Business process management (BPM) standards: a survey

Ryan K.L. Ko

Advanced Design and Modelling Laboratory,
School of Mechanical and Aerospace Engineering,
Nanyang Technological University, Singapore

Stephen S.G. Lee

School of Mechanical and Aerospace Engineering,
Nanyang Technological University, Singapore, and

Eng Wah Lee

Planning and Operations Management Research Group,
Agency for Science and Technology,
Singapore Institute of Manufacturing Technology, Singapore

Abstract

Purpose – In the last two decades, a proliferation of business process management (BPM) modeling languages, standards and software systems has given rise to much confusion and obstacles to adoption. Since new BPM languages and notation terminologies were not well defined, duplicate features are common. This paper seeks to make sense of the myriad BPM standards, organising them in a classification framework, and to identify key industry trends.

Design/methodology/approach – An extensive literature review is conducted and relevant BPM notations, languages and standards are referenced against the proposed BPM Standards Classification Framework, which lists each standard's distinct features, strengths and weaknesses.

Findings – The paper is unaware of any classification of BPM languages. An attempt is made to classify BPM languages, standards and notations into four main groups: execution, interchange, graphical, and diagnosis standards. At the present time, there is a lack of established diagnosis standards. It is hoped that such a classification facilitates the meaningful adoption of BPM languages, standards and notations.

Practical implications – The paper differentiates BPM standards, thereby resolving common misconceptions; establishes the need for diagnosis standards; identifies the strengths and limitations of current standards; and highlights current knowledge gaps and future trends. Researchers and practitioners may wish to position their work around this review.

Originality/value – Currently, to the best of one's knowledge, such an overview and such an analysis of BPM standards have not so far been undertaken.

Keywords Process management, Standards, Work flow

Paper type Literature review

1. Introduction

1.1 The growth of business process management
With intensified globalisation, the effective management of an organisation's business processes became ever more important. Many factors such as:

- the rise in frequency of goods ordered;
- the need for fast information transfer;
- quick decision making;
- the need to adapt to change in demand;
- more international competitors; and
- demands for shorter cycle times (Simchi-Levi *et al.*, 2000)

are challenging the profitability and survival of big and small companies.

In a bid to deal with these challenges, information technology (IT) was harnessed to manage business processes (Davenport, 1993; Georgakopoulos *et al.*, 1995). Over the past two decades, previously manual hand-filled forms were increasingly replaced by their "paperless" electronic counterparts. This eventually evolved into what is known as business process management (BPM) today.

According to prominent BPM researcher van der Aalst *et al.* (2003), BPM is defined as "supporting business processes using methods, techniques and software to design, enact, control and analyze operational processes involving humans, organizations, applications, documents and other sources of information" (for more discussions on BPM definitions, see Section 2.2). Software tools supporting the management of such operational processes became known as business process management systems (BPMS).

At the end of 2006, the BPMS market reached nearly US$1.7 billion in total software revenue (Hill *et al.*, 2007a, b) and began to exhibit the characteristics of an early mainstream software market, i.e. proven technology, stable vendors, vendor consolidation and rapid user adoption. The BPMS market is also the second fastest-growing middleware (a type of integrative software) market segment; Gartner research estimates that the BPMS market will have a compound annual growth rate of more than 24 per cent from 2006 to 2011 (Hill *et al.*, 2007a, b).

1.2 The proliferation of BPM languages, standards and software systems
Naturally, interest in BPM from practitioners and researchers grew rapidly. A wide variety of paradigms and methodologies from organization management theory, computer science, mathematics, linguistics, semiotics, and philosophy were adopted, making BPM a cross-disciplinary "theory in practice" subject.

However, this unprecedented growth is a double-edged sword. Many new BPM terminologies and technologies are often not well defined and understood by many practitioners and researchers using them (Havey, 2005; Hill *et al.*, 2008). New languages and notations proposed often contain duplicating features for similar concepts (Mendling and Neumann, 2005), and loosely claim to be based on theoretical formalisms such as Pi-calculus and Petri nets (Havey, 2005). Most of them have also not been validated (van der Aalst, 2004a, b, c), especially in a real business and office environment (Winograd and Flores, 1986).

1.3 Motivation of this paper
Because of the afore-mentioned reasons and the fact that an framework to evaluate BPM standards is non-existent at the time of writing (Recker, 2007), a taxonomy to rationalize, classify and evaluate BPM standards is timely. This paper's goal is to leave

the reader (from the novice researcher to the experienced practitioner) with some semblance of order out of a disparate collection of specifications, white papers, journal publications, conference publications and workshop notes to be consolidated as a single paper. More specifically, this survey paper attempts to:

- discuss and rationalize the terminologies associated with BPM and its standards;
- systematically categorize/classify BPM standards;
- discuss the current strengths and limitations of each standard;
- clarify, the differences of theoretical underpinnings of prominent BPM standards; and
- explore the gaps of knowledge of current BPM standards and how these may be bridged.

This paper is structured as follows: Section 2 briefly discusses main BPM concepts to clarify BPM terminologies and the BPM life cycle. Section 3 then introduces the categories of BPM standards and the role of BPM standards from the perspective of the BPM life cycle. Section 4 discusses the perceived gaps in current BPM standards. Section 5 highlights the upcoming BPM research trends and lastly, Section 6 explores highlights related and preceding foundation works in BPM.

Figure 1 was created to help the reader to meander through this paper.

2. BPM basics
Before exploring the BPM standards and related fields, it is always good to begin with an overview of BPM basics. While it may seem unbelievable a discipline with a history of about three decades has yet to clarify basic BPM terminologies like business process, BPM vs workflow management (WfM), workflow, business process reengineering (BPR). This chapter aims to address this gap. Let us begin with basic BPM concepts and terminologies.

2.1 The BPM life cycle
As previously discussed, BPM is mainly a cross-discipline "theory in practice" subject with many views, definitions and perspectives. Because of its multi-disciplinary nature, it is often easy to find business process research materials across many subjects' databases.

To effectively understand the terminologies and features of BPM, one should start from an appreciation of the BPM life cycle. There are many views of the generic BPM life cycle (Havey, 2005; Hill *et al.*, 2006; van der Aalst, 2004a, b, c; van der Aalst *et al.*, 2003) but we adopt van der Aaslt *et al.*'s (Figure 2) because of its succinctness and relevance. According to them, the BPM life cycle consists of van der Aalst *et al.* (2003):

- *Process design.* In this stage, fax- or paper-based as-is business processes are electronically modeled into BPMS. Graphical standards are dominant in this stage.
- *System configuration.* This stage configures the BPMS and the underlying system infrastructure (e.g. synchronization of roles and organization charts from the employee's accounts in the company's active directory (Lowe-Norris and
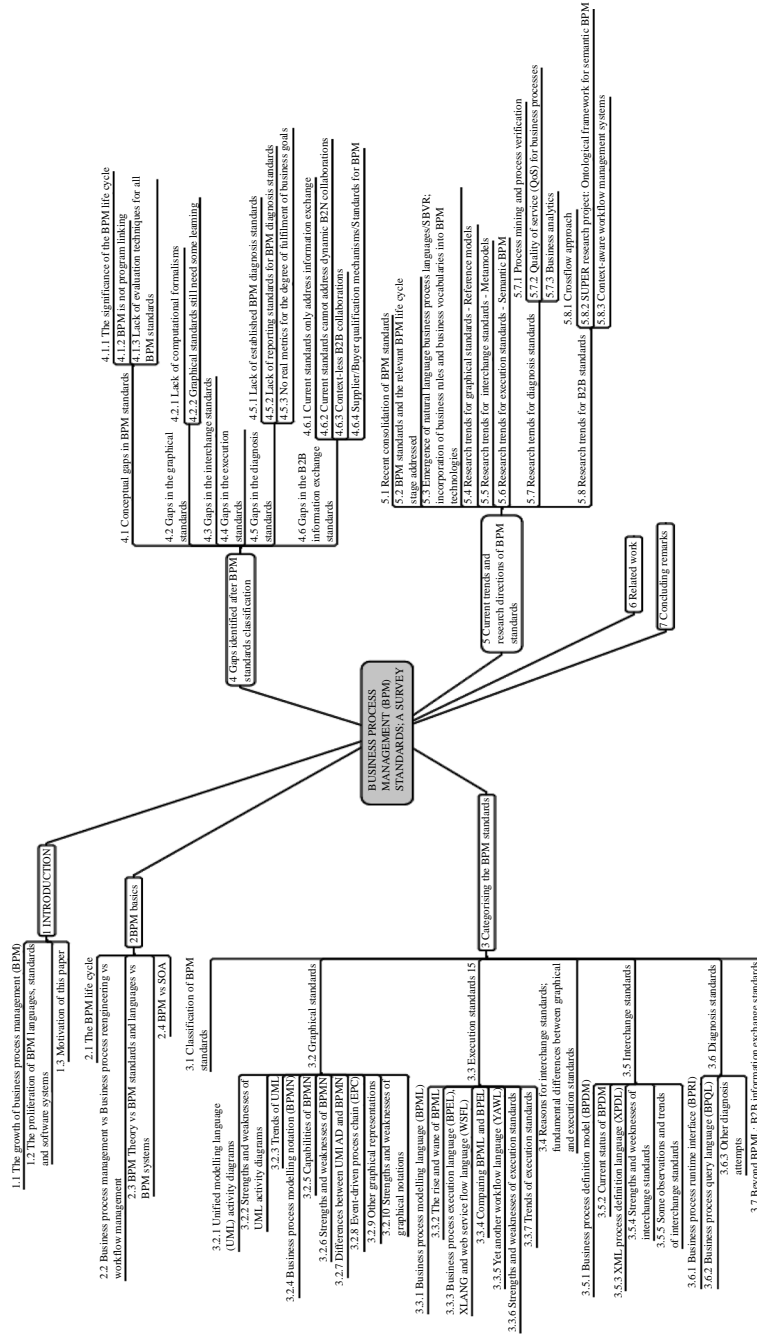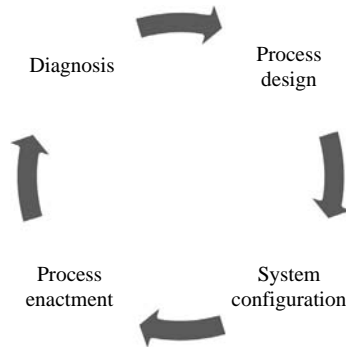
BUSINESS PROCESS MANAGEMENT (BPM) STANDARDS: A SURVEY

1 INTRODUCTION
- 1.1 The growth of business process management (BPM)
- 1.2 The proliferation of BPM languages, standards and software systems
- 1.3 Motivation of this paper

2 BPM basics
- 2.1 The BPM life cycle
- 2.2 Business process management vs Business process reengineering vs workflow management
- 2.3 BPM Theory vs BPM standards and languages vs BPM systems
- 2.4 BPM vs SOA

3 Categorising the BPM standards
- 3.1 Classification of BPM standards
- 3.2 Graphical standards
  - 3.2.1 Unified modelling language (UML) activity diagrams
  - 3.2.2 Strengths and weaknesses of UML activity diagrams
  - 3.2.3 Trends of UML
  - 3.2.4 Business process modelling notation (BPMN)
  - 3.2.5 Capabilities of BPMN
  - 3.2.6 Strengths and weaknesses of BPMN
  - 3.2.7 Differences between UML AD and BPMN
  - 3.2.8 Event-driven process chain (EPC)
  - 3.2.9 Other graphical representations
  - 3.2.10 Strengths and weaknesses of graphical notations
- 3.3 Execution standards 15
  - 3.3.1 Business process modelling language (BPML)
  - 3.3.2 The rise and wane of BPML
  - 3.3.3 Business process execution language (BPEL), XLANG and web service flow language (WSFL)
  - 3.3.4 Comparing BPML and BPEL
  - 3.3.5 Yet another workflow language (YAWL)
  - 3.3.6 Strengths and weaknesses of execution standards
  - 3.3.7 Trends of execution standards
- 3.4 Reasons for interchange standards; fundamental differences between graphical and execution standards
- 3.5 Interchange standards
  - 3.5.1 Business process definition model (BPDM)
  - 3.5.2 Current status of BPDM
  - 3.5.3 XML process definition language (XPDL)
  - 3.5.4 Strengths and weaknesses of interchange standards
  - 3.5.5 Some observations and trends of interchange standards
- 3.6 Diagnosis standards
  - 3.6.1 Business process runtime interface (BPRI)
  - 3.6.2 Business process query language (BPQL)
  - 3.6.3 Other diagnosis attempts
- 3.7 Beyond BPML: B2B information exchange standards

4 Gaps identified after BPM standards classification
- 4.1 Conceptual gaps in BPM standards
  - 4.1.1 The significance of the BPM life cycle
  - 4.1.2 BPM is not program linking
  - 4.1.3 Lack of evaluation techniques for all BPM standards
- 4.2 Gaps in the graphical standards
  - 4.2.1 Lack of computational formalisms
  - 4.2.2 Graphical standards still need some learning
- 4.3 Gaps in the interchange standards
- 4.4 Gaps in the execution standards
- 4.5 Gaps in the diagnosis standards
  - 4.5.1 Lack of established BPM diagnosis standards
  - 4.5.2 Lack of reporting standards for BPM diagnosis standards
  - 4.5.3 No real metrics for the degree of fulfilment of business goals
- 4.6 Gaps in the B2B information exchange standards
  - 4.6.1 Current standards only address information exchange
  - 4.6.2 Current standards cannot address dynamic B2N collaborations
  - 4.6.3 Context-less B2B collaborations
  - 4.6.4 Supplier/Buyer qualification mechanisms/Standards for BPM

5 Current trends and research directions of BPM standards
- 5.1 Recent consolidation of BPM standards
- 5.2 BPM standards and the relevant BPM life cycle stage addressed
- 5.3 Emergence of natural language/business process languages/SBVR; incorporation of business rules and business vocabularies into BPM technologies
- 5.4 Research trends for graphical standards – Reference models
- 5.5 Research trends for interchange standards – Metamodels
- 5.6 Research trends for execution standards – Semantic BPM
- 5.7 Research trends for diagnosis standards
  - 5.7.1 Process mining and process verification
  - 5.7.2 Quality of service (QoS) for business processes
  - 5.7.3 Business analytics
- 5.8 Research trends for B2B standards
  - 5.8.1 Crossflow approach
  - 5.8.2 SUPER research project: Ontological framework for semantic BPM
  - 5.8.3 Context-aware workflow management systems

6 Related work

7 Concluding remarks

Figure 1.
Organisation and
coverage of survey paper

Figure 2.
van der Aalst *et al.*'s
BPM life cycle

Denn, 2000). This stage is hard to standardize due to the differing IT architectures of different enterprises.

- *Process enactment.* Electronically modeled business processes are deployed in BPMS engines. Execution standards dominate this stage.

- *Diagnosis.* Given appropriate analysis and monitoring tools, the BPM analyst can identify and improve on bottlenecks and potential fraudulent loopholes in the business processes. The tools to do this are embodied in diagnosis standards.

With this appreciation of the BPM life cycle stages, we are now able to distinguish the features of BPM from its predecessors BPR and WfM.

2.2 BPM vs BPR vs WfM

The influence of IT in the managing of business processes can be traced back to Hammer and Champy's (1992, 1993) BPR paradigm and Davenport's (1993) book on how process innovation can facilitate BPR. However, BPM and BPR are not the same: whereas BPR calls for a radical obliteration of existing business processes, its descendant BPM is more practical, iterative and incremental in fine-tuning business processes.

Another two terminologies often used loosely are "Workflow management (WfM)" and "Business process management (BPM)". There are mainly two differing viewpoints. One viewpoint by Gartner research views BPM as a management discipline with WfM supporting it as a technology (Hill *et al.*, 2008). According to their report (Hill *et al.*, 2008):

Business process management (BPM) is a process-oriented management discipline. It is not a technology. Workflow is a flow management technology found in business process management suites (BPMSs) and other product categories.

Another viewpoint from academics is that the features stated in WfM according to Georgakopoulos *et al.* (1995) is a subset of BPM defined by van der Aalst *et al.* (2003) (Table I), with the diagnosis stage of the BPM life cycle as the main difference. However, in reality, to our best knowledge, many BPMS are still very much workflow management systems (WfMS) and have not yet matured in the support of the BPM diagnosis. In recent years, the authors have observed that many vendors have updated their products' names from "WfM" to the more contemporary "BPM". One example is

Metastorm's product name change from Metastorm E-Work Version 6 to Metastorm BPM Version 7 in 2005 (Metastorm, 2007). Noticeably, the change of name was not accompanied by a maturity of the Diagnosis portion of its suites (i.e. WfM to BPM) (Table I). Instead, the visible changes from Versions 6 to 7 are their system's adaptation of Microsoft SQL Server 2005, the obsolescence of simulation features and an aesthetically appealing graphical user interface (GUI).

