Association for Information Systems
AIS Electronic Library (AISeL)

MCIS 2011 Proceedings

Mediterranean Conference on Information Systems (MCIS)

2011

SEMANTIC REUSE OF BUSINESS PROCESS MODELS VIA GENERALISATION

Laden Aldin
Department of Information Systems and Computing Brunel University Uxbridge, laden.aldin@brookes.ac.uk

Sergio de Cesare
Brunel University, sergio.deCesare@Brunel.ac.uk

SEMANTIC REUSE OF BUSINESS PROCESS MODELS VIA GENERALISATION

Laden Aldin, Oxford Brookes University, School of Technology, Department of Computing, Wheatley Campus, Wheatley, Oxford, OX33 1HX, U.K., laden.aldin@brookes.ac.uk

Sergio de Cesare, Brunel University, Department of Information Systems and Computing, Uxbridge, Middlesex, UB8 3PH, U.K., sergio.decesare@brunel.ac.uk

Abstract

Business process models do not explicitly guide their reuse process. An important conceptual mechanism for enabling reuse is generalisation. Generalisation (or specialisation) enables a high level of flexibility and variability when classifying business process models at different levels of abstraction. Object-oriented analysis and design methods take full advantage of the generalisation hierarchy when it comes to modelling the classes in a system. As a result, the motivation of this paper is to propose an approach that facilitates reuse of business process models through generalisation. The approach proposed adopts a business process ontology to drive the generalisation. Six types of process generalisation are identified and their use demonstrated. Finally, a critical discussion of the approach is presented along with suggestions for improvement and future work.

Keywords: Business process, generalisation, semantics, business process model.

1 INTRODUCTION

In modern enterprises, global competition and increasing customer expectations require organisations to increase the levels of efficiency while maintaining flexibility to streamline the changes in their environment. As a consequence organisations must improve their understanding and management of their business process models so as to identify the potential for process model reusability. While reusability has been widely researched and applied to varying levels of success in software engineering, its application to the management and modelling of business processes has been minimal. The different emphasis on reuse in Business Process Management (BPM), as opposed to software engineering, may be due to the longer history of software engineering as a recognised discipline as well as the limited penetration of BPM as a systematic and important activity within and across organisations in these past years. This paper focuses on generalisation/specialisation as a particular mechanism for enabling and facilitating the reuse of process models.

In general terms generalisation/specialisation is similar to super/sub set relationships in set theory whereby if A is a subset of B then any member of A is also a member B ($A \subseteq B$). In the Unified Modelling Language (UML) generalisation is defined as: “A taxonomic relationship between a more general and a more specific element. The more specific element is fully consistent with the more general element and it contains additional information” (Rumbaugh et al., 2004, p.370). Such a definition complies with the principles of the object-oriented paradigm in which subclasses inherit the attributes, methods and relationships of their respective superclasses.

While generalisation (as a relationship between classes) is commonly applied to so-called static or structural models, such as UML class diagrams, this modelling construct is seldom applied to behaviour; for example, business processes (as a form of organisational behaviour) or Web services (as a form of software behaviour). In relation to business processes there are certain challenges that make generalisation more difficult. First, a definition of business process must be adopted (or agreed). This represents the starting point for being able to identify the commonalities between two different processes. While a process is generally defined of a set of interrelated activities that transform inputs into outputs in order to achieve a specified goal, such a definition needs to be refined so as to extract those specific elements typical to organisational behaviour that form the basis of identifying commonalities. Second, generalisation of processes can be defined in multiple ways, as the literature demonstrates. These different ways can be either theoretically rooted in the definition of generalisation/specialisation itself (Wyner and Lee, 2002) or more pragmatically in the selection of process elements that the business architect deems most appropriate in relation to the domain in which the organisation operates (this will be explained further in the paper). This latter and more pragmatic conception of business process generalisation forms the core of this work and it recognises the principle of interpreting models based on their real-world (rather than formal) semantics.

Consequently, the research is aimed at identifying different types of process generalisation that a business architect could adopt with the goal of increasing model reusability. The research approach adopted conforms to the principles of Design Science (Hevner et al., 2004) whereby the development of artefacts (conceptual or physical) contribute to resolving a recognised problem. The research proceeded as follows: (1) development of a Business Process Ontology (BPO) derived from the literature and modelled with the principle of conciseness in mind, (2) apply combinations of the modelled BPO elements to generalise previously developed process models, (3) critically discuss the types of business process generalisation identified and applied.

