information Systems (CoopIS)*, ed. O. Etzion en P. Scheuermann, *Lecture Notes in Computer Science*, Springer-Verlag, Heidelberg, Berlin, pp. 18.

van der, A., A.H.M.Ter, H., Kiepuszewski, B. & Barros, A.P. 2003, "Workflow Patterns", *QUT Technical report, FIT-TR-2002-02, Queensland University of Technology, Brisbane, 2002 (also see http://www.tm.tue.nl/it/research/patterns). To appear in Distributed and Parallel Databases*

van der, A. & Hee, K. 2002, *Workflow management: models, methods, and systems,* Cambridge, MA, USA: MIT Press.

van Dongen, B.F. 2007, *Process Mining and Verification*, published PhD, University Microfilms International, USA.

van Dongen, B.F. & van der Aalst, W.M.P. 2005, "A meta model for process mining data", In: J. Casto, E. Teniente (Eds.), Proceedings of the CAiSE'05 Workshops (EMOI-INTEROP Workshop), vol. 2, FEUP, Porto, Portugal, pp. 309–320.

Veryard, R. 2000, *Component-based Business: Business Patterns*, Great Britain, UK: Athenaeum Press Ltd.

Wand, Y., Monarchi, D.E., Parsons, J. & Woo, C.C. 1995a, "Theoretical foundations for conceptual modelling in information systems development", *Decision Support Systems,* vol. 15, no. 4, pp. 285-304.

Wand, Y. & Weber, R. 1990, "An ontological model of an information system", *Software Engineering, IEEE Transactions,* vol. 16, no. 11, pp. 1282-1292.

Weber, B., Reijers, H.A., Zugal, S. & Wild, W. 2009, "The Declarative Approach to Business Process Execution: An Empirical Test", ed. P. van Eck, J. Gordijn, and R. Wieringa, Berlin, Heidelberg, Springer-Verlag, pp. 470–485.

Weber, R. 1997, *Ontological foundations of information systems,* Coopers & Lybrand, Accounting Association of Australia and New Zealand.

Weber, R. & Zhang, Y. 1996, "An analytical evaluation of NIAM's grammar for conceptual schema diagrams", *Information Systems Journal,* vol. 6, no. 2, pp. 147-170.

Wetzstein, B., Ma, Z., Filipowska, A., Kaczmarek, M., Bhiri, S., Losada, S., Lopez-Cob, J. & Cicurel, L. 2007, "Semantic Business Process Management: A Lifecycle Based Requirements Analysis", *Semantic Business Process and Product Lifecycle Management 07*, ed. 3rd European Semantic Web Conference, CEUR-WS.org, Innsbruck, Austria, pp. 1.

White, S. A., 2004, "Introduction to BPMN"*,* IBM Cooperation.

Winter, R. 2008, "Design Science Research in Europe", *European Journal of Information Systems*, vol. 17, no.5, pp. 470-475.

Yang, N. & Zhang, Y. 2005, "The application of reusable pattern-based business modelling in equipment management", *Proceedings of the 7th international conference on Electronic commerce,* Xi'an, China, August 15-17, pp. 490.

Zlatkin, S. & Kaschek, R. 2005, "Towards Amplifying Business Process Reuse", *Perspectives in Conceptual Modelling,* Lecture Notes in Computer Science, vol. 3770, pp. 364-374.

Appendix A **Business Process Models**

Preparation of Legacy Assets - Reverse Engineering of Legacy Assets (BP Models)

1. Financial Services Domain

- **Retail Banking**
  - Define Retail Banking Client

Define Client Information	
Terms	**BP Elements**
Client Information Created or Maintained	Event
Set up Client Type	Activity
Add Client Details Form	Activity
Complete Client Employment Details Form	Activity
Complete Client Financial Details	Activity
Set up Maintainable Client Details	Activity
Attach Product Template to Client	Activity
New client	Precondition
Client	Resources
Retail Banking Assistant	Resources
Client Details Form	Resources
Client Maintenance Form	Resources
Client Employment Form	Resources
Client Financial Form	Resources
Client Information Report	Resources

  - Information Define Retail Banking Product Templates

Define Product Templates	
Terms	**BPElements**
NewRetailBankingProduct definition	Event
Select Product Category	Activity
Create Product Template	Activity
Set up Fixed Rules	Activity
Record Product Amendable Details	Activity
Set up Product Interest Condition	Activity
Complete Product Interest Type Form	Activity
Set up Product Status	Activity
Produce Product Report	Activity
New Market and Customer Need New Product	Precondition
Product to meet customer needs	Post condition
Retail Banking Admin	Resources
Product Interest Type Form	Resources
Product Interest Condition Form	Resources
Product Template Form	Resources
Fix Rule Guide	Resources
Product Report	Resources

  - Open Retail Banking Accounts

Open Retail Banking Accounts	
Terms	**BP Elements**
New Client Account Defined	Event
Select Client information	Activity
Select Product Template	Activity
Fill in Account Details	Activity
Review Product Interest Condition	Activity
Set up attached Product interest Condition	Activity
Set Account Status	Activity
Produce Account Report	Activity
Supply Bank Statement	Activity
Client account for retail banking created first	Procondition
Account Avaliable for customer	Post condition
Client	Resources
Retail Banking Assistant	Resources
Client Information Form	Resources
R Banking Account Form	Resources
Product Interest Condition Form	Resources
Product Template Form	Resources
Retail Banking Report	Resources
Bank Statement	Resources

- Issue Retail Client Cards

IssueRetailClientCards	
Terms	**BP Elements**
Card Expiry Approached	Event
Fill in Card Template Details	Task
Define Institution Card Parameters	Task
Add an Issue of Card	Task
Order Issue of Card	Task
Attach Card to Account	Task
Send Letter Card to Client	Task
One set of parameter for each bank	Precondition
once the card issue you can inquire on it	Post condition
Retail Banking Admin	Resources
Card Template Form	Resources
Card Letter	Resources

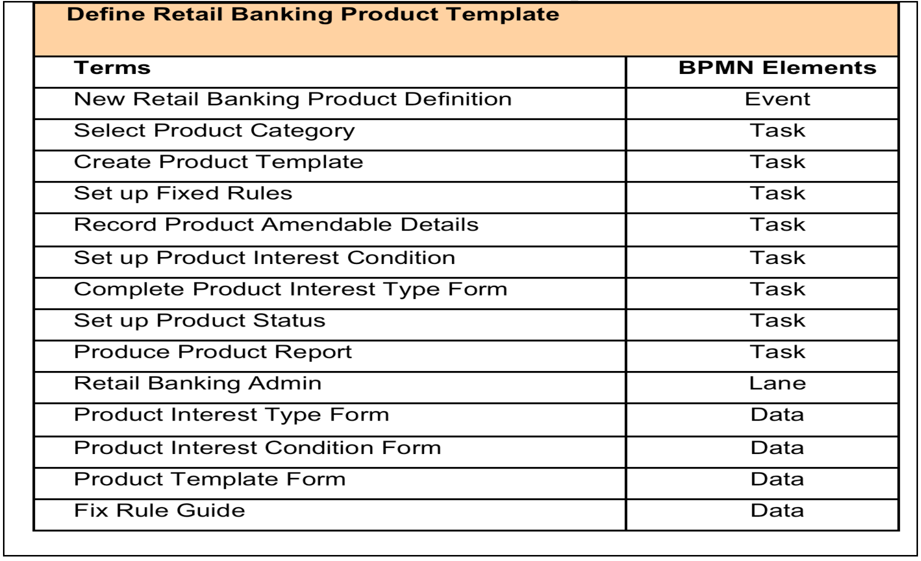
- Define Cheques For Retail Banking

Define Cheques For Retail Banking	
Terms	**BP Elements**
New Retail Account opened or renewed	Event
Select Client information	Activity
Fill in Cheque book Details	Activity
Choose Vault Code	Activity
Set Cheque Type	Activity
Set Cheque Status	Activity
Produce Client Cheque Book Report	Activity
Vault cheque code set	Precondition
Maintain cheque book/ Inqury	Postcondition
Retail Banking Admin	Resources
Cheque Book Detail Form	Resources
Client Cheque Report	Resources

- **Insurance**
  - Define Insurance Client Details

Define Insurance Client Details	
Terms	**BP Elements**
Client Information Created or Maintained	Event
Set up Client Name Details	Activity
Set up Client Address Details	Activity
Set up Financial Details	Activity
Complete Client Employment Details	Activity
Define Underwriting Details	Activity
Attach an Agent to Client name	Activity
Attach Product Template to Client	Activity
Produce Client Reports	Activity
New or exist clien	Precondition
Client has direct and indirect relationship with other system.	Post condition
Client	Resources
Insurance Assistance	Resources
Client Details Form	Resources
Client Maintenance Form	Resources
Client Employment Form	Resources
Client Financial Form	Resources
Client Details Report	Resources

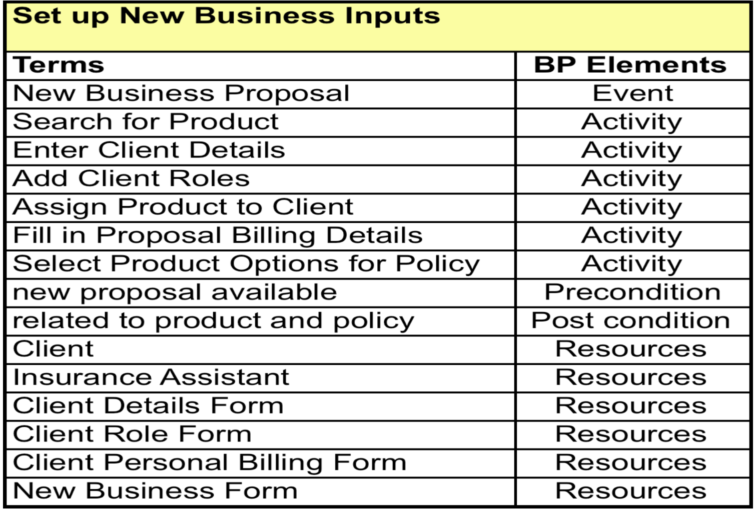
- Define Insurance Product Templates



Define Retail Banking Product Template

Terms	BPMN Elements
New Retail Banking Product Definition	Event
Select Product Category	Task
Create Product Template	Task
Set up Fixed Rules	Task
Record Product Amendable Details	Task
Set up Product Interest Condition	Task
Complete Product Interest Type Form	Task
Set up Product Status	Task
Produce Product Report	Task
Retail Banking Admin	Lane
Product Interest Type Form	Data
Product Interest Condition Form	Data
Product Template Form	Data
Fix Rule Guide	Data

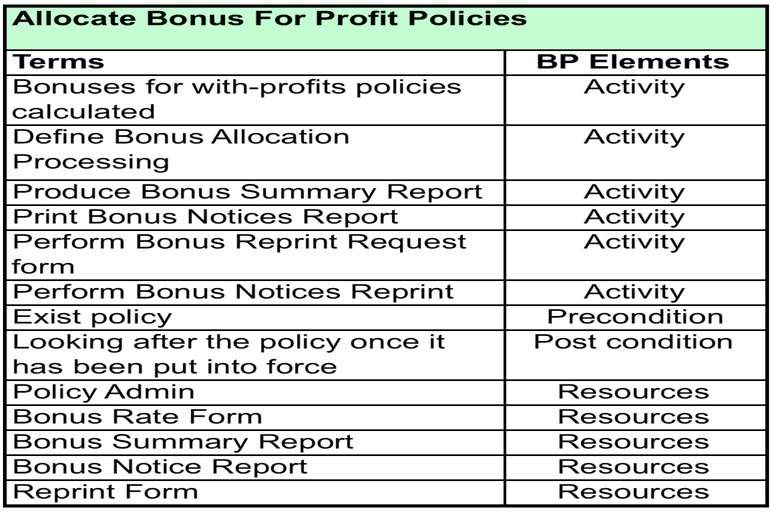
- Set up New Business Inputs



Set up New Business Inputs

Terms	BP Elements
New Business Proposal	Event
Search for Product	Activity
Enter Client Details	Activity
Add Client Roles	Activity
Assign Product to Client	Activity
Fill in Proposal Billing Details	Activity
Select Product Options for Policy	Activity
new proposal available	Precondition
related to product and policy	Post condition
Client	Resources
Insurance Assistant	Resources
Client Details Form	Resources
Client Role Form	Resources
Client Personal Billing Form	Resources
New Business Form	Resources

- Allocate Bonus For Profit Policies



Allocate Bonus For Profit Policies

Terms	BP Elements
Bonuses for with-profits policies calculated	Activity
Define Bonus Allocation Processing	Activity
Produce Bonus Summary Report	Activity
Print Bonus Notices Report	Activity
Perform Bonus Reprint Request form	Activity
Perform Bonus Notices Reprint	Activity
Exist policy	Precondition
Looking after the policy once it has been put into force	Post condition
Policy Admin	Resources
Bonus Rate Form	Resources
Bonus Summary Report	Resources
Bonus Notice Report	Resources
Reprint Form	Resources

- Define Financial Administration Processes

Define Financial Administration Processes	
Terms	**BP Elements**
New Business Financial calculation	Event
Set up New Business Financial control	Activity
Generate Regular Premium	Activity
Start Premium Collection	Activity
Match Cash Received to Premium Due	Activity
Release Premium Paid Commission	Activity
Record Unit Allocation Trigger	Activity
Complete Record of Financial Transaction Form	Activity
Create General Ledger Transactions	Activity
Exist finance	Precondition
direct and indirect relationship policy, product and clients	Post condition
Policy Admin	Lane
Financial Transaction Form	Data
General Ledger Transactions Form	Data

- **Mortgages**
  - Set up Nominal Accounts

Set up Nominal Accounts	
Terms	**BP Elements**
New Mortgage Account installation	Event
Set up Nominal Account Code Form	Activity
Define sub and Major Account Code	Activity
Allocate Account Type	Activity
Ensure account Indicator set correctly	Activity
Attach GL Code to identify Transaction	Activity
Produce Nominal Account Reports	Activity
Exist account	Precondition
can use for transaction	Post condition
Mortgage Admin	Resources
Nominal Account Form	Resources
Nominal Account Report	Resources

  - Define Bank Account Set up

Define Bank Account Set up	
Terms	**BP Elements**
New Account	Event
Set up Bank Branch Details	Activity
Define Account Code Identifier	Activity
Allocate Bank Transaction Record	Activity
Set Account Amendable Details	Activity
Produce Bank Account Reports	Activity
New market needs	Precondition
used for transaction	Postcondition
Client	Resources
Mortgage Admin	Resources
Bank Account Form	Resources
Bank Account Report	Resources

- Arrange Daily Returns Processing

Arrange Daily Returns Processing	
Terms	**BP Elements**
End of Day Calculation	Event
Produce the Daily Return Book	Activity
Summaries Branches Returns	Activity
Post Overall Cash Movement	Activity
Produce an analysis of Branch banking for the Day	Activity
Update Cash Figure Form	Activity
Marks the Branch Balance sheet Record	Activity
Produce Daily Returns Report	Activity
Account no. and Rate	Precondition
Enable transaction and daily update	postcondition
Mortgage Admin	Resources
Daily Return Book Form	Resources
Cash Figure Form	Resources
Branch Balance Sheet	Resources
Daily Returns Report	Resources

- Define Mortgage Products

Define Mortgage Products	
Terms	**BP Elements**
New Mortgage Product	Event
Create Product Shell	Activity
Define Product Specific Rules	Activity
Apply Product Processing Rules	Activity
Apply Interest Rate Rules	Activity
Define Product amendable Details	Activity
Set Mortgage Product Effective Date	Activity
Applicant need mortgage and satisfy rules	Precondition
Set or cancel mortgage	Post condition
Mortgage Admin	Resources
Product Shell	Resources
Product interest Rate Form	Resources
Interest Rate Guide	Resources
Mortgage Report	Resources

- Process Mortgage Applications

Process Mortgage Applications	
Terms	**BP Elements**
New Client request Mortgages	Event
Establish Basic Loan Application Form	Activity
Record Customer Details	Activity
Add Occupational Details	Activity
Record Financial Details	Activity
Issue Reference Request Letter	Activity
Hand in Mortgage User Guide	Activity
Issue Summary of all Reference request Letters	Activity
Record Property against Application	Activity
Initiates Dairy for certain Event	Activity
Set up Customer Inquiry and update condition	Activity
Record Receipt of the application	Activity
Define Cancellation Condition	Activity
Proces rues defined	Precondition
Effective date reached	Post condition
Customer	Resources
Mortgage Assistant	Resources
Loan Application Form	Resources
Receipt	Resources
Customer Details Form	Resources
Customer Occupational Form	Resources
Customer Financial Form	Resources

- Change Mortgage Interest Rate Process

Change Mortgage Interest Rate Process	
Terms	BP Elements
New Interest Rate - Market changed	Event
Enter new Interest Rate for Each Mortgage Product Code	Activity
Run Rate Change Report to Calculate the New Payment for the Effective Date	Activity
Run the Rate Change Report to Produce Notification to Customer Letter	Activity
Run the Report to bring the New Rate and Payment into Effect	Activity
Account no. and Rate	Precondition
New customer/ account type code/ interest rate applied/ catogeries setup	Postcondition
Customer	Resources
Mortgage Admin	Resources
Mortgage Interest Rate Form	Resources
New Payment Calculation Report	Resources
Customer Notification Report	Resources
New Rate and Payment Report	Resources

- Create Mortgage Account Types

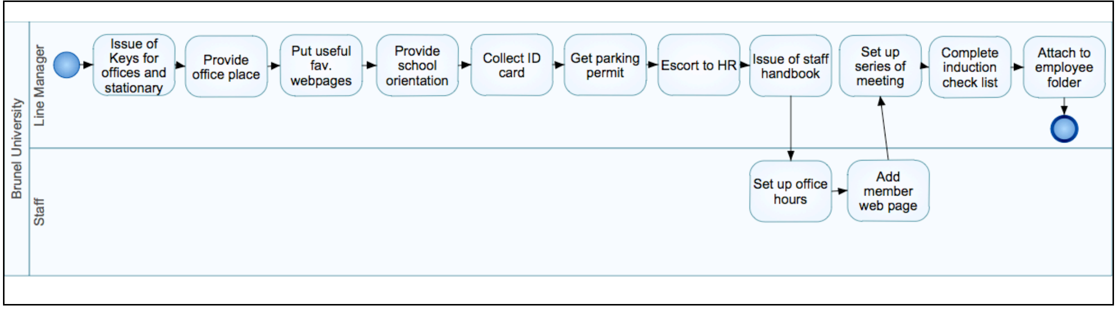
Create Mortgage Account Types	
Terms	BP Elements
New mortgage Account Opened	Event
Complete Account Type Details	Activity
Define Account Opening Categories	Activity
Apply Interest Rate Rules	Activity
Restrict Access to the Account Type at Specific Branch	Activity
Set Account Bonuses Details	Activity
Set Transaction and Charges Details	Activity
Print Account statement	Activity
Set up Account Type Status	Activity
Specify Transfer Rules for the account	Activity
Exist fund company	Precondition
Set up done	Postcondition
Customer	Resources
Mortgage Assistant	Resources
Mortgage Interest Rate Form	Resources
Bonuses Form	Resources
Interest Rate Guide	Resources
Mortgage Account Type Form	Resources
Account Opening Form	Resources

- Create Investment Accounts

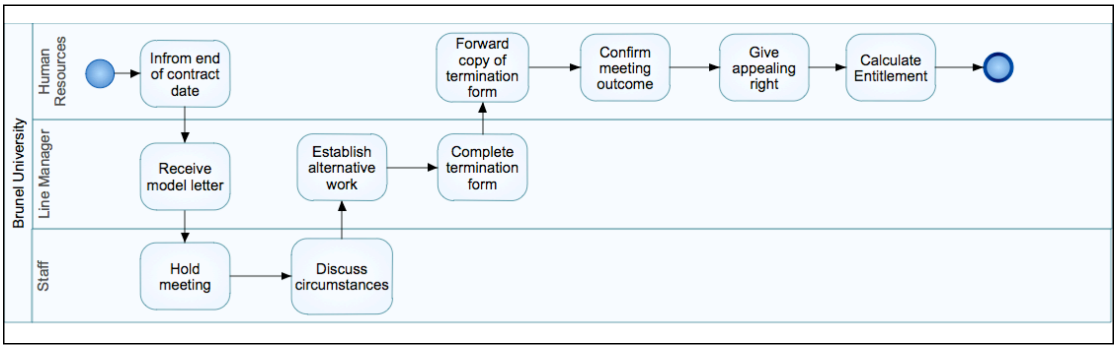
CreateInvestmentAccounts	
Terms	BP Elements
New investment Customer	Event
Complete Customer Details	Activity
Complete Occupational Details	Activity
Add Customer Financial Details	Activity
Check Credit Reference	Activity
Produce Receipt of Credit Reference	Activity
Attach Product to Customer	Activity
Identify Authorisation Link	Activity
Identify Beneficiaries Link	Activity
Set Customer self Certification	Activity
Set Account Status	Activity
Exist customer details	Precondition
Enable account rules	Post condition
Customer	Resources
Mortgage Assistant	Resources
Customer Details Form	Resources
Customer Product Form	Resources
Customer Occupational Form	Resources
Customer Financial Form	Resources
Customer Certification	Resources
Credit Receipt	Resources

2. Educational Domain

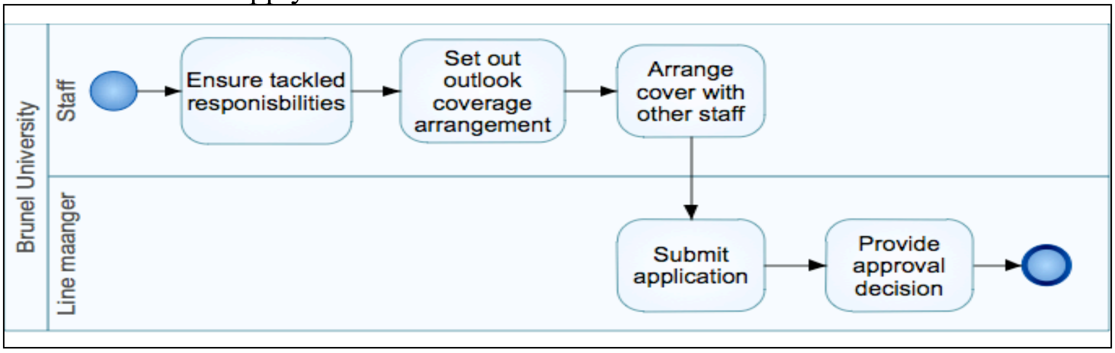
- **Brunel University Staff Handbook**
  - Arrange Staff Induction



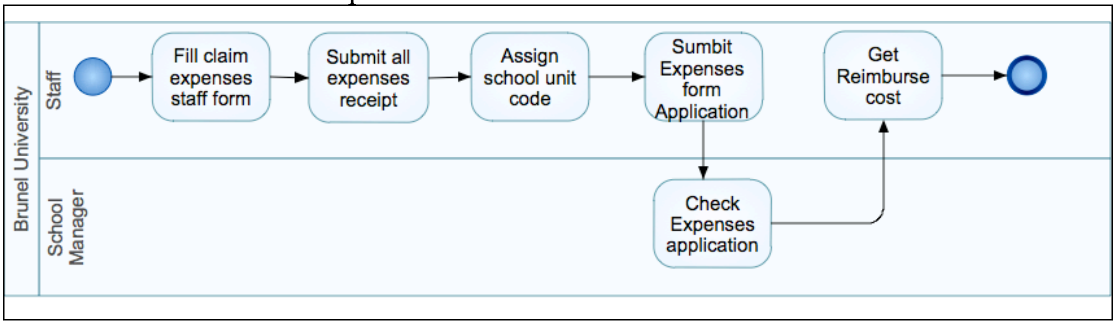
  - Terminate of fix term Contract



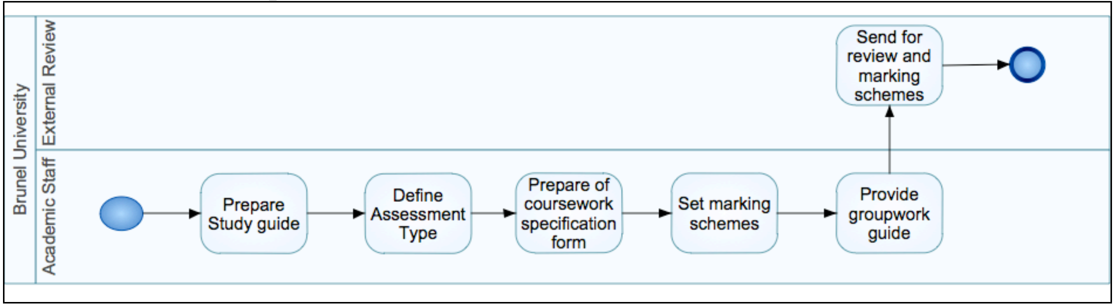
  - Apply for Leave


Brunel University
Staff
Line maanger
Ensure tackled responisbilities
Set out outlook coverage arrangement
Arrange cover with other staff
Submit application
Provide approval decision

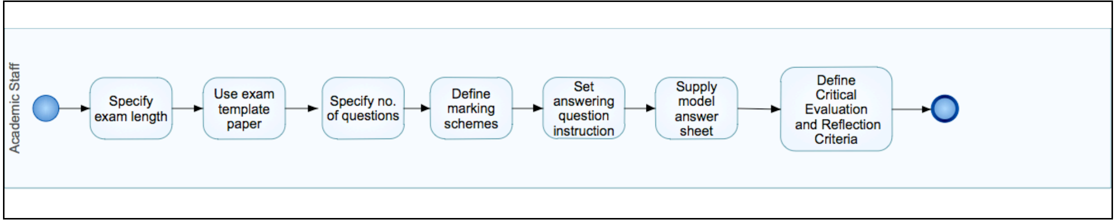
  - Claim Expenses


Brunel University
Staff
School Manager
Fill claim expenses staff form
Submit all expenses receipt
Assign school unit code
Sumbit Expenses form Application
Get Reimburse cost
Check Expenses application

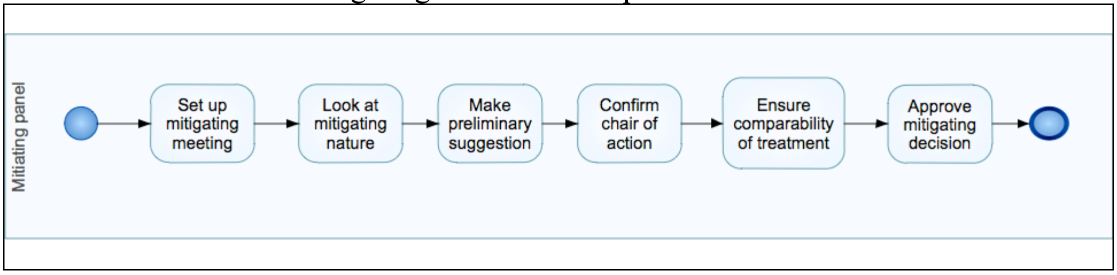
- Prepare Assessment Process



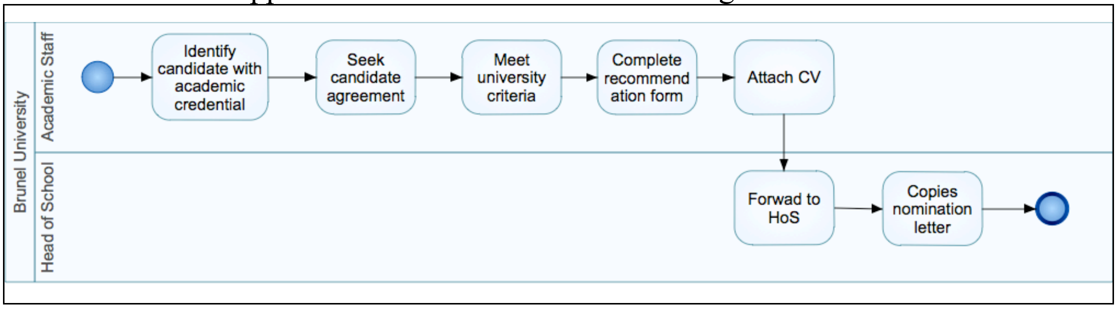
- Prepare Examination Paper


Academic Staff
Specify exam length
Use exam template paper
Specify no. of questions
Define marking schemes
Set answering question instruction
Supply model answer sheet
Define Critical Evaluation and Reflection Criteria

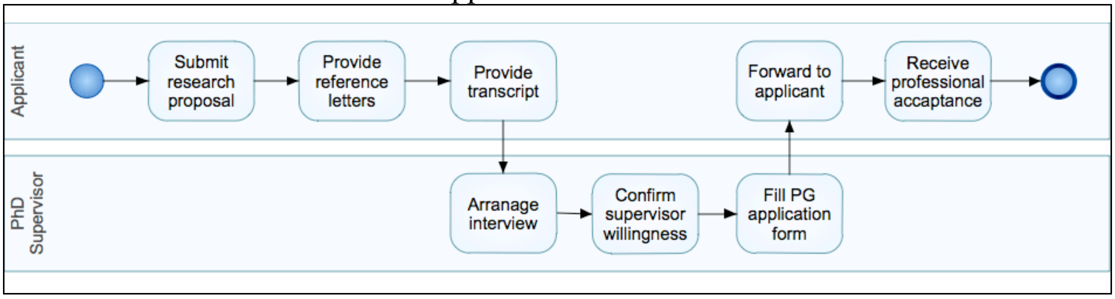
- Hold mitigating Circumstance panel


Mitiating panel
Set up mitigating meeting
Look at mitigating nature
Make preliminary suggestion
Confirm chair of action
Ensure comparability of treatment
Approve mitigating decision

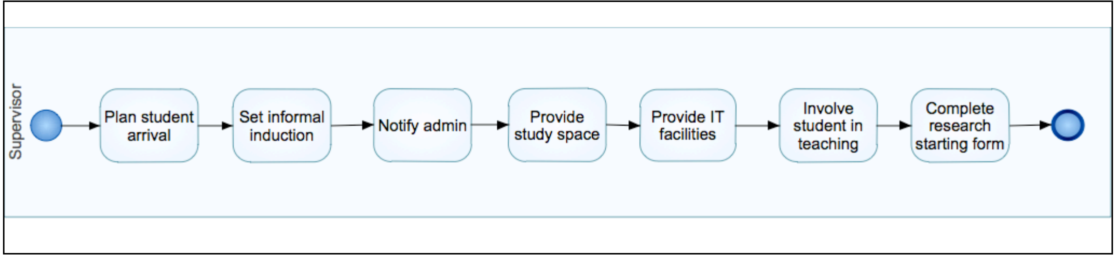
- Appoint of External Examiner for Taught course



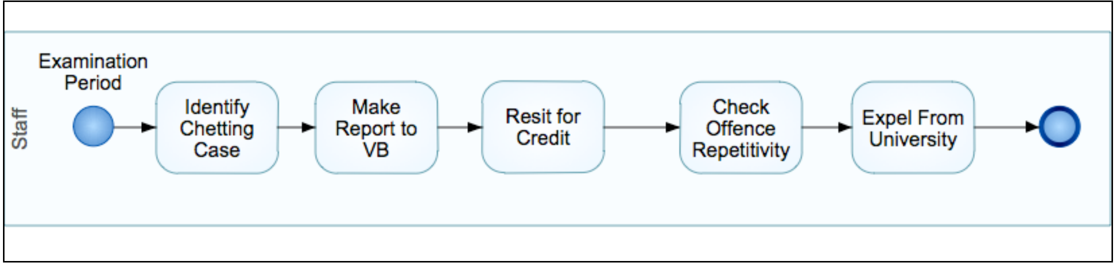
- Process of PGR application



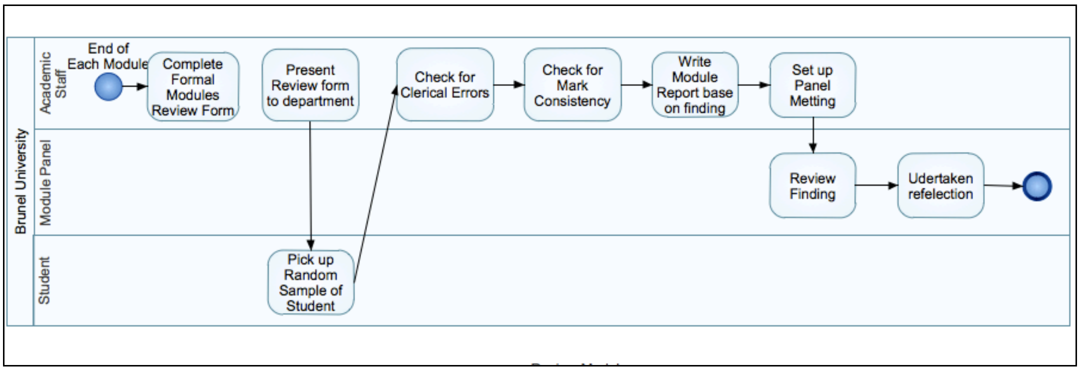
- Arrange PhD induction



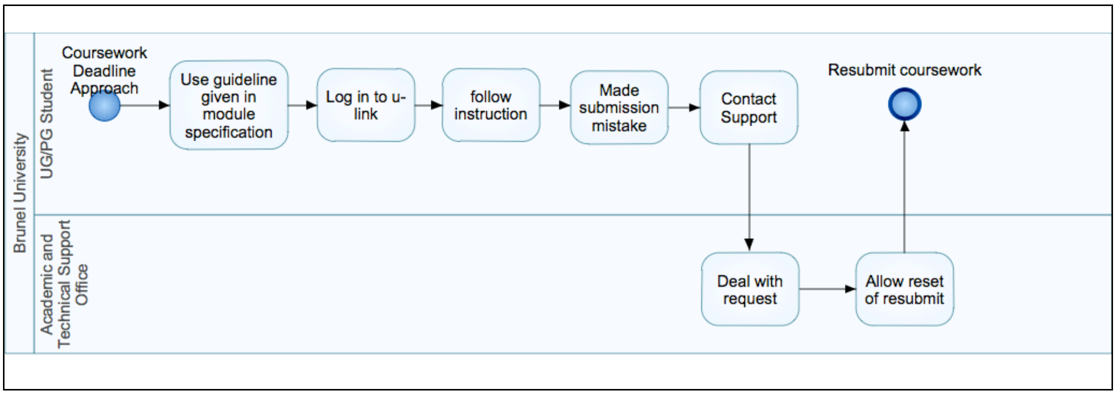
- **Brunel University Students Handbook**
  - Identify Cheating Case



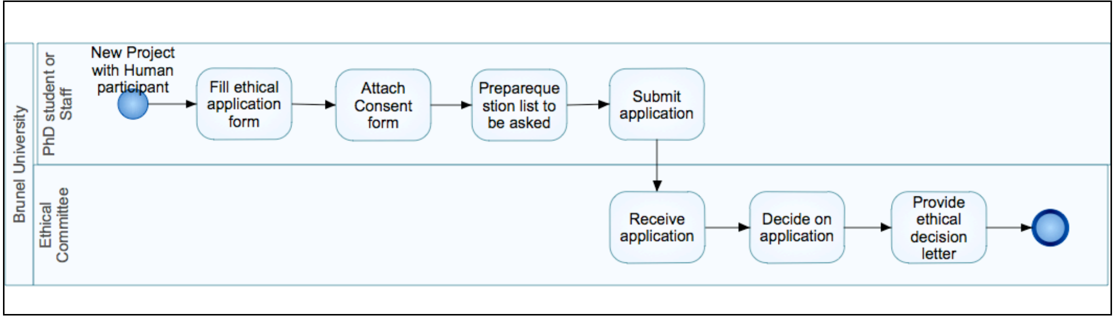
- Review Module Processes


Brunel University
Academic Staff
Module Panel
Student
End of Each Module
Complete Formal Modules Review Form
Present Review form to department
Check for Clerical Errors
Check for Mark Consistency
Write Module Report base on finding
Set up Panel Metting
Review Finding
Udertaken refelection
Pick up Random Sample of Student

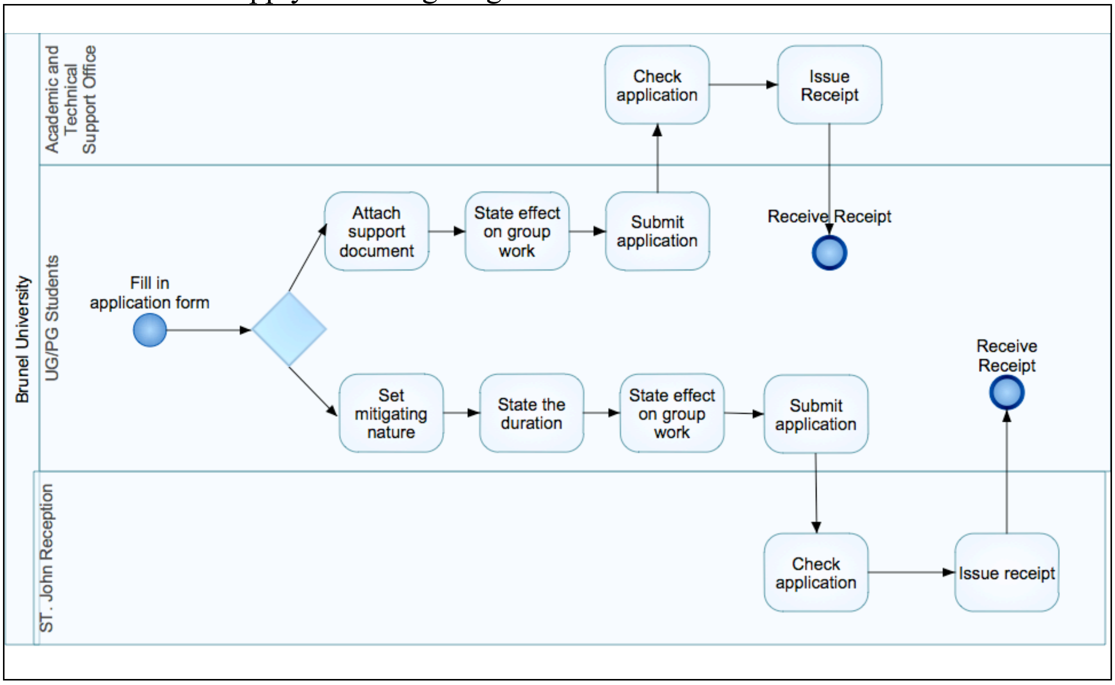
- Submit Coursework Processes



- Apply For Ethical Approval Students/Staff



- Apply For Mitigating Circumstances



Appendix B BP Ontology Models

Semantic Analysis of BP Models - Interpretation (BP Ontology Models)

1. Financial Services Domain

- **Retail Banking**
  - Define Retail Banking Client

DefineRetailBankingClientInformations

Terms	BPMN Elements	Interpretation	BPO Elements
Client Information Created or Maintained	Event	NewClientWillingToOpenRetailBankingAccounts	BP Initiating Event
Set up Client Type	Task	ChooseClientTypes	Activities
Add Client Details Form	Task	FillinRetailClientDetailForms	Activities
Complete Client Employment Details Form	Task	FillinRetailClientEmploymentForms	Activities
Complete Client Financial Details	Task	FillinRetailClientFinancialDetailForms	Activities
Set up Maintainable Client Details	Task	SetupRetailClientMaintainableDetails	Activities
Attach Product Template to Client	Task	ChoosePreferableRetailProductToClients	Activities
Produce Client Information Reports	Task	CheckProvidedInformationWithClients	Activities
		ProduceRetailClientInformationReport	Activities
		RetailBankingClientInformationCreated	BP Goal Event
Client	Lane	Clients	BP Participant
Retail Banking	Lane	RetailBankingAssistants	BP Participant
Client Details Form	Data	RetailClientDetailForms	Input
Client Maintenance Form	Data	Client Maintenance Forms	Input
Client Employment Form	Data	RetailClientEmploymentForms	Input
Client Financial Form	Data	RetailClientFinancialForms	Input
Client Information Report	Data	RetailClientInformationReports	Output

  - Information Define Retail Banking Product Templates

DefineRetailBankingProductTemplates

Terms	BPMN Elements	Interpretation	BPO Elements
NewRetailBankingProduct definition	Event	NewRetailBankingProductIntroductory	BP Initiating Event
Select Product Category	Task	AssignCategoryForTheNew RetailProducts	Activities
Create Product Template	Task	FillinRetailProductTemplate Forms	Activities
Set up Fixed Rules	Task	SelectFixedRulesForTheRetailProductFromRulesGuides	Activities
Record Product Amendable Details	Task	DefineWhichRetailProductDetailCanBeAmended	Activities
Set up Product Interest Condition	Task	AssignInterestConditionsFormsToRetailProducts	Activities
Complete Product Interest Type Form	Task	FillinRetailProductInterestTypeForms	Activities
Set up Product Status	Task	ChooseLifeStatusToLaunch RetailProducts	Activities
Produce Product Report	Task	ProduceRetailProductReports	Activities
		NewRetailBankingProductLaunchedToMarket	BP Goal Event
Retail Banking Admin	Lane	RetailBankingAdmins	BP Participants
Product Interest Type Form	Data	RetailProductInterestTypeForms	Input
Product Interest Condition Form	Data	RetailProductInterestConditionForms	Input
Product Template Form	Data	RetailProductTemplateForms	Input
Fix Rule Guide	Data	RetailProductFixRuleGuides	Input
Product Report	Data	RetailProductReports	Output

- Open Retail Banking Accounts

OpenRetailBankingAccounts			
Terms	**BPMN Elements**	**Interpretations**	**BPO Elements**
New Client Account Defined	Event	OpenNewRetailBankingAccountForExistingClients	BP Initiating Event
Select Client information	Task	SelectExistingRetailCustomerInformations	Activities
Select Product Template	Task	SelectRetailProductTemplateThatClientInterestedIn	Activities
Fill in Account Details	Task	FillinRetailAccountDetailForms	Activities
Review Product Interest Condition	Task	ChooseRetailProductInterestConditionForms	Activities
Set up attached Product interest Condition	Task	ReviewRetailProductInterestConditionForms	Activities
Set Account Status	Task	ChooseLifeStatusToMakeAccountExistToCustomerUse	Activities
Produce Account Report	Task	ProduceRetailAccountReports	Activities
Supply Bank Statement	Task	IssueBankStatements	Activities
		NewAccountOpeningCompleted	BP Goal Events
Client	Lane	Clients	BP Participants
Retail Banking Assistant	Lane	RetailBankingAssistants	BP Participants
Client Information Form	Data	Client Information Form	Input
R Banking Account Form	Data	RetailAccountDetailForms	Input
Product Interest Condition Form	Data	RetailInterestConditionForms	Input
Product Template Form	Data	Product Template Form	Input
Retail Banking Report	Data	RetailAccountReports	Output
Bank Statement	Data	BankStatements	Output

- Issue Retail Client Cards

IssueRetailClientCards			
Terms	**BPMN Elements**	**Interpretation**	**BPO Element**
Card Expiry Approached	Event	RetailCardExpiryReached	BP Initiating Event
Fill in Card Template Details	Task	FillinRetailCardTemplateForms	Activities
Define Institution Card Parameters	Task	SetupInstitutionCardParameters	Activities
Add an Issue of Card	Task	AssignIssueForTheRetailCards	Activities
Order Issue of Card	Task	MakeanOrderForTheIssuedCards	Activities
Attach Card to Account	Task	AttachTheNewCardForExistRetailClientAccounts	Activities
Send Letter Card to Client	Task	SendNewRetailCardLetters	Activities
		NewRetailCardReadyToBeUsed	BP Goal Event
Retail Banking Admin	Lane	RetailBankingAdmins	BP Participant
Card Template Form	Data	RetailCardTemplateForms	Input
Card Letter	Data	RetailCardCustomerLetters	Output

- Define Cheques For Retail Banking

DefineChequesForRetailBanking			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Retail Account opened or renewed	Event	SpecificRetailClientAccountOpened	BP Initiating Event
Select Client information	Task	SelectExistingRetailClientDetails	Activities
Fill in Cheque book Details	Task	FillinChequeBookDetailsForms	Activities
Choose Vault Code	Task	AssignSpecificVuaIltCodeToChequeTypes	Activities
Set Cheque Type	Task	DefineChequeTypes	Activities
Set Cheque Status	Task	ChooseLifeStatusToReleaseChequeBooks	Activities
Produce Client Cheque Book Report	Task	ProduceRetailClientChequeBookReports	Activities
		NewRetailChequeBookProduced	BP Goal Event
Retail Banking Admin	Lane	RetailBankingAdmins	BP Participants
Cheque Book Detail Form	Data	ChequeBookDetailForms	Input
Client Cheque Report	Data	RetailClientChequeBookReports	Output

- **Insurance**
  - Define Insurance Client Details

DefineInsuranceClientDetails			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
Client Information Created or Maintained	Event	NewClientWillingToHaveInsuranceAccountCreated	BP Initiating Events
Set up Client Name Details	Task	ChooseTheTypeOfClientInsurances	Activities
		FillinInsuranceClientDetailForms	Activities
Set up Client Address Details	Task	AddClientAddressDetails	Activities
Set up Financial Details	Task	FillinInsuranceClientFinancialDetailsForms	Activities
Complete Client Employment Details	Task	FillinInsuranceClientEmploymentDetailForms	Activities
Define Underwriting Details	Task	SetupInsuranceClientAmendableDetails	Activities
Attach an Agent to Client name	Task	AttachAnAgentForClientInsuranceAccounts	Activities
Attach Product Template to Client	Task	ChoosePreferableInsuranceProductForCustomer	Activities
Produce Client Reports	Task	ProduceInsuranceClientDetailReports	Activities
		InsuranceClientDetailCreated	BP Goal Event
Client	Lane	Clients	BP Participant
Unisure Assistance	Lane	InsuranceAssistants	BP Participant
Client Details Form	Data	InsuranceClientEmploymentForms	Input
Client Maintenance Form	Data	Client Maintenance Form	Input
Client Employment Form	Data	InsuranceClientForms	Input
Client Financial Form	Data	InsuranceClientFinancialForms	Input
Client Details Report	Data	InsuranceClientDetailReports	Output

- Define Insurance Product Templates

DefineInsuranceProductTemplates			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Product Introductory or Existing Product maintenance	Event	NewInsuranceProductIntroductory	BP Initiating Event
Define Product Category	Task	ChooseCategoryForTheNewInsuranceProduct	Activities
Define Product Structure using Product Template	Task	FillinInsuranceProductTemplateForms	Activities
Set up Investment Options	Task	DecideOnTheTypeofInvestmentForms	Activities
Attach Set of Product Rules Record	Task	SelectInsuranceRulesFromRuleGuide	Activities
Define Product Transaction Process Type	Task	AssignTransactionFormProcess	Activities
Set up Product Status	Task	ChooseLifeStatusToReleaseProductToMarket	Activities
Produce Product Report	Task	ProduceInsuranceProductReports	Activities
		NewInsuranceProductLuanchedToMarket	BP Goal Events
Client	Lane	Clients	BP Participant
Unisure	Lane	InsuranceAdmins	BP Participant
Product Investment Form	Data	InsuranceInvestmentForms	Input
Product Template Form	Data	InsuranceProductTemplateForms	Input
		InsuranceTransactionForms	Input
Product Transaction Form	Data	InsuranceRuleGuides	Input
Product Report	Data	InsuranceProductReports	Output

- Set up New Business Inputs

SetupNewBusinessInputs			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Business Proposal	Event	NewBusinessProposalPutInPlaces	BP Initiating Event
Search for Product	Task	ChooseProductMatchNewBusinessRequirements	Activities
Enter Client Details	Task	FillinNewBusinessClientDetailForms	Activities
Add Client Roles	Task	FillinInsuranceRoleForms	Activities
Assign Product to Client	Task	AssignClinetToTheNewBusinesses	Activities
Fill in Proposal Billing Details	Task	FillinBillingFormsForTheNewBusinesses	Activities
Select Product Options for Policy	Task	SelectRulesThatMatchTheNewBusinessProposals	Activities
		NewBusinessDefinitionCompleted	BP Goal Event
Client	Lane	Clients	BP Participant
Unisure	Lane	InsuranceAssistants	BP Participant
Client Details Form	Data	NewBusinessClientForms	Input
Client Role Form	Data	NewBusinessClientRoleForms	Input
Client Personal Billing Form	Data	ClientPersonalBillingForms	Input
New Business Form	Data	NewBusinessForms	Output

- Allocate Bonus For Profit Policies

AllocateBonusForProfitPolicies			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
Bonuses for with-profits policies calculated	Event	NewBonusesRateIntroduced	BP Initiating Event
Define Bonus Allocation Processing	Task	ChooseTheWayForNewBonusesApplications	Activities
		FillinBonusesRatesForms	Activities
Produce Bonus Summary Report	Task	ProduceEntitledBonusesAccounts	Activities
Print Bonus Notices Report	Task	PrintBonusesNoticeReports	Activities
Perform Bonus Reprint Request form	Task	ProduceNewReportForAppliedBonuses	Activities
Perform Bonus Notices Reprint	Task	ReprintBonusSummaryReport	Activities
		NewBonusesApplied	BP Goal Event
Policy Admin	Lane	Policy Admin	BP Participant
Bonus Rate Form	Data	Bonus Rate Form	Input
Bonus Summary Report	Data	Bonus Summary Report	Output
Bonus Notice Report	Data	Bonus Notice Report	Input
Reprint Form	Data	Reprint Form	Output

- Define Financial Administration Processes

DefineFinancialAdministrationProcesses			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Business Financial calculation	Event	InsuranceTransactionCalculated	BP Initiating Event
Set up New Business Financial control	Task	CreateControlForNewBusinessesFinance	Activities
Generate Regular Premium	Task	AssignRegularPremuimPayments	Activities
Start Premium Collection	Task	BeginCollectionFromEachAccount	Activities
Match Cash Received to Premium Due	Task	MatchCollectedCashWithPremuim	Activities
Release Premium Paid Commission	Task	TakeoffComissionPayments	Activities
Record Unit Allocation Trigger	Task	DocumentUnitAllocationBusinessTrigger	Activities
Complete Record of Financial Transaction Form	Task	FillinFinancialTransactionForms	Activities
Create General Ledger Transactions	Task	ProduceNewBusinessGeneralLedgerTransactionForm	Activities
		InsuranceFinancialTransactionCompleted	BP Goal Event
Policy Admin	Lane	InsurancePolicyAdmins	BP Participant
Financial Transaction Form	Data	FinancialTransactionForms	Input
General Ledger Transactions Form	Data	GeneralLedgerTransaction Forms	Output

- **Mortgages**
  - Set up Nominal Accounts

SetupNominalAccounts			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Mortgage Account installation	Event	NewNominalAccountPlanned	BP Initiating Event
Set up Nominal Account Code Form	Task	FillinNominalAccountForms	Activities
Define sub and Major Account Code	Task	OrganiseRelatedAccounts	Activities
Allocate Account Type	Task	AssignTypeForTheNewAccounts	Activities
Ensure account Indicator set correctly	Task	SetupAccountIndicators	Activities
Attach GL Code to identify Transaction	Task	AttachGLCodeForTransactions	Activities
Produce Nominal Account Reports	Task	ProduceNominalAccountReports	Activities
		NominalAccountLaunchedForUse	BP Goal Event
UFSS Admin	Lane	MortgagesAdmins	BP Participant
Nominal Account Form	Data	NominalAccountForms	Input
Nominal Account Report	Data	NominalAccountReports	Output

  - Define Bank Account Set up

DefineBankAccountSetup			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Account	Event	NewTypeMortgageNeeded	BP Initiating
Set up Bank Branch Details	Task	FillinBankBranchDetailForms	Activities
Define Account Code Identifier	Task	SetupCodeIdentifierForTheAccounts	Activities
Allocate Bank Transaction Record	Task	AllocateTransactionRecords	Activities
Set Account Amendable Details	Task	DefineAmedableDetailofTheAccount	Activities
Produce Bank Account Reports	Task	ProduceBankAccountReports	Activities
		NewMortgageAccountDefined	BP Goal Event
Client	Lane	Clients	BP Participant
UFSS Admin	Lane	MortgagesAdmins	BP Participant
Bank Account Form	Data	BankBranchDetailForms	Input
Bank Account Report	Data	BankAccountReports	Output

  - Arrange Daily Returns Processing

Arrange Daily Returns Processing			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
End of Day Calculation	Event	End of the Day Money Return Calculation	BP Initiating Event
Produce the Daily Return Book	Task	Produce A Book with the Daily Branch Returns Forms	Activities
Summaries Branches Returns	Task	Produce a Summary of the Returns	Activities
Post Overall Cash Movement	Task	Post All Cash for the Daily Returns	Activities
Produce an analysis of Branch banking for the Day	Task	Analyse The Daily Return of Each Branch	Activities
Update Cash Figure Form	Task	Update Cash of each Branch using Cash figure Form	Activities
Marks the Branch Balance sheet Record	Task	Mark the Balance Sheet of Each Branch	Activities
Produce Daily Returns Report	Task	Produce Daily Returns Report	Activities
		Daily Returns Completed	BP Goal Event
UFSS Admin	Lane	UFSS Admins	BP Participant
Daily Return Book Form	Data	Daily Return Book Form	Input
Cash Figure Form	Data	Cash Figure Form	Input
Branch Balance Sheet	Data	Branch Balance Sheet	Input
Daily Returns Report	Data	Daily Returns Report	Output

- Define Mortgage Products

DefineMortgageProducts			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Mortgage Product	Event	NewMortgageProductIntroductory	BP Initiating Event
Create Product Shell	Task	FillinMortgageProductTemplateForms	Activities
Define Product Specific Rules	Task	SelectMortgageSpecificRules	Activities
Apply Product Processing Rules	Task	AssignMortgageProcessingRules	Activities
Apply Interest Rate Rules	Task	AssignInterestRateUsingInterestRateForms	Activities
Define Product amendable Details	Task	DefineAmendableMortgageProductDetails	Activities
Set Mortgage Product Effective Date	Task	ChooseLifeDateToReleaseProducts	Activities
		ProduceMortgageProductReports	Activities
		NewMortgageProductLaunchedToMarkets	BP Goal Event
UFSS Admin	Lane	MortgagesAdmins	BP Participant
Product Shell	Data	MortgageProductTemplateForms	Input
Product interest Rate Form	Data	MortgageInterestRateForms	Input
Interest Rate Guide	Data	MortgageInterestRateGuides	Input
Mortgage Report	Data	MortgageProductReports	Output

- Process Mortgage Applications

ProcessMortgageApplications			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Client request Mortgages	Event	NewClientRequestsMortgages	BP Initiating Event
Establish Basic Loan Application Form	Task	FillinBasicLoanApplicationForms	Activities
Record Customer Details	Task	FillinMortgageCustomerDetailForms	Activities
Add Occupational Details	Task	FillinOccupationalDetailForms	Activities
Record Financial Details	Task	FillinFinancialDetailsforms	Activities
Issue Reference Request Letter	Task	IssueReferenceRequestLetters	Activities
Hand in Mortgage User Guide	Task	HandinMortgageUserGuides	Activities
Issue Summary of all Reference request Letters	Task	PrintSummaryOfAllReferenceRequestLetters	Activities
Record Property against Application	Task	AssignPropertyToMortgageLoans	Activities
Initiates Dairy for certain Event	Task	EnableDairyForCertainMortgagesAccountEvents	Activities
Set up Customer Inquiry and update condition	Task	ProduceCustomerInquiryAndConditionUpdates	Activities
Record Receipt of the application	Task	IssueReceiptForTheApplications	Activities
Define Cancellation Condition	Task	SetupCancellationCriteriaForCustomers	Activities
		MortgageLoanProvidedForCustomers	BP Goal Event
Customer	Lane	Clients	BP Participant
UFSS Assistant	Lane	MortgagesAssistants	BP Participant
Loan Application Form	Data	BasicLoanApplicationForms	Input
Receipt	Data	MortgageUserGuides	Input
		MortgageCutomerFinancialForms	Input
Customer Details Form	Data	MortgageCutomerDetailForms	Input
Customer Occupational Form	Data	MortgageCustomerOccupationalForms	Input
		MortgageApplicationReceipts	Output
		ReferenceRequestLetters	Output
Customer Financial Form	Data	ReferenceSummaryReports	Output

- Change Mortgage Interest Rate Process

ChangeMortgageInterestRateProcess			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New Interest Rate - Market changed	Event	MarketInterestRateChanged	BP Initiating Event
Enter new Interest Rate for Each Mortgage Product Code	Task	FillinMortgageInterestRatesForms	Activities
		UpdateEachMortgageWithNewInterestRates	Activities
Run Rate Change Report to Calculate the New Payment for the Effective Date	Task	ProduceReportForAffectedAccounts	Activities
Run the Rate Change Report to Produce Notification to Customer Letter	Task	ProduceCustomerReportForTheNewChanges	Activities
Run the Report to bring the New Rate and Payment into Effect	Task	IssueNewReportAfterInterestApplications	Activities
		NewMortgagesInterestRateApplied	BP Goal Events
Customer	Lane	Clients	BP Participant
UFSS Admin	Lane	MortgagesAdmins	BP Participant
Mortgage Interest Rate Form	Data	MortgageInterestRateForms	Input
New Payment Calculation Report	Data	AccountAffectedInterestRateReports	Output
Customer Notification Report	Data	CustomerReports	Output
New Rate and Payment Report	Data	InterestRateApplicationReports	Output

- Create Mortgage Account Types

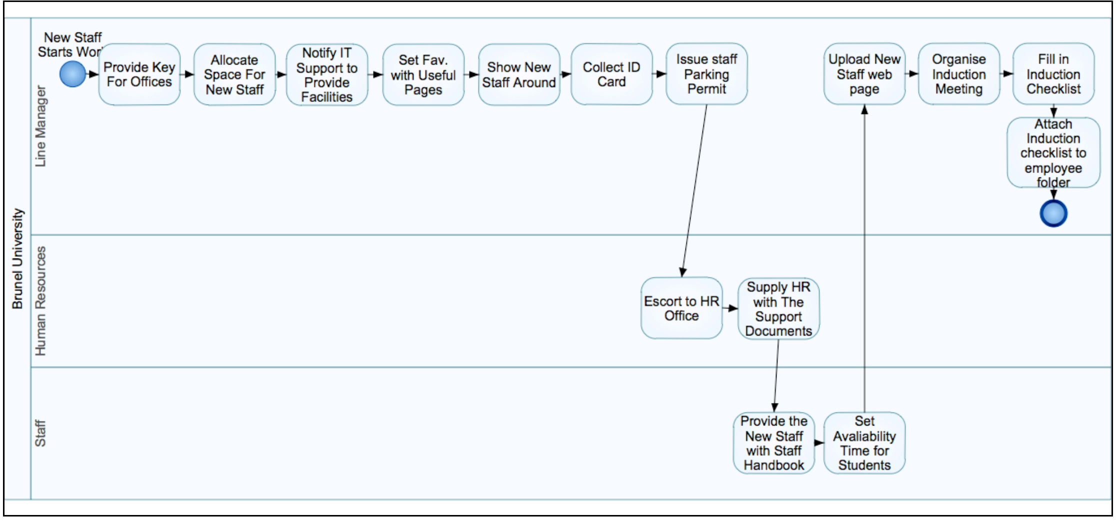
CreateMortgageAccountTypes			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New mortgage Account Opened	Event	NewMortgageAccountTypeRequested	BP Initiating Event
Complete Account Type Details	Task	FillinMortgageAccountTypeDetails	Activities
Define Account Opening Categories	Task	ChooseCategoryForAccountTypes	Activities
Apply Interest Rate Rules	Task	AssignInterestRateFromRuleGuides	Activities
		FillinInterestRateForms	Activities
Restrict Access to the Account Type at Specific Branch	Task	ApplyConstraintOnAccountAccessibilities	Activities
Set Account Bonuses Details	Task	CompleteAccountBonusesForms	Activities
Set Transaction and Charges Details	Task	ApplyTransactionChargesDetails	Activities
Print Account statement	Task	ProduceAccountOpeningForms	Activities
Set up Account Type Status	Task	ChooseLifeStatusToBeAvaliableForCustomers	Activities
Specify Transfer Rules for the account	Task	ConditionTransferMoneyForTheAccounts	Activities
		MortgageAccountTypeOpened	BP Goal Event
Customer	Lane	Clients	BP Participant
UFSS Assistant	Lane	MortgagesAdmins	BP Participant
Mortgage Interest Rate Form	Data	MortgageInterestRateForms	Input
Bonuses Form	Data	BonusesForms	Input
Interest Rate Guide	Data	InterestRateGuides	Input
Mortgage Account Type Form	Data	MortgageAccountDetailForms	Input
Account Opening Form	Data	MortgageAccountOpeningForms	Output

- Create Investment Accounts

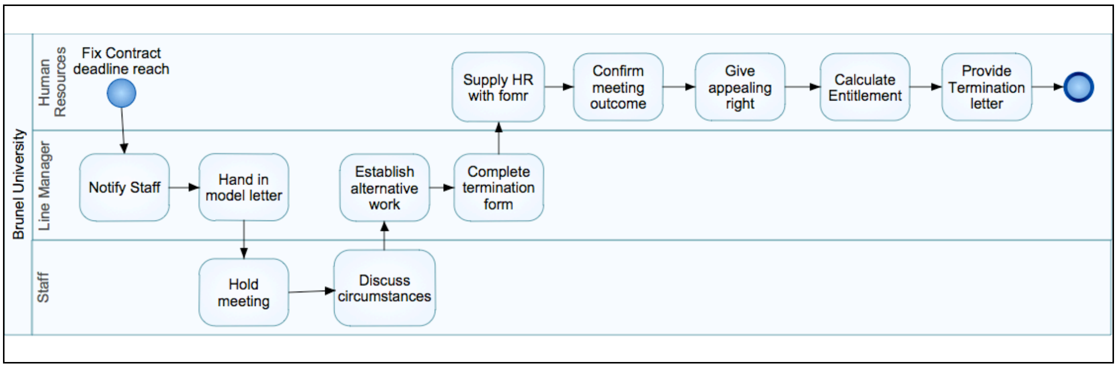
CreateInvestmentAccounts			
Terms	**BPMN Elements**	**Interpretation**	**BPO Elements**
New investment Customer	Event	CustomerWillingToHaveInvestmentAccounts	BP Initiating Event
Complete Customer Details	Task	FillinInvestmentCustomerDetailForms	Activities
Complete Occupational Details	Task	FillinInvestmentCustomerOccupationalForms	Activities
Add Customer Financial Details	Task	FillinInvestmentCustomerFinancialForms	Activities
Check Credit Reference	Task	CheckCustomerCredits	Activities
Produce Receipt of Credit Reference	Task	ProvideReceiptofInvestmentCredits	Activities
Attach Product to Customer	Task	ChooseInvestmentSuitsCustomerNeeds	Activities
Identify Authorisation Link	Task	SetAuthorisationLinks	Activities
Identify Beneficiaries Link	Task	SelectCustomerBeneficials	Activities
Set Customer self Certification	Task	SetCustomerSelfCertificatesAccessibility	Activities
Set Account Status	Task	ChooseLifeStatusForInvestmentAccounts	Activities
		InvestmentAccountCreatedForCustomers	BP Goal Event
Customer	Lane	Clients	BP Participant
UFSS Assistant	Lane	MortgagesAssistants	BP Participant
Customer Details Form	Data	InvestmentCustomerDetailForms	Input
Customer Product Form	Data	InvestmentCustomerFinancialForms	Input
Customer Occupational Form	Data	InvestmentCustomerOccupationalForms	Input
Customer Financial Form	Data	Customer Financial Form	Input
Customer Certification	Data	CustomerCertifications	Output
Credit Receipt	Data	CreditReceipts	Output

2. Educational Domain

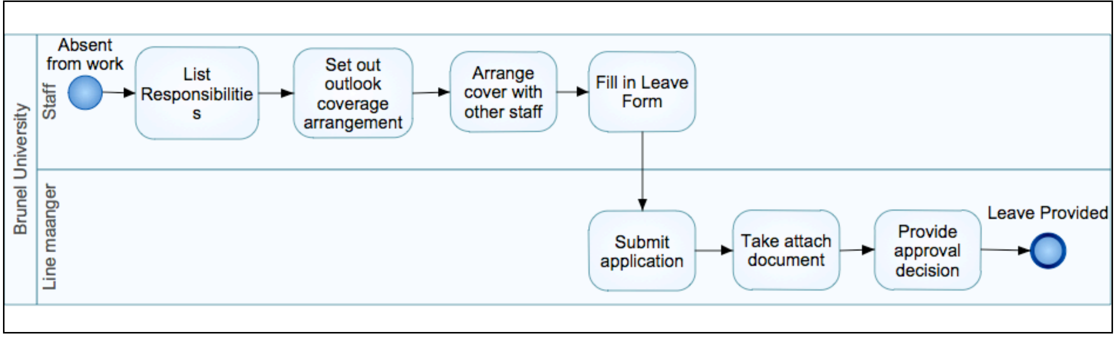
- **Brunel University Staff Handbook**
  - Arrange Staff Induction


Brunel University
Line Manager
Human Resources
Staff
New Staff Starts Wor
Provide Key For Offices
Allocate Space For New Staff
Notify IT Support to Provide Facilities
Set Fav. with Useful Pages
Show New Staff Around
Collect ID Card
Issue staff Parking Permit
Upload New Staff web page
Organise Induction Meeting
Fill in Induction Checklist
Attach Induction checklist to employee folder
Escort to HR Office
Supply HR with The Support Documents
Provide the New Staff with Staff Handbook
Set Avaliability Time for Students

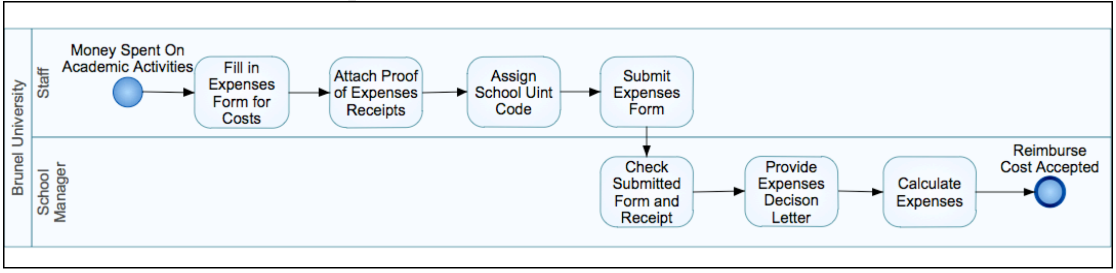
  - Terminate of fix term Contract



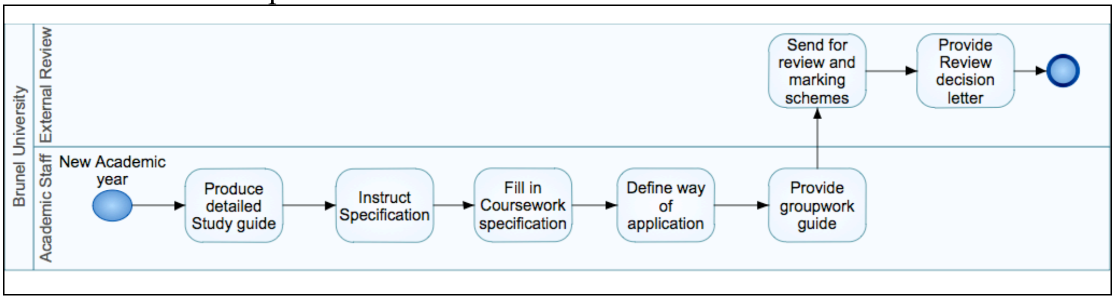
  - Apply for Leave



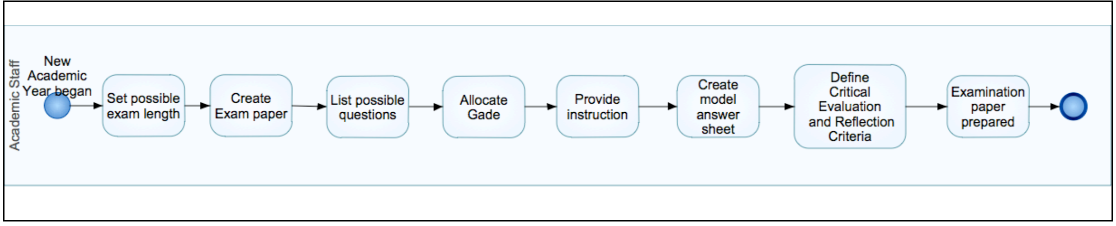
- Claim Expenses


Brunel University
Staff
School Manager
Money Spent On Academic Activities
Fill in Expenses Form for Costs
Attach Proof of Expenses Receipts
Assign School Uint Code
Submit Expenses Form
Check Submitted Form and Receipt
Provide Expenses Decison Letter
Calculate Expenses
Reimburse Cost Accepted

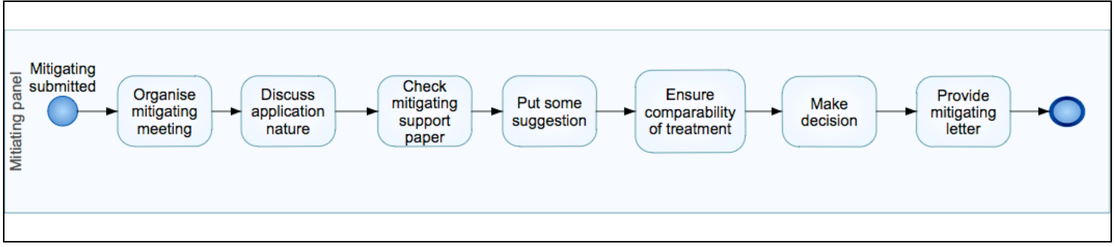
- Prepare Assessment Process


Brunel University
External Review
Academic Staff
New Academic year
Produce detailed Study guide
Instruct Specification
Fill in Coursework specification
Define way of application
Provide groupwork guide
Send for review and marking schemes
Provide Review decision letter

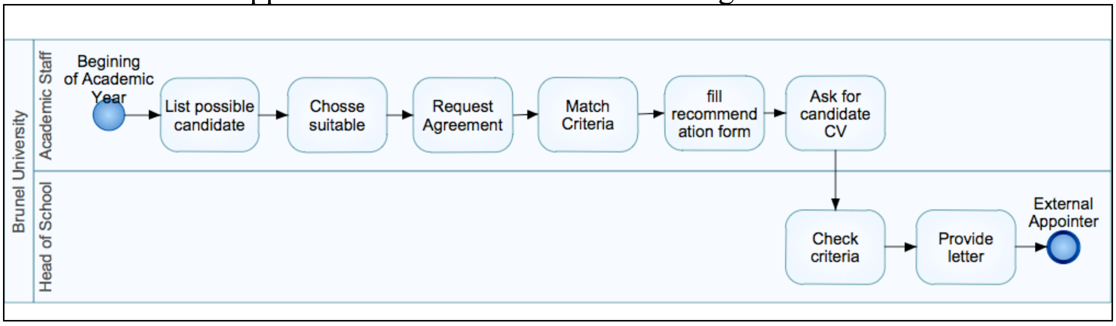
- Prepare Examination Paper



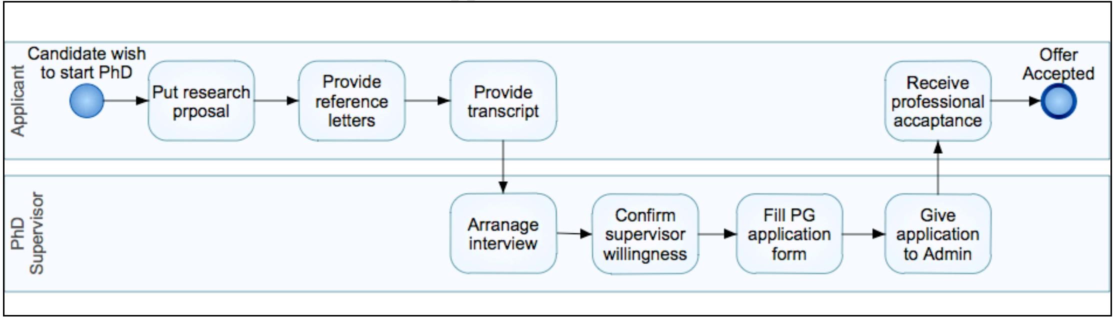
- Hold mitigating Circumstance panel


Mitiating panel
Mitigating submitted
Organise mitigating meeting
Discuss application nature
Check mitigating support paper
Put some suggestion
Ensure comparability of treatment
Make decision
Provide mitigating letter

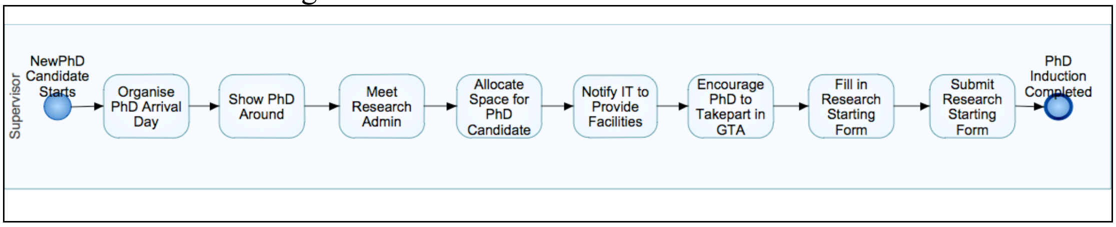
- Appoint of External Examiner for Taught course



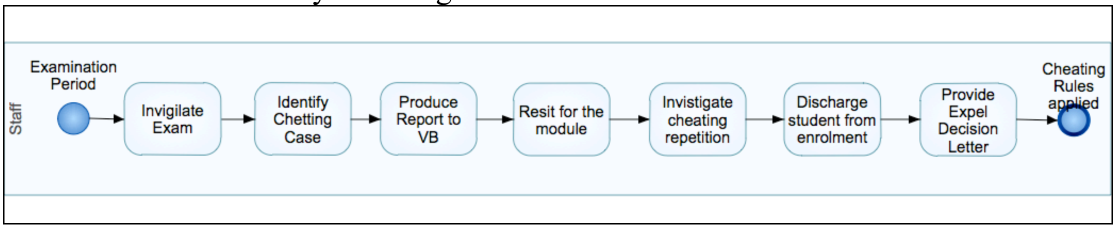
  - Process of PGR application


Applicant
PhD Supervisor
Candidate wish to start PhD
Put research prposal
Provide reference letters
Provide transcript
Receive professional accaptance
Offer Accepted
Arranage interview
Confirm supervisor willingness
Fill PG application form
Give application to Admin

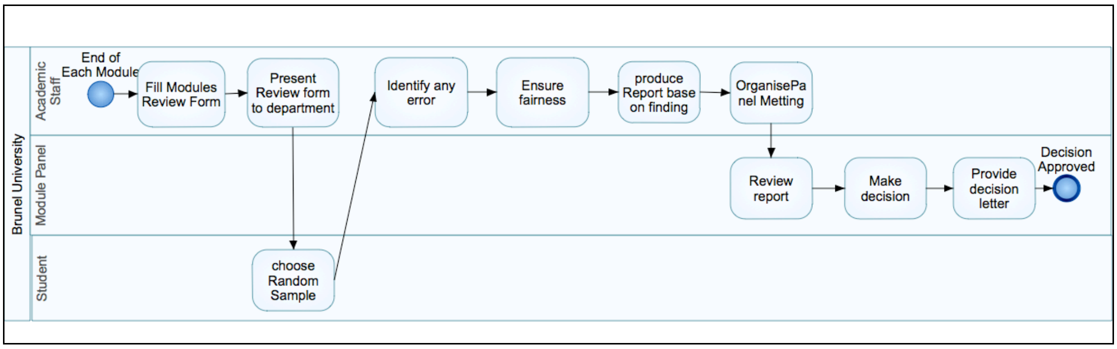
  - Arrange PhD induction


Supervisor
NewPhD Candidate Starts
Organise PhD Arrival Day
Show PhD Around
Meet Research Admin
Allocate Space for PhD Candidate
Notify IT to Provide Facilities
Encourage PhD to Takepart in GTA
Fill in Research Starting Form
Submit Research Starting Form
PhD Induction Completed

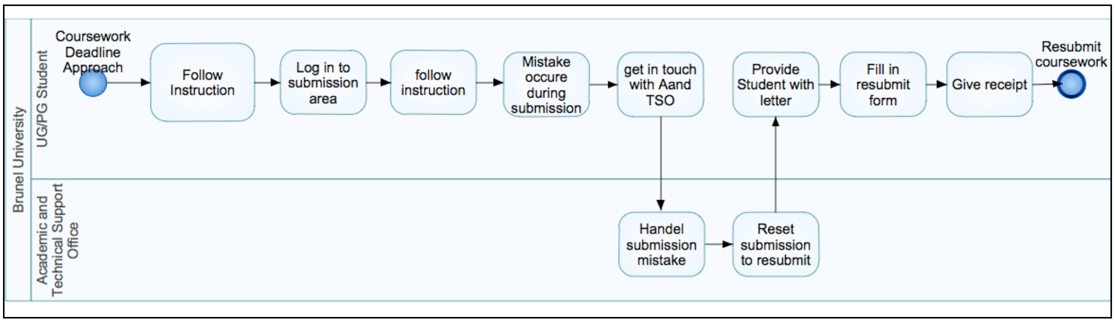
- Brunel University Students Handbook
  - Identify Cheating Case



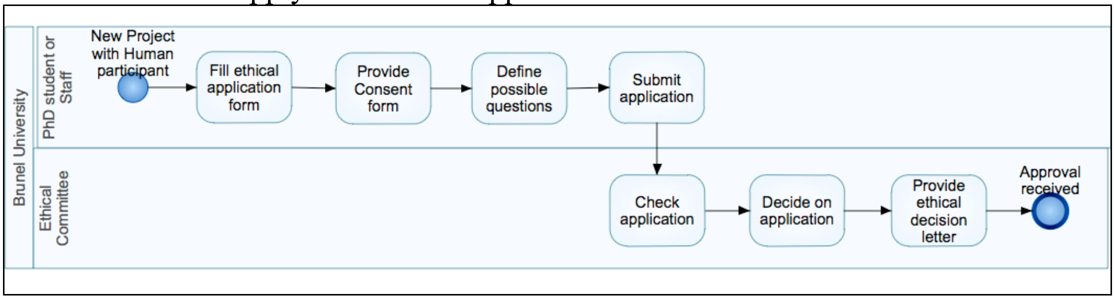
  - Review Module Processes



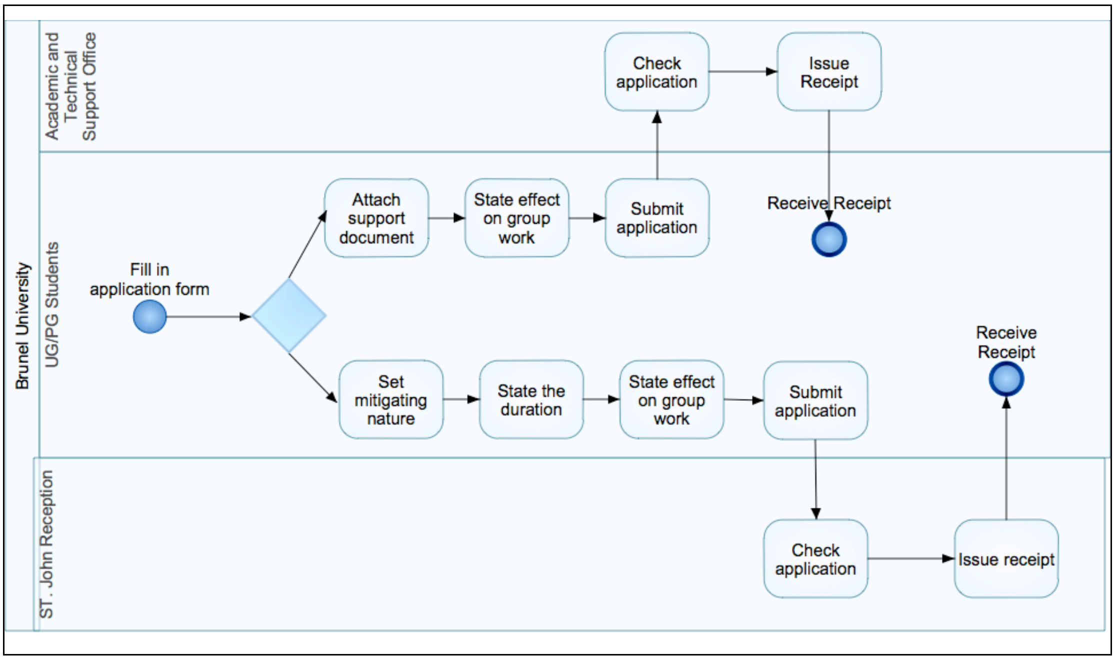
- Submit Coursework Processes


Brunel University
UG/PG Student
Academic and Technical Support Office
Coursework Deadline Approach
Follow Instruction
Log in to submission area
follow instruction
Mistake occure during submission
get in touch with Aand TSO
Provide Student with letter
Fill in resubmit form
Give receipt
Resubmit coursework
Handel submission mistake
Reset submission to resubmit

- Apply For Ethical Approval Students



- Apply For Mitigating Circumstances

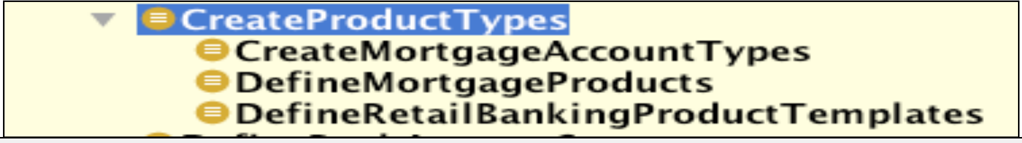


Appendix C BP Ontology Generalisation Models

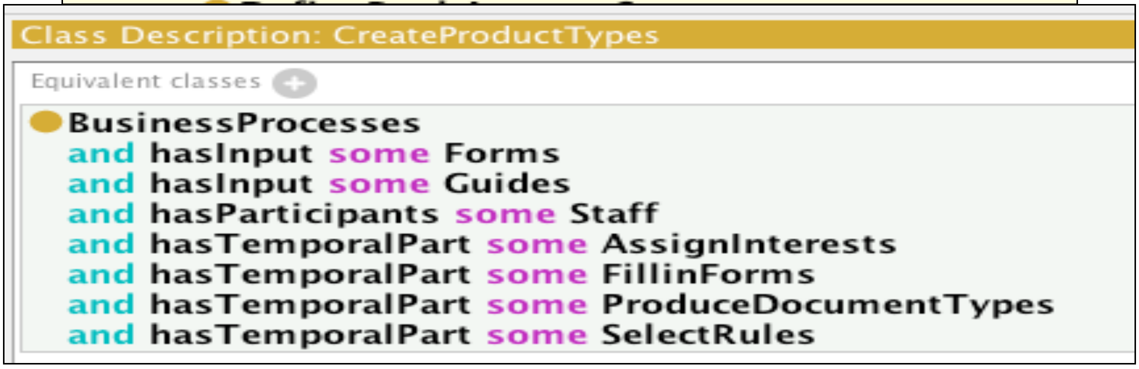
Semantic Enhancement of BP Models

First: Generalise Similar Type Business Processes

- Create Product Types



CreateProductTypes
- CreateMortgageAccountTypes
- DefineMortgageProducts
- DefineRetailBankingProductTemplates

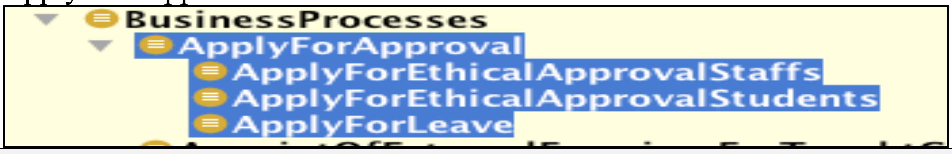


Class Description: CreateProductTypes

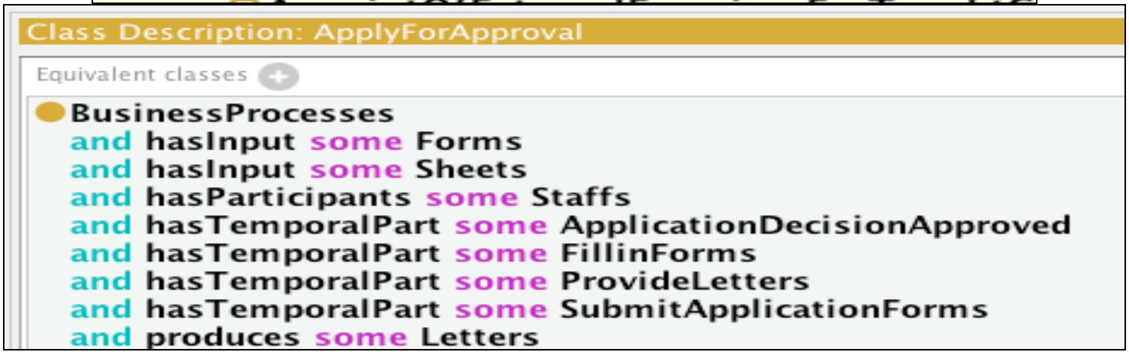
Equivalent classes

```
BusinessProcesses
 and hasInput some Forms
 and hasInput some Guides
 and hasParticipants some Staff
 and hasTemporalPart some AssignInterests
 and hasTemporalPart some FillinForms
 and hasTemporalPart some ProduceDocumentTypes
 and hasTemporalPart some SelectRules
```

- Apply For Approval



BusinessProcesses
- ApplyForApproval
  - ApplyForEthicalApprovalStaffs
  - ApplyForEthicalApprovalStudents
  - ApplyForLeave

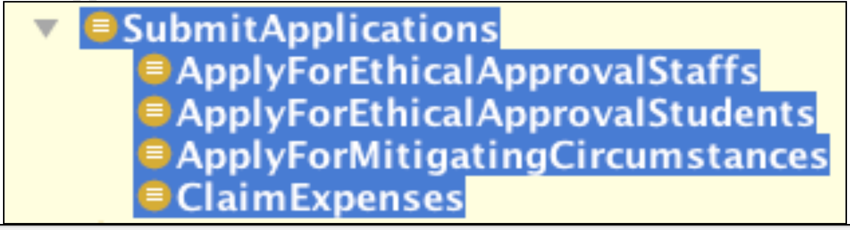


Class Description: ApplyForApproval

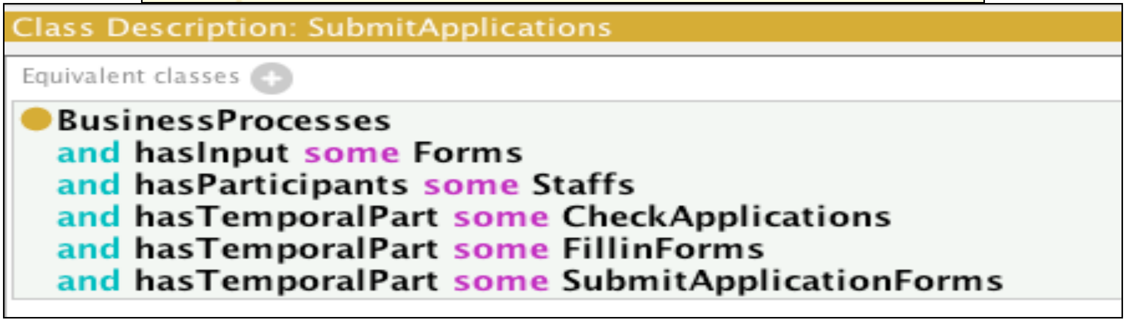
Equivalent classes

```
BusinessProcesses
 and hasInput some Forms
 and hasInput some Sheets
 and hasParticipants some Staffs
 and hasTemporalPart some ApplicationDecisionApproved
 and hasTemporalPart some FillinForms
 and hasTemporalPart some ProvideLetters
 and hasTemporalPart some SubmitApplicationForms
 and produces some Letters
```

- Submit Applications



- SubmitApplications
  - ApplyForEthicalApprovalStaffs
  - ApplyForEthicalApprovalStudents
  - ApplyForMitigatingCircumstances
  - ClaimExpenses



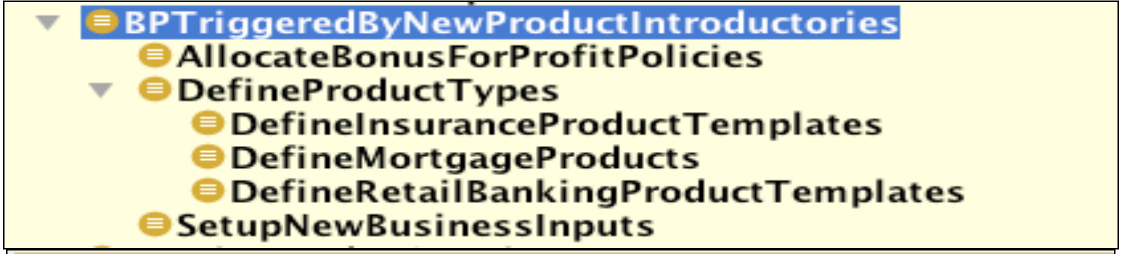
Class Description: SubmitApplications

Equivalent classes

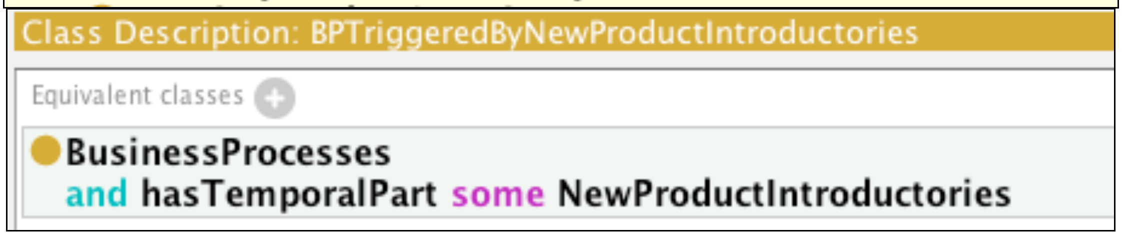
```
BusinessProcesses
 and hasInput some Forms
 and hasParticipants some Staffs
 and hasTemporalPart some CheckApplications
 and hasTemporalPart some FillinForms
 and hasTemporalPart some SubmitApplicationForms
```

Second: Generalisation of Similar Business Processes Initiating Event:

- BP Triggered By New Product Introductories



- BPTriggeredByNewProductIntroductories
  - AllocateBonusForProfitPolicies
  - DefineProductTypes
    - DefineInsuranceProductTemplates
    - DefineMortgageProducts
    - DefineRetailBankingProductTemplates
  - SetupNewBusinessInputs

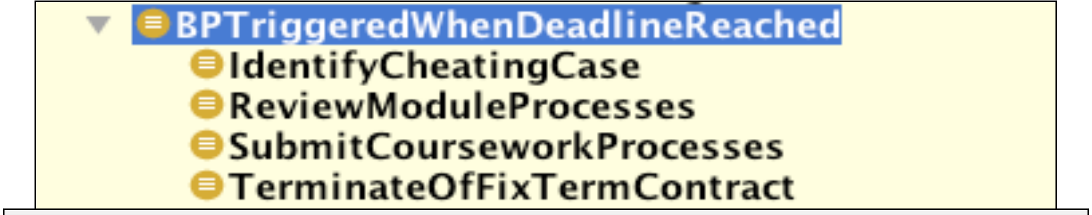


Class Description: BPTriggeredByNewProductIntroductories

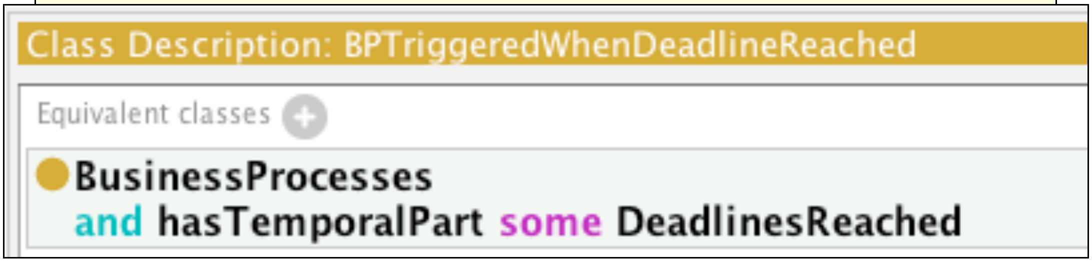
Equivalent classes

```
BusinessProcesses
 and hasTemporalPart some NewProductIntroductories
```

- BP Triggered When Deadline Reached.



- BPTriggeredWhenDeadlineReached
  - IdentifyCheatingCase
  - ReviewModuleProcesses
  - SubmitCourseworkProcesses
  - TerminateOfFixTermContract

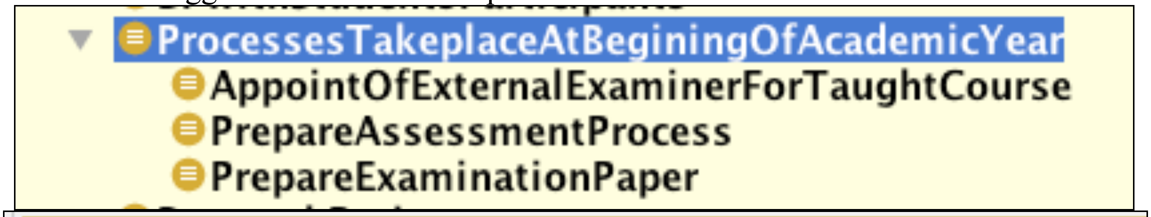


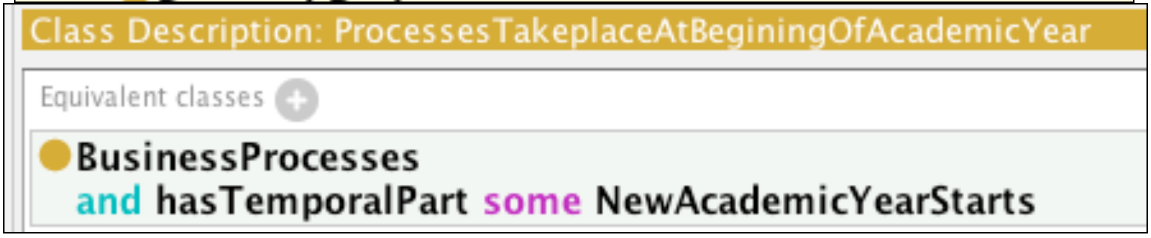
Class Description: BPTriggeredWhenDeadlineReached

Equivalent classes

```
BusinessProcesses
 and hasTemporalPart some DeadlinesReached
```

- BP Trigger When New Participant Started.

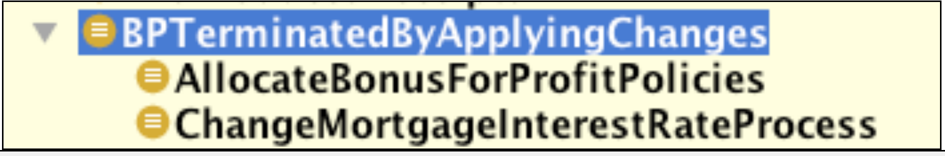

ProcessesTakeplaceAtBeginingOfAcademicYear
- AppointOfExternalExaminerForTaughtCourse
- PrepareAssessmentProcess
- PrepareExaminationPaper

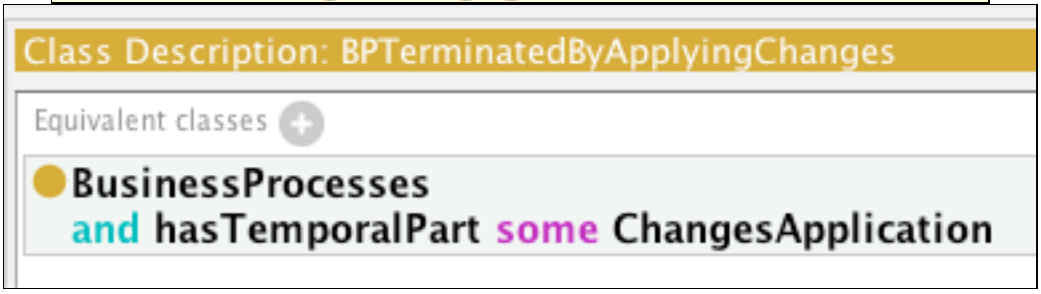

Class Description: ProcessesTakeplaceAtBeginingOfAcademicYear

```
BusinessProcesses
 and hasTemporalPart some NewAcademicYearStarts
```

Third: Generalisation of Similar Business Processes Goal Event:

- BP Terminated By Applying Changes

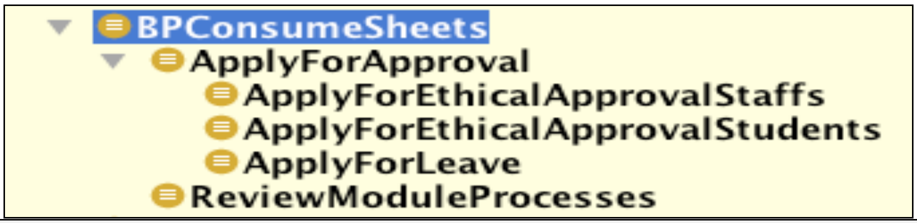

BPTerminatedByApplyingChanges
- AllocateBonusForProfitPolicies
- ChangeMortgageInterestRateProcess

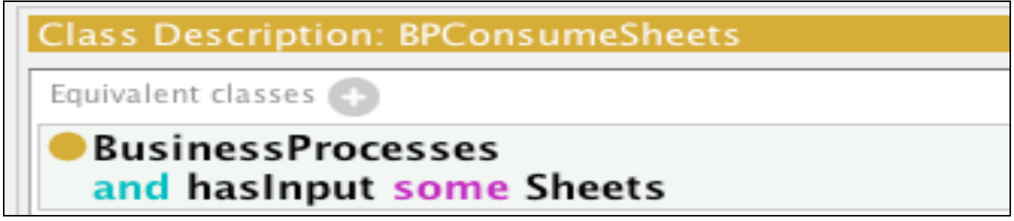

Class Description: BPTerminatedByApplyingChanges

```
BusinessProcesses
 and hasTemporalPart some ChangesApplication
```

Forth: Generalisation of Similar Business Processes Inputs:

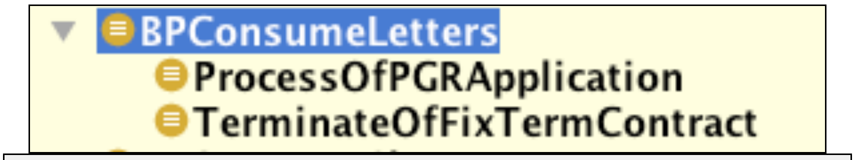
- BP Consume Sheets


BPConsumeSheets
- ApplyForApproval
  - ApplyForEthicalApprovalStaffs
  - ApplyForEthicalApprovalStudents
  - ApplyForLeave
- ReviewModuleProcesses

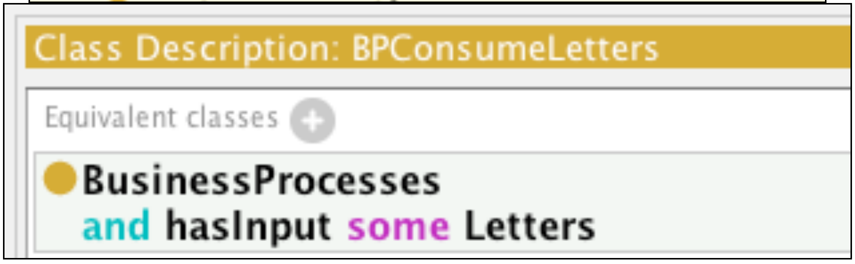

Class Description: BPConsumeSheets

```
BusinessProcesses
 and hasInput some Sheets
```

- BP Consume Letters



BPConsumeLetters
- ProcessOfPGRApplication
- TerminateOfFixTermContract

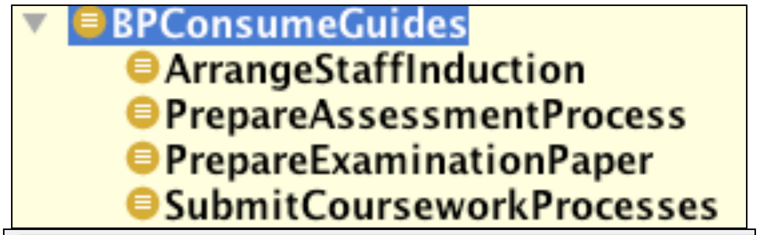


Class Description: BPConsumeLetters

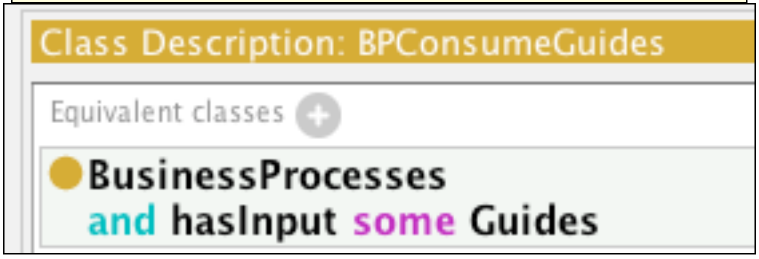
Equivalent classes

BusinessProcesses
and hasInput some Letters

- BP Consume Guides.



BPConsumeGuides
- ArrangeStaffInduction
- PrepareAssessmentProcess
- PrepareExaminationPaper
- SubmitCourseworkProcesses



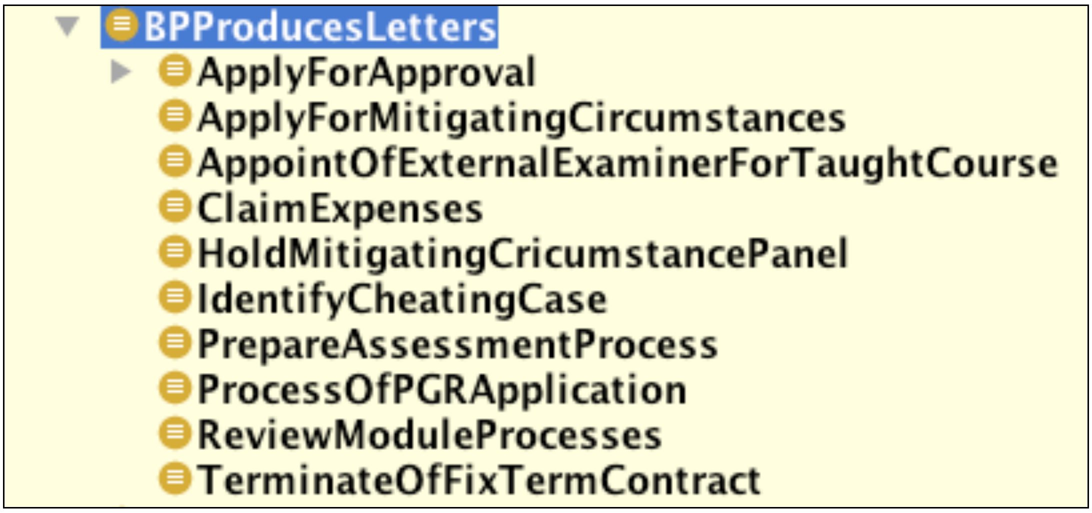
Class Description: BPConsumeGuides

Equivalent classes

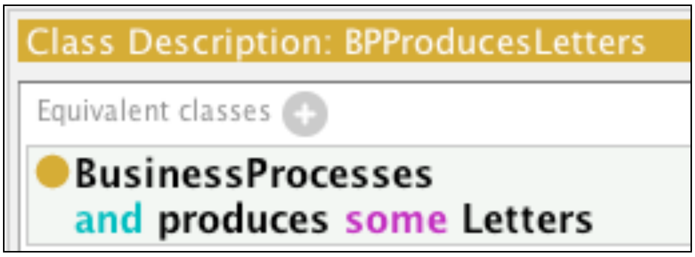
BusinessProcesses
and hasInput some Guides

Fifth: Generalisation of Similar Business Processes Outputs:

- BP Produces Letters



BPProducesLetters
- ApplyForApproval
- ApplyForMitigatingCircumstances
- AppointOfExternalExaminerForTaughtCourse
- ClaimExpenses
- HoldMitigatingCricumstancePanel
- IdentifyCheatingCase
- PrepareAssessmentProcess
- ProcessOfPGRApplication
- ReviewModuleProcesses
- TerminateOfFixTermContract

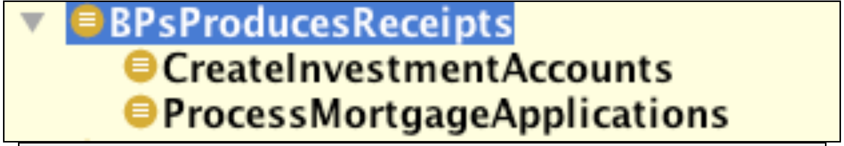


Class Description: BPProducesLetters

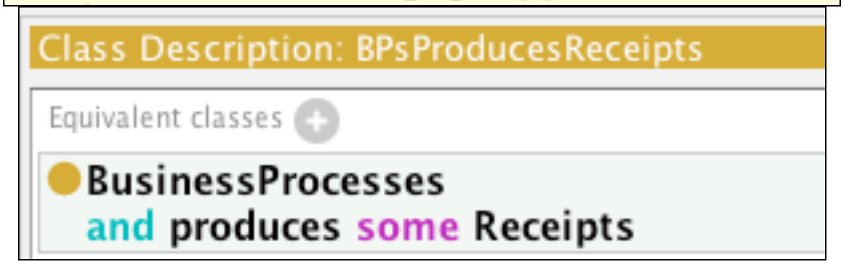
Equivalent classes ⊕

● BusinessProcesses
and produces some Letters

- BP Produces Receipts



▼ BPsProducesReceipts
CreateInvestmentAccounts
ProcessMortgageApplications



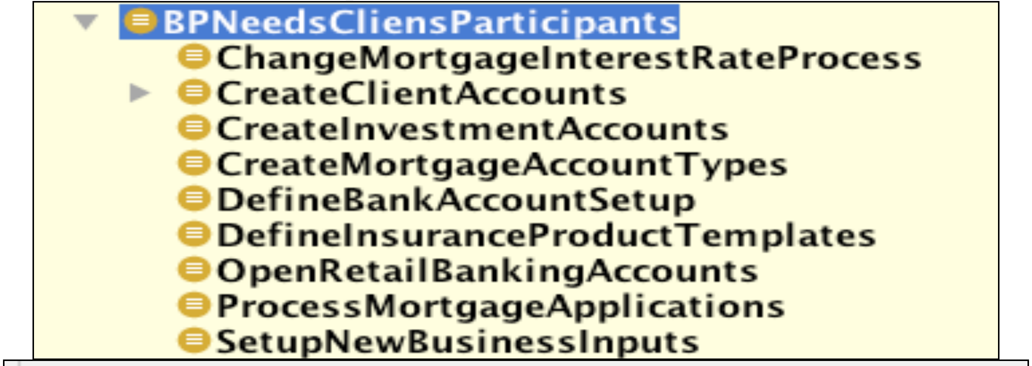
Class Description: BPsProducesReceipts

Equivalent classes ⊕

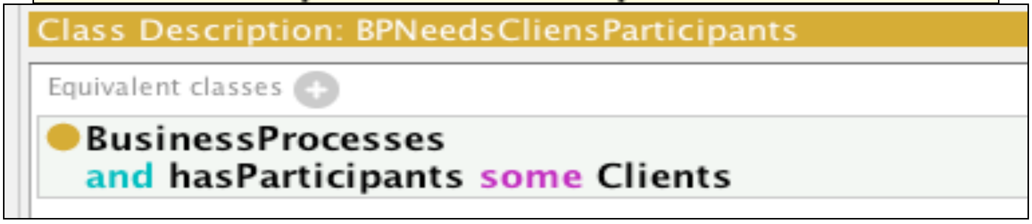
● BusinessProcesses
and produces some Receipts

Sixth: Generalisation of Similar Business Processes Participants:

- BP Needs Client Participants



▼ BPNeedsCliensParticipants
ChangeMortgageInterestRateProcess
▶ CreateClientAccounts
CreateInvestmentAccounts
CreateMortgageAccountTypes
DefineBankAccountSetup
DefineInsuranceProductTemplates
OpenRetailBankingAccounts
ProcessMortgageApplications
SetupNewBusinessInputs

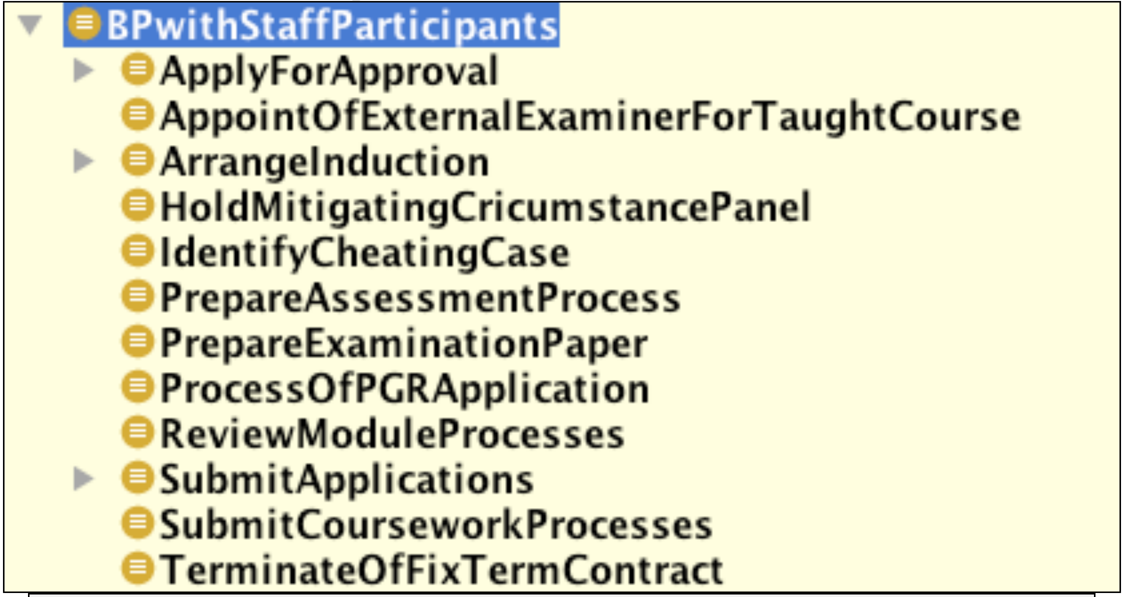


Class Description: BPNeedsCliensParticipants

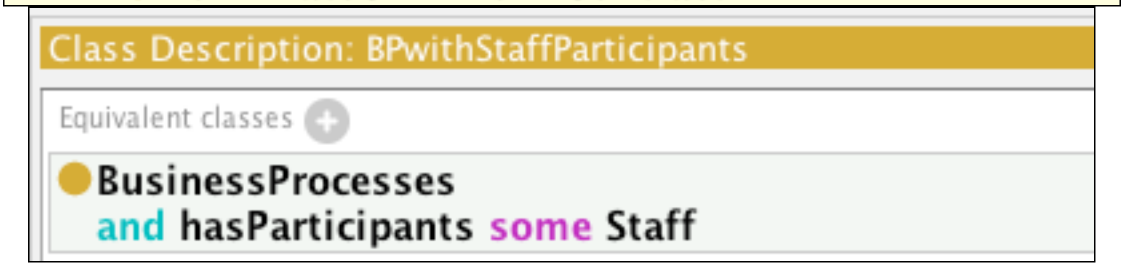
Equivalent classes ⊕

● BusinessProcesses
and hasParticipants some Clients

- BP with Staff Participants



- BPwithStaffParticipants
  - ApplyForApproval
  - AppointOfExternalExaminerForTaughtCourse
  - ArrangeInduction
  - HoldMitigatingCricumstancePanel
  - IdentifyCheatingCase
  - PrepareAssessmentProcess
  - PrepareExaminationPaper
  - ProcessOfPGRApplication
  - ReviewModuleProcesses
  - SubmitApplications
  - SubmitCourseworkProcesses
  - TerminateOfFixTermContract

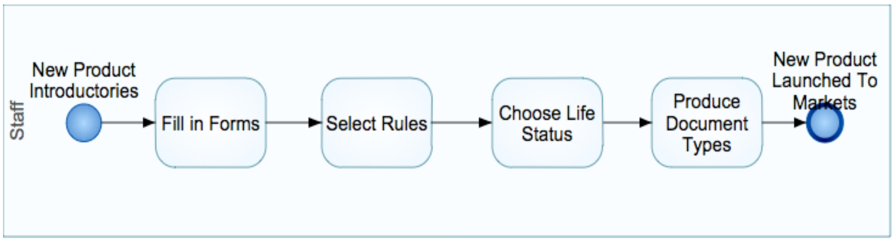


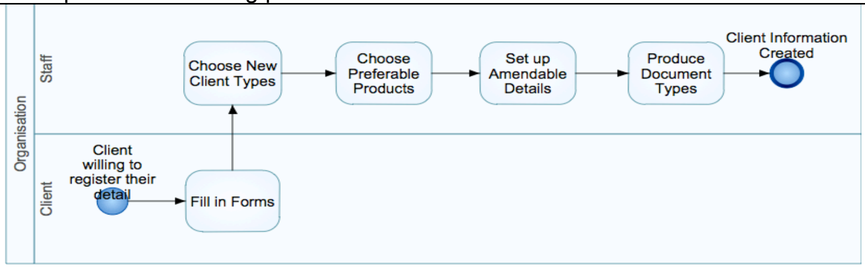
Class Description: BPwithStaffParticipants

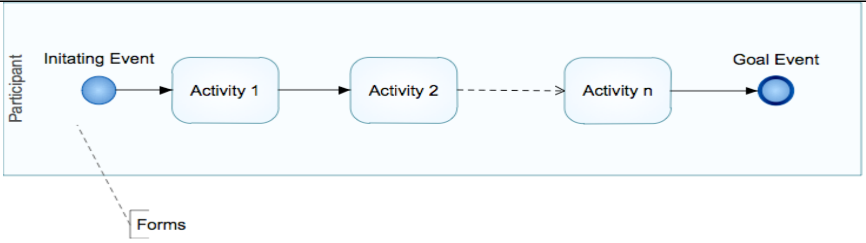
Equivalent classes

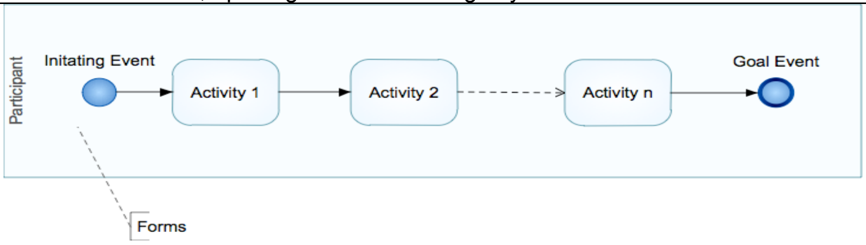
BusinessProcesses
and hasParticipants some Staff

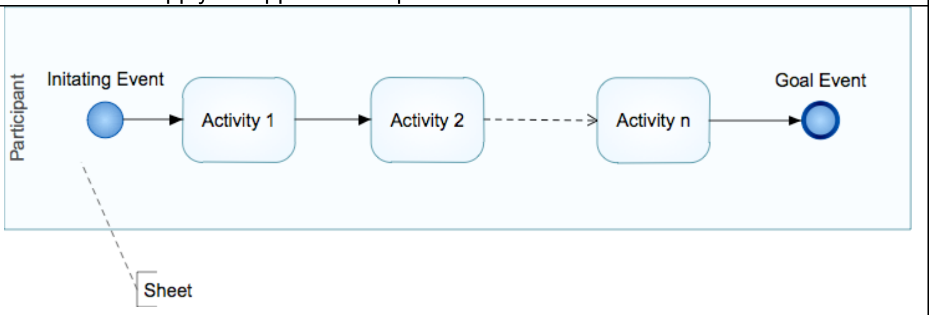
Appendix D BP Patterns Documentation

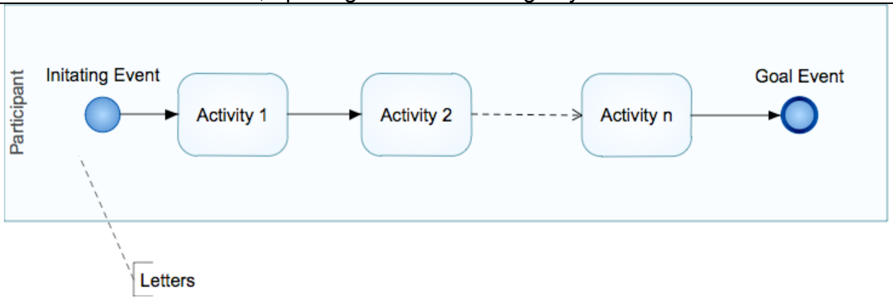
Pattern One	
Name	Define Product Type
Intent	This pattern is a business process pattern that describes the way in which products can be defined, such as new mortgage product or new university course.
Motivation	Suppose that ABC product is required to be defined as a new product at XYZ organisation to keep XYZ organisation up to date with markets need. This ABC product definition has start and end event on when the need for a new product introductory to XYZ organisation take place, and when launching the ABC product to XYZ organisation can be consider fully defined. You define these products to the organisation in the form of sequence of steps. This ensures the accurate representation for the product. Certain rules for the ABC product can be selected. These rules that you set for a product that define its nature. ABC product is set to life status to release the product to the market and make it available for customers use, as well produce document like report, guides, forms that support its definition, this pattern can be used with any type of product definition.
Applicability	Define Product Types pattern lays the foundation for the definition of any product within an organisation in a flexible and high-quality model. The Define Product Types pattern can be implemented to clarify the product structures within an organisation, or to build an information system that define information about product and its structure.
Structure	 Staff New Product Introductories Fill in Forms Select Rules Choose Life Status Produce Document Types New Product Launched To Markets
Participants	Input: Forms Output: Reports Participant: Staff Initiating Event: New product Introductory Goal Event: New product launched to markets Activities: Fill in Forms, Select Rules, Choose Life Status and Produce Document Types.
Consequences	Using Define Product Types pattern provides a proven and a clear architecture for process modelling that facilitates the definition, improvement and amendment of existing or new product in an organisation. Thus, A new product is ready to be launched by the company.

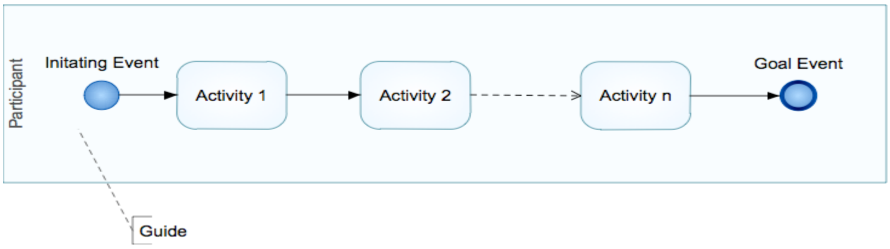
Pattern Two	
Name	Create Client Information
Intent	This pattern is a business process pattern that helps to describe the way on which client details information can be created within an organisation, such as Staff, Clients and Students. Client information is attached to any product in an organisation. Thus, it should well define.
Motivation	This pattern offers a simple way to model client information in any business processes. Interestingly this business processes interact with other business processes, typically via the exchange of resources or information between the processes. For example, this business process patterns need product type creation pattern to be defined so a product can be attached to the creation of the client information pattern.
Applicability	This pattern can be used to model all organisations, as far as the author knows. It is particularly powerful when the definition of client is required. At the university student information is require to be defined to attach it to the defined course at the university. Also, for opening bank account a client pattern is necessary and can be attached to a bank product i.e. saving premium.
Structure	 Organisation Staff Client Choose New Client Types Choose Preferable Products Set up Amendable Details Produce Document Types Client Information Created Client willing to register their detail Fill in Forms
Participants	Input: Forms Output: Reports Participant: Staff, client Initiating Event: Client willing to register their details Goal Event: Client Information Created Activities: Fill in forms, Choose client types, choose preferable products, set up amendable details, produce document types.
Consequences	Organisational models produced using this pattern are built upon a solid foundation for defining their client information that allows for changes in the definition easy to adopt.

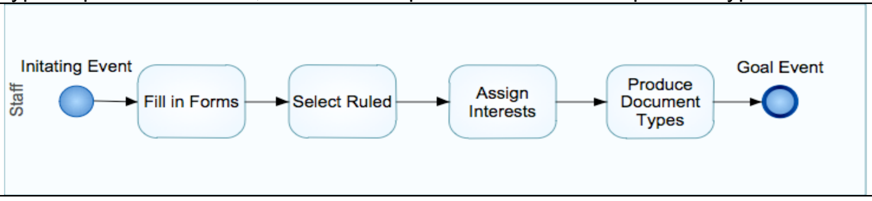
Pattern Three	
Name	BP Uses Forms
Intent	This pattern helps to bring all the processes of an organisation that uses form as an Input to precede its business processes. Forms are used in lots of businesses. The intent of the BP uses forms pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Form is type of a document that is to be used as an input for some business processes and without this form a business process will not be able to proceed in the process, i.e. applying for leave, opening bank account, etc. each of those business processes need special type of form as an input to proceed in their business process.
Applicability	Forms are used in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the form used such as enrolment, opening account and registry.
Structure	 Participant Initating Event Activity 1 Activity 2 Activity n Goal Event Forms
Participants	Input: Forms
Consequences	The advantage of use this pattern is that helps to classify the processes according there type of form they use, so it differentiated from the other processes that uses sheets or guide.

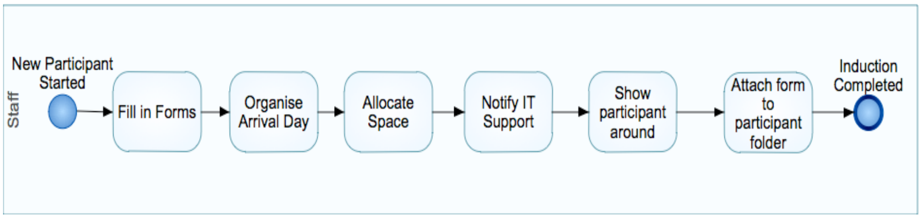
Pattern Four	
Name	BP Consume Forms
Intent	This pattern helps to bring all the processes of an organisation the uses form as an Input to precede its business processes. Forms are used in lots of businesses. The intent of the BP uses forms pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Form is type of a document that is to be used as an input for some business processes and without this form a business process will not be able to proceed in the process, i.e. applying for leave, opening bank account, etc. each of those business processes need special type of form as an input to proceed in their business process.
Applicability	Forms are used in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the form used such as enrolment, opening account and registry.
Structure	 Participant Initating Event Activity 1 Activity 2 Activity n Goal Event Forms
Participants	Input: Forms
Consequences	The advantage of use this pattern is that helps to classify the processes according there type of form they use, so it differentiated from the other processes that uses sheets or guide.

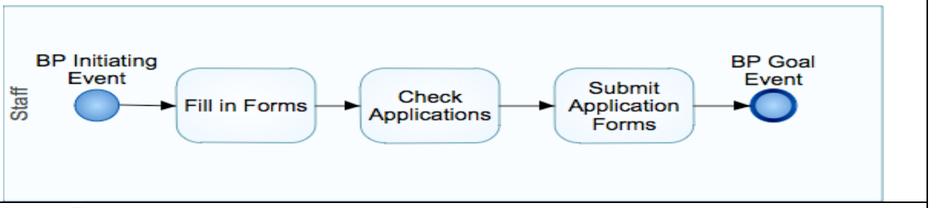
Pattern Five	
Name	BP Consume Sheets
Intent	This pattern helps to bring all the processes of an organisation the uses sheet as an Input to precede its business processes. Sheets are used in lots of businesses. The intent of the BP uses sheets pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Sheet is type of a document that is to be used as an input for some business processes and without this sheet a business process will not be able to proceed in the process, i.e. apply for approval, review module, etc. each of those business processes need special type of sheet as an input to proceed in their business process.
Applicability	Sheets are used in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the sheet used such as apply for approval and process review sheet.
Structure	 Participant Initating Event Activity 1 Activity 2 Activity n Goal Event Sheet
Participants	Input: Sheets
Consequences	The advantage of use this pattern is that helps to classify the processes according there type of sheet they use, so it differentiated from the other processes that uses forms or guide.

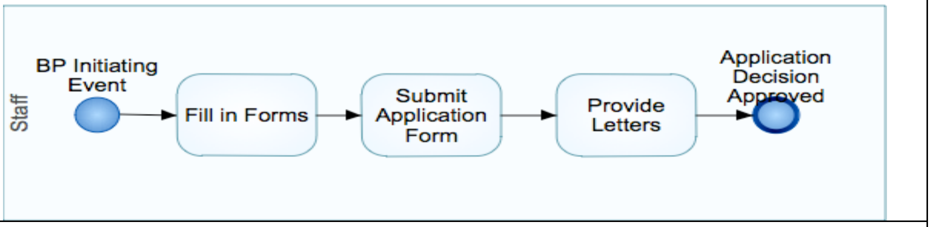
Pattern Six	
Name	BP Consume Letters
Intent	This pattern helps to bring all the processes of an organisation the uses letters as an Input to precede its business processes. Letters are used in lots of businesses. The intent of the BP uses letters pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Letter is type of a document that is to be used as an input for some business processes and without this letter a business process will not be able to proceed in the process, i.e. apply for approval, review module, etc. each of those business processes need special type of letter as an input to proceed in their business process.
Applicability	Letters are used in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the letter used such as enrolment, opening account and registry.
Structure	 Participant Initating Event Activity 1 Activity 2 Activity n Goal Event Letters
Participants	Input: Letters
Consequences	The advantage of use this pattern is that helps to classify the processes according there type of letter they use, so it differentiated from the other processes that uses forms or guide.

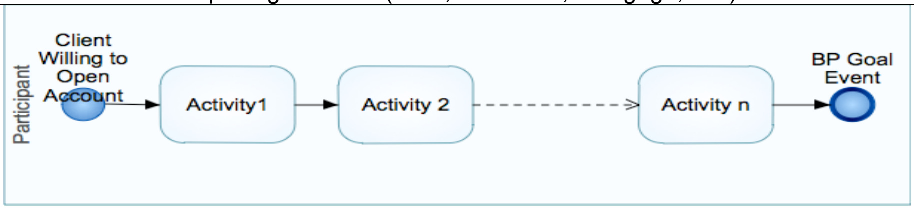
Pattern Seven	
Name	BP Consume Guides
Intent	This pattern helps to bring all the processes of an organisation the uses letters as an Input to precede its business processes. Guides are used in lots of businesses. The intent of the BP uses guides pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Guide is type of a document that is to be used as an input for some business processes and without this guide a business process will miss one of its input to proceed in the process, i.e. arrange staff induction, this business processes need special type of guide as an input to proceed in their business process.
Applicability	Guides are used in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the guide used such as enrolment, and opening account.
Structure	 Participant Initating Event Activity 1 Activity 2 Activity n Goal Event Guide
Participants	Input: Guide
Consequences	The advantage of use this pattern is that helps to classify the processes according there type of guide they use, so it differentiated from the other processes that uses forms or letters.

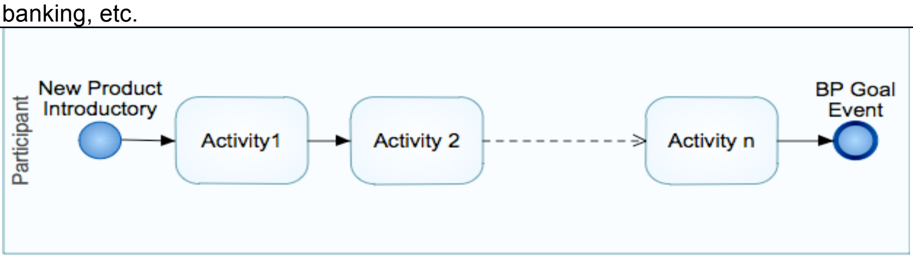
Pattern Eight	
Name	Create Product Types
Intent	This pattern states the steps for creating a product in an organisation, product such as a new course definition at the university, or new type of mortgages to be introduced to the market.
Motivation	This pattern is a business process pattern that describes the way in which products can be created, such as new mortgage product or new university course.
Applicability	Suppose that ABC is a new product that required to be created to meet the market needs at XYZ organisation. This ABC product creation has start and end event. You need to create the new product before enabling its definition and use in the market. The creation in an organisation goes over a sequence of steps. This ensures the accurate creation for the product. Certain rules for the ABC product can be selected. These rules that you set for a product that define its nature. ABC produce document like report, guides, forms that support its creation, this pattern can be used with any type of product creation, which will help for the definition of product type.
Structure	 Staff Initating Event Fill in Forms Select Ruled Assign Interests Produce Document Types Goal Event
Participants	Input: Forms Output: Reports Participant: Staff Activities: Fill in Forms, Select Rules, Assign Interest and Produce Document Types.
Consequences	Using Create Product Types pattern provides a proven and a clear architecture for process modelling that facilitates the creation of a new product and make it ready to be launched by the organisation.

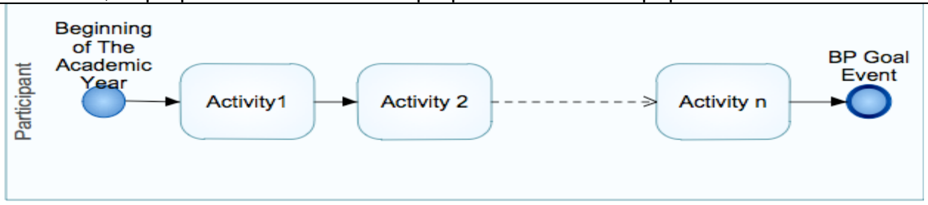
Pattern Nine	
Name	Arrange Induction
Intent	Arrange Induction falls under the Generalise Similar Type Business Processes pattern category. It provides the basic structure for making advance arrangement for welcome and induction of new person in an organisation.
Motivation	This pattern is a business process pattern that suggests a standard format for new participant Induction in an organisation. It should be used as a guide only. The induction arranged should be suitable for the individual's particular role and is dependant on the nature of the member's role e.g. such as arrange new academic staff induction, arrange PhD student induction.
Applicability	These guidelines are designed to assist the manager with the induction of a new member of staff. The aim of induction is to help the newcomer to adjust as quickly as possible to the new working environment, in order to achieve maximum working efficiency in the shortest possible time. It is important to remember that induction is a process that should take place over a number of weeks; it is not a one-day event. The checklists that follow are designed to suggest general issues/ topics that may need to be included in the induction programme in order to assist the new employee settle in as easily and effectively as possible. So this pattern starts when a newcomer starting and end up when full induction provided.
Structure	 Staff New Participant Started Fill in Forms Organise Arrival Day Allocate Space Notify IT Support Show participant around Attach form to participant folder Induction Completed
Participants	Input: Forms Output: Reports Participant: Staff BP Initiating Event: New Participant Started BP Goal Event: Induction Completed Activities: Fill in Forms, Organise Arrival Day, Notify IT Support, Show Participant Around and Attach forms to participant folder.
Consequences	This pattern provides a powerful structure to guide any organisations on what required to be done when new participant or candidate starting their first day in the organisation. By identifying possible actions, and necessary prerequisites, problems can be eliminated and the goals of successful induction of the new candidate can be achieved.

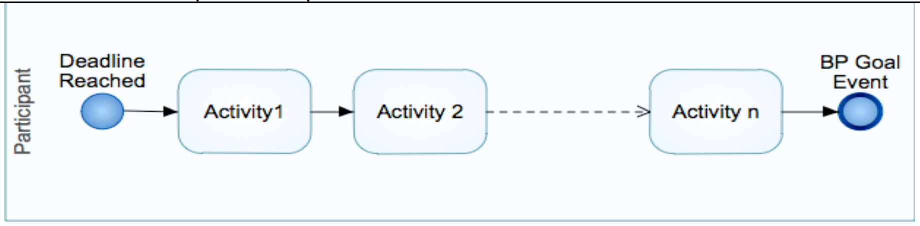
Pattern Ten	
Name	Submit Applications
Intent	Submit Application falls under the Generalise Similar Type Business Processes pattern category. It provides the main basic steps for submitting an application within any organisation.
Motivation	This pattern is a business process pattern that consider to be part of the daily process in many organisations such as banking, universities, NHS, etc.
Applicability	This process pattern is applicable to all situations where the submitting an application is involved, for example, when modelling a student applying for mitigating circumstance in relation to submitting a coursework or attend examination.
Structure	 Staff BP Initiating Event Fill in Forms Check Applications Submit Application Forms BP Goal Event
Participants	Input: Forms Participant: Staff Activities: Fill in Forms, Check Application, and Submit Application Form.
Consequences	This pattern provides a powerful structure to guide any organisations on what are the main steps to submit an application. Then it is down to the organisation to specialise it to the level of detail that meet that organisation needs. By offering only the main there activities in the process help to eliminate the problem of redundant processes and the goals of successful application submission can be achieved.

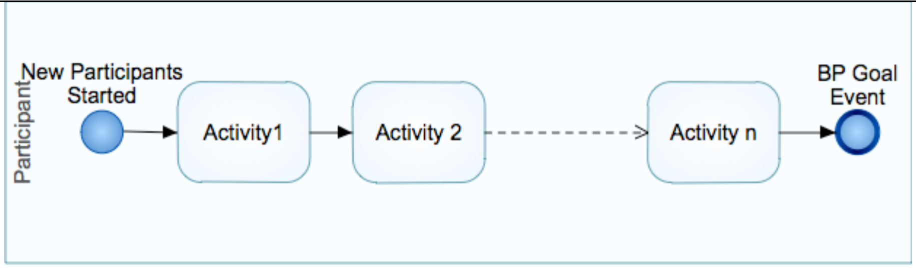
Pattern Eleven	
Name	Apply For Approval
Intent	Apply For Approval falls under the Generalise Similar Type Business Processes pattern category. It provides the main basic steps for apply for approval within any organisation.
Motivation	This pattern is a business process pattern that consider to be part of the daily process in some organisations more than other such as universities doing research, NHS, etc.
Applicability	This process pattern is applicable to all situations where the approval is required when submitting an application. This pattern makes it possible to avoid un necessary step or resources involve with this type of modelling. An example for this type of model that research group is willing to start a new project that requires some approval to commence their research. Thus, this pattern helps that group to be aware of the type of resources and steps to reach the point of receiving approval. This pattern then can be specialised further to meet the policy of that specific organisation.
Structure	 Staff BP Initiating Event Fill in Forms Submit Application Form Provide Letters Application Decision Approved
Participants	Input: Forms Input: Sheets Output: Letter Participant: Staff Activities: Fill in Forms, Submit Application Form, and Provide Letter. BP Goal Event: Application Decision Approved.
Consequences	This pattern provides a guide for any organisations on the steps to be undertaken when applying for approval as well it tells the type of document modeller should expect for and from using this type of processes.

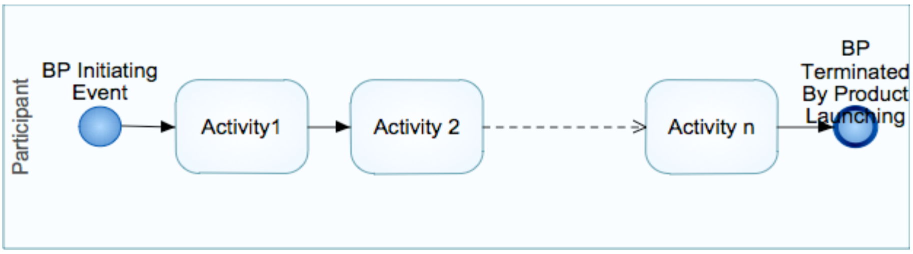
Pattern Twelve	
Name	BP Triggered by Opening Account
Intent	This pattern is a business process pattern that falls under processes of a similar initiating event. This pattern helps to track signification initiating event within an organisation, by bringing all the processes that could take part when opening account event take place to trigger its associated processes.
Motivation	This pattern offers opening account initiating event to support the decision that take place to start related business processes, which will help to modeller to make better business decisions. Often certain actions have to be taken, and things have to be done on or before this opening account event. i.e. consider candidate John Smith approach BBB bank and willing to open an account, having this point the willingness of John Smith initiate an event for open an account process, and by now this event modeller enable the modeller to distinguish what processes can be brought with having this event. This pattern supports the classification of the different processes and eliminates un related processes to take part.
Applicability	This pattern can be used to model all financial organisations that need processes related to account opening, as far as the author knows. It is particularly powerful when the definition of client opening account (bank, insurance, mortgage, etc.).
Structure	 Participant Client Willing to Open Account Activity1 Activity 2 Activity n BP Goal Event
Participants	Initiating Event: Client willing to open account
Consequences	The advantage of using this pattern is that helps to classify the processes according there type of initiating event they use, so it differentiated from the other processes that triggered by different event.

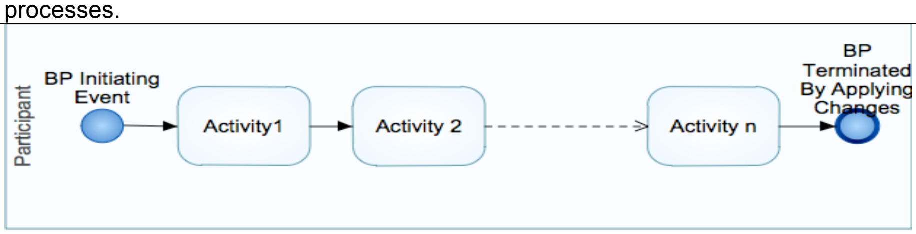
Pattern Thirteen	
Name	BP Triggered By New Product Introductory
Intent	This pattern is a business process pattern that falls under processes of a similar initiating event. This pattern helps to track signification initiating event within an organisation, by bringing all the processes that could take part when new product introductory event take place to trigger its associated processes.
Motivation	This pattern offers a previously defined decision to start this type of processes. An example is that an organisation ABC need a new product a decision has to be taken to enable the initiating event of takes part to trigger related processes.
Applicability	This pattern can be used to model business processes in any organisation, as far as the researcher knows. This because introducing new thing in an organisation is required to keep it up to date with market changes and needs. Examples of when this type of initiating event can be thought of, new courses at the university, new account type in banking, etc.
Structure	 Participant New Product Introductory Activity1 Activity 2 Activity n BP Goal Event
Participants	Initiating Event: New Product Introductory
Consequences	The advantage of using this pattern is that helps to classify the processes according there type of initiating event they use, so it differentiated from the other processes that triggered by different event.

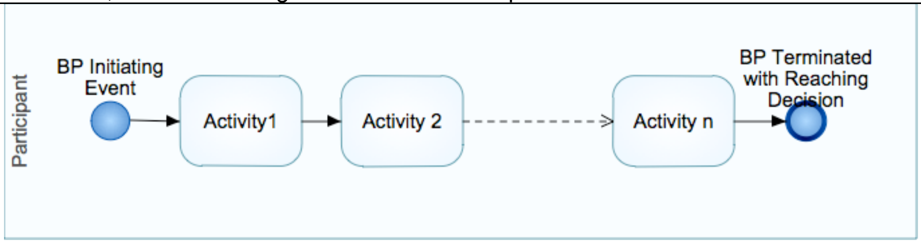
Pattern Fourteen	
Name	Processes Take place at The beginning of The Academic Year
Intent	This pattern is a business process pattern that falls under processes of a similar initiating event. This pattern helps to track signification initiating event within an academic organisation, by bringing all the processes that could take part when the academic year event take place to trigger its associated processes.
Motivation	This pattern offers new academic year initiating event to support the decision that take place to start related business processes, which will help to modeller to make better business decisions. Often certain actions have to be taken, and things have to be done on or before this academic year event. i.e. consider the university ABC is planning its processes that should take part at the beginning of the academic year and any delay in those processes may lead to delay in running the university processes, such as prepare examination paper for courses, appointed for the external examiner. As far as the researcher awareness that this pattern works within the academic domain unless if we were able to discover similar pattern in the financial domain when specific processes should take part at the beginning of the financial year.
Applicability	This pattern can be used to model all educational domain that need processes related to the start of the academic year. It is particularly powerful when the appointed external examiner, or prepare assessment and prepare examination paper.
Structure	 Participant Beginning of The Academic Year Activity1 Activity 2 Activity n BP Goal Event
Participants	Initiating Event: New Academic Year Started
Consequences	The advantage of using this pattern is that helps to classify the processes according there type of initiating event they use, so it differentiated from the other processes that triggered by different event.

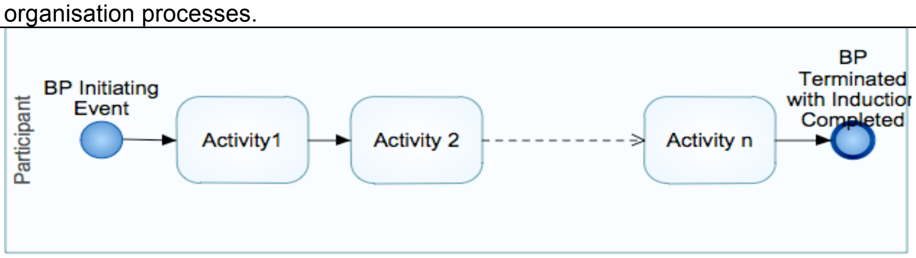
Pattern Fifteen	
Name	BP Trigger When Deadline Reached
Intent	This pattern is a business process pattern that falls under processes of a similar initiating event. This pattern helps to track signification initiating event within any organisation, by bringing all the processes that could take part when deadline event reached to trigger its associated processes.
Motivation	This pattern use to trigger all the business processes that should take part when Due date reached, deadline is the time by which something must be finished or completed. This pattern provide a powerful structure, as due date or deadline event is part of every organization daily processes, such as submit coursework, organise meeting or issue or re-issue for bank card. An example of a process that has been trigger by this type of event is terminating employee contract. This event triggers the termination process to take place where a decision will be made related to the termination process.
Applicability	This pattern is applicable to all domains, as far as the research can predict. It is particularly powerful because this pattern helps modellers to be aware of the type of resources and steps to take part when deadline reached.
Structure	 Participant Deadline Reached Activity1 Activity 2 Activity n BP Goal Event
Participants	Initiating Event: Deadline Reached
Consequences	The advantage of using this pattern is that helps to classify the processes according there type of initiating event they use, so it differentiated from the other processes that triggered by different event.

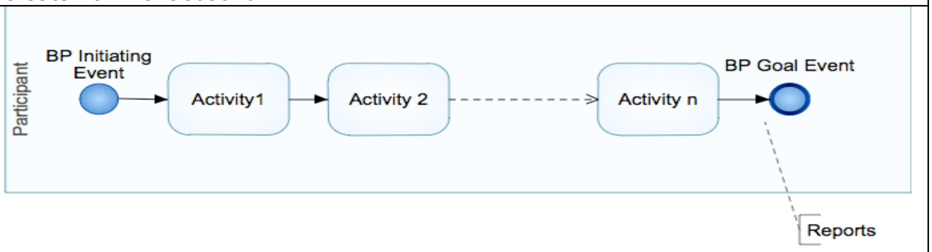
Pattern Sixteen	
Name	BP Trigger When New Participant Started
Intent	This pattern is a business process pattern that falls under processes of a similar initiating event. This pattern helps to track signification initiating event within any organisation, by providing the basic structure of all the processes that the organisation have to trigger when this event start.
Motivation	This pattern uses to trigger all the business processes that should take part when new participant starts, new participant event related to any organisation that use the process of arranging induction for its new starter employee. Such as new lecture starts or new doctor appointed.
Applicability	This pattern is applicable to all domains, as far as the research can predict. It is particularly powerful because this pattern helps modellers to be aware of the type of resources and steps to take part when new participant is starting.
Structure	 Participant New Participants Started Activity1 Activity 2 Activity n BP Goal Event
Participants	Initiating Event: New Participant Started
Consequences	The advantage of using this pattern is that helps to classify the processes according there type of initiating event they use, so it differentiated from the other processes that triggered by different event.

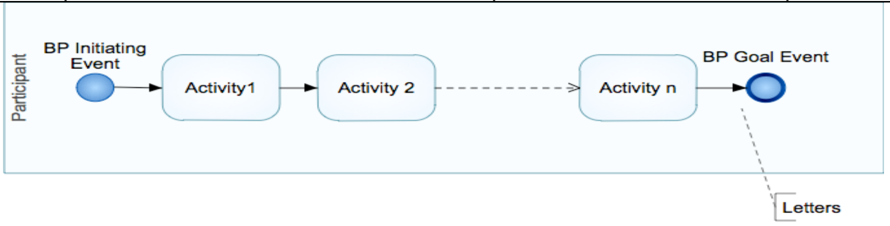
Pattern Seventeen	
Name	BP Terminated By Product Launching
Intent	This pattern is a business process pattern that falls under processes of a similar goal event. This pattern helps to track signification goal event within any organisation, by providing the basic structure of all the processes that the organisations have to trigger when this event start.
Motivation	This pattern uses to terminate all the business processes that should take part when products are launched.
Applicability	This pattern is applicable to all domains that uses product, as far as the research can predict. It is particularly powerful with organisation defining or creating new product and would like to terminate all the process within that organisation when the definition of the creation of a product took part.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Terminated By Product Launching
Participants	Goal Event: Product Launching
Consequences	The advantage of using this pattern is that helps to classify the processes according there type of goal event they use, so it differentiated from the other processes that terminated by different event.

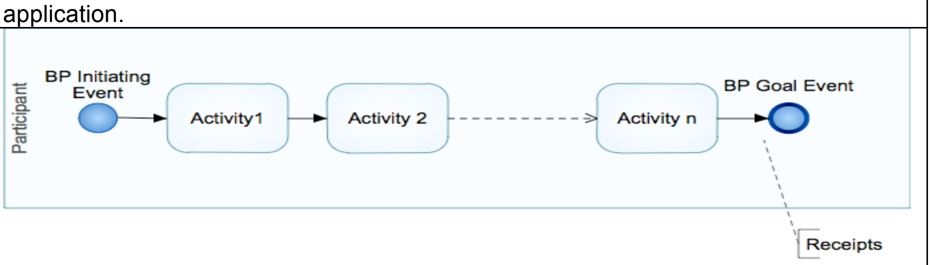
Pattern Eighteen	
Name	BP Terminated By Applying Changes
Intent	This pattern is a business process pattern that brings processes of similar termination event type. In this pattern, applying changes is defined as a goal event that terminate the all business processes once the changes applied for specific organisation process i.e. apply interest rate. If the termination event is not specified, a problem can arise, such as interest rate may not be applied, as the process does not have termination event.
Motivation	Suppose that a specific account type in a banking organisation has decided to change their interest type to meet market changes requirements, and update that account interest rate that might be other processes that take part after or before commencing in this account updating process. Having a termination event i.e. applying changes goal event that as soon as the new interest rate applied the process should be terminated.
Applicability	This pattern solves problems of terminating processes when changes are applied for any processes in any organisations. This pattern can be used as well to classify all the processes that terminated when changes event happened and applied to their processes.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Terminated By Applying Changes
Participants	Goal Event: Changes Applied
Consequences	The advantage in using the pattern is that using termination event brings various organisation processes together. So it differentiated from the other processes that terminated by different event.

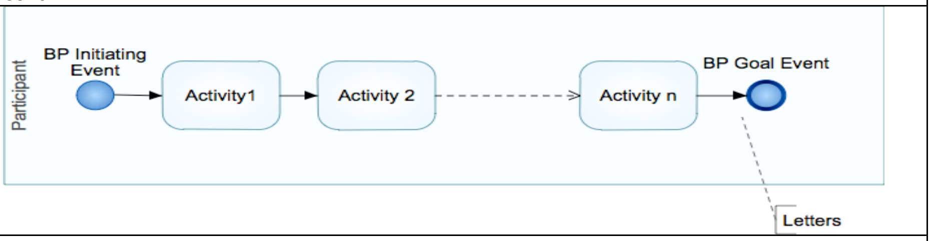
Pattern Nineteen	
Name	BP Terminated With Reaching Decision
Intent	This pattern is a business process pattern that brings processes of similar termination event type. In this pattern, reaching decision is defined as a goal event that terminate the all business processes once the decision for specific organisation process is reached i.e. apply for approval. If the termination event is not specified, a problem can arise, such as what is the end of that approval is a decision reached whether it is positive or negative but at least this event specify the termination of that business process.
Motivation	This pattern uses to terminate all the business processes that took part and had to provide a decision at the end of their processes, reaching decision event related to any organisation that use the process of give a decision at the end of its processes such as apply for ethical approval, apply for leave.
Applicability	This pattern can be used in all businesses that its last activity is providing a decision, all those type of business processes need to be terminated, thus reaching decision goal event support the termination all those processes. This pattern can be used in finance, academic, NHS and all organisations that uses processes with decision.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Terminated with Reaching Decision
Participants	Goal Event: Reaching Decision
Consequences	Using this pattern provides a proven and a clear architecture for process modelling that facilitates the definition, improvement and amendment of reaching decision in an organisation.

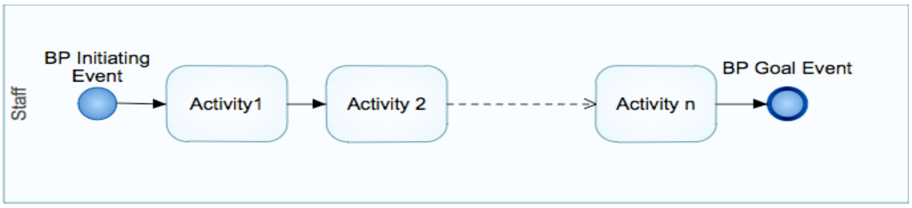
Pattern Twenty	
Name	BP Terminates with Induction Completed
Intent	Arrange Induction falls under the Generalise Similar Business Processes Termination Events types pattern category. It provides the basic structure for terminating the induction of new person in an organisation when the induction completed.
Motivation	This pattern is a business process pattern that suggests a termination for new participant Induction in an organisation when that process completed. e.g. such as arrange new academic staff induction, arrange PhD student induction.
Applicability	This type of pattern is applicable to be used in any organisations; However, it is particularly powerful in organisation that has the induction process as a part of its organisation processes.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Terminated with Inductior Completed
Participants	Goal Event: Induction completed
Consequences	The BP Terminates with Induction Completed pattern facilitates the design of flexible business processes and support systems that handle inductions and recruitments.

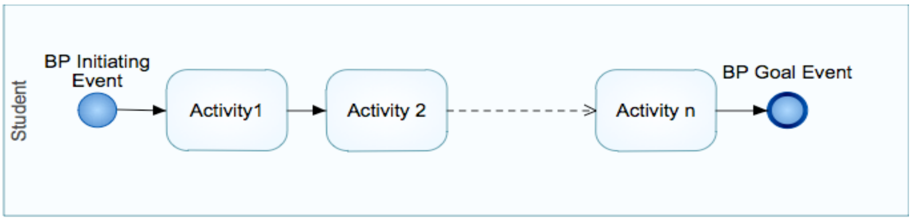
Pattern Twenty One	
Name	BP Produces Reports
Intent	This pattern helps to bring all the processes of an organisation the produces reports as an output to precede its business processes. Reports are used in lots of businesses. The intent of the BP produces report pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Report is type of a document that is to be used as an output or an input for some business processes and without this report a business process will be unable to deliver its outcome of this business, i.e. arrange daily cash return process, this business processes need to produce a specialised type of report to reflect on the amount of cash and related calculation outcome from running this daily cash process.
Applicability	Reports are produced in almost all businesses; this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the report produces as an outcome of its business processes such as Allocate bonuses, create nominal account.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event Reports
Participants	Output: Reports
Consequences	The advantage of use this pattern is that helps to classify the processes according there type of output they use, so it differentiated from the other processes that produce report.

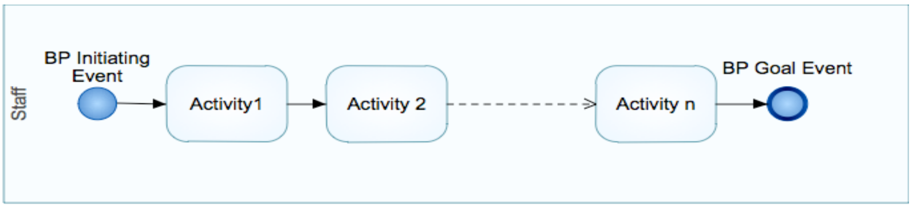
Pattern Twenty Two	
Name	BP Produces Letters
Intent	This pattern helps to bring all the processes of an organisation the produces letters as an output to precede its business processes. Letters are used in lots of businesses. The intent of the BP produces letters pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Letter is type of a document that is to be used as an output or an input for some business processes and without this letter a business process will be unable to deliver its outcome of this business, i.e. process of postgraduate application, this business processes need to produce a specialised type of letter to enable student to finish his/her enrolment as a postgraduate student.
Applicability	Letters are produced in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the letter produces as an outcome of its business processes such as claim expenses.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event Letters
Participants	Output: Letters
Consequences	The advantage of use this pattern is that helps to classify the processes according their type of output they use, so it differentiated from the other processes that produce letters.

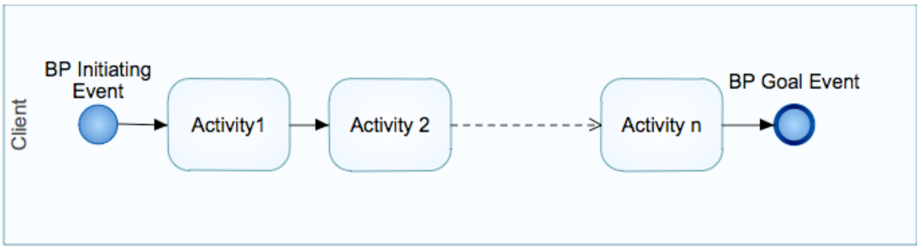
Pattern Twenty Three	
Name	BP Produces Receipts
Intent	This pattern helps to bring all the processes of an organisation the produces receipts as an output to precede its business processes. Receipts are used in lots of businesses. The intent of the BP produces receipts pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	Receipt is type of a document that is to be used as an output for some business processes and without this receipt a business process will be unable to deliver its outcome of this business, i.e. create investment account, this business processes need to produce a specialised type of receipt to enable client to finish his/her account opening.
Applicability	Receipts are produced in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the receipt produces as an outcome of its business processes such as process mortgage application.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event Receipts
Participants	Output: Receipts
Consequences	The advantage of use this pattern is that helps to classify the processes according their type of output they use, so it differentiated from the other processes that produce receipts.

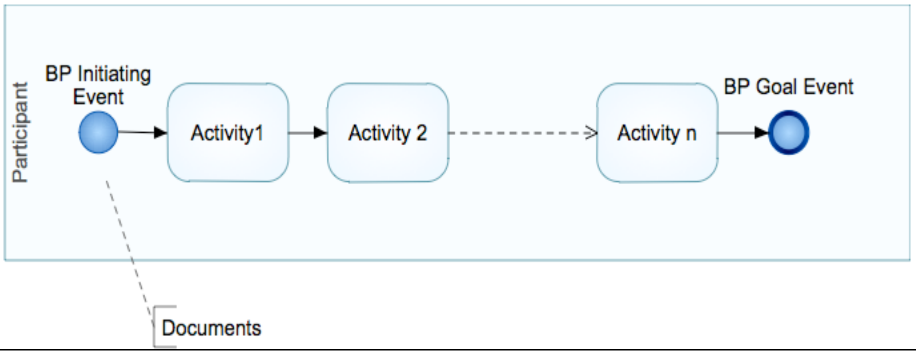
Pattern Twenty Four	
Name	BP Produces Letters
Intent	This pattern helps to bring all the processes of an organisation the produces letters as an output to precede its business processes. Letters are used in lots of businesses. The intent of the BP produces letters pattern is to provide a practical way to approach the issues on which type of document should be use with this business processes, including its different versions and copies.
Motivation	A Letter is type of a document that is to be used as an output or an input for some business processes and without this letter a business process will be unable to deliver its outcome of this business, i.e. process mortgage application, this business processes need to produce a specialised type of letter to enable mortgage application to precede further and get the mortgage approval.
Applicability	Letters are produced in almost all businesses, this pattern can serve as a starting point and either simplified or extended as necessary to fit the processes where the letter produces as an outcome of its business processes such as issue retail client card.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event Letters
Participants	Output: Letters
Consequences	The advantage of use this pattern is that helps to classify the processes according their type of output they use, so it differentiated from the other processes that produce letters.

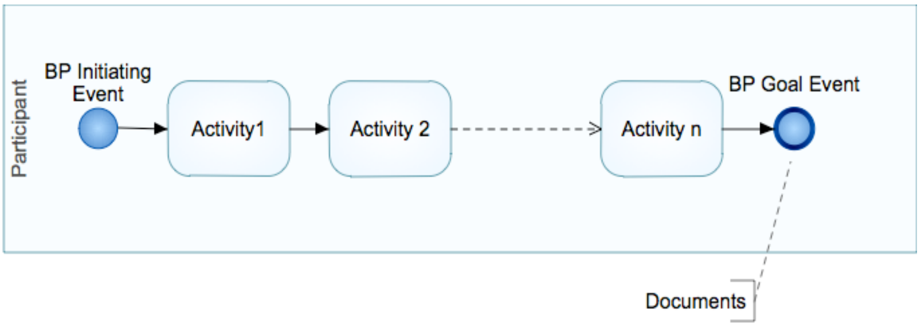
Pattern Twenty Five	
Name	BP with Staff Participants
Intent	The staff participant pattern is a business process pattern that generalised processes according to their participant. It purposes to help structure the organisation processes and assign the most suitable participant to accomplish specific processes.
Motivation	Suppose that a person is employed in an organisation, that person has to be responsible on running some organisation processes. Thus, knowing which staff participant in which process will help to smooth run and tasks achievement within any organisation.
Applicability	This process pattern is applicable to all organisation processes where staff participation in required. This pattern makes it possible to avoid un necessary involvement from staff that do not related to specific processes. An example for this type of model that research group is willing to start a new project that requires some approval to commence their research. Thus, this pattern helps to bring the right staff to do the right job. This pattern then can be specialised further to meet the specific organisation employment structure.
Structure	 Staff BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event
Participants	Participant: Staff
Consequences	Organisational models produced using this pattern are built upon a solid foundation for defining their staff participation to run the organisation various processes and allows for changes in the staff easy to adopt.

Pattern Twenty Six	
Name	BP with Students Participants
Intent	The student participant pattern is a business process pattern that generalised processes according to their participant. It purposes to help structure the organisation processes and assign the most suitable participant to accomplish specific processes.
Motivation	Suppose that a person a student in an academic organisation, that person has to be responsible on running some organisation processes. Thus, knowing which student participant in which process will help to smooth run and tasks achievement within any organisation.
Applicability	This process pattern is applicable to all organisation processes where student participation in required. This pattern makes it possible to avoid un necessary involvement from students that do not related to specific processes. An example for this type of model that phd research group is willing to start a new project that requires some approval to commence their research. Thus, this pattern helps to bring the right student to do the right job.
Structure	 Student BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event
Participants	Participant: Students
Consequences	Organisational models produced using this pattern are built upon a solid foundation for defining their student participation to run the organisation various processes and allows for changes in the student easy to adopt.

Pattern Twenty Seven	
Name	BP Needs Staff Participants
Intent	The staff participant pattern is a business process pattern that generalised processes according to their participant. It purposes to help structure the organisation processes and assign the most suitable participant to accomplish specific processes.
Motivation	Suppose that a person is employed in an organisation, that person has to be responsible on running some organisation processes. Thus, knowing which staff participant in which process will help to smooth run and tasks achievement within any organisation.
Applicability	This process pattern is applicable to all organisation processes where staff participation in required. This pattern makes it possible to avoid un necessary involvement from staff that do not related to specific processes. An example for this type of model that research group is willing to start a new project that requires some approval to commence their research. Thus, this pattern helps to bring the right staff to do the right job. This pattern then can be specialised further to meet the specific organisation employment structure.
Structure	 Staff BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event
Participants	Participant: Staff
Consequences	Organisational models produced using this pattern are built upon a solid foundation for defining their staff participation to run the organisation various processes and allows for changes in the staff easy to adopt.

Pattern Twenty Eight	
Name	BP Needs Client Participants
Intent	The Client participant pattern is a business process pattern that generalised processes according to their participant. It purposes to help structure the organisation processes and assign the most suitable participant to accomplish specific processes.
Motivation	Suppose that a person is a new client in an organisation that person has to take part in some of the organisation processes, such as create client information processes, choose product. A client pattern enables bringing all the processes that has to have a client as a participant to commence some of its business processes.
Applicability	This process pattern is applicable to all organisation processes where client participation in required. This pattern makes it possible to avoid un necessary involvement from client that do not related to specific processes. An example for this type of models that organisation is willing to start a new product, thus it need some client participant to take part in the definition of the product.
Structure	 Client BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event
Participants	Participant: Client
Consequences	Organisational models produced using this pattern are built upon a solid foundation for defining their client participation to run the organisation various processes and allows for changes in the client needs easy to adopt.

Pattern Twenty Nine	
Name	Pattern Utilises Document Types
Intent	Documents are used in all businesses, and they can cause a lot of confusion for modeller. One common problem is which type of documents should be used with specific business processes. Does the process uses form, sheet, etc. or other type of resources.
Motivation	Constructing this pattern helps to bring all business processes and process patterns that use any type of documents as a resource (input) to commence the process work within the organisation. It also provides a way to classify business processes according to the type of resources it uses.
Applicability	Documents consider being part of any organisation daily processes, such as process application, seek agreement, define product, and arrange induction and so on. This pattern can serve as a starting point and either simplified or extended as necessary to fit the situations.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event Documents
Participants	Input: Documents
Consequences	The advantage of using utilises documents pattern is that it helps to understand the different type of document that a specific organisation utilised for its various business processes.

Pattern Thirty	
Name	Pattern Provides Document Types
Intent	Documents are used in all businesses, and they can cause a lot of confusion for modeller. One common problem is which type of documents should be delivered with specific business processes. Does the process deliver letters, receipts etc. or other type of resources.
Motivation	Constructing this pattern helps to bring all business processes and process patterns that deliver any type of documents as a resource (output) to complete the process work within the organisation. It also provides a way to classify business processes according to the type of resources it uses.
Applicability	Documents consider being part of any organisation daily processes, such as process application, seek agreement, define product, and arrange induction and so on. This pattern can serve as a starting point and either simplified or extended as necessary to fit the situations.
Structure	 Participant BP Initiating Event Activity1 Activity 2 Activity n BP Goal Event Documents
Participants	Output: Document
Consequences	The advantage of using provides documents pattern is that it helps to understand the different type of document that a specific organisation delivers for its various business processes. It is a powerful pattern, which can be further specialised to specific type of output like cash daily report.

Appendix E **Business Process Ontology in OWL**

In this appendix presents only BPO in OWL.

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
   <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
   <!ENTITY swrl "http://www.w3.org/2003/11/swrl#" >
   <!ENTITY swrlb "http://www.w3.org/2003/11/swrlb#" >
   <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
   <!ENTITY owl2xml "http://www.w3.org/2006/12/owl2-xml#" >
   <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
   <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
   <!ENTITY protege "http://protege.stanford.edu/plugins/owl/protege#" >
   <!ENTITY xsp "http://www.owl-ontologies.com/2005/08/07/xsp.owl#" >
   <!ENTITY TopLevelOntology "http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#" >
   <!ENTITY BusinessProcessOntology "http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#" >
]>

<rdf:RDF xmlns="http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#"
     xml:base="http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:TopLevelOntology="http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#"
     xmlns:swrl="http://www.w3.org/2003/11/swrl#"
     xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"
     xmlns:owl2xml="http://www.w3.org/2006/12/owl2-xml#"
     xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:BusinessProcessOntology="http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#">
   <owl:Ontology rdf:about="">
      <owl:imports rdf:resource="http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl"/>
   </owl:Ontology>

  <!--
   ///////////////////////////////////////////////////////////////////////////////////////
   //
   // Object Properties
   //
   ///////////////////////////////////////////////////////////////////////////////////////
    -->
   <!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#creates -->

   <owl:ObjectProperty rdf:about="&TopLevelOntology;creates"/>

   <!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#dissolves -->

   <owl:ObjectProperty rdf:about="&TopLevelOntology;dissolves"/>

   <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#hasInput -->

   <owl:ObjectProperty rdf:about="#hasInput"/>

   <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#hasParticipants -->

   <owl:ObjectProperty rdf:about="#hasParticipants">
      <owl:inverseOf rdf:resource="#participatesIn"/>
   </owl:ObjectProperty>

   <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#hasTemporalPart -->

   <owl:ObjectProperty rdf:about="#hasTemporalPart">
      <owl:inverseOf rdf:resource="#temporalPartOf"/>
   </owl:ObjectProperty>

   <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#immediatePrecedes -->
```

```
<owl:ObjectProperty rdf:about="#immediatePrecedes"/>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#isInputOf -->

<owl:ObjectProperty rdf:about="#isInputOf">
  <owl:inverseOf rdf:resource="#hasInput"/>
</owl:ObjectProperty>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#participatesIn -->

<owl:ObjectProperty rdf:about="#participatesIn"/>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#producedBy -->

<owl:ObjectProperty rdf:about="#producedBy"/>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#produces -->

<owl:ObjectProperty rdf:about="#produces">
  <owl:inverseOf rdf:resource="#producedBy"/>
</owl:ObjectProperty>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#temporalPartOf -->

<owl:ObjectProperty rdf:about="#temporalPartOf"/>

<!--
///////////////////////////////////////////////////////////////////////////////////////
//
// Classes
//
///////////////////////////////////////////////////////////////////////////////////////
 -->
<!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#CreationEvents -->

<owl:Class rdf:about="&TopLevelOntology;CreationEvents"/>


<!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#DissolutionEvents -->

<owl:Class rdf:about="&TopLevelOntology;DissolutionEvents"/>

<!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#Events -->

<owl:Class rdf:about="&TopLevelOntology;Events"/>

<!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#Individuals -->

<owl:Class rdf:about="&TopLevelOntology;Individuals">
  <rdfs:subClassOf rdf:resource="&owl;Thing"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#States -->

<owl:Class rdf:about="&TopLevelOntology;States"/>

<!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#TemporalParts -->

<owl:Class rdf:about="&TopLevelOntology;TemporalParts"/>

<!-- http://www.semanticweb.org/ontologies/2010/3/TopLevelOntology.owl#TimeInstants -->

<owl:Class rdf:about="&TopLevelOntology;TimeInstants">
  <rdfs:subClassOf rdf:resource="&owl;Thing"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#Activities -->

<owl:Class rdf:about="#Activities">
  <owl:equivalentClass>
    <owl:Class>
```

```
         <owl:intersectionOf rdf:parseType="Collection">
            <rdf:Description rdf:about="&TopLevelOntology;States"/>
            <rdf:Description rdf:about="#OrganisationalTemporalParts"/>
            <owl:Restriction>
               <owl:onProperty rdf:resource="#temporalPartOf"/>
               <owl:someValuesFrom rdf:resource="#BusinessProcesses"/>
            </owl:Restriction>
         </owl:intersectionOf>
      </owl:Class>
   </owl:equivalentClass>
   <owl:disjointWith rdf:resource="#BPEvents"/>
   <owl:disjointWith rdf:resource="#BusinessProcesses"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#BPEvents -->

<owl:Class rdf:about="#BPEvents">
   <owl:equivalentClass>
      <owl:Class>
         <owl:intersectionOf rdf:parseType="Collection">
            <rdf:Description rdf:about="&TopLevelOntology;Events"/>
            <rdf:Description rdf:about="#OrganisationalTemporalParts"/>
            <owl:Restriction>
               <owl:onProperty rdf:resource="#temporalPartOf"/>
               <owl:someValuesFrom rdf:resource="#BusinessProcesses"/>
            </owl:Restriction>
         </owl:intersectionOf>
      </owl:Class>
   </owl:equivalentClass>
   <owl:disjointWith rdf:resource="#BusinessProcesses"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#BPGoalEvents -->

<owl:Class rdf:about="#BPGoalEvents">
   <owl:equivalentClass>
      <owl:Class>
         <owl:intersectionOf rdf:parseType="Collection">
            <rdf:Description rdf:about="&TopLevelOntology;DissolutionEvents"/>
            <rdf:Description rdf:about="#BPEvents"/>
            <owl:Restriction>
               <owl:onProperty rdf:resource="&TopLevelOntology;dissolves"/>
               <owl:someValuesFrom rdf:resource="#BusinessProcesses"/>
            </owl:Restriction>
         </owl:intersectionOf>
      </owl:Class>
   </owl:equivalentClass>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#BPInitiatingEvents -->

<owl:Class rdf:about="#BPInitiatingEvents">
   <owl:equivalentClass>
      <owl:Class>
         <owl:intersectionOf rdf:parseType="Collection">
            <rdf:Description rdf:about="&TopLevelOntology;CreationEvents"/>
            <rdf:Description rdf:about="#BPEvents"/>
            <owl:Restriction>
               <owl:onProperty rdf:resource="&TopLevelOntology;creates"/>
               <owl:someValuesFrom rdf:resource="#BusinessProcesses"/>
            </owl:Restriction>
         </owl:intersectionOf>
      </owl:Class>
   </owl:equivalentClass>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#BPparticipants -->

<owl:Class rdf:about="#BPparticipants">
   <owl:equivalentClass>
      <owl:Class>
         <owl:intersectionOf rdf:parseType="Collection">
```

```
                <rdf:Description rdf:about="#Persons"/>
                <owl:Restriction>
                    <owl:onProperty rdf:resource="#participatesIn"/>
                    <owl:someValuesFrom rdf:resource="#BusinessProcesses"/>
                </owl:Restriction>
            </owl:intersectionOf>
        </owl:Class>
    </owl:equivalentClass>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#BusinessProcesses -->

<owl:Class rdf:about="#BusinessProcesses">
    <rdfs:subClassOf rdf:resource="#OrganisationalTemporalParts"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#hasTemporalPart"/>
            <owl:someValuesFrom rdf:resource="#BPInitiatingEvents"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#produces"/>
            <owl:someValuesFrom rdf:resource="#Outputs"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#hasParticipants"/>
            <owl:someValuesFrom rdf:resource="#BPparticipants"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#hasTemporalPart"/>
            <owl:someValuesFrom rdf:resource="#BPGoalEvents"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#hasInput"/>
            <owl:someValuesFrom rdf:resource="#Inputs"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#hasTemporalPart"/>
            <owl:someValuesFrom rdf:resource="#Activities"/>
        </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#Inputs -->

<owl:Class rdf:about="#Inputs">
    <owl:equivalentClass>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="&TopLevelOntology;Individuals"/>
                <owl:Restriction>
                    <owl:onProperty rdf:resource="#isInputOf"/>
                    <owl:someValuesFrom rdf:resource="#Activities"/>
                </owl:Restriction>
            </owl:intersectionOf>
        </owl:Class>
    </owl:equivalentClass>
    <owl:equivalentClass>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="&TopLevelOntology;Individuals"/>
                <owl:Restriction>
                    <owl:onProperty rdf:resource="#isInputOf"/>
```

```
                        <owl:someValuesFrom rdf:resource="#BusinessProcesses"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
    </owl:Class>

    <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#OrganisationalTemporalParts -->

    <owl:Class rdf:about="#OrganisationalTemporalParts">
        <rdfs:subClassOf rdf:resource="&TopLevelOntology;TemporalParts"/>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="#temporalPartOf"/>
                <owl:someValuesFrom rdf:resource="#Organisations"/>
            </owl:Restriction>
        </rdfs:subClassOf>
    </owl:Class>

    <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#Organisations -->

    <owl:Class rdf:about="#Organisations">
        <rdfs:subClassOf rdf:resource="#Persons"/>
    </owl:Class>

    <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#Outputs -->

    <owl:Class rdf:about="#Outputs">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="&TopLevelOntology;Individuals"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#producedBy"/>
                        <owl:someValuesFrom rdf:resource="#Activities"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="&TopLevelOntology;Individuals"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="#producedBy"/>
                        <owl:someValuesFrom rdf:resource="#BusinessProcesses"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
    </owl:Class>

    <!-- http://www.semanticweb.org/ontologies/2010/5/BusinessProcessOntology.owl#Persons -->

    <owl:Class rdf:about="#Persons">
        <rdfs:subClassOf rdf:resource="&TopLevelOntology;Individuals"/>
    </owl:Class>

    <!-- http://www.w3.org/2002/07/owl#Thing -->
    <owl:Class rdf:about="&owl;Thing"/>
</rdf:RDF>
<!-- Generated by the OWL API (version 2.2.1.842) http://owlapi.sourceforge.net -->
```