SEMANTIC DISCOVERY AND REUSE OF BUSINESS PROCESS PATTERNS

Laden Aldin, Sergio de Cesare and Mark Lycett
Department of Information Systems and Computing
Brunel University
Uxbridge, U.K.
Email: {laden.aldin|sergio.decesare|mark.lycett}@brunel.ac.uk

Abstract. Patterns currently play an important role in modern information systems (IS) development and their use has mainly been restricted to the design and implementation phases of the development lifecycle. Given the increasing significance of business modelling in IS development, patterns have the potential of providing a viable solution for promoting reusability of recurrent generalized models in the very early stages of development. As a statement of research-in-progress this paper focuses on business process patterns and proposes an initial methodological framework for the discovery and reuse of business process patterns within the IS development lifecycle. The framework borrows ideas from the domain engineering literature and proposes the use of semantics to drive both the discovery of patterns as well as their reuse.

Keywords: Pattern, information systems development, business process pattern, domain engineering.

1 INTRODUCTION

Business modelling is assuming increasing significance in information systems (IS) development. Evidence of this phenomenon is highlighted, for example, by the introduction of a business modelling phase in methodologies like the Rational Unified Process, the recent definition of the Business Process Modelling Notation (BPMN) and the emergence of service-oriented approaches in which services are combined to realize business processes. Despite these positive signs modelling business processes remains problematic due to the evolutionary nature of organizations. Business processes evolve throughout an organization's lifetime in order to meet dynamic and changing business requirements (Hammer and Champy, 2001). It is essential that such changes are represented systematically and their impact is clearly understood (Morgan, 2007). When developing computer-based information systems, it is necessary to understand the role they play in giving support to their business context. To reach such understanding there is a need to create business process models (Lindsay et al., 2003). Business process modelling (BPM) is frequently used to control the execution of organizational processes and to ensure consistency and thoroughness in capturing relevant processes to improve efficiency and productivity (Aguilar-Savén, 2004). The achievement of greater agility and flexibility within BPM represents a key goal for organizations. One of the reasons that impede BPM to achieve this goal is the lack of systematic reuse of business models. In IS development business modellers may encounter similar and recurrent patterns of behaviour. Being able to reuse previously modelled behaviour can have a beneficial impact on the quality and efficiency of the overall IS development process and also improve the effectiveness of an organization's business processes (Ericksson and Penker 2000, Caetano et al. 2005).

The representation of organizational processes has been the focus of much research in past years. Only some of it has focused on modelling business-related patterns (Kaisler, 2005). This paper provides a contribution in this sense. More specifically, this study focuses on business process

patterns. A pattern is a reusable model of a solution to a recurrent class of problems. It offers a solution based on previous success in resolving a similar type of business problem. The aim of this research is to develop a methodological framework for empirically deriving ontological patterns of business processes from organizational knowledge sources.

Given the above, this paper presents early outcomes of research in progress, which develops a research agenda to direct work on business process patterns discovery and reuse. In achieving this aim, the paper is structured as follows: the following section provides an overview of the background related to patterns in IS development and business process modelling. Section 3 presents an initial proposition of a semantic-based framework for the identification of business process patterns as well as their reuse. Finally, section 4 presents conclusions and an outline of future.

2 BACKGROUND

The concept of patterns was introduced by the architect Christopher Alexander in 1977. Alexander et al. (1977) refer to patterns in the following way: "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". 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 et al. (1999) later adopted the term archetype to indicate "a form from which all classes of the same kind more or less follow" (p.3). 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 organizations. 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 underdeveloped. Furthermore Fowler's work tends to be directed toward software designers. As a result his analysis patterns in many areas refer more to software artefacts rather than to generic business domain structures and behaviour.

Eriksson and Penker (2000) later developed a set of business patterns, which came closer to a generic representation of organizational structures and processes. Although these patterns like the previous (Fowler and Hay) are ultimately aimed toward the facilitation of realizing software artefacts that will help to effectively and efficiently develop and 'run' information systems, Eriksson and Penker's business patterns are modelled and described from a perspective that is closer to that of the enterprise rather than the software developer.

More recently there has 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. (2003) produced a set of so called workflow patterns. 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 of van der Aalst et al. (2003) (2000) 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 organizational processes.

Besides the debatable business nature of the patterns discussed above, a more important limitation can be identified. In the patterns literature the way in which patterns are discovered is not clear. The literature states that patterns derive from experience and that a model constitutes a pattern if it has been used in multiple instances to resolve the same type of problem. Within the business domain, knowledge and experience tends to be dispersed among diverse and numerous sources (e.g., people, documents, legacy applications, designs and data, etc.). Often such knowledge is implicit and/or even informal and business behaviour is not just designed, but is in good part emergent.

With more and more researchers and practitioners recognizing the importance of reusability in business process modelling (Di Duo, 2007), it is essential to explore new viable solutions that can provide successful ways to reuse. This paper proposes the adoption of semantics in order to discover new business process patterns and subsequently apply such patterns when modelling businesses. This study aims at overcoming two problems with previous solutions: (1) as highlighted above, limited work has been carried out by other authors on business processes patterns, and (2) none of the previous work provides guidelines to modellers as to how business process patterns can be discovered. The following section proposes a semantic-based methodological framework that can help overcome such problems.

3 SDR FRAMEWORK

This paper proposes a methodological framework for the semantic discovery and reuse of business process patterns. Patterns are initially discovered from legacy sources and then applied during business modelling. The framework is based on a dual lifecycle model as proposed by the domain engineering literature (Prieto-Daz, 1990). This model defines two interrelated lifecycles (Figure 1): (1) a lifecycle aimed at generating business process patterns and (2) a lifecycle aimed at producing business process models. To model an organization in terms of its information rather than simply the data flowing through it requires understanding of the meaning of that information, its semantics. Semantics play a key role in this framework and are modelled through ontologies. While ontologies are used to represent the process patterns in the former lifecycle, the patterns' semantics then drive subsequent business modelling efforts during the latter lifecycle.

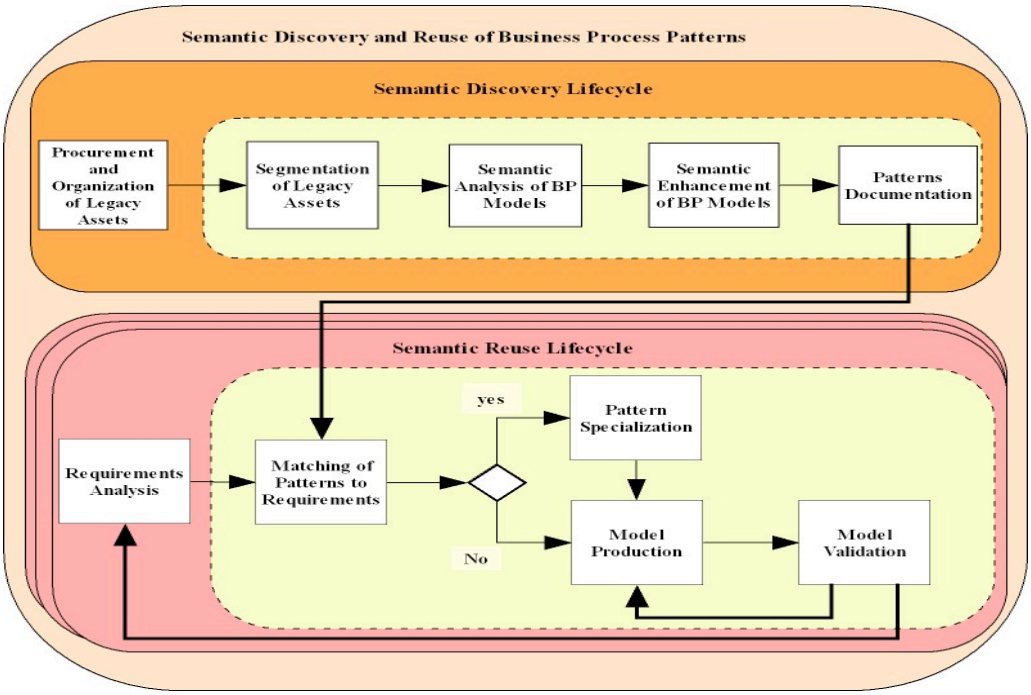

Fig. 1. *SDR Framework (adapted from (Foreman, 1996))*

Theoretically speaking a semantics-based approach to modelling must ensure that there is evidence of mapping between elements of a model and the real-world things that those modelling elements refer to. This concept of mapping is integral to most definitions of semantics whereby there is a relation between a signifier (sign or symbol) and the signified (the thing being represented). Evidence of such mapping within the proposed framework derives from legacy source data. In this study legacy sources represent any body of knowledge (system application data, documentation, models, expert knowledge, observations, etc.), which provides confirmation of the existence of certain behaviour and types of behaviour in an organization. For example, from organizational documentation of a bank a modeller may elicit behaviour corresponding to the withdrawal of money from an account. This behaviour can be detailed into a series of steps that lead to a certain outcome (e.g., an account being debited).

The Semantic Discovery Lifecycle (SDL) initiates with the procurement and organization 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 are as follows:

- *Procurement and Organization of Legacy Assets (POLA)*: 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 generalized recurrence across multiple organizations. SDL, in this sense, is similar to the way scientific theories are discovered from scientific data. Only model types which have been previously and demonstrably adopted by organizations and/or workflow systems can be modelled and become part of the patterns repository. Therefore, acquiring legacy assets and organizing them in a repository is an essential initial step.
- *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.
- *Semantic Analysis of BP Models (SA)*: This phase along with the following represent the core of SDL. In SA business process models are extracted from the legacy asset fragments. These models

are typical process flow diagrams such as UML activity diagrams or BPMN diagrams. The elements of the process diagrams are then semantically interpreted in order to derive more precise ontological models of the processes themselves.

- *Semantic Enhancement of BP Models (SE)*: This phase takes the ontological models created in SA and aims at generalizing them to existing patterns or to newly developed patterns.
- *Pattern Documentation (PD)*: The pattern(s) derived from a cycle of SDL are finally documented and catalogued in the patterns repository.

The 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 1 are purely indicative. An organization can adopt any business modelling process it prefers 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 specialization) 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 organizations. In this case the organization performing the SDL would be specialized in the management and supply of process patterns, while its clients would consume the discovered patterns. The typical phases of the SRL 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 Specialization*: 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.

4 CONCLUSIONS AND FUTURE WORK

This research-in-progress paper presented a methodological framework for the semantic discovery and reuse (SDR) of business process patterns. The framework defines a dual lifecycle model. The first lifecycle is aimed at deriving business process patterns from legacy content through the use of ontologies. The second lifecycle is aimed at business modelling and reuses the patterns defined in the previous lifecycle.

The SDR methodological framework overcomes two limitations of previous research on business process patterns. Firstly, the workflow patterns defined by van der Aalst et al. (2003) model common control structures of workflow languages are not aimed at modelling generic processes of a business domain (like an industrial sector). Secondly, the patterns research community to date has dedicated limited attention to the process of patterns discovery. The unique features of the SDR methodological framework are its dual lifecycle model, its use of semantics and the grounding in real world legacy

models and data to derive the patterns. This last point is of particular importance because it underlines the fact that the modelled patterns must be based on evidence of their actual existence.

The work presented here is ongoing. The following phases of our research will be to: (1) continue discovering business process patterns from legacy systems; (2) continually test the existing patterns against legacy models and data and (3) define a maturity model of business process patterns based on the type of testing that the patterns have undergone (e.g., tested against one legacy system, against multiple systems of one domain and, finally, multiple systems across multiple domains).

REFERENCES

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

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

Beck, K. & and Cunningham, W. 1987, "Using Pattern Languages for Object-Oriented Program", *Workshop on Specification and Design for Object-Oriented Programming* OOPSLA.

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* ACM, New York, NY, USA, pp. 1308.

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 Modeling Color with Uml: Enterprise Components and Process with CDrom,* Prentice Hall PTR Upper Saddle River, NJ, USA.

Di Dio, G. 2007, "ARWOPS: A Framework for Searching Workflow Patterns Candidate to be Reused", *Internet and Web Applications and Services. ICIW '07. Second International Conference* IEEE CNF, Internet and Web Applications and Services. ICIW '07. Second International Conference, pp. 33.

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

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

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

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

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

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

Kaisler, S.H. 2005, *Software paradigms,* Hoboken, N.J. John Wiley, c2005.

Lindsay, A., Downs, D. & Lunn, K. 2003, "Business processes—attempts to find a definition", *Information and Software Technology,* vol. 45, no. 15, pp. 1015-1019.

Morgan, T. 2007, "Business Process Modeling and ORM", *LECTURE NOTES IN COMPUTER SCIENCE,* vol. 4805, pp. 581.

Prieto-Daz, R. 1990, "Domain analysis: an introduction", *SIGSOFT Software Engineering Notes,* vol. 15, no. 2, pp. 47-54.

Thom, L.H., Iochpe, C. & Reichert, M. 2007, "Workflow Patterns for Business Process Modeling", *Proceedings of Workshops and Doctoral Consortium of the 19th International Conference on Advanced Information Systems Engineering (CAiSE),* vol. 1, pp. 349-358.

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

van der, A., ter Hofstede, A., Kiepuszewski, B. & Barros, A. 2003, "Workflow Patterns", *Distributed and Parallel Databases,* vol. 14, no. 1, pp. 5-51.