The paper is structured as follows. Section 2 discusses existing business processes generalisation approaches. Section 3 clarifies the meaning of business process and defines the concepts used for generalising the processes themselves. These concepts form the basis of the proposed BPO. Section 4 proposes different types of business process generalisation. A discussion then follows on the generalisation types identified along with conclusions and avenues for further research.

2 BACKGROUND

The subject of business process generalisation/specialisation represents a topic that has been discussed at various stages among information systems (IS), organisational theory, and business process modelling researchers and practitioners.

The MIT Process Handbook project started in 1991 with the aim to establish an online library for sharing knowledge about business processes. The process knowledge in the Process Handbook presented a redesign methodology based on concepts such as process specialisation, dependencies, and coordinating mechanisms (Malone et al., 2003). 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’. Furthermore, the MIT Process Handbook can similarly be seen as a catalogue of common business patterns. The patterns movement can be seen to provide a set of ‘good practice’ building blocks that extend well beyond software development to describe implementation solutions for generic business process problems. 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. Finally, the MIT process handbook has inspired several projects, among them Peristeras and Tarabanis (2000) used the MIT Process Handbook to propose a Public Administration General Process Model.

Pentland (2003) describes business process as a ‘generative structure’. It is not fixed, but varies according to type of input, personnel involved in execution, etc. Pentland suggests that variety in business processes can be described in three dimensions: (1) variety in the range of tasks performed (task variety), (2) variety in the order that these tasks are performed in (sequential variety) and (3) variety in the inputs and outputs of the process (content variety). The work is carried out across four sub-units of a major US bank to demonstrate that the sub-units, which had high task variety, also had low sequential variety while those with low task variety had high sequential variety. Pentland concludes in his study that process variation is not in itself a problem, there may be many ways of achieving a desired outcome successfully and there may be a genuine need to respond to local circumstances in different ways.

Dumas and García-Bañuelos (2009) define similarity search queries with respect to a similarity measure between pairs of process models. The similarity of process models is measured on the basis of three complementary aspects of process models: (1) the labels attached to tasks, events and other model elements, (2) their graph structure (Structural Similarity), and (3) their execution semantics (Behavioural Similarity). The work is carried out using 100 process models. The conclusions of their work were that existing process model similarity search techniques focus on process models composed of atomic tasks and connectors, while little attention was paid to other process modelling constructs such as sub-process invocation and exception handlers. Furthermore limiting is the fact that existing process model similarity search techniques tend to focus on the control-flow view of process models, neglecting data manipulation (e.g. data inputs/outputs) and resource allocation (Dumas and García-Bañuelos, 2009).

Finally, the review of business process modelling generalisation/specialisation literature highlights the existence of some critical points in existing approaches as presented in the following points:

(1) The shortcomings of ad-hoc description languages offered by the MIT process handbook is preventing it from fully supporting collaboration, synchronisation, sharing and analysis, and certainly not execution (Handayan, et al., 2009).

(2) Different Representations: As there is no single standard for modelling a business process, despite of the various modelling languages available (e.g., BPMN), different organisations will probably use various notations. Thus, if generalisation of processes was dependent only on its graphical labels, this will lead to limitations in understanding of the process elements.

(3) Different Constructs: Existing approaches to process modelling lack an adequate specification of the semantics of the terminology of the underlying process models, which leads to inconsistent interpretations and usage of knowledge. People from different departments or organisations do not always use the same vocabulary for entities of the real world. “Also in different companies the identifiers for process actions are different: ‘Pay an account’ in one company might be the same as ‘Settle a bill’ with both addressing an ‘invoice’ ” (Lautenbacher et al., 2008).

(4) The limitation of most approaches is to focus on identifying, analysing and measuring current processes to redesign a new model. This results in incremental and non-sensible improvements (Stoddard and Jarvenpaa, 1995) since redesign is too oriented towards modelling the “as is” to refine existing processes rather than define really new models (Margherita and Petti, 2010).

The work presented here attempts to address the above points, or at the very least, contribute toward suggesting the general overarching framework of an approach that can address such limitations. The basis of the proposed generalisation approach is ontology and the representation of real-world semantics. While we accept that formal representations are fundamental for enabling the automation of any use case related to BPM, we also recognise that previous research has been predominantly preoccupied with logical and internal consistency of models rather than the real world semantics of such representations. By real-world semantics we intend the mapping between symbols in a model and things in the real-world (e.g., a business organisation). This is consistent with Lowe’s definition of ontology according to which an ontology is "the set of things whose existence is acknowledged by a particular theory or system of thought" (E.J. Lowe in the Oxford Companion to Philosophy).

In this research this philosophical definition is applied in conjunction with the current Semantic Web notion of ontology. When modelling the business processes we do so with the aim of accurately identifying the individual process elements, as they exist in an organisation. These real-world models are then represented in the Web Ontology Language (OWL) in order to allow for automatic inferencing of generalised process patterns. Interested readers can refer to Aldin et al. (2009) for a more complete explanation of the method that was developed for the modelling of process patterns. In the sections that follow we will only focus on generalisation/specialisation.

Therefore, this mix of philosophical and computational ontologies is an attempt at addressing the issues presented above. More specifically the use of philosophical (real-world) ontologies can help alleviate the problems related to (2) and (3) above and OWL ontologies can help resolve the automation problem highlighted in (1). As for (4) the method described in Aldin et al. (2009) suggests a way for designing new business processes from existing process patterns.

3 BUSINESS PROCESS CONCEPTS AND GENERALISATION

Many definitions of business process have been proposed. Table 1 summarises five definitions found in the literature and extracts the main concepts emphasised by the respective authors (Aldin and de Cesare, 2011).

Definitions	**Concepts Identified**
A **business process** is the set of internal activities performed to serve a customer (Jacobson 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 organisation, 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 a goal)

Table 1. Definitions of business process drawn from the literature.

The concepts identified in Table 1 represent those elements that the business process community commonly and generally accepts as being fundamental in characterising business processes (Aldin et al., 2009). These elements include:

- ***Process***: A set of activities, events, etc. that together and cohesively delivers a service and/or a product.
- ***Activity:*** Specific behaviour carried out in an organisation.
- ***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.
- ***Participant:*** The types of actors or agents that take part in processes.
- ***Goal:*** The aim of a process. Goals are a special type of dissolution event normally corresponding to a predefined type of outcome for a business process. For example, the approval of a mortgage application.
- ***Event:*** An occurrence that takes place at a specific point in time and that is capable of inducing some observable behaviour (activity or process). Two types of events can be defined. Initiating events which trigger the start of a process or an activity and dissolution events which terminate a business process or an activity.
- ***Resource:*** Tangible (e.g., raw materials) or intangible (e.g., specific documents or information) things that are processed, manipulated, transformed, etc. by processes/activities. Resources that are required at the start of a process or activity are considered inputs, while resources that are produced by a process or an activity are considered outputs. Hence the relationships hasInput and hasOutput.

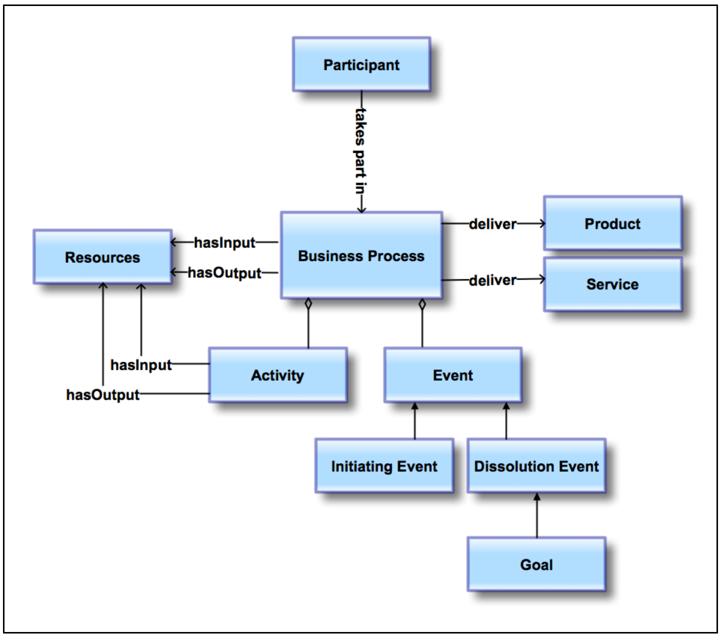

Figure 1. Business Process Ontology.

The BPO was initially represented informally in UML (Figure 1) and then converted to OWL in Protégé 4.0. Business process models of two domains (financial and educational) were derived from legacy system design documents and user manuals for the former and from staff/student handbooks for the latter. The method of extraction and interpretation followed to derive such models is documented in Aldin et al. (2009). The models were then represented in OWL based on the ontology in Figure 1 and subsequently generalised as described in Section 4. Process generalisation was carried out by identifying commonalities based on a mix of the core process elements of the BPO.

4 APPLICATION OF THE GENERALISATION APPROACH

The business process models that were used in this study derived from the financial and higher education domains. For the financial domain, process models were extracted from user and design documentation of three legacy systems each representing a specific subdomain; these were retail banking, insurance and mortgages. For the domain of higher education staff and student handbooks represented the main source of information. Overall about 50 process models were extracted. In broad terms the derivation of the process models followed three main phases: (1) data collection and organisation of the documentation, (2) interpretation of the documentation in order to derive business process diagrams and (3) discovery of patterns. It was in the final phase in which generalisation/specialisation played a key role and it is this aspect of the overall approach that this section will focus on.

As mentioned previously, the literature proposes various definitions of process generalisation. Most definitions focus on the possible traces through a process diagram. Possible traces through a process diagram reflect the possible execution sets of a business process class. A business process diagram is the representation of a process class (P) while a specific process execution (p_1) occurring in the business is an instance of P. Due to limitation of space, here we will refer only to two possible definitions of 'process specialisation' as referred to by Wyner and Lee (2002). The authors refer to *maximal execution set semantics* and *minimal execution set semantics*. These are defined as follows:

1. "one might define a process class as including all systems whose execution sets are subsets of some *maximal execution set* i.e. whose execution sets can include at most the behaviors specified by the maximal execution set" (p.138)

2. "a [process] class might be defined as including all systems whose execution sets are supersets of some *minimal execution set*, i.e., whose execution sets must include at least all the behaviors specified by the minimal execution set" (p.138)

For example, according to the first definition the process class P (in Figure 2) would be considered a generalisation of Q because any execution of Q would also be an instance of P. Whereas according to the second definition, Q would be the generalised process class because P at the very least includes the behaviour specified in Q. We concur with Wyner and Lee's (2002) preference in adopting *maximal execution set semantics*. Such a decision appears theoretically sounder and provides a more intuitive approach to the problem (i.e., a specialised process class as a behavioural subset of the generalised class).

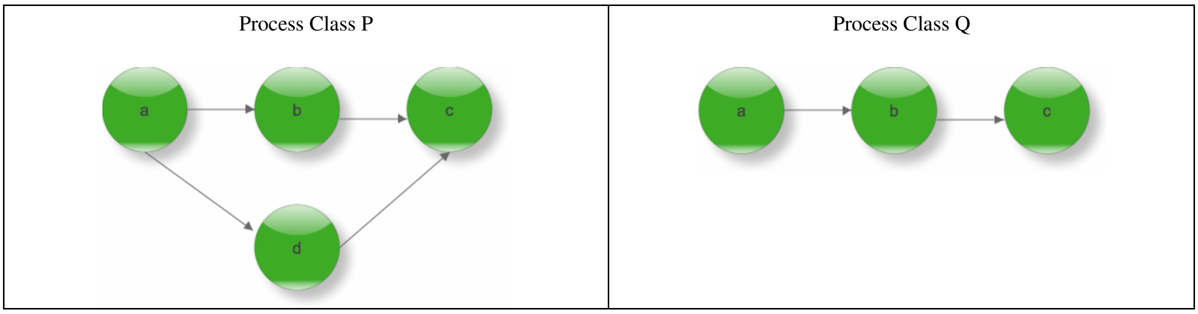

Figure 2. Simple example of two process classes.

However, while process specialisation based on execution set semantics can work well when purely considering possible traces through a process model, as the BPO of section 3 shows, activities (and events) are not the only elements that define a business process; there are others such as participants, input and output resources, services, products and goals. For example, one may consider two processes producing similar types of outputs, but having limited overlap in the internal activities, as being specialisations of a more generic type.

The remainder of this section exemplifies a pragmatic approach to process generalisation. The BPO was used to drive the generalisation process. The BPO as well as the processes extracted during data collection were represented in OWL and the ontologies modelled in Protégé 4.0 (Corcho et al. 2003). The identification of possible generalised process classes was carried out manually and their formal definitions were axiomatised in OWL. The OWL reasoner FaCT++ was executed within Protégé 4.0 to produce the specialisation hierarchies.

The following subsections provide examples of six types of process generalisation tested in this study. The generalised process classes were defined on the basis of property restrictions related to the different elements of the BPO.

(1) Generalisation based on similar business process elements

DefineProductTypes (Table 2 and Figure 3) represents a generalised process class that defines a process for creating new products in an organisation. The general type was derived from three processes as they were defined for the insurance, mortgage and retail banking subdomains (Table 3). The generalisation of the three original processes took into account as many elements as possible of the BPO. After running FaCT++ the original three process models were inferred as subclasses of *DefineProductTypes* (Figure 4).

BPO	DefineProductTypes
Participant	Staff
Input Resource	Forms
	Guides
Output Resource	Reports
Initiating Event	New Product Introductory
Goal	New Product Launched To Markets
Activities	Fill in Forms
	Select Rules
	Choose Life Status
	Produce Document Types

Table 2. Define Product Types.

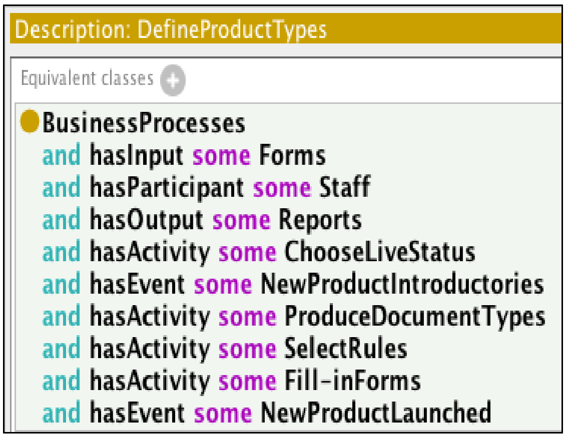

Figure 3. DefineProductTypes in Protégé.

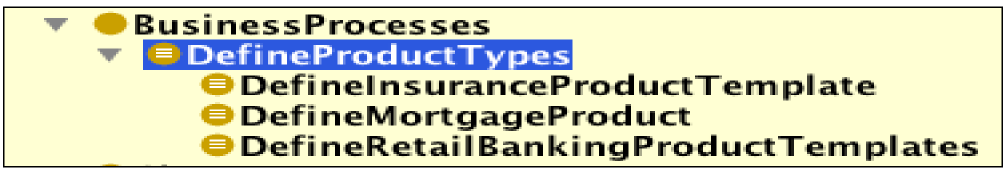

Figure 4. DefineProductTypes and its subclasses.

Table 3 shows how the specialisation of *DefineProductTypes* occurred. The three original models were represented side by side in a tabular manner allowing for commonalities to be visually spotted. Table 3 is the post-generalisation version and Table 4 summarises parts of the original processes that were eliminated from the generalised class.

This type of generalisation can be considered as the most complete since it uses the whole (or most of) BPO to derive a general process class. While this approach can be applied is those cases in which there is much commonality across the original process models, sometimes it may be useful to simply focus on commonalities that lie only in one or a few elements of the BPO. Generalisation types 2-6 (that follow) provide examples of this kind.

BPO Elements	***Define Product Types***	**Define Insurance Product Template**	**Define Mortgage Product**	**Define Retail Banking Product Templates**
Participant	*Staff*	Insurance Admin	Mortgages Admin	Retail Banking Admin
Input	*Forms*	Insurance Product Template Forms	Research Starting Form	Retail Product Template Forms
	Guides	Insurance Rule Guides	Mortgage Interest Rate Guides	Retail Product Fixed Rule Guides
Output	*Reports*	Insurance Product Reports	Mortgage Product Reports	Retail Product Reports
Initiating Event	*New Product Introductory*	New Insurance Product Introductory	New Mortgage Product Introductory	New Retail Banking Product Introductory
Goal	*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 3. Define Product Types with similar processes.

BPO Elements	**Non Generalised Elements of *Define Insurance Product Template***	**Non Generalised Elements of *Define Mortgage Product***	**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 4. Non-generalised business process elements.

(2) Generalisation based on similar Initiating Events

This type of generalisation depends on the Initiating Event of a business process. The reason for developing this type of generalisation is because it enables organisations to keep a record of all the important initiating events; in fact many organisations keep a record of their process events in an event log. Thus, subsuming all the processes that have similar starting events helps to keep a record of when a process should take place. It can be easily stated 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. Figures 5 and 6 demonstrate the application of this generalisation rule.

ProcessesTakePlaceAtTheBeginningOfTheAcademicYear is a generalised business process defined via a property restriction on the initiating event of a new academic year starting. Such an event triggers the processes shown in Figure 4 resulting from the inferencing carried out by the reasoner FaCT++.

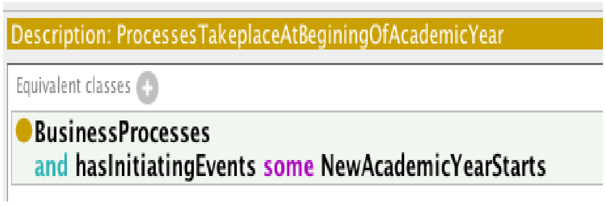

Figure 5. Class of processes triggered at the start of an academic year.

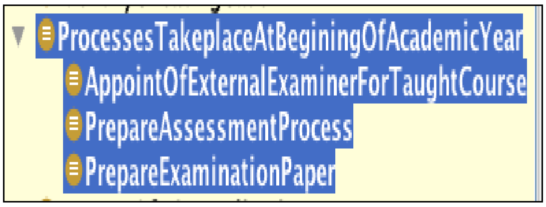

Figure 6. Subclasses of processes triggered at the start of an academic year.

(3) Generalisation based on similar Goals

This type of generalisation depends on the goal event of a business process. Thus, it generalises all the business processes that have a similar type of goal event to terminate the process. The reason to define this type of generalisation is that a business process exists for a reason, i.e. it strives to achieve a goal. Thus, any business process without a clear goal may require redesign. 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.

Figures 5 and 6 provide an example. *BPTerminatedWithReachingDecision* is a general process defined by one restriction that all processes should have a similar type of goal event, i.e. *ApplicationDecisionApproved*.

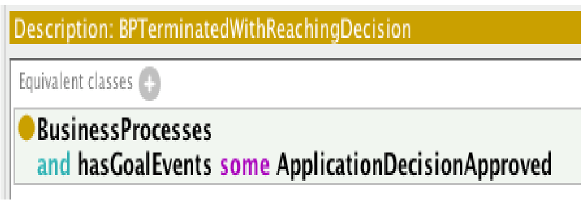

Figure 7. Class of processes that terminate with a decision.

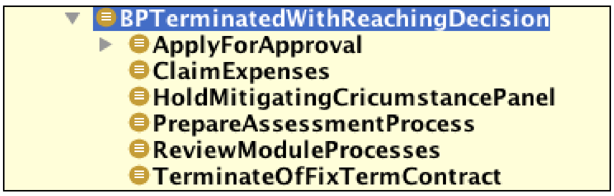

Figure 8. Subclasses of the general process.

(4) Generalisation based on similar Inputs

This type of generalisation depends on the inputs that a business process requires as resources in order to achieve its ultimate goal(s). 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 of 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, this type of model is important to understand how 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. Figures 9 and 10 provide an example of a generalised process class (i.e., *BPUsesForms*) obtained from this type of generalisation rule.

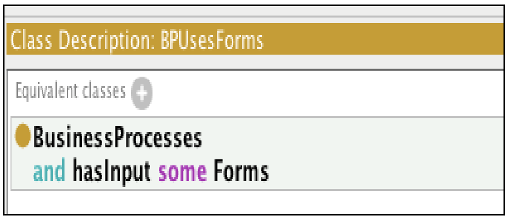

Figure 9. BPUsesForms defined in Protégé.

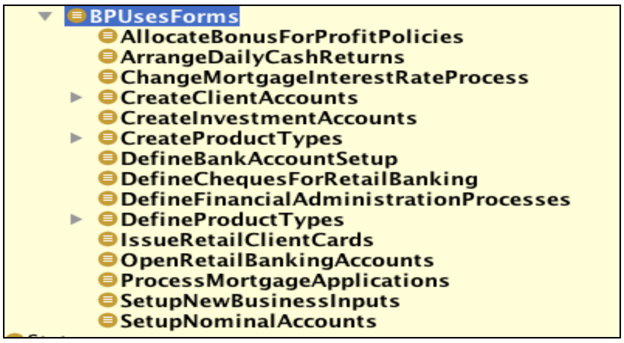

Figure 10. BPUsesForms and it subprocesses.

(5) Generalisation based on similar 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. This type of generalisation is quite important especially if combined with the type of goal processes produce. Unlike generalisation on inputs it is very likely that the similarity of different processes is higher if they produce similar outputs (for example, two processes that ultimately produce a weather bulletin). Below is a generalised process class achieved via this generalisation rule. *BPProducesReports* is defined by a property restriction on the type of resource that the process produces (e.g., reports as in Figures 11 and 12).

Figure 11. BPProducesReports as defined in Protégé.

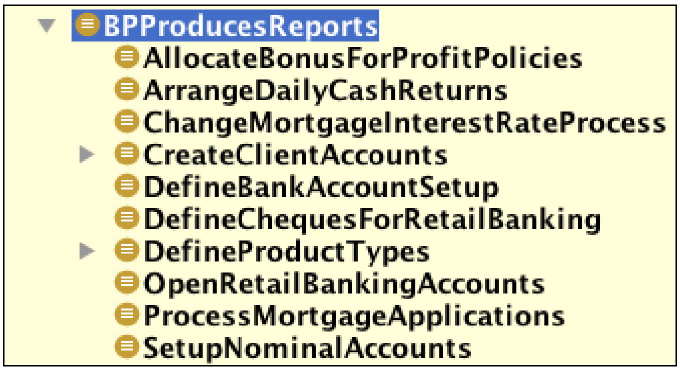

Figure 12. BPProducesReports and its subprocesses.

(6) Generalisation based on similar Participants

This type of generalisation depends on the participant(s) that take part in a business process. Thus, it generalises all the business processes that have similar participant types. For example, in most organisations the general types of participant could be client and staff. These general types could be specialised even further giving rise to more meaningful process hierarchies. This type of generalisation may be useful in the case of wanting to know which processes involve certain roles. The aim of this generalisation type is to enable a better connection 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) defined the actor-role pattern to be used in all problem situations in which there is a need to separate actors from roles. Figures 13 and 14 show an example of a generalised process class defined with this type of generalisation rule. The common participant in the example is Staff.

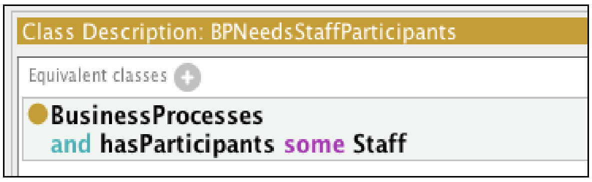

Figure 13. BPNeedsStaffParticipants as defined in Protégé.

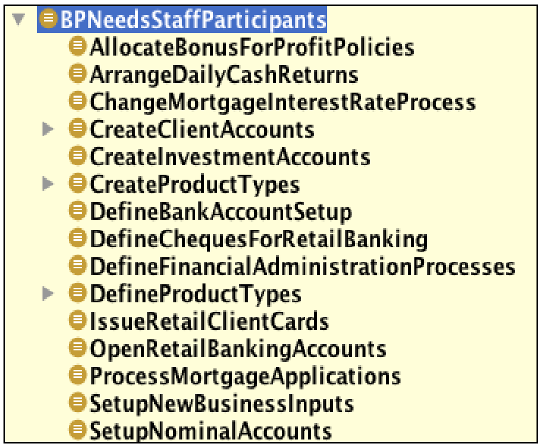

Figure 14. BPNeedsStaffParticipants and its subprocesses.

5 DISCUSSION AND CONCLUSION

This paper has presented a general approach to generalising business processes via the use of ontologies. The approach utilises at its core Business Process Ontology (BPO) derived mainly from various business process definitions found in the literature. The BPO was kept intentionally simple for the purposes of this paper in order to allow for a clear exposition of how its elements can be used for deriving generalised classes of organisational processes thus creating different types of hierarchies dependent on which elements of the BPO are used.

The paper presented six types of process generalisation. While the generalisation approach adopts the informal ontology of Figure 1, the ontology was formalised in OWL and the class hierarchies were automatically inferred by the reasoned FaCT++ in Protégé 4.0. The approach presented in this paper deviates from the more formal approaches proposed by previous research including, for example, Wyner and Lee (2002). The less formal nature of our approach can be viewed as both a limitation and a benefit. It does represent a limitation because formal representations have the benefit of being mathematically sound and therefore capable of automation. However one must consider that most representations of business processes are currently in informal or semiformal models and/or documents. Moreover previous literature tends to not take into account the real-world semantics embedded in such representations; real-world semantics that can currently only be interpreted adequately by people and not machines. This situation therefore demands that any approach used to reengineer business knowledge from organisational assets must also provide pragmatic support especially when reusability of process models is the aim.

The different generalisation types presented in this paper are obviously not devoid of problems. While the research presented here demonstrates the use of a pragmatic approach to the problem of process generalisation, it has not however answered the question of which generalisation types are most useful within an organisational context. For example, as stated above, generalisation on goals and/or output types may provide more meaningful process hierarchies (meaningful from a business perspective) than hierarchies developed purely based on inputs. In fact many processes may have some similar types of inputs (e.g., raw materials), but composed of completely different chains of activities and producing completely different outputs, making such a process class hierarchy of very limited utility for the purpose of reuse. Hence further research into which types of generalisation are more meaningful for business stakeholders will be required.

In conjunction with the previous limitation it must be noted that any type of semantic analysis should have at its basis an ontology that closely maps to the real world domain it is representing. The BPO proposed in this paper has demonstrated its potential, however it would be preferable to evolve such a BPO from an analysis and interpretation of business processes in an empirical setting, or at the very least, test the current BPO against further business process data in order to determine its level of robustness against a greater number of business process instances.

6 REFERENCES

Rumbaugh, J., Jacobson, I. & Booch, G., 2004, *The Unified Modeling Language Reference Manual, Second Edition*, Addison-Wesley.

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

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

Pentland, B.T., 2003a, Sequential variety in work processes, Organization Science, Vol. 14, no. 5, pp. 528–540.

Dumas, M., and García-Bañuelos, L., Dijkman, R.M., 2009, Similarity search of business process models. Bulletin of the Technical Committee on Data Engineering, vol. 32, no. 3, pp. 23-28.

Aldin, L., de Cesare, S. & Lycett, M., 2009b,"A Semantic-Based Framework for Discovering Business Process Patterns", In Proceedings of the 6th International Workshop on Ontology Driven Software Engineering (ODiSE) in conjunction with the Object-Oriented Programming, Systems, Languages and Application Conference (OOPSLA), Orlando, Florida, USA.

Houy, C., Fettke, P. und Loos, P., 2010a, Empirical Research in Business Process Management - Analysis of an emerging field of research; Business Process Management Journal, 16. Jg., H. 4; S. 619-661.

Peristeras, V., Tarabanis, K., 2000, Towards an enterprise architecture for public administration using a top-down approach, European Journal of Information Systems, Vol.9, pp. 252–260.

Pentland, B.T., 2003b, Conceptualizing and measuring variety in the execution of organizational work processes, Management Science Vol. 49, no.7, pp.857–870.

Handayani, P. et al., 2010, Implementing A Modern Process Management System Based On The MIT Process Handbook, Journal System Information MTI UI, Volume 5, No 2, ISBN 1412 – 8896.

Lautenbacher, F.; Bauer, B.; and Seitz, C., 2008, Semantic Business Process Modelling - Benefits and Capability. In AAAI Spring Symposium, Stanford University, California: AAAI.

Stoddard, D.B., Jarvenpaa, S.L., 1995, Business process redesign: Tactics for managing radical change. Journal of Management Information Systems 12 (1), pp. 81–107.

Margherita, Alessandro, and Claudio Petti., 2010. ICT-enabled and process-based changes: an integrative roadmap. Business Process Management Journal, Vol. 16, no.3, pp. 473-491.

Aldin, L. and de Cesare, S. 2011, "A Literature Review on Business Process Modelling: New Frontiers of Reusability", Enterprise Information System Journal, Vol. 6, no. 2.

Aldin, L. & and de Cesare, S., 2009a, "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.

Wyner G.M. and Lee, J., 2002, Process Specialization: Defining Specialization for State Diagrams, Computational & Mathematical Organization Theory, 8, pp. 133-155.

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.

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

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