In the first author's work experience and observations from technical forum contributors, many of these WfMS-turned-BPMS have yet to offer rich diagnosis features. Although many software suites offer business activity monitoring (BAM) dashboards, the creation of useful audit trails, and the churning of meaningful reports displaying process trends still requires external specialized reporting tools like Microsoft Reporting Server or Crystal Reports.

With new research interests in the BPM diagnosis sub-topics business process analysis and BAM, the diagnosis component of the BPM life cycle is starting to gain more attention from software vendors. This paves the way for the development of true BPM.

2.3 BPM theory vs BPM standards and languages vs BPMS

At the time of writing, there are more than ten formal groups working on BPM standards (zur Muehlen, 2007), seven of which are dedicated to modelling definitions (Ghalimi and McGoveran, 2005). Hence, it is no surprise that the BPM landscape became very fragmented from the late 1990s onwards. The confusion was so bad that even theory was confused for standards and standards for BPMSs, when the three are in a nested relationship as shown in Figure 3.

BPM life cycle stage	Workflow management (WfM)	Business process management (BPM)
Process design	Yes	Yes
System configuration	Yes	Yes
Process enactment	Yes	Yes
Diagnosis	Weak	Yes

Table I.
WfM and BPM compared
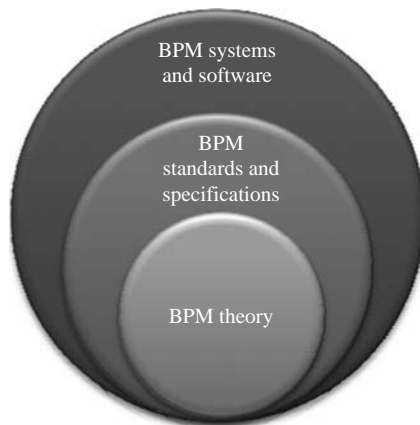
Figure 3.
The relationship between BPM theory, standards and systems

As shown in Figure 3, BPM standards and specifications (e.g. Business Process Execution Language – BPEL (Andrews *et al.*, 2003)) are based on established BPM theory (e.g. Pi-calculus (Milner, 1982, 1999) and Petri nets (Petri, 1962) and are eventually adopted into software and systems (e.g. *Intalio Designer* (Intalio, 1999-2007), KAISHA-Tec ActiveModeler (KAISHA-Tec Co., 2008), etc.). BPM standards and systems are also what Gartner (Hill *et al.*, 2007a, b, 2006) describes as "BPM-enabling technologies".

2.4 BPM vs service oriented architecture

In the industry, there is a growing awareness of the emerging service-oriented architecture (SOA). For example, SAP AG has migrated from the traditional ABAP-based R/3 system's SAPGUI front end to the Java-based SAP NetWeaver Portal which is supported by SAP Web Dynpro Technology, in the design, configuration and the linkage of web services.

BPM is a process-oriented management discipline aided by IT while SOA is an IT architectural paradigm. According to Gartner (Hill *et al.*, 2006), BPM "organizes *people* for greater agility" while SOA "organizes *technology* for greater agility" (Figure 4). In other words, processes in SOA (e.g. linked web services) enable the coordination of distributed systems supporting business processes and should never be confused with business processes.

With so many standards in the market, it can be a daunting task for a practitioner or even seasoned researchers to make sense of the myriad of standards. This can greatly stifle the adoption rate of BPM standards in BPMS, and may cause a plateau in knowledge development.

In the authors' opinion, the immediate need for BPM is a classification of BPM standards into a meaningful taxonomy. This is likened to the classification of *flora* and *fauna* species, and even the classification of gene sequences.

Hence, by grouping standards of similar attributes, we are able to appreciate the strengths and current limitations of each group, note the gaps, or even start the systematic evaluation criteria of standards from each group.

3. Categorising the BPM standards

In view of the above-mentioned disparate understanding and confusion of BPM standards, the authors attempt to categorize current standards by both their features
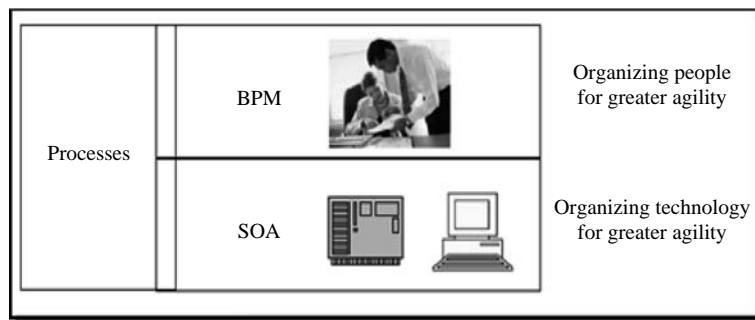
Source: Gartner, February (2006)

and their BPM life cycle perspective. This categorisation will enable us to filter out web service standards, business-to-business (B2B) standards from BPM standards, and also allow us to further classify BPM standards into graphical, execution, interchange and diagnosis standards. Such categorisation will preempt duplication and pin-point the impact of current and new BPM standards, but also pre-empt duplication and identify improvements.

3.1 Classification of BPM standards

The most logical way to make sense of the myriad of BPM standards is to categorise them into groups with similar functions and characteristics. From a high level point of view, a sharp observer should be able to infer that many of these standards actually address at least one of four phases of the BPM life cycle: process design, system configuration, process enactment and diagnosis. For example, business process modelling notation (BPMN) actually address the process design while BPEL mainly enables process enactment. There are also some languages such as XML Process Definition Language (XPDL) and Yet Another Workflow Language (YAWL) that spans across both. It is not easy to place them into strictly the process design or the process enactment phase.

For this reason, the authors propose a cleaner separation of features found in standards addressing the process design and process enactment phase into three clear-cut types of standards:

(1) *Graphical standards.* This allows users to express business processes and their possible flows and transitions in a diagrammatic way.

(2) *Execution standards.* It computerizes the deployment and automation of business processes.

(3) *Interchange standards.* It facilitates portability of data, e.g. the portability of business process designs in different graphical standards across BPMS; different execution standards across disparate BPMS, and the context-less translation of graphical standards to execution standards and vice versa.

BPM suites often have these three categories of process design and enactment standards, but often overlook one type of standard that makes BPM different from BPR and WfM: standards facilitating the diagnosis phase:

(4) *Diagnosis standards.* It provides administrative and monitoring (such as runtime and post-modeling) capabilities. These standards can identify bottlenecks, audit and query real-time the business processes in a company.

Figure 5 shows a flow chart demonstrating the authors' classification process of current standards into BPM standards (graphical, execution, interchange and diagnosis), B2B standards and web service/SOA standards. Web service standards and B2B standards are initially filtered out. After which, the BPM standards are further categorised into four relevant groups by their attributes and impact stage in the BPM life cycle. The graphical, execution and interchange standards address the process design and process enactment stage of the BPM life cycle while diagnosis standards address the diagnosis stage of the BPM life cycle.

As mentioned, the heterogeneity of business process languages is a notorious problem for BPM (Mendling and Neumann, 2005). Table II shows prominent standards,
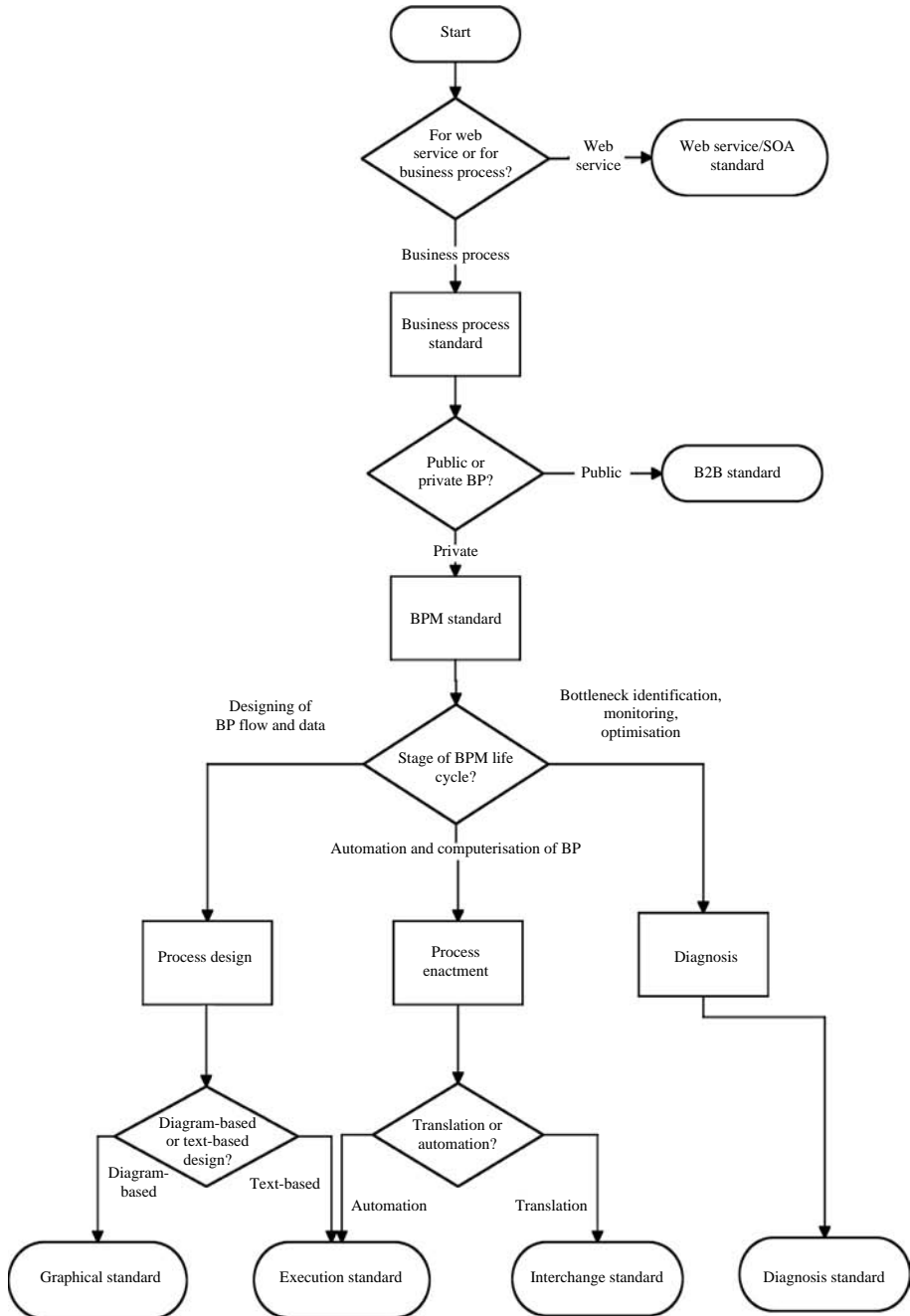
Figure 5.
Flow chart demonstrating the classification process of current standards into business process management standards (graphical, execution, interchange and diagnosis), B2B standards and web service/SOA standards

	BPM/SOA/B2B	Background	Theory/graphical/interchange/execution/diagnosis/B2B info exchange	Standardised?	Current status
BPDM	BPM	Industry	Interchange	Yes	Unfinished
BPEL	BPM	Industry	Execution	Yes	Popular
BPML	BPM	Industry	Execution	Yes	Obsolete
BPQL	BPM	Industry	Diagnosis	Yes	Unfinished
BPRI	BPM	Industry	Diagnosis	Yes	Unfinished
ebXML BPSS	B2B	Industry	B2B info exchange	Yes	Popular
EDI	B2B	Industry	B2B info exchange	Yes	Stable
EPC	BPM	Academic	Graphical	No	Legacy
Petri Net	All	Academic	Theory/graphical	NA	Popular
Pi-Calculus	All	Academic	Theory/execution	NA	Popular
Rosetta-Net	B2B	Industry	B2B info exchange	Yes	Popular
UBL	B2B	Industry	B2B info exchange	Yes	Stable
UML AD	BPM	Industry	Graphical	Yes	Popular
WSCI	SOA	Industry	Execution	Yes	Obsolete
WSCL	SOA	Industry	Execution	Yes	Obsolete
WS-CDL	SOA	Industry	Execution	Yes	Popular
WSFL	BPM	Industry	Execution	No	Obsolete
XLANG	BPM	Industry	Execution	No	Obsolete
XPDL	BPM	Industry	Execution/interchange	Yes	Stable
YAWL	BPM	Academic	Graphical/execution	No	Stable

Table II.
Prominent BPM
standards, languages,
notations, and theory and
their status

languages and theory that are often quoted as "standards supporting business processes". It also outlines for each row (standard/language/theory), its applicability (BPM, B2B or SOA), background, current status and if it is standardized.

When we process each row in Table II according to the flow chart in Figure 5, we can identify the groups these standards belong to. For example, BPEL will eventually be grouped into the execution standards group as it is a BPM standard, with a focus on text-based process design and enactment (recall BPM life cycle) via automation of business processes. For this reason, web service orchestration and choreography standards like WS-CDL, and the now-obsolete WSCI and WSCL, are not considered within the context of BPM standards.

From another perspective, graphical standards are currently the highest level of expression of business processes (i.e. most natural to human beings) while the lowest level (i.e. the most technical) are the execution standards (Figure 6). Even though the interchange standards aim to bridge the graphical standard to the execution standard or vice versa, the translation can sometimes be imperfect, as both standards are conceptually different (Recker and Mendling, 2006). As system configuration is a company-based (internal) process, having a standards category in the framework for this phase of the BPM life cycle will not make sense.

In this paper, our discussions will focus predominantly on BPM standards, with short sections discussing B2B standards. In the authors' opinion, brief discussions on B2B standards is important because the challenges from globalisation will eventually
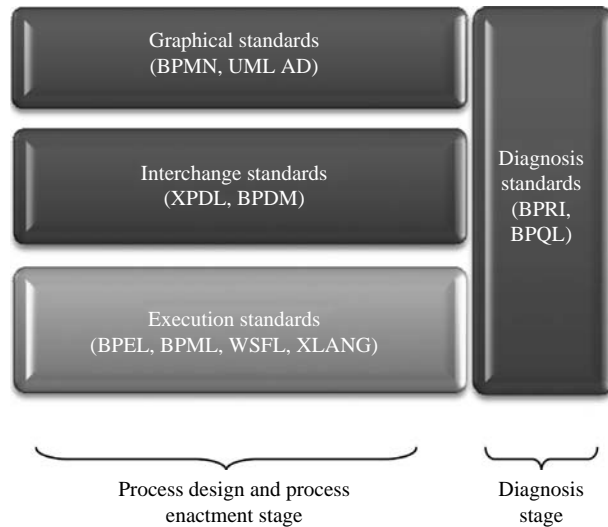
Figure 6.
The categories of current
BPM standards in relation
to the BPM life cycle

forge the integration of intra-company BP management (via BPM standards) and inter-company BP management (via B2B standards).

The ensuing sections expand on the afore-mentioned standards, starting from graphical standards.

3.2 Graphical standards

Graphical standards allow users to express the information flow, decision points and the roles of business processes in a diagrammatic way. Amongst the four categories of standards as mentioned in Section 3.1, graphical standards are currently the most human-readable and easiest to comprehend without prior technical training. Unified Modelling Language activity diagrams – UML AD (Object Management Group – OMG, 2004b), BPMN (OMG, 2004a), event-driven process chains – EPC (Scheer, 1992), role-activity diagrams (RADs) and flow charts are common techniques used to model business processes graphically.

These techniques range from common notations (e.g. flow charts) to standards (e.g. BPMN). And of the standards, UML AD and BPMN are currently the two most expressive, easiest for integration with the interchange and execution level, and possibly the most influential in the near future. For this reason, we will focus more on UML AD and BPMN, followed by a brief description of the other graphical business process modelling techniques.

3.2.1 Unified Modelling Language AD. OMG's (2004b) UML (Version 2.0), standardised in 2004, is the backbone of the object-oriented software engineering computing paradigm that superseded the structural programming paradigm. Broadly speaking, UML is a suite of 13 object-oriented notations that captures all attributes and behaviour of the objects modelled (Ambler, 2004). A few examples of these notations include the use case diagram (for documenting high-level user requirements), the sequence diagram (for documenting program sequence), and the AD, etc. Of the two,

the AD is most commonly used to model business processes in a graphical way (Russell *et al.*, 2006).

The UML AD is both a flowcharting technique and a special kind of state machine whose activities are states and interactivity links trigger-less transitions (Havey, 2005). In our opinion, UML AD are more like the "object-oriented equivalent" of flow charts and data flow diagrams from the structural programming paradigm. This view is also shared by critics of the UML (Version 2.0) like Bell (2004). If one subscribes to the view of the UML AD being like an extended flow-chart, decision points are similarly notated by diamonds and behaviour by action, which are the rounded rectangles in the diagram. The circular notations denote the start and the end points of the process (Wohed *et al.*, 2006).

The UML AD is rooted in Petri net token semantics (OMG, 2004b) and is logically based on the UML state diagrams, which in turn is based upon Harel's (1987) extension of state machines concept (Havey, 2005).

3.2.2 Strengths and weaknesses of UML AD. To the authors' best knowledge, there is currently no standard framework for evaluating BP modelling notations. The most common approach is a comparison of the BP modelling notation's features against the Workflow Patterns Framework (van der Aalst *et al.*, 2000) (a collection of generic, recurring constructs originally devised to evaluate workflow systems, but later adopted to evaluate workflow standards, business process languages). In 2001, Dumas *et al.* investigated the expressiveness and the adequacy of ADs for workflow specification, by systematically evaluating their ability to capture a collection of workflow patterns (Dumas and ter Hofstede, 2001). Dumas' work was a response to White's (2004b) publication *Workflow Handbook* evaluating the BP modelling notations (BPMN and UML AD) against workflow patterns.

From their evaluation, Dumas and ter Hofstede (2001) concluded that in the context of workflow specification, the strong points of UML AD with respect to alternative languages provided by commercial WFMS are essentially the following (Dumas and ter Hofstede, 2001; Russell *et al.*, 2006):

- They support signal sending and receiving at the conceptual level.
- They support both waiting and processing states.
- They provide a seamless mechanism for decomposing an activity into sub-activities. The combination of this decomposition capability with signal sending yields a powerful approach to handling activity interruptions.

On the other hand, the weaknesses of UML AD are (Wohed, 2004; Russell *et al.*, 2006):

- Some of the UML AD constructs lack a precise syntax and semantics. For instance, the "well-formedness" rules linking forks with joins are not fully defined, nor are the concepts of dynamic invocation and deferred events, among others.
- They do not fully capture important kinds of synchronisation such as the discriminator and the *N*-out-of-*M* join. Similarly, they do not fully support the producer-consumer pattern with termination activity.

The suitability of UML as a BP modelling technique was assessed by Russell *et al.* (2006). He concluded that:

- UML AD offers comprehensive support for the control-flow and data perspectives allowing the majority of the constructs encountered when analysing these perspectives to be directly captured.
- However, UML ADs are extremely limited in modelling resource-related or organisational aspects of business processes. It is interesting to note that UML ADs cannot capture many of the natural constructs encountered in business processes such as cases and the notion of interaction with the operational environment in which the process functions.

These limitations observed by Dumas and ter Hofstede (2001) and Russell *et al.* (2006) are common to many other business process modelling formalisms and reflect the overwhelming emphasis that has been placed on the control-flow and data perspectives in contemporary modelling notations (Russell *et al.*, 2006).

3.2.3 Trends of UML. While the first version of UML was hailed as the *de facto* software modelling standard for object-oriented programs, its second version had many critics and detractors.

In the author's opinion, not only did the 13 diagrammatic notations of the UML Version 2.0 not add value, but they also increased complexity. This was not helped by the aesthetically inconsistent terms across the 13 diagrams. For example, a two-dimensional box may mean different things across different diagrams of the UML Version 2.0. This view is shared by Bell (2004) as the "UML fever".

While UML AD are functional, business analysts somehow cannot use them without prior technical knowledge. In the author's working experience, business analysts are more at home with flowcharts even though these are limited to the modelling of single processes, not levels of decompositions of BPs. It is also quite challenging for one to design business processes at different levels of detail. In other words, while the UML AD is very good for designing single level business processes, sub processes cannot be easily notated in a UML AD. A business analyst cannot model a business process and its sub-processes from the highest level to the lowest level of detail in an UML AD.

From the authors' observation and (Koskela and Haajanen, 2007), UML AD are increasingly losing favour with practitioners (although there are currently several projects working on UML-to-BPEL translations by IBM and OMG). This is mainly due to industry's growing consolidation of BPMN as the *de facto* standard for BP modelling.

3.2.4 Business process modelling notation. First released in May 2004 by Business Process Management Initiative (BPMI.org.), the graphical, flowchart-based BPMN is a recent BP modelling language that is already gaining wide acceptance (Koskela and Haajanen, 2007). Being a graphical notation, BPMN hopes to bridge the gap between IT and business analysts (OMG, 2007).

BPMI first expounded BPMN as a graphical representation of the Business Process Modelling Language (BPML), an XML-based process execution language. However, when BPML lost favour, its closest rival BPEL became the *de rigeur* BPEL. Despite this, BPMN held its own as the graphical representation of BPEL. There are currently 44 BPMN implementations (i.e. software tools) endorsed by OMG and four upcoming implementations (OMG, 2004a, b), many of which can generate BPEL code (OMG, 2008a, b, c, d).

3.2.5 Capabilities of BPMN. The BPMN elements (like activities, events, gateways, flows, etc.) in business process diagrams (BPDs) are compliant with most flow-charting notations but offer much more precise flow control semantics (Figure 7). Notably, BPMN is able to model private (internal) processes, public (abstract) processes (Mous *et al.*, 2007) and collaboration (global) processes at different levels of granularity. For example, roles (a.k.a swimlanes in BPMN) may be modelled from either the perspective of key stakeholders or from an inter-department perspective. Most BPMN models can be mapped to execution code (e.g. BPEL) which is its main strength over UML AD.

The theoretical underpinnings of BPMN are largely based on Petri nets, adopting the same token passing for flow control (White, 2004a; Havey, 2005).

3.2.6 Strengths and weaknesses of BPMN. BPMN enables roles to be defined at various levels of granularity through pools and swimlanes unlike UML AD (White, 2004a). For example, a business analyst designing BPMN processes may choose to represent the processes across departments, or across roles of different departments, or even across companies.

BPMN has a drawback. Since there is no XML interchange format for BPMN diagrams, OMG has introduced the business process definition metamodel – BPDM (Frank *et al.*, 2004; OMG, 2008a, b, c, d) specification but it is not yet supported by existing tools. Currently, XPDL (Workflow Management Coalition – WfMC, 2002) is the *de facto* interchange format due to its long history, stability and strong industry support.

The initial intent for BPMN to visualize BPEL has not succeeded (White, 2004a) because of irreconcilable differences between BPMN and BPEL, making it very difficult, and in some cases impossible, to faithfully generate BPEL code from BPMN models. Even more difficult is the synchronization of the original BPMN model and the generated BPEL code (White, 2004a). Furthermore, BPMN elements are hard to sketch on paper unlike UML AD or flowcharts (Wohed *et al.*, 2006). In spite of its widespread adoption, BPMN 1.0 is still incomplete and the request for proposal (RFP) of BPMN 2.0 is under way.
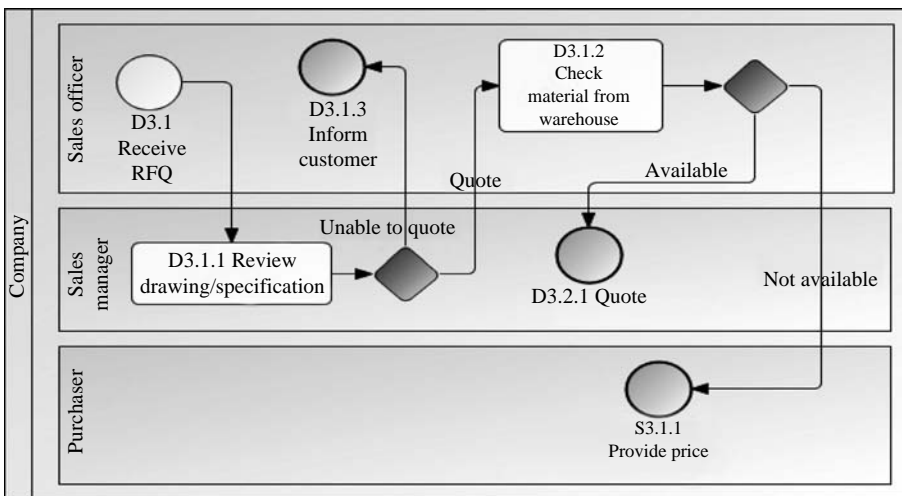
Figure 7.
An example of receive request for quotation (receive RFQ) business process notated in BPMN

3.2.7 Differences between UML AD and BPMN. BPMN has fewer core objects with variations to encompass complex processes (White, 2004a). Another difference between the two notations is terminology (Koskela and Haajanen, 2007; White, 2004a). For example, a UML AD has a start node while the BPMN's BPD has a start event (Koskela and Haajanen, 2007).

The BPMN BPD shows details at multiple levels of business processes. The granularity of roles can also be freely assigned through pools and swimlanes (OMG, 2004a; Wohed *et al.*, 2006). On the other hand, a UML AD is at a single-level perspective (Russell *et al.*, 2006; White, 2004a, b; Wohed, 2004). This, coupled with terminologies inherited from UML, is causing UML AD's popularity to wane.

The differences between BPMN and UML can be understood by considering the intended users of both notations. While BPMN was targeted at business analysts, UML (its AD) was primarily targeted for software development. Although the UML 2.0 development upgraded the AD to accomodate business analysts, it is still technically oriented.

According to White (2004a, b), since the BPD and AD have very similar views (i.e. higher level representation) of the same meta-model in BPDM, it is foreseeable that they will converge in the future. The OMG is determined to address the concerns of higher-than-software-development levels of business modelling, including the formulation of business rules and design of business processes. This is also the thrust of the authors' research.

Even though originally developed within BPMI, the future of a BPD may be part of the high-level business modelling infrastructure being developed within the OMG.

3.2.8 Event-driven process chain. Aside from BPMN and UML AD, there is the EPC (Scheer, 1992), which was developed by the Institute for Information Systems (IWi) at the University of Saarland, Germany. It is a language that is widely used in the ARIS Toolset of IDS Scheer AG and the workflow component of the SAP R/3 System. It was quite influential as a modeling notation in the 1990s.

An EPC is simple and easy for non-technical users to pick up. It works as an ordered graph of events and functions and supports parallel execution of processes. A notable feature of EPC is its logical operators (e.g. OR, AND and XOR). However, the semantics and syntax of the EPC are apparently not well defined (van der Aalst, 1999; Kindler, 2004). Because of these limitations and the absence of a standardization process, the EPC will not be classified as a graphical standard.

3.2.9 Other graphical representations. RADs and flowcharts are strictly not standards but tools to display the temporal transitions of business processes. One must never confuse them with standards. However, in the first author's own industry experience, end-users often fall back on flowcharts to depict the business processes because of the notational simplicity of flowcharts and RADs.

3.2.10 Strengths and weaknesses of graphical notations. The notable use of the Workflow Pattern Framework to evaluate BPMN and UML AD (White, 2004a; Russell *et al.*, 2006; Wohed *et al.*, 2006; Koskela and Haajanen, 2007) demonstrate that both notations could adequately model most of the workflow patterns.

The only exception was the absence of an adequate graphical representation of the interleaved parallel routing pattern in UML AD, even though the underlying UML AD metamodel has the appropriate structure to create the pattern (Dumas and ter Hofstede, 2001; Wohed *et al.*, 2006). The fact that both notations provide similar flow control

solutions to most of the patterns underscore their similarities. The UML AD and the BPMN share many graphical symbols (e.g. rounded rectangles for activities, diamonds for decisions, etc.). These similarities are understandable because both UML AD and BPMN are designed to represent procedural business processes.

Graphical notations like UML AD and BPMN are easy for non-technical business users to understand and use. Compared to the text-based execution-level standards like BPEL, graphical standards visually reveal patterns, loopholes and bottlenecks of a business process. However, the finite set of process diagram elements may restrict design freedom somewhat.

As mentioned earlier, because of the absence of semantic and computational formalisms in graphical notations, their models will never be able to fully translate into executable code. There will always be some loss of data or semantics of the control flow.

Although graphical standards provide a high-level representation of business processes, its focus is on flow control. Graphical standards are weak on the formulation, evaluation and measurement of the fulfillment of goals. Goal-based notations or language intrinsic to the language are desirable.

3.3 Execution standards
Execution standards enable business process designs to be deployed in BPMS and their instances executed by the BPMS engine. There are currently two prominent execution standards: BPML and BPEL. Of the two, BPEL is more widely adopted in several prominent software suites (e.g. IBM Webshere, BEA AquaLogic BPM Suite, SAP Netweaver, etc.) even though BPML can better address business process semantics.

3.3.1 Business Process Modelling Language. The BPML is an eXtensible Markup Language (XML) process definition language that describes the structural representation of a process and the semantics of its execution (Havey, 2005). Business processes modelled in BPML are run on an engine element by element, according to precisely defined semantics. Despite BPML being an XML-based code, it has a good balance of graphical and block-oriented paradigms, making it one of a few formally complete languages (Shapiro, 2002). Therefore, the code of a BPML process has not only graph-oriented constructs such as loops and parallel paths, but also block-oriented constructs such as variables, recursive blocks and structured exception handlings (Arkin, 2002).

The block-oriented constructs enable a BPML business process to be programmed, making BPML the leading light of the process-oriented programming paradigm. It is important for BPM practitioners to note that, in BPML, recursive block structures play a significant role in scoping issues that are relevant for declarations, definitions and process execution (Shapiro, 2002). Flow control is also handled entirely by block structure concepts (e.g. executing all the activities in the block sequentially) (Shapiro, 2002).

BPML was designed for business processes to be executed in contemporary web service-based BPMS (e.g. Intalio BPM). BPML's theoretical underpinnings are rooted in Pi-calculus (Arkin, 2000; Havey, 2005). Pi-calculus examines the interaction of two processes based on the flow of messages between them. Because of the underlying Pi-calculus formalisms, each participant in BPML may be flexibly defined; it can be as

simple as a stateless one-method service or as complex as a process with a well-defined message exchange.

The strengths of BPML are as follows (Shapiro, 2002; Smith, 2004; Koskela and Haajanen, 2007):

- BPML supports the concept of zero-code design driven deployment. This means that programmers do not have to dwell on low-level programming languages but focus on defining processes and their execution sequences (i.e. flow control). This is also known to practitioners as "programming in the large".

- BPML encourages reusability and scalability being an open standard for all BPMS just as the entity relation diagram and SQL are for relational database management systems (DBMS). As its backbone is XML, its components can be easily reused and parsed.

- Being formally complete, BPML can express complete end-to-end executable processes. This is its edge over the competing BPEL, which is formally incomplete and is limited in expressing business process semantics.

- BPML supports transactions (i.e. small modular activities) with ACID properties within and outside a process. This is another of BPML's edge over BPEL which does not support transactions. BMPL's model of transaction supports nested transactions, within nested processes, and across multiple BPML processes (i.e. within an end-to-end process).

BPML has its limitations (Shapiro, 2002; Smith, 2004; Koskela and Haajanen, 2007):

- The temporal component of a process is not evident in a BPML process definition as it is coded in XML. However, because of the graphical features, building a graphical BPML modeller is not difficult, as shown in earlier versions of the open sourced Intalio BPM suite (Shapiro, 2002).

- As it is a higher level of programming abstraction, BPML can only be supported in systems of pure BPMS vendors but not by dominant market products like Microsoft's BizTalk and IBM's MQServer and Websphere. Ironically, these dominant commercial products needed only a simpler structure and were not ready to adopt BPML when BPML was first introduced. This led Microsoft and IBM to develop their own languages XLANG and Web Service Flow Language (WSFL), respectively, (Smith, 2004).

Despite BPML being a formally complete business process standard (Arkin, 2002; Shapiro, 2002; van der Aalst *et al.*, 2002a, b; Smith, 2004), it is no longer supported by its founding organisation BPMI after its merger with OMG in 2005 (OMG, 2008a, b, c, d).

3.3.2 The rise and wane of BPML. As shown in Figure 8, the idea of an open standard for the modelling of business processes was first mooted in 1999 by practitioners, developers and many software vendors, including giants like Microsoft and IBM. In 2001, Microsoft and IBM released their own BP execution languages XLANG (Thatte, 2001) and WSFL (Leymann, 2001) to enhance their products with workflow capabilities. Both XLANG and WSFL were viewed skeptically in industry circles as Microsoft's BizTalk and IBM's MQServer systems stop-gap measures as both were not yet ready for the rich language of BPML (Smith, 2004).
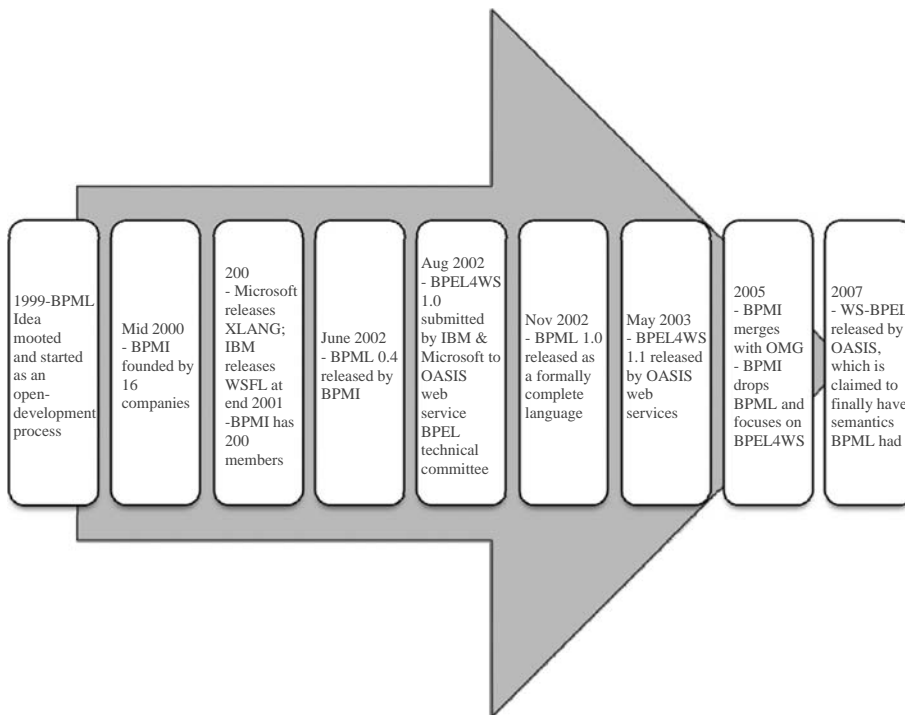
In June 2002, BPML 0.4 was finalised after three years of hard work by the BPM community, led by one Arkin (2002). In August 2002, three months before BPMI's release of BPML 1.0 in November 2002, IBM and Microsoft merged XLANG and WSFL into Business Process Execution Language for Web Services (BPEL4WS) and submitted it to Organization for the Advancement of Structured Information Standards (OASIS) for consideration of adoption. Following this move, many small BPMS vendors were torn between adopting BPML or BPEL's as the execution standard of their software (Smith, 2004).

In May 2003, BPEL4WS 1.1 was released by OASIS coinciding with the rise of the SOA paradigm (Andrews *et al.*, 2003). With large organizations like IBM, Microsoft, BEA and SAP leading the way, the BPM market (including smaller BPM vendors) consolidated towards BPEL.

This trend of "going with the flow" was identified by many industry observers and even academics (van der Aalst, 2003a, b). In the authors' opinion, this consolidation was a step backwards for BPM. BPEL is in many ways not a complete language even today. For instance, BPEL's second version (i.e. Web Service Business Process Execution Language – WS-BPEL) had to be semantically enriched in the form of the extension WS-HumanTask in order to model human participation in business processes (Smith, 2004). In contrast, BPML can intrinsically model human participation.

The execution standard level of business process modelling is now dominated by the formally incomplete BPEL. However, ironically BPEL is beginning to look more like BPML with patchy extensions.

3.3.3 BPEL, XLANG and WSFL. BPEL, an XML-based language for specifying business processes in the web service environment, is a collective term for both its versions (Andrews *et al.*, 2003; OASIS, 2007):

- BPEL4WS Version 1.1; and
- WS-BPEL Version 2.0.

BPEL is currently the most influential execution standard in the market, as such, we will be devoting more time on it. It is used together with Web Service Definition Language (WSDL) and other related technologies. This means that BPEL is used to define how the business process is built from invocations of existing web services and the kind of interaction of the process with external participants.

Technically, BPEL can be seen as an XML-programming language for web service compositions. In-depth understanding of BPEL requires software development competence as well as knowledge of the underlying web service technologies. The first version, BPEL4WS 1.0, was originally submitted to OASIS (2007) WSBPEL Technical Committee by Microsoft and IBM in July 2002. BPEL4WS combined properties from Microsoft's WSFL (Leymann, 2001) and IBM's XLANG (Thatte, 2001) (Figure 9).

The revised version of BPEL4WS (i.e. Version 1.1) (Andrews *et al.*, 2003) has been widely adopted by tool vendors. In Version 2.0, the language was renamed WS-BPEL and approved as an OASIS (2007) standard. In this report, the acronym BPEL is generally used to refer to both versions because the older version cannot be considered obsolete. Where necessary, clear distinctions between the versions are made. The new version involves syntactic changes and improved alignment with other XML technologies such as XPath (Koskela and Haajanen, 2007). Hence, it is important to note that newer versions of BPEL processes are not backward-compatible.

From the viewpoint of the creators of BPEL (Andrews *et al.*, 2003), business processes can be described in two ways (Shapiro, 2002):

(1) *Executable business processes*. Model actual details and behaviour of a participant in a business interaction.

(2) *Business protocols*. In contrast, use process descriptions that specify the mutually visible message exchange behaviour of each of the parties involved in the protocol, without revealing their internal behaviour and details. The process descriptions for business protocols are called abstract processes (Havey, 2005).

In other words, the full implementation logic of the business process is defined via executable processes, while only the message exchange between process participants (i.e. business protocols) is modelled in abstract processes. A business process written in BPEL consists of two types of files (Andrews *et al.*, 2003):
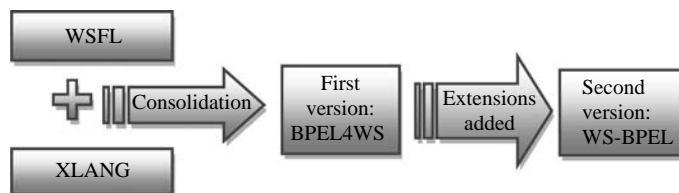

Figure 9.
Evolution of BPEL

(1) The BPEL file, encoded in XML, forms the "stateful" definition of a process, including its main activities, partner links, variables, and event handlers.

(2) The accompanying WSDL files specify the "stateless" web service interfaces that are of interest to the process defined in the BPEL file (e.g. services implemented by and called by the process).

Structured in XML, the core elements of a BPEL document are greatly influenced by web service concepts, and include (Koskela and Haajanen, 2007; OASIS, 2007):

- roles of the process participants;

- port types required from the participants;

- orchestration, which is the actual process flow; and

- correlation information, the definition of how messages can be routed to correct composition instances.

BPEL activities can be either basic or structured activities (Andrews *et al.*, 2003; OASIS, 2007). Basic activities correspond to actual components in a business process. These activities are realized through web service interactions (i.e. through invocations of WSDL operations). On the other hand, the structured activities resemble control structures of conventional programming language. They constitute the block-oriented part of BPEL, which originated from XLANG (Thatte, 2001). Additionally, BPEL specifies handlers for events and faults. For each handler, an event, a scope and a corresponding activity to handle the event are defined (OASIS, 2007).

The order of execution inside a <flow> element can be controlled using <link> elements. This defines the limited, acyclic graph-oriented nature of BPEL, which originates from IBM's WSFL. Consequently, BPEL links are crucial when BPMN's BPDs are transformed into executable processes. However, BPEL links (OASIS, 2007):

- cannot cross the boundaries of repeatable constructs such as <while>, and in WS-BPEL 2.0 only outbound links can cross <catch>, <catchAll> (OASIS, 2007); and

- <terminationHandler> scopes a <link> declared in <flow> cannot create a control cycle.

The strengths of BPEL are:

- Most popular and does not have any serious competitors in the industry (Havey, 2005; van der Aalst *et al.*, 2005a, b; Woodley and Gagnon, 2005; Koskela and Haajanen, 2007). This means that BPEL-compatible products are stable and the risk of obsolescence is minimal. As it has been adopted by major software vendors, portability is not an issue with products of small BPMS vendors.

- It focuses on processes rather than low-level programming constructs. In comparison with conventional programming languages such as Java, BPEL can model typical business process interactions such as long-term transactions, asynchronous messaging and parallel activities. It would need much more effort and lines of code to express the same process in a conventional programming language (van der Aalst *et al.*, 2005a, b).

- It subscribes to the web services paradigm. This means BPEL capitalises on the dynamic and highly adaptive nature of web services. BPEL incorporates a number of specialized features for web services development including direct support for XML data definition and manipulation, a dynamic binding mechanism based on the explicit manipulation of endpoint references, a declarative mechanism for correlating incoming messages to process instances, which is essential for asynchronous communication. As such, BPEL may be seen as an attractive alternative to conventional programming languages when it comes to developing web services (van der Aalst *et al.*, 2005a, b).

Some of BPEL's weaknesses include:

(1) *Complex syntax and difficult to implement.* Whilst undeniably a powerful language, BPEL is difficult to deploy. Its XML representation is very verbose and only readable to the trained eye (van der Aalst *et al.*, 2005a, b). It offers many constructs and so different implementations can yield the same end result (e.g. using links and the flow construct or using sequences and switches). The choice of the best construct depends on experience. Although several vendors (Oracle Process Manager, Intalio BPM) offer a graphical interface that generates BPEL code, these interfaces are often just a direct reflection of the BPEL code and are not intuitive to end-users. Therefore, BPEL is more akin to classical programming languages than the more user-friendly WfMS today like Metastorm BPM (van der Aalst *et al.*, 2005a, b).

(2) *Restrictive syntax.* This severely limits its modelling capabilities and is a source of many problems in the BPMN-to-BPEL transformations. Some examples of this limitation are as follows:

- *Incomplete standard.* Although BPEL is an important standard, some critical features like an overarching framework and supporting tools for behavioural service mediation are missing (van der Aalst *et al.*, 2005a, b).

- *Limited graphical support: cannot support cyclic processes (Smith, 2004; van der Aalst et al., 2005a, b; Koskela and Haajanen, 2007).* BPEL's Achilles' heel must surely be BPEL's acyclic nature consistent with the link sequences of web services (Shapiro, 2002). In the author's experience, real-life business processes often contain cyclical components (e.g. negotiation procedures). BPEL will not be able to model them properly.

- *Abstract business processes and is not really adopted in the industry (van der Aalst, 2003a, b).* Although abstract BPEL processes model business protocols, abstract BPEL processes only model the perspective of just one party of the collaboration (van der Aalst *et al.*, 2005a, b). This is clear in Figure 10, which shows the executable and the abstract process codes in BPEL. It is clear that in both cases, the work is seen from the perspective of one of the partners (van der Aalst *et al.*, 2005a, b). In the authors' and van der Aalst's *et al.* (2005a, b) opinion, BPEL is grossly inadequate as a language for modelling abstract processes.

(3) *BPEL does not model human involvement in business processes well.* This is a serious setback in BPM. WfMS have always been able to model human participation in business processes and if execution languages like BPEL

(a) Executable business process model
(b) Abstract business process model
Source: van der Aalst *et al.* (2005)
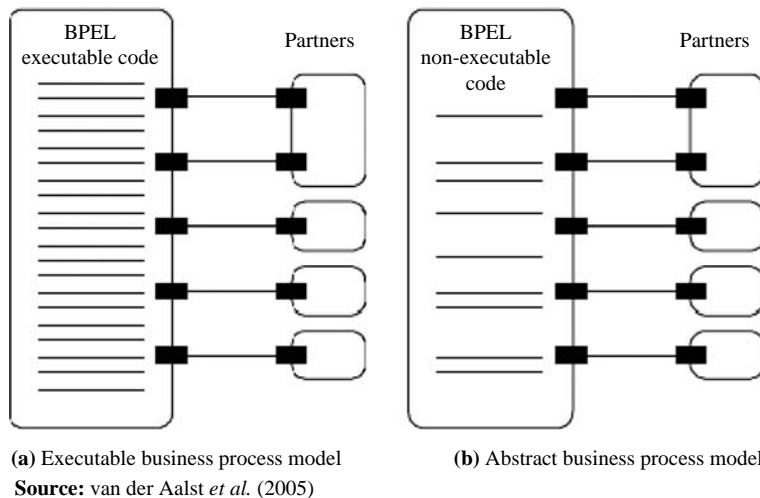
Figure 10.
Demonstrating that work
can only be seen from the
perspective of one
collaborator in both
abstract and executable
BPEL

cannot do so, this is a severe drawback. The gap was exposed with recent extensions of BPEL (e.g. BPEL4People and WS-HumanTask) aiming to model human involvements in business processes (Smith, 2004).

(4) *BPEL does not have some process constructs (Koskela and Haajanen, 2007).* Because of this, it is not possible to express all conceivable business processes in BPEL. For this reason, BPEL is often used in conjunction with programming languages, for example Java, or embellished by proprietary scripting languages inherent to commercial implementations of workflow or integration broker engines (Koskela and Haajanen, 2007).

(5) *BPEL emphasizes web service definitions at the expense of work and resource distribution.* In every BPEL file, a considerable amount of coding is needed to ensure that web services are properly coordinated and linked. This is in contrast to the design interfaces of WfMS (i.e. Metastorm e-Work (Metastorm, 2007), Savvion BPM Software (Savvion, 2008), SAP Business Workflow (SAP, 2008b)) that the first author has worked with. In these systems, there is an emphasis on the ease of design of the flow, roles and information passed, and less so on making the back-end coding work. The back-end coding is necessary for customisation but is never a complex action like that of BPEL. In the authors' opinion, despite BPEL's alignment with the web services platform, it is unable to address the needs of BPM directly.

(6) *Lack of support for B2B collaboration.* BPEL as a standalone cannot support B2B collaborations as it is fundamentally designed from a single perspective of the business process (Figure 10). For process execution in electronic business networks, better support for collaborative processes is required. Choreography languages such as WSCDL address this need, freeing BPEL to implement the internal processes (Wohed *et al.*, 2003a, b; van der Aalst *et al.*, 2005a, b).

Regardless of its weaknesses, BPEL is still the most popular execution language and is maturing into a semantically richer execution standard with its WS-BPEL version.

3.3.4 Comparing BPML and BPEL. Although there is a consensus in industry (Ghalimi and McGoveran, 2005; Havey, 2005; Khan, 2005) that BPEL and BPML are similar except for their syntax, both execution standards are actually quite different. Despite BPML's eventual demise, it still has an indirect influence on BPEL's developments. Unknowing to detractors of BPEL, BPML actually serves as a benchmark for the comparison of BPEL's features as an executable standard. Without BPML, industry would not be able to assess the flow control constructs and business process semantics of BPEL.

Both BPEL and BPML focus on issues important in defining web services. This is reflected in several ways (Shapiro, 2002):

- activity types specifically for message interchange, event handling, compensation (in case of failure), and delay;
- attributes to support instance correlation, extraction of parts of messages, locating service instances; and
- support for transactions, utilizing the block structure context, exception handling and compensation.

As BPEL omits certain process constructs, not all conceivable business processes can be expressed in BPEL (Shapiro, 2002). For this reason, BPEL is often used in conjunction with programming languages, for example Java, or extended by the proprietary scripting languages inherent to commercial implementations of workflow or integration broker engines (Shapiro, 2002). Indeed, in practical applications, there is nearly always the need to extend the language with other programming tools.

In contrast, BPML, being a complete language, can implement additional semantics as processes, rather than adding new "tags" to the XML as in the case of BPEL. The scalability of BPML over BPEL is therefore evident.

In the view of van der Aalst *et al.* (2005a, b) organizations do not need to agree on a common execution language. Instead, they argued that there are more important issues to be addressed:

- developing a higher-level language to describe both processes and interactions; and
- monitoring the execution of composite web services/choreographies (van der Aalst *et al.*, 2005a, b).

3.3.5 Yet Another Workflow Language. Proposed by academics from TU Eindhoven and Queensland University of Technology (van der Aalst *et al.*, 2004; van der Aalst and ter Hofstede, 2005), YAWL is a comprehensive workflow execution language based on all workflow patterns (van der Aalst *et al.*, 2002a, b). Although technically not a "standard", YAWL has rich control flow constructs. It is taught in many universities and employed by the hotel and telecommunications industry (YAWL, 2007). YAWL is XML based and is supported on an open-source software package that includes an engine, graphical editor and a work-list handler. YAWL can handle more control flow constructs than Petri nets, the basis of basic workflow patterns.

3.3.6 Strengths and weaknesses of execution standards. The key strengths of execution standards are as follows:

- *Ideal for automation by computers.* Syntax-based and block-oriented execution standards facilitate business process automation in IT systems. Currently, many execution standards (e.g. BPEL, BPML, etc.) are based on the well-structured XML. XML is easily customisable and scalable and yet has rich block-oriented features like nesting, structures, and good parsing capability. Older versions of business process models can be easily modified without the need for a drastic overhaul.

- *Captures many hidden semantics that graphical standards cannot.* With a clear set of syntax, nested features and a formalised method of expressing business processes, execution standards can encapsulate logical details succinctly.

However, execution standards are limited in the following ways:

- *They are not high-level and relationships are not visibly obvious.* Like assembly languages in computing, the sequence, activities and the linkages in and between business processes are not visually obvious in execution standards.

- *They require some technical knowledge and web-services know-how.* Unlike the flowchart-like graphical standards, the execution standards require some technical knowledge; in recent years, primarily web services. This is a veritable obstacle for process-owners and business analysts who are conversant in business process design but not the technical implementation.

3.3.7 Trends of execution standards. Prior to the industry's recent consolidation towards BPEL, there were several proposals to streamline business process execution. These proposals can be classified as follows:

- *Extensions of programming languages.* After the ascendancy of WfM in the late 1990s, many programming languages began to cater to the design of web services. A prominent example is the Java Business Process Management, which is supported on many Java systems like the community developed JBOSS (Koenig, 2004).

- Consequently, the problems and limitations of programming languages were directly inherited by business process designs. For example, in order to link with other programming language-oriented business process extensions, one will need to build interfaces (e.g. application program interfaces), which are often not dynamic, scalable and manageable.

- *Web service-based proposals.* The other category of execution standards are those based on web services. The web services paradigm focuses on service-oriented concepts, meaning that programs or business processes can be created by mixing and matching modular programs each serving a particular function. Service-oriented programs are reusable and are very adaptive to changes in requirements and computing environments.

Because of the growing dominance of the SOA, the web service-based proposals are currently the most influential. Execution standards based on adaptive paradigms are

better received due to the user's need to survive in an increasingly globalised business climate.

The authors acknowledge the important contributions of web service standards to BPM (e.g. WS-CDL (Kavantzas *et al.*, 2004)). However, as they are low level and not BPM specific, web service standards are not included in our discussions.

Significant web service-based business process products include the BEA AquaLogic BPM Suite (BEA, 2008), IBM WebSphere (WebSphere, 2005), SAP Netweaver (Woods and Word, 2004), Intalio BPM Suite (Intalio, 2008), Oracle BPM (Oracle, 2008a), IDS Scheer ARIS Platform (Scheer, 1992), Savvion BPM (Savvion, 2008), etc.

3.4 Reasons for interchange standards: fundamental differences between graphical and execution standards

By now, the reader should be aware that graphical and execution standards have some fundamental differences: diagrammatic graphical standards are graph-oriented while the syntax-based execution standards are block-oriented (Koskela and Haajanen, 2007). While graph-oriented graphical standards represent temporal progression and logical flow through nodes and inter-connecting arcs, block-oriented execution standards control flow by nesting different kinds of syntactical control primitives (e.g. XML) (Koskela and Haajanen, 2007).

Because of this, information is often lost in transformations from graph-oriented graphical standards to block-oriented execution standards and vice versa (Koskela and Haajanen, 2007). The semantics of business processes expressed intrinsically in graph-oriented models cannot be adequately expressed by the limited constructs of block-oriented standards.

An example is the expression of temporal sequence. Temporal aspects of business processes expressed in graphical standards via nodes and arcs cannot be faithfully displayed in a text-based block-oriented standard.

Standardisation groups (e.g. OMG) which pioneered interchange standards often claim their creations as the missing link between the business analyst and the IT specialist (Frank *et al.*, 2004). In the authors' opinion, this is only half true as there are still many aspects of business process modelling (e.g. goals, context, semantics, ontological matching, role definitions, etc.) that current standards fail to address. In fact, it is more accurate to say that interchange standards are the non-contextual translator between graphical standards and execution standards.

3.5 Interchange standards

As mentioned earlier, interchange standards are needed to:

- translate graphical standards to execution standards; and
- exchange business process models between different BPMS's (Mendling and Neumann, 2005).

Although many practitioners thought these interchange standards as "the link between business and IT", the authors do not totally agree with this assertion because (Koskela and Haajanen):

- an interchange standard is essentially a translator from a graphical standard to an execution standard;

- the translation from execution standards to graphical standards is hardly achievable (Mendling and Neumann, 2005); and
- in the authors' opinion, interchange standards currently focus only on the flow control aspect of business processes and hardly address other aspects (e.g. level of fulfilment of business goals, human roles, other business process semantics, etc.).

There are currently two prominent interchange standards:

(1) BPDM by OMG.

(2) XPDL by the WfMC.

3.5.1 Business Process Definition Metamodel. The BPDM is an XML-based proposal by the OMG. It was initiated following a RFPs issued on 31 January 2003 and is still in its formative years. At the time of writing, the finalization of the specifications is underway (OMG, 2008a, b, c, d). BPDM provides the capability to represent and model business processes independent of notation or methodology, thus bringing different approaches together into a cohesive capability (OMG, 2008a, b, c, d).

As its name suggests, the BPDM was meant to be the authoritative meta-object facility (Frankel, 2003) (an abstract modelling language by the OMG) metamodel for the common elements in process definitions (Havey, 2005).

According to OMG (2008a, b, c, d):

The metamodel behind BPDM captures business processes in a very general way and provides a XML syntax for storing and transferring business process models between tools and infrastructures. Various tools, methods and technologies can then map their way to view, understand and implement processes to and through BPDM.

Simply put, a metamodel is basically an abstract model that contains basic elements found across many modelling languages. In this case, the "metamodel" feature in BPDM is an abstraction of basic and common elements found across BPEL, BPMN, XPDL, XLANG, WSFL and UML AD.

This means that BPDM works like a multi-lingual standards translator with a common platform. This qualifies BPDM as an interchange standard. Havey (2005) summarizes the current interchange capabilities of BPDM as shown in Figure 11.

According to Harmon (2004), BPDM is not as concerned with graphical notation as with semantics. It is conceivable that vendors will choose to maintain their existing notations but use the OMG BP metamodel to facilitate the transfer of information to
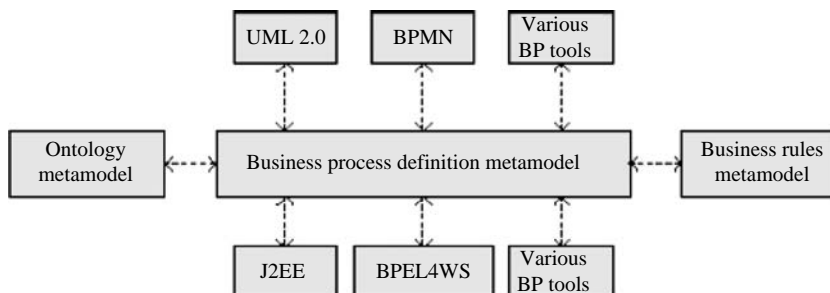
Source: Harmon (2004)

Figure 11.
BPMD serving as an
interchange mechanism

other tools and models. In other words, a variety of different notations can continue to thrive in the OMG BP metamodel. In the long-run, however, the OMG will probably move most companies toward UML AD (Harmon, 2004).

According to the text of the RFP, BPDM:

[...] will define a set of abstract business process definition elements for specification of executable business processes that execute within an enterprise, and may collaborate between otherwise-independent business processes executing in different business units or enterprises (OMG, 2008a, b, c, d).

3.5.2 Current status of BPDM. From a higher perspective, BPDM is nurturing a unified theory of process definition. Given this tall order, it is not a surprise that in many online technical forums, BPDM is criticised as a complex and user-unfriendly standard. As the BPDM is relatively immature with no software tool using it, there will be little emphasis on this standard in this report. The challenge posed by WfMC's well-supported and more stable XPDL to BPDM has yet to play out.

3.5.3 XML Process Definition Language. In the fast-changing world of BPM standards, many standards do not even get to see the light of day (van der Aalst, 2003a, b; van der Aalst *et al.*, 2005a, b). However, the XML-based XPDL stood the test of time and will mark its tenth year anniversary in 2008. XPDL started in 1995 when the WfMC published the workflow reference model identifying five key interfaces necessary for any WfMS (WfMC, 1995). In this reference model, one of the interfaces was for defining business processes. It includes a process definition expression language developed *via* a programmatic interface (i.e. process definition tool) to transfer the process definition to/from the workflow management system.

Initial Foundations as an Execution Standard. The first version of such a process definition language was called the Workflow Process Definition Language (WPDL) (WfMC, 1999) published in 1998 (van der Aalst, 2003a, b). WPDL contained all the key concepts required to support workflow expression and automation using the then state of the art: URL encoding (Fielding, 1995).

However, by 1998, the first standards based on XML began to emerge. The first move to establish XML as an interchange language had happened. The WfMC Working Group 1 produced an updated process definition expression language called XPDL, now known as XPDL 1.0 (WfMC, XML, 2002). This second revision was an XML-based interchange language that contained many similar concepts of WPDL, but with some improvements. XPDL 1.0 (Marin, 2001) was eventually approved by the WfMC in 2002, and was subsequently implemented by more than two dozen workflow software products (e.g. Metastorm e-Work, Lombardi, etc.) as a process definition exchange mechanism.

How XPDL matured into an interchange standard. From 2002 to 2004, XPDL was an influential standard for the interchange of process design. This was especially so after WfMC endorsed BPMN as a graphical standard in 2004, after it was enhanced to represent the concepts present in a BPMN diagram in XML.

This extension made XPDL ideal not only as a definition (i.e. execution) standard for business processes, but also as an interchange format between BPMN and XML-based execution standards (e.g. BPEL). The third revision of XPDL (XPDL 2.0) was released by the WfMC in 2005 (WfMC, 1995). Today, there are about 70 different BPM-related software based on XPDL (2008).

Shift of focus from process definition (i.e. execution) to process interchange. Many practitioners have downplayed the process interchange capabilities of XPDL which they view as an execution standard (Pyke, 2007). As its flow control features cannot be compared to that of BPEL and BPML (Shapiro, 2002; van der Aalst, 2003a, b), the main strength of XPDL still remains in its interchange capabilities, which is its selling point. There are currently over 70 products and applications that leverage XPDL on Java, Microsoft.NET Framework, or Linux. Some examples include Oracle 9i Warehouse Builder, IDS Scheer Business Architect, BEA Enterprise Repository and BPM Suite, etc.

3.5.4 Strengths and weaknesses of interchange standards. The strengths of interchange standards include:

- interchange standards offer a "globally accepted" file format to save process definitions. Business process models in different BPMS are perfectly compatible; and
- XPDL is well-accepted and stable, having had a ten-year history.

The shortcomings of interchange standards include:

- Owing to fundamental differences in graph-oriented graphical and block-oriented execution standards, the quality of transformation of the interchange standards is limited by different syntax and structures. For instance, a cyclical and temporal implication in a graphical standard cannot be easily transformed into an execution standard. The translation of recursive capabilities from an execution standard to a graphical standard is an even more challenging task.
- Currently in the industry, translation from graphical to execution is easier than that from execution to graphical standards. This applies to XPDL and even BPDM. This limitation raises doubts as to whether the "bridge between the business analyst and the IT specialist" is near in sight.
- In the first author's work experience, the need to exchange business process models amongst systems is not an everyday activity and XPDL formats sufficient for most purposes. Also, many WfMS (and BPMS) already have the capability to translate (in a governed fashion) their graphical designs into XML execution codes.

3.5.5 Some observations and trends of interchange standards. Recalling the earlier discussions about block-oriented and graph-oriented paradigms of business process standards, it is a brave attempt by BPDM to mediate amongst so many forms of graphical and execution standards. Given their fundamental differences, it is impractical to expect a perfect transformation from the graphical to the execution standards and vice versa.

In the authors' opinion, a BPDM standard is not necessary if each level of the BPM standards had been consolidated. After all, that is what a standard is supposed to do. The BPDM is but a patch work that breeds a multitude of "yet-another" standard. At the moment, the industry's consolidation to XPDL as the interchange standard offers the best hope for a universal interchange standard. XPDL can save both the graphical and the execution forms of business process designs digitally in XML. This runs counter to industry's move to digress from the already capable XPDL to BPMN, BPML and BPEL. Some pro-BPEL programmers can argue that the flow control features of XPDL are not

as rich as those of BPEL and BPML (Mendling and Neumann, 2005) but in the authors' opinion, these are just technical features that can embellish the next version of XPDL.

3.6 Diagnosis standards

The core difference between WfM and contemporary BPM lies in the diagnosis portion of the BPM life cycle (van der Aalst *et al.*, 2003). Diagnosis standards monitor and optimize business processes running in and across companies' BPMS. Audit trails, real-time business process information, trend analysis and bottleneck identification are just some of the important diagnostic tools that aid a company manage their post-modeling phase of the BPM life cycle. However, in the authors' observation, it was only recently that diagnosis standards attracted due attention.

After the WorldCom auditing scandal and Enron's fraud, many companies tightened the management of their business processes via strict corporate governance. The US senate passed the Sarbanes-Oxley (SOX) Act of 2002 (Washington DC Government Printing Office, 2002) to regulate financial business processes and IT processes in all US companies. Many software companies (e.g. CrystalReports.com, 2008; SAP, 2008a; Oracle, 2008b, etc.) started to market their process related products as SOX tools. This section introduces the beginnings of BPM diagnosis standards and highlights their potential. Though an important component of BPM, diagnostic standards are not the main focus of the authors' research and so the literature review will be brief.

3.6.1 Business Process Runtime Interface. The Business Process Runtime Interfaces – BPRI (OMG, 2002) is a RFP initiated by OMG in 2002 to define a common interface for the execution engines of different vendors. This interface, it is hoped, will facilitate human interaction and drive down costs, and create a new market for environments that will use those execution engines (Havey, 2005). As of the time of writing, BPRI is still not finalised.

3.6.2 Business Process Query Language. The Business Process Query Language (BPQL) is currently being developed by BPMI.org (i.e. OMG due to the 2005 merger) and will be the first standards-based query language for business processes. BPQL will support the deployment of business processes onto a process server and the real-time querying of process instances (Ghalimi and McGoveran, 2005). This facilitates process mining and efficient runtime management of business processes; for instance, "Which stage of the business process is the current bottleneck of the business process?" To the authors' best knowledge, BPQL has been on hold after the merger of OMG and BPMI.

3.6.3 Other diagnosis attempts. There are other lesser known and less influential languages and tools to facilitate business process post-modelling diagnosis. These include (Beeri *et al.*, 2005, 2007; Ghalimi and McGoveran, 2005):

- Business Activity Monitoring Language.
- Business Process Audit Trail Schema.
- Business Process Query Project by Catriel Beeri and the Israel Science Foundation.

In the authors' observation and experience, the current market trend leaves software vendors provide the administration, monitoring and analysis tools as built-in features of their BPMS. Although the diagnosis portion of the BPM life cycle is an important process that makes BPM different from WfM, the industry's push for standards for the diagnosis stage is not aggressive at the moment.

In our opinion, if the diagnosis standards cannot be established, the advancement from WfM to BPM will only be a simple name change and not an advancement of a field.

3.7 Beyond BPM: B2B information exchange standards
While the management of internal business processes via BPM is crucial, a serious BPM practitioner and researcher should never overlook how organisations collaborate with each other. One must remember companies exist mainly to turn in a profit, via the fundamental activities of buying and selling of products or services, Processes exist within and across companies to support this high-level goal of making profits.

Hence, an efficient methodology to support collaborative business processes in B2B collaborations is crucial. Since the late 1970s, standards which facilitate information exchange in B2B collaborations (a.k.a. collaborative business processes or B2B integration) have been introduced. A few examples of these B2B information exchange standards include:

- *Electronic data interchange.* Electronic data interchange – EDI (ANSI, 1979; UN/EDIFACT, 1990), one of the early B2B information exchange standards, was created for communications between different proprietary formats of collaborating partners. There are two predominant forms of EDI; the American National Standards Institute X12 standards (ANSI, 1979) and the European UN/EDIFACT standards. In 1987, the International Organisation for Standardisation (ISO) adopted the EDIFACT standard. EDI serves to facilitate document exchange between companies. It is a medium for exchanging business documents with external entities, and integrating the data from those documents into the company's internal systems. This is done via a value-added network, which is like a post office that forwards the data bundles to their designated businesses for a service fee.

- *ebXML BPSS.* The Electronic Business using eXtensible Markup Language (ebXML) (OASIS and UN/CEFACT, 2001a, b, 2002) was formalised in 2001 as a joint initiative between the United Nations Centre for Trade Facilitation and Electronic Business – UN/CEFACT and OASIS (2002). Presently, it is a full set of ISO standards maintained by its two contributing organisations (Mertz, 2001). ebXML's stated objective was to make it possible for any business of any size in any industry to do business with any other companies anywhere in the world. The initial hope was that the presence of an accepted international e-business standard would motivate small business software developers to support ebXML. Compared to RosettaNet, ebXML is a collection of general standards which are not specific to any business (i.e. horizontal standards) while RosettaNet comprises specific standards, thereby making a thorough coverage (i.e. vertical standards). ebXML is adopted at much lower cost as compared to RosettaNet.

- *RosettaNet.* RosettaNet (RosettaNet, 1998), launched in June 1998, aims to standardise supply chain interactions by creating interoperable collaborative business processes (Gibb and Damodaran, 2002). Member companies transact billions of dollars within their trading networks using partner interface process (PIP) specifications (RosettaNet, 1998). PIPs are system-to-system, XML-based dialogues that represent operational-level collaborative business processes,

e.g. request for quote. Each PIP defines how two specific processes, running in two different partners' organizations, are standardized and interfaced across the entire supply chain. PIPs include all business logic, message flow, and message contents to align the two business processes (RosettaNet Program Office, 2007). The entire scope of RosettaNet processes is divided into seven clusters containing all supply chain processes (RosettaNet Program Office, 2007): partner product and service review, product information, order management, inventory management, marketing information management, service and support, and manufacturing.

RosettaNet distinguishes between internal (i.e. private (Mous *et al.*, 2007)) and external (i.e. public) business processes. RosettaNet specifies a wide range of public processes using the PIP specifications. A PIP specifies information content and a sequence of message exchange and is based on a peer-to-peer business exchange between trading partners. Either partner can initiate a business process, addressing another partner directly.

- *Universal Business Language*. Universal Business Language – UBL (Meadows and Seaburg, 2004) is a royalty-free library of XML-based, commonly used business documents such as purchasing orders, invoices, legal documents, etc. It is an international effort by OASIS, designed to eliminate the re-keying of data in existing fax- and paper-based business correspondence and provide an entry point into electronic commerce for small and medium-sized businesses. Its second version, UBL 2.0, was released in 2006.

B2B information exchange standards like those mentioned above essentially standardize information exchange but still do not address the real needs of a dynamic business process collaborations like those discussed by Tan *et al.* (2006) and Wombacher *et al.* (2003). To our best knowledge, current B2B information exchange standards are still unable to dynamically formulate or update collaborative business processes according to real-time business goals. It is the authors' belief that the challenges in increased globalisation will pave the way for a conceptual merger of B2B Standards and BPM Standards. When such a state is realised, companies will be more responsive, and can react on-the-fly to adverse changes in demands or supply chain contingencies.

4. Gaps identified after BPM standards classification

Following the classification of BPM Standards, it became apparent to the authors that the following issues remain to be resolved.

4.1 Conceptual gaps in BPM standards

4.1.1 The significance of the BPM life cycle. Before creating "yet-another" BPM standard, one would need to consider the stage of the BPM life cycle that the proposed new standard is meant to address. Then, instead of proposing yet another BPM standard, the gaps in existing standards should first be plugged. Thus, the BPM life cycle compels practitioners and researchers to improve on existing standards rather than invent new ones.

4.1.2 BPM is not program linking. Perhaps, due to the recent increased interest in SOA, there is currently an over-emphasis on web services choreography and orchestration and not on the actual needs of BPM.

BPM is not mere agglomeration of programs. The linking of modular programs like web services merely process information (Medina-Mora *et al.*, 1992). Medina-Mora *et al.* assert that business processes are more than mere information transmission.

Notwithstanding the power of web services, an over-emphasis on web services means an over-emphasis on the flow control aspect of business processes at the expense of other aspects like business goals (Koubarakis and Plexousakis, 1999; Andersson *et al.*, 2005).

4.1.3 Lack of evaluation techniques for all BPM standards. A comprehensive evaluation technique is needed to assess the robustness of BPM standards. Right now, it is not possible to ascertain the quality of BPM graphical, interchange, diagnosis or execution standard.

To date, the most notable evaluation technique is the Workflow Patterns Framework (Dumas and ter Hofstede, 2001; van der Aalst *et al.*, 2000, 2002a, b; Wohed, 2004; Wohed *et al.*, 2003a, b) which evaluates BPM graphical and execution standards like UML, BPMN, BPEL and BPML. Besides, control flow, the fulfillment of business objectives, the usage of computing resources of the standards' underlying infrastructure, the ability to integrate human participation, etc. are other important metrics.

4.2 Gaps in the graphical standards

4.2.1 Lack of computational formalisms. Graphical standards often trace their theoretical roots to Petri nets but the actual underlying formalism is often not clear (Wohed *et al.*, 2006). If there are to be new graphical formalisms, these should be more theoretical.

4.2.2 Graphical standards still need some learning. Although graphical standards are easy to use and compactly depict business processes, their notation are not the most intuitive to humans. This causes end-users to fall back on the less expressive but easy-to-use flow charts and RADs. Greater effort is needed to foster more widespread learning and adoption of BPMN and UML AD notations and symbols.

4.3 Gaps in the interchange standards

BPDM was proposed as a metamodel of influential modeling notations and standards so that this metamodel can serve as an interchange mechanism. While the notion to have a metamodel of influential business process standards is a noble idea, the scale of covering the modeling features of all aspects of the targeted standards is overwhelming and would not be adaptable to new versions and extensions to the targeted standards.

The BPM field needs a new proposal for a scalable, flexible and embodying interchange methodology standard that is not at the mercy of new standards in the market or a change in versions of the standards to be interchanged.

4.4 Gaps in the execution standards

Although some execution standards (e.g. BPEL and BPML) claim that their theoretical basis is on established process algebra or formal notations, to our best knowledge, this is largely unsubstantiated. This highlights the need for assessment techniques highlighting the degree of influence of a formalism on an execution standard and how the formalism's constructs address execution issues. Such assessment techniques will also highlight the metrics of evaluating how well the execution standards perform the efficiency of real business environments.

4.5 Gaps in the diagnosis standards

4.5.1 Lack of established BPM diagnosis standards. There is a dearth of complete diagnosis standards. BPMI joined OMG in 2005, work on BPQL and BPRI appear to have stalled. Currently, BPMS suites provide limited diagnosis activities like monitoring, business intelligence and analytics like audit trail recordings, but a universal diagnosis standard is still needed.

4.5.2 Lack of reporting standards for BPM diagnosis standards. In the first author's experience evaluating BPM suites for his former employer, many BPMS offer inbuilt diagnosis tools that do not have many adequate reporting features. Companies have to rely on external reporting tools like Crystal Reports and Microsoft Reporting Services. A complete industry standard would be very beneficial BPM. The current process mining research (van der Aalst *et al.*, 2005a, b) can be applied to address the needs of diagnosis standards.

4.5.3 No real metrics for the degree of fulfillment of business goals. Present-day BPM standards and systems are unable to ascertain the extent of fulfillment of high-level business goals, key performance indices (KPIs) and targets in strategic business processes. Currently, one infers if a business goal has been realized only at the final end stage or the withdrawn stage of a business process. In the first author's experience, the end node of a business process does not necessarily imply that all the associated tasks have been accomplished.

4.6 Gaps in the B2B information exchange standards

As mentioned in Section 3.7, B2B standards are envisioned to be more closely integrated with BPM Standards to support the real needs of increased globalisation. The following sections discuss the current gaps that need to be first addressed.

4.6.1 Current standards only address information exchange. Collaborative business processes (a.k.a. B2B process integration) have recently attracted much interest from the BPM research community. Collaborative business processes fulfill the common business goals of the partners. However, current B2B information exchange standards are merely just that – information exchange – and have not yet addressed the higher-level collaboration semantics of true collaborative business processes.

4.6.2 Current standards cannot address dynamic B2B collaborations. Current B2B information exchange standards are predefined static specifications and so cannot accommodate collaborative business processes which are dynamically formed on-the-fly. In reality, the identification of suppliers or service providers is still mostly executed manually, even if B2B information exchange standards are available in the market.

Furthermore, the establishment of automated standardized B2B information exchange requires a costly setup prior to the actual usage. This means that before fully reaping the benefits and efficiency of the B2B information exchange standards, companies have to evaluate their existing legacy systems, IT infrastructure and licensing schemes prior to utilizing these standards. For example, in RosettaNet, the RosettaNet Implementation Framework must first be established before "business signals" and RosettaNet PIPs may be dispatched.

4.6.3 Context-less B2B collaborations. Currently, B2B information exchange standards do not embody contextual information. If some context can be embedded in web services supporting the underlying methods, a first move to context-aware business process systems will have been made.

4.6.4 Supplier/buyer qualification mechanisms/standards for BPM. To facilitate the dynamic formulation of collaborative business processes, an automatic buyer/supplier qualification procedure is necessary. In industry, the "trust-worthiness" of potential suppliers and buyers is pre-qualified before the B2B collaboration information exchange interfaces are developed. It is not difficult to automate the pre-qualification of non-subjective criteria such as fulfilment rates of deadlines, contract specifications, etc. Examples of current work into trust-based collaborative filtering may be found in Weng *et al.* (2006, 2005).

5. Current trends and research directions of BPM standards

5.1 Recent consolidation of BPM standards

In the early 2000s, there were many standards which eventually lost favour with practitioners (Pyke, 2007). Fortunately in recent years, these consolidated into essentially three key standards in the modeling, interchange and the execution of business processes (Khan, 2005;Pyke, 2007):

(1) *BPMN (*OMG, 2004a*)*. By the former BPMI, now a part of the OMG, the BPMN represents the high-level notation/graphical representation of business processes easily understood by business analysts, and especially useful in communicating business requirements.

(2) *BPEL (*Andrews *et al.*, 2003*; OASIS, 2007)*. By the OASIS, the execution level BPEL allows automation of business processes and makes use of the web services platform.

(3) *XPDL (WfMC, XML, 2002)*. By the WfMC, the XPDL functions as a file format and acts as a popular interchange language for the easy translation either:

- between different software using the BPMN notations without a loss of information integrity; or

- more importantly from the notational BPMN to executable BPEL.

5.2 BPM standards and the relevant BPM life cycle stage addressed

After the classification of BPM standards through our framework, a clear perspective of how each standard address the needs of the relevant stage of the BPM life cycle were revealed (Table III).

It can be seen that graphical standards like BPMN and UML AD are based on Petri nets, and address the process design stage of the BPM Life cycle.

From Table III, it is also shown that the Interchange standards do not have much academic or theoretical formalism as their basis. Furthermore, execution standards are mainly based on either Pi-calculus or Petri nets or both. Process mining research form the basis for BAM, business intelligence and business analytics for the diagnosis standards that address the diagnosis stage of the BPM life cycle. The theoretical basis of XPDL and BPDM are not obvious. To our best knowledge, there are currently no diagnosis standards that are completed and implemented in BPM systems.

BPM standards category	Current standards/ notations	Academic/theoretical background	Creator	BPM life cycle stage
Graphical	BPMN	Petri nets	BPMI/OMG	Process design
	UML AD	Petri nets	OMG	Process design
Interchange	XPDL	–	WfMC	Process enactment
	BPDM	–	OMG	–
Execution	BPEL	Pi-calculus and Petri nets	OASIS	Process enactment
	BPML	Pi-calculus	BPMI	Process enactment
	YAWL	Workflow patterns (Petri nets)	W. van der Aalst *et al.*	Process enactment
Diagnosis	BPQL	Process mining	BPMI/OMG	Diagnosis
	BPRI	–	OMG	System configuration/ diagnosis

Table III.
BPM standards
classification table

5.3 Emergence of natural language business process languages/SBVR: incorporation of business rules and business vocabularies into BPM technologies
While graphical standards are high level, natural language is the other form of high-level communication that allows us to model business processes. The advantage of natural languages like structured English sentences over graphical notations is the ability to be easily translated and "mechanized" in computers (i.e. execution standards in our context) (Figure 12).

Recently, in February 2008, the Semantics of Business Vocabulary and Business Rules – SBVR (OMG, 2008a, b, c, d) Version 1.0 specification is a formal logic with a natural language interface, and received final approval at the December 2007 OMG
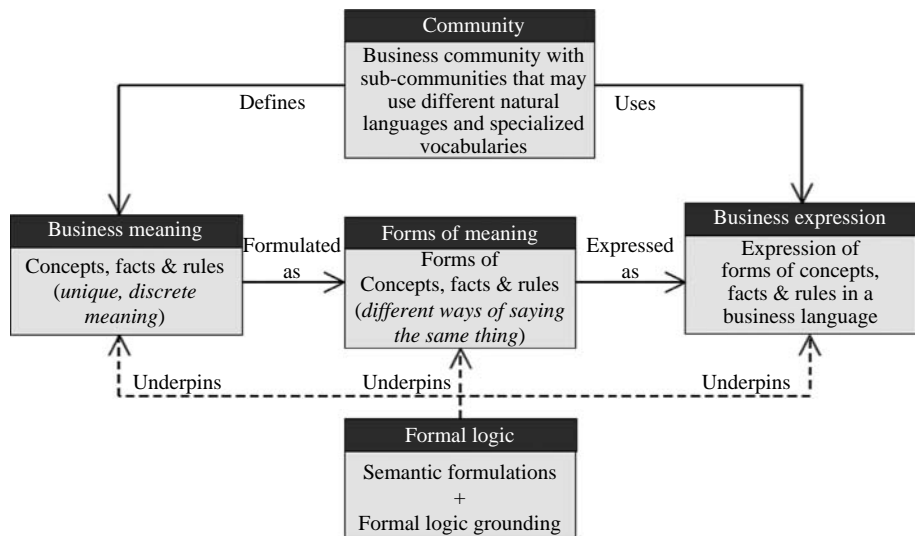
Figure 12.
Overview of SBVR

Source: www.omg.org/news/meetings/ThinkTank/past-events/2006/presentations/04-WS1-2_Hall.pdf

Meeting in Burlingame, California, and its specifications are now available on the OMG web site (OMG, 2008a, b, c,d).

The SBVR allows a "Business community" (e.g. a company) to define the concepts, facts, and rules of its daily business operations and business processes. Given these definitions, the Semantics of Business Vocabulary (i.e. taxonomy, thesaurus and ontology) and the Business Rules (i.e. operative business rules) can be created.

As shown in Figure 13, SBVR facilitates the expression of business rules in natural textual structure English expressions like "It is prohibited that a barred driver is a driver of a rental" in a fictitious car rental company (Figure 13). "Barred driver", "driver" and "rental" are identified as symbols in the Business Vocabulary of that company. As shown in Steps 3 and 4 of Figure 13, these symbols are parsed through language rules and facts of logical formulations (e.g. "It is prohibited [...]" or "It is obligatory [...]") are formed. Facts of logical formulations are represented as objects (object-oriented programming). These objects are eventually written as XML definitions.

SBVR is a landmark milestone for OMG and is heavily influenced by the Business Rules research (Bajec and Krisper, 2005; Hall, 2005), as it allows the non-graphical natural language to be used as a notation for its predominantly graphical standards (e.g. UML). The SBVR is dubbed to be used in other OMG standards like BPDM and knowledge discovery metamodel – KDM (OMG, 2008a, b, c, d).

5.4 Research trends for graphical standards – reference models

Reference models are also called universal models, generic models or model patterns. A process reference model represents dynamic aspects of an enterprise, e.g. activity
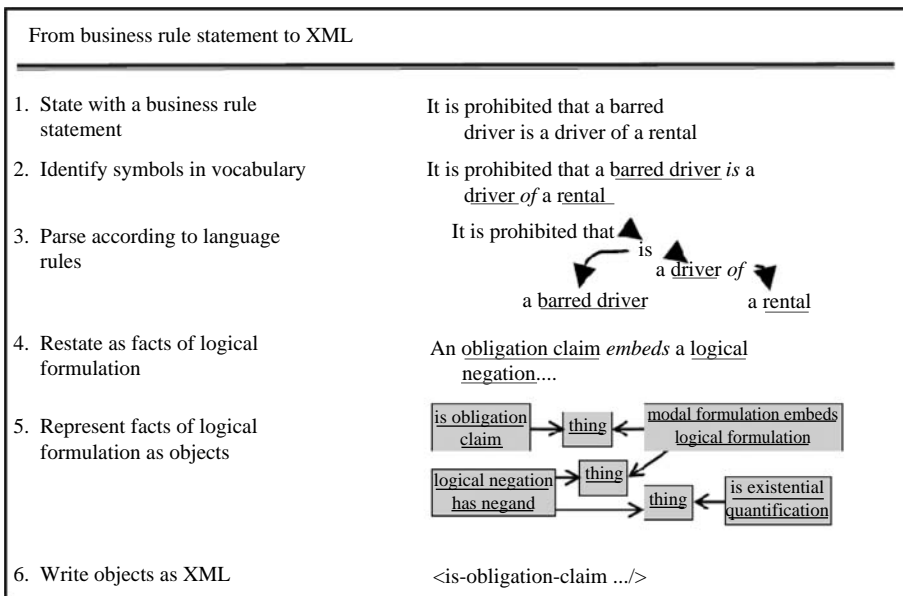

Figure 13.
SBVR facilitates
translation from English
statements to XML

Source: www.omg.org/news/meetings/ThinkTank/past-events/2006/presentations/04-WS1-2_Hall.pdf

sequences, organizational activities required to satisfy customer needs, control-flow between activities, particular dependency constraints, etc.

One example of a business process reference model is AMFIBIA, which is a formalism-independent meta-model that formalizes the essential aspects and concepts of business processes (Axenath *et al.*, 2007).

5.5 Research trends for interchange standards – metamodels

While XPDL is a well-adopted interchange standard, it is mainly useful for storing and interchanging BPML (in the past), BPMN and BPEL. XPDL is vulnerable to new graphical or execution standards that may surface, and is not adaptable to new versions or drastic extensions. In the end, XPDL will always be chasing the graphical and execution standards trend.

From the OMG side, there is the drive to use the model-driven architecture (Frankel, 2003) to tackle the interchangeability problem. This is done via defining "metamodels" with the meta-object framework in BPDM and pattern based integration of process driven SOA models (Zdun and Dustdar, 2007).

In our opinion, the usefulness of metamodels to address the problem of heterogeneity of graphical and execution standards depend on whether metamodels are:

- derived from existing standards; or
- foundations from which to develop future standards.

If metamodels are created from existing standards, then it is just a short-term solution that cannot accomodate new versions and changes to the existing standards. However, if metamodels are initially created as a basis for future standards to be built on them, metamodels can be powerful long-term solutions for the interchangeability problem.

5.6 Research trends for execution standards – semantic BPM

Researchers from the semantic web community have also identified that the modeling of higher level semantics of business processes are currently limited.

By hybridizing semantic web services and business process modeling, Koschmider and Oberweis (2008) and Hepp *et al.* (2005) highlighted the main semantic limitation of execution standards as the lack of machine accessible semantics, and argued that the modeling constructs of semantic web services frameworks are a natural solution to this.

5.7 Research trends for diagnosis standards

5.7.1 Process mining and process verification. Process mining aims to diagnose business processes by mining event logs to understand process, control, data, organizational and social structures. This allows analysts to compare the real behavior of an information system or its users with the intended or expected behavior (van der Aalst).

In recent years, the focus has shifted from mere process mining to verification of the processes mined. Earlier work on process mining primarily focused on process discovery, by automatically constructing models of knowledge extracted from event logs (van der Aalst *et al.*, 2005a, b). However, in view of recent focus on good corporate governance, there is a need to also spot and pre-empt unexpected behavior observed in the event logs.

5.7.2 Quality of service for business processes. The "quality of service (QoS)" in BPM is different from that of computer networking. According to Cardoso *et al.* (2004), the management of quality of service (QoS) metrics such as products or services to be delivered, deadlines, quality of products, and cost of services directly impacts the success of organizations participating in e-commerce.

When services or products are created or managed using workflows or web processes, the underlying workflow engine must accept the specifications and be able to estimate, monitor, and control the QoS rendered to customers.

Cardoso *et al.* presented a predictive QoS model that makes it possible to automatically compute the QoS for workflows based on atomic task QoS attributes.

5.7.3 Business analytics. From a business viewpoint, business analytics (Davenport and Harris, 2006) underscores the need for the diagnosis stage of the BPM life cycle. In their paper, Davenport and Harris observe that successful companies are often those which keep a history of their data (e.g. business process, customer relationship, customer preferences, etc.), analyze them and act on them. In the authors' view, diagnosis urgently needs diagnostic formalisms and methodologies, i.e. business analytics born out of process mining, process verification, and QoS metrics.

5.8 Research trends for B2B standards

The gaps mentioned in Sections 3.7 and 4.6 are recognised by researchers, and are currently addressed in the following projects and efforts.

5.8.1 CrossFlow approach. CrossFlow is a European research project for cross-organizational WfM in virtual enterprises (Grefen, 2000). Most active in the late 1990s and the early 2000s, the project's view on contract frameworks and virtual enterprises are still sound and worth revisiting.

5.8.2 SUPER research project: ontological framework for semantic BPM. The major objective of semantics utilized for process management within and between enterprises (SUPER) is to "raise Business Process Management (BPM) to the business level, where it belongs, from the IT level where it mostly resides now" (Hepp and Roman, 2007). This project addresses the semantic gaps of current web service-based business processes.

5.8.3 Context aware WfMS. Another school of thought leans heavily on context awareness in business processes. The main difference between context and semantics/ontological matching is the argument that ontology's of different companies can never be matched.

Saidani and Nurcan (2007) introduced a basic taxonomy of context which captures most common CRK and two structures for modeling and categorizing the context:

(1) the context tree (CT) depicting contextual characteristics; and

(2) adapted CT (ACT) applicable to a specific domain.

In our opinion, the ACT is not scalable and hence not practical to manage. Furthermore, its basis, the CT, needs to be validated with industry business processes.

Ardissono *et al.* (2007) proposed the Context Aware Workflow Execution Framework for the development of applications composing web service suppliers in context-sensitive workflows. With this framework, context-sensitive workflows can be executed in conventional workflow engines.

Tan *et al.* (2007) visualized B2B collaborations aided by a Context Aware Framework that categorizes contextual information into user (i.e. the company),

temporal and location. In their view, most B2B transactions are dominated by the exchange of business documents which are rich in contextual information in the form of user, temporal and location.

Not everyone is excited about embellishing BPM with context. According to Weigand (2006), "computing machines, which are purposely designed to process symbols independent of their context, have no hopes of becoming experts". His concern with the context-less fundamentals of computers is understandable.

In the authors' view, the future of context awareness very much depends on a strong theoretical breakthrough in the underlying enabling infrastructure and not just extensions of current architectures. History has shown that is advances have been marked by a progress to what is convenient and natural to use (especially in computer science). For instance, programming languages have evolved from structural to object-oriented programming and to the present day web-based programming (i.e. assembly languages to FORTRAN to C to Java and.NET).

6. Related work

This review paper aims to complement the following high-impact ground-work on BPM concepts, in particular clarifying the current BPM standards classification and analysis:

- Medina-Mora *et al.* 's (1992) fundamental take on business process vs information processes vs physical processes.
- Georgakopolous *et al.*'s review on WfM and WfMS in Georgakopoulos *et al.* (1995).
- van der Aalst's reviews on definitions and perspectives of BPM concepts in van der Aalst (2004a, b, c) and van der Aalst *et al.* (2003).
- Koskela and Haajanen's (2007) and Recker and Mendling's (2006) identification of conceptual mismatches between BPEL and BPMN.

This review also hopes to address the common misconceptions in BPM standards classifications; for example the classification of "collaborative" business processes (Roser and Bauer, 2005). In their paper, the programming platform J2EE, the B2B information exchange standard ebXML BPSS and the methodology ARIS were classified as "modeling languages and approaches" without clarifying their fundamental differences and their place in the BPM life cycle. The authors also failed to distinguish between BPM systems and standards. For instance, ARIS is a methodology used in IDS Scheer BPM suites and not a standard.

Another example on business process modeling was unclear about the role of modeling in the BPM life cycle (Wang *et al.*, 2006). Also, Petri nets (theory) and IDEF0 (B2B information exchange standard) were superficially treated together as "BPM modeling standards" in their publication.

7. Concluding remarks

In this paper, a method to categorise current BPM standards into graphical, interchange, execution and diagnosis standards is proposed. This categorisation facilitates the:

- *Discrimination of BPM standards, thereby resolving common misconceptions.* Graphical Standards are currently the highest level of expression of business

processes (and the most natural to human beings) while at the lowest (i.e. technical) level are execution standards. Interchange standards currently translate graphical standards to execution standards so business process models in different BPMS are interchangeable. However, the fundamental differences between graphical and execution standards severely limit any translation.

- *Standardization of the diagnosis stage of the BPM life cycle.* Although diagnosis standards govern the management and optimisation of business processes as envisioned in BPM, they are the most under-developed of all standards. Two notable but uncompleted diagnosis standards are BPRI and BPQL. The current research into process mining and business analytics can perhaps drive the formalization of diagnosis standards.

- *Identification of current strengths and limitations of each standard.* Graphical standards are easily interpreted by business analysts but lack computational formalisms. Execution standards enable business process automation (i.e. the process enactment stage of the BPM life cycle). However, they are rather limited in expressing loops and cycles commonly found in real-life business processes. BPEL is beginning to embody many of the capabilities of BPML; its second version supports human involvements. Interchange standards translate graphical to execution standards and vice versa. Prominent interchange standards are XPDL and BPDM. Currently, XPDL is more widely used as BPDM is still a fledgling standard with not many software adaptations.

  In summary, graphical and execution standards lack computational formalisms whereas interchange standards should be scalable and flexible to accommodate new standards and versions. Lastly, diagnosis standards are lacking in metrics for diagnosis and reporting standards for the management of business processes. The industry is currently consolidating towards BPMN as the graphical standard, XPDL as an interchange standard, and BPEL as an execution standard. These three standards address the process design and process enactment stages of the BPM life cycle.

- *Evaluation of BPM standards.* The most prevalent method to evaluate standards is the Workflow Patterns Framework, which is a collection of generic, recurring constructs originally devised to evaluate workflow systems. However, it is unable to assess high-level business goals, and QoS (i.e. the business process' actual achievement of promised business services) of the business processes.

- *Understanding gaps and future trends.* Many trends were revealed, and most notably, the emergence of natural language standards like SBVR can potentially revolutionize the way high-level business processes are formulated and translated into low-level execution code via business rules and fact identifications. In the authors' opinion, natural language standards are at the same level as graphical standards, as shown in Figure 14.

- Another trend concerns the growing need for B2B collaboration. With increased globalisation aided by technology, the authors foresee the eventual conceptual integration of BPM and B2B standards. At present, B2B standards only address information exchange but not dynamic, on-the-fly communications. This can potentially be addressed by trust-based buyer-supplier qualifications and context-aware capabilities.
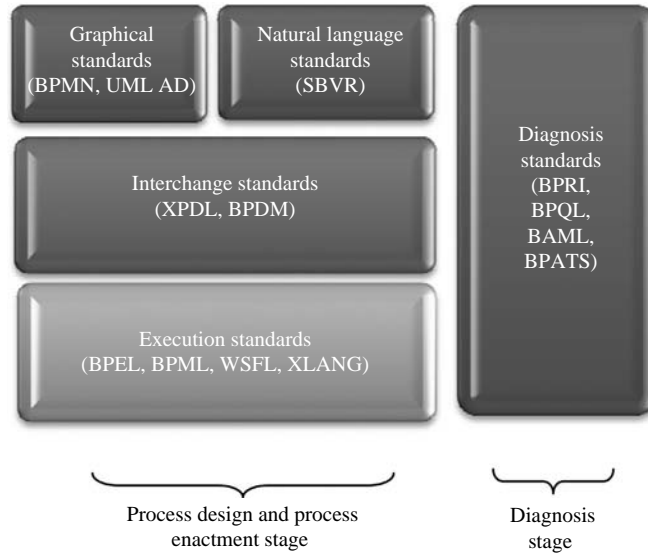
Figure 14.
Future BPM categories

The taxonomy in our proposed categorisation of BPM languages, standards and theory clarifies BPM terminologies and reinforces a universal understanding of standards. However, to move both internal and collaborative BPM research and standardisation forward, there are important standardisation lessons to be learnt from internet standardisation (Nickerson and zur Muehlen, 2006) and DBMS (van der Aalst, 2003a, b).

References

Ambler, S.W. (2004), *The Object Primer: Agile Model Driven Development with UML 2*, Cambridge University Press, Cambridge.

Andersson, B., Bider, I., Johannesson, P. and Perjons, E. (2005), "Towards a formal definition of goal-oriented business process patterns", *Business Process Management Journal*, Vol. 11 No. 6, pp. 650-62.

Andrews, T., Curbera, F., Dholakia, H., Goland, Y., Klein, J., Leymann, F., Liu, K., Roller, D., Smith, D. and Thatte, S. (2003), *Business Process Execution Language for Web Services, Version 1.1. Specification, BEA Systems*, IBM Corp., Microsoft Corp., SAP AG, Siebel Systems, New York, NY.

ANSI (1979), "X12, electronic data interchange X12 standards", Draft Version, ANSI, Washington, DC.

Ardissono, L., Furnari, R., Goy, A., Petrone, G. and Segnan, M. (2007), "A framework for the management of context-aware workflow systems", *Proceedings of the 3rd International Conference on Web Information Systems and Technologies* (*WEBIST 2007*), pp. 1-9.

Arkin, A. (2000), *BPML 0.4 – Assaf Arkin Discusses Relationship between BPML and Pi Calculus*, Intalio, Palo Alto, CA.

Arkin, A. (2002), *Business Process Modeling Language* (*BPML*), BPMI, New York, NY.

Axenath, B., Kindler, E. and Rubin, V. (2007), "AMFIBIA: a meta-model for integrating business process modelling aspects", *International Journal of Business Process Integration and Management*, Vol. 2 No. 2, pp. 120-31.

Bajec, M. and Krisper, M. (2005), "A methodology and tool support for managing business rules in organisations", *Information Systems*, Vol. 30 No. 6, pp. 423-43.

BEA (2008), "BEA AquaLogic BPM suite", available at: www.bea.com/bpm (accessed 20 February 2008).

Beeri, C., Eyal, A., Kamenkovich, S. and Milo, T. (2005), "Querying business processes with BP-QL", *Proceedings of the 31st International Conference on Very Large Data Bases*, pp. 1255-8.

Beeri, C., Eyal, A., Milo, T. and Pilberg, A. (2007), "Monitoring business processes with queries", *Proceedings of the 33rd International Conference on Very Large Data Bases 2007* (*VLDB 2007*)*, Vienna, Austria*, pp. 603-14.

Bell, A.E. (2004), "Death by UML fever: self-diagnosis and early treatment are crucial in the fight against UML fever", *ACM Queue*, Vol. 2 No. 1, pp. 72-80.

Cardoso, J., Sheth, A., Miller, J., Arnold, J. and Kochut, K. (2004), "Quality of service for workflows and web service processes", *Web Semantics: Science, Services and Agents on the World Wide Web*, Vol. 1 No. 3, pp. 281-308.

CrystalReports.com (2008), *Crystal Reports*, available at: www.crystalreports.com (accessed 2 March 2008).

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

Davenport, T.H. and Harris, J.G. (2006), *Competing on Analytics*, Harvard Business School Press, Boston, MA.

Dumas, M. and ter Hofstede, A.H.M. (2001), "UML activity diagrams as a workflow specification language", *UML 2001: 4th International Conference on the Unified Modeling Language: Modeling Languages, Concepts, and Tools, Toronto*, pp. 76-90.

Fielding, R. (1995), "RFC1808: relative uniform resource locators", *Internet RFCs*, June.

Frank, J.H., Gardner, T.A., Johnston, S.K., White, S.A. and Iyengar, S. (2004), "Business processes definition metamodel concepts and overview", paper presented at the IBM's Proposal in Response to the OMG's RFP for a Business Process Definition Metamodel.

Frankel, D. (2003), *Model Driven Architecture: Applying MDA to Enterprise Computing*, Wiley, New York, NY.

Georgakopoulos, D., Hornick, M. and Sheth, A. (1995), "An overview of workflow management: from process modeling to workflow automation infrastructure", *Distributed and Parallel Databases*, Vol. 3 No. 2, pp. 119-53.

Ghalimi, I. and McGoveran, D. (2005), "Standards and BPM", *BPM.Com*, available at: www.bpm.com/FeatureRO.asp?FeatureId=145

Gibb, B.K. and Damodaran, S. (2002), *ebXML: Concepts and Application*, Wiley, New York, NY.

Grefen, P. (2000), *CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises*, Centre for Telematics and Information Technology, University of Twente Host, University of Twente, Enschede.

Hall, J. (2005), "Business rules boot camp (tutorial)", *Proceedings of the 6th European Business Rules Conference* (*EBRC 2007*), *Dusseldorf*, pp. 1-67.

Hammer, M. and Champy, J. (1992), "What is reengineering?", *InformationWeek*, Vol. 372, pp. 10, 14, 18, 20, 24.

Hammer, M. and Champy, J. (1993), *Reengineering the Corporation: A Manifesto for Business Revolution*, HarperBusiness, New York, NY.

Harel, D. (1987), "Statecharts: a visual formalism for complex systems", *Science of Computer Programming*, Vol. 8 No. 3, pp. 231-74.

Harmon, P. (2004), "Business process notation", *Business Process Trends*, Vol. 2 No. 6, available at: www.bptrends.com

Havey, M. (2005), *Essential Business Process Modeling*, 1st ed., O'Reilly Media, Sebastopol, CA.

Hepp, M. and Roman, D. (2007), "An ontology framework for semantic business process management", *Proceedings of Wirtschaftsinformatik 2007, Karlsruhe, Germany*, pp. 1-18.

Hepp, M., Leymann, F., Domingue, J., Wahler, A. and Fensel, D. (2005), "Semantic business process management: a vision towards using semantic web services for business process management", *Proceedings of the IEEE ICEBE*, pp. 18-20.

Hill, J.B., Kerremans, M. and Bell, T. (2007a), *Cool Vendors in Business Process Management, 2007*, Gartner Research, Stamford, CT.

Hill, J.B., Pezzini, M. and Natis, Y.V. (2008), "Findings: confusion remains regarding BPM terminologies", Vol. ID No. G00155817, Gartner Research, Stamford, CT.

Hill, J.B., Cantara, M., Deitert, E. and Kerremans, M. (2007b), "Magic quadrant for business process management suites", Vol. ID No. G001252906, Gartner Research, Stamford, CT.

Hill, J.B., Sinur, J., Flint, D. and Melenovsky, M.J. (2006), "Gartner's position on business process management, 2006", *Business Issues*, Gartner, Stamford, CT.

Intalio (1999-2007), *Intalio Designer*, available at: www.intalio.com/products/designer/ (accessed 2 February 2008).

Intalio (2008), *Intalio BPM Suite*, available at: www.intalio.com/ (accessed 20 February 2008).

KAISHA-Tec Co. (2008), *ActiveModeler Advantage*, available at: www.activemodeler.com/ (accessed 2 February 2008).

Kavantzas, N., Olsson, G., Mischkinsky, J. and Chapman, M. (2004), "Web service choreography description language (WSCDL) 1.0", W3C Working Draft, Biot.

Khan, R.M. (2005), "What standards really matter for BPM?", Business Process Trends, Newton, MA.

Kindler, E. (2004), "On the semantics of EPCs: a framework for resolving the vicious circle", paper presented at Business Process Management: 2nd International Conference, BPM 2004, Potsdam.

Koenig, J. (2004), "JBoss jBPM", White Paper, JBoss Labs, Atlanta, GA.

Koschmider, A. and Oberweis, A. (2008), "Modeling semantic business process models", in Rittgen, P. (Ed.), *Handbook of Ontologies for Business Interaction*, Idea Group, Harpenden.

Koskela, M. and Haajanen, J. (2007), "Business process modeling and execution: tools and technologies report for the SOAMeS project", VTT Research Notes No. 2407, VTT Technical Research Centre of Finland, Espoo.

Koubarakis, M. and Plexousakis, D. (1999), "Business process modelling and design – a formal model and methodology", *BT Technology Journal*, Vol. 17 No. 4, pp. 23-35.

Leymann, F. (2001), "WSFL – Web Services Flow Language", White Paper, IBM Software Group, Armonk, NY.

Lowe-Norris, A.G. and Denn, R. (2000), *Windows 2000 Active Directory*, O'Reilly & Associates, Sebastopol, CA.

Marin, M. (2001), *Workflow Process Definition Interface – XML Process Definition Language*, The Workflow Management Coalition Specification, Burnsville, MN, pp. 15-31.

Meadows, B. and Seaburg, L. (2004), Universal Business Language 1.0. Organization for the Advancement of Structured Information Standards (OASIS), Committee Draft, Rotterdam.

Medina-Mora, R., Winograd, T., Flores, R. and Flores, F. (1992), "The action workflow approach to workflow management technology", *Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work*, pp. 281-8.

Mendling, J. and Neumann, G. (2005), "A comparison of XML interchange formats for business process modeling", *Workflow Handbook*, Future Strategies, Lighthouse Point, FL.

Mertz, D. (2001), "Understanding ebXML", available at: www.ibm.com/developerworks/xml/library/x-ebxml/ (accessed 28 February 2008).

Metastorm (2007), *Metastorm Inc. Corporate Website*, available at: www.metastorm.com (accessed 20 February 2008).

Milner, R.R. (1982), *A Calculus of Communicating Systems*, Springer, New York, NY.

Milner, R.R. (1999), *Communicating and Mobile Systems: The Pi Calculus*, Cambridge University Press, Cambridge.

Mous, K., Ko, K.L., Lee, S.G., Tan, P.S. and Lee, E.W. (2007), "High-level business processes for agile B2B collaboration", *Proceedings of the 35th International MATADOR Conference, Taipei*, Springer, London.

Nickerson, J.V. and zur Muehlen, M. (2006), "The ecology of standards processes: lessons from internet standardization", *MIS Quarterly*, Vol. 30 No. 5, pp. 467-88.

OASIS (2007), Web Services Business Process Execution Language (WSBPEL), available at: www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel (accessed 14 September 2007).

OASIS and UN/CEFACT (2001a), *Business Process Specification Schema v1.01*, OASIS and UN/CEFACT, Boston, MA.

OASIS and UN/CEFACT (2001b), *ebXML Technical Architecture Specification v1.04*, OASIS and UN/CEFACT, Boston, MA.

OMG (2002), *RFP: Business Process Runtime Interfaces* (*BPRI*), *Platform Independent Model* (*PIM*), OMG, Needham, MA.

OMG (2004a), *Business Process Modeling Notation* (*BPMN*), OMG, Needham, MA, available at: www.bpmn.org/ (accessed 14 September 2007).

OMG (2004b), *UML 2.0 Superstructure Specification*, OMG, Needham, MA, available at: www.omg.org/cgi-bin/doc?ptc/2004-10-02 (accessed 10 February 2008).

OMG (2007), *Business Process Modeling Notation* (*BPMN*), OMG, Needham, MA, available at: www.bpmn.org/ (accessed 14 September 2007).

OMG (2008a), *List of BPMN Implementors*, OMG, Needham, MA, available at: www.bpmn.org/ (accessed 15 February 2008).

OMG (2008b), *Object Management Group* (*OMG*), OMG, Needham, MA, available at: www.omg.org/ (accessed 20 February 2008).

OMG (2008c), *OMG Business Modeling Specifications*, OMG, Needham, MA, available at: www.omg.org/ (accessed 20 February 2008).

OMG (2008d), *Semantics of Business Vocabulary and Business Rules* (*SBVR*), OMG, Needham, MA, available at: www.omg.org/docs/formal/08-01-02.pdf (accessed 3 March 2008).

Oracle (2008a), "Oracle", available at: www.oracle.com/index.html (accessed 20 February 2008).

Oracle (2008b), "Oracle Business Process Management", available at: www.oracle.com/technologies/bpm/index.html (accessed 20 February 2008).

Petri, C.A. (1962), "Kommunikation mit Automaten", PhD thesis, Rheinisch-Westfälisches Institut f. instrumentelle Mathematik and University, Essen.

Pyke, J. (2007), "XPDL – the silent workhorse of BPM", *BPM.Com*, available at: www.bpm.com/FeatureRO.asp?FeatureID=232

Recker, J.C. (2007), "A study on the decision to continue using a modeling grammar", paper presented at the 13th Americas Conference on Information Systems, Keystone, CO.

Recker, J.C. and Mendling, J. (2006), "On the translation between BPMN and BPEL: conceptual mismatch between process modeling languages", in Latour, T. and Petit, M. (Eds), *Proceedings of the 18th International Conference on Advanced Information Systems Engineering, Luxembourg*, pp. 521-32.

Roser, S. and Bauer, B. (2005), "A categorization of collaborative business process modeling techniques", *Proceedings of the 7th IEEE International Conference on e-Commerce Technology Workshops*, pp. 43-54.

RosettaNet (1998), *RosettaNet Implementation Framework: Core Specification*, RosettaNet, Lawrenceville, NJ.

RosettaNet Program Office (2007), *Overview – Cluster, Segments, and PIPs* (*Version 02.10.00*), RosettaNet Program Office, Lawrenceville, NJ.

Russell, N., van der Aalst, W.M.P., ter Hofstede, A.H.M. and Wohed, P. (2006), "On the suitability of UML 2.0 activity diagrams for business process modeling", *Proceedings of the 3rd Asia-Pacific Conference on Conceptual Modelling*, Vol. 53, pp. 95-104.

Saidani, O. and Nurcan, S. (2007), "Towards context aware business process modeling", *8th Workshop on Business Process Modelling, Development, and Support (BPMDS'07 in Association with CAISE'07), Trondheim, Norway, 11-12 June,* Springer.

SAP (2008a), "SAP AG", available at: www.sap.com/ (accessed 3 March 2008).

SAP (2008b), "SAP business workflow", available at: www.sdn.sap.com/irj/sdn/nw-bpm?rid=/webcontent/uuid/10ff0453-ae33-2a10-7984-9d8df609d8f9 (accessed 3 March 2008).

Savvion (2008), "Savvion BPM suite", available at: www.savvion.com/ (accessed 20 February 2008).

Scheer, I.D.S. (1992), *ARIS* (*Architecture of Integrated Information Systems*), Springer, New York, NY.

Shapiro, R. (2002), "A comparison of XPDL, BPML, and BPEL4WS", OASIS XML Cover Pages, Rotterdam.

Simchi-Levi, D., Kaminsky, P. and Simchi-Levi, E. (2000), *Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies*, McGraw-Hill/Irwin, New York, NY.

Smith, H. (2004), "Enough is enough in the field of BPM: we don't need BPELJ: BPML semantics are just fine", unpublished paper, BPM3, available at: www.bpm3.com/bpelj/BPELJ-Enough-Is-Enough.pdf

Tan, P.S., Goh, A.E.S., Lee, S.S.G. and Lee, E.W. (2006), "Issues and approaches to dynamic, service-oriented multi-enterprise collaboration", *Proceedings of the 2006 IEEE International Conference on Industrial Informatics*, pp. 399-404.

Tan, P.S., Lee, S.S.G., Goh, A.E.S. and Lee, E.W. (2007), "Context-enabled B2B collaborations", *Proceedings of the IEEE International Conference on Services Computing, 2007* (*SCC 2007*), pp. 240-3.

Thatte, S. (2001), *XLANG: Web Services for Business Process Design*, Microsoft Corporation, Redmond, WA.

UN/CEFACT and OASIS (2002), *ebXML, XML BPSS Version 1.01 Technical Specification*, United Nations CEFACT/OASIS, Toyako, Geneva/Rotterdam.

UN/EDIFACT (1990), United Nations Directories for Electronic Data Interchange for Administration Commerce and Transport, available at: www.unece.org/trade/untdid/texts/d100_d.htm (accessed 14 September 2007).

van der Aalst, W.M.P. (1999), "Formalization and verification of event-driven process chains", *Information and Software Technology*, Vol. 41 No. 10, pp. 639-50.

van der Aalst, W.M.P. (2003a), "Don't go with the flow: web services composition standards exposed", *IEEE Intelligent Systems*, Vol. 18 No. 1, pp. 72-6.

van der Aalst, W.M.P. (2003b), "Patterns and XPDL: a critical evaluation of the XML process definition language", QUT Technical Report, FIT-TR-2003-06, Queensland University of Technology, Brisbane.

van der Aalst, W.M.P. (2004a), "Business alignment: using process mining as a tool for Delta analysis", *Proceedings of the 5th Workshop on Business Process Modeling, Development and Support* (*BPMDS'04*), Vol. 2, pp. 138-45.

van der Aalst, W.M.P. (2004b), "Business process management: a personal view", *Business Process Management Journal*, Vol. 10 No. 2, p. 5.

van der Aalst, W.M.P. (2004c), "Business process management demystified: a tutorial on models, systems and standards for workflow management", *Lecture Notes in Computer Science, Lectures on Concurrency and Petri Nets*, Vol. 3098, pp. 1-65.

van der Aalst, W.M.P. and ter Hofstede, A.H.M. (2005), "YAWL: yet another workflow language", *Information Systems*, Vol. 30 No. 4, pp. 245-75.

van der Aalst, W.M.P., de Beer, H.T. and van Dongen, B.F. (2005a), *Process Mining and Verification of Properties: An Approach Based on Temporal Logic*, Springer, New York, NY.

van der Aalst, W.M.P., ter Hofstede, A.H.M. and Weske, M. (2003), "Business process management: a survey", *Proceedings of the International Conference on Business Process Management, BPM 2003, Eindhoven, The Netherlands, 26-27 June*.

van der Aalst, W.M.P., Aldred, L., Dumas, M. and ter Hofstede, A.H.M. (2004), "Design and implementation of the YAWL system", *Proceedings of the 16th International Conference on Advanced Information Systems Engineering, CAiSE 2004, Riga, Latvia, 7-11 June*.

van der Aalst, W.M.P., Barros, A., ter Hofstede, A. and Kiepuszewski, B. (2000), "Advanced workflow patterns", *Proceedings of the 7th International Conference on Cooperative Information Systems* (*CoopIS 2000*), Vol. 1901, pp. 18-29.

van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M. and Wohed, P. (2002a), "Pattern-based analysis of BPML (and WSCI)", QUT Technical Report FIT-TR-2002-05, Queensland University of Technology, Brisbane.

van der Aalst, W.M.P., ter Hofstede, A.H.M., Kiepuszewski, B. and Barros, A.P. (2002b), "Workflow patterns", QUT Technical Report, FIT-TR-2002-02, Queensland University of Technology, Brisbane.

van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M., Russell, N., Verbeek, H.M.W. and Wohed, P. (2005b), "Life after BPEL?", *Lecture Notes in Computer Science*, Vol. 3670, pp. 35-50.

Wang, W., Ding, H., Dong, J. and Ren, C. (2006), "A comparison of business process modeling methods", *Proceedings of the IEEE International Conference on Service Operations and Logistics, and Informatics, SOLI'06*, pp. 1136-41.

Washington DC Government Printing Office (2002), Public Law No. 107-204 (Sarbanes-Oxley Act), Washington, DC.

WebSphere (2005), "IBM WebSphere Software", available at: www.306.ibm.com/software/info/bpmsoa/index.html

Weigand, H. (2006), "Two decades of the language action perspective", *Communications of the ACM*, Vol. 49 No. 5, pp. 45-6.

Weng, J., Miao, C. and Goh, A. (2006), "Improving collaborative filtering with trust-based metrics", paper presented at the 21st Annual ACM Symposium on Applied Computing (SAC), Dijon.

Weng, J., Miao, C., Goh, A. and Li, D. (2005), "Trust-based collaborative filtering", *ACM 14th Conference on Information and Knowledge Management (CIKM 2005)*, pp. 299-300.

WfMC (1995), The Workflow Management Coalition (WfMC), available at: www.wfmc.org/ (accessed 20 January 2008).

WfMC (1999), "Workflow management coalition workflow standard: interface 1 – process definition Interchange Process Model (WFMC-TC-1016)", Technical Report, Workflow Management Coalition, Lighthouse Point, FL.

WfMC, XML (2002), *Process Definition Language-XPDL 1.0*, Workflow Management Coalition, Lighthouse Point, FL, December.

White, S. (2004a), *Introduction to BPMN*, BPMI, New York, NY.

White, S. (2004b), "Process modeling notations and workflow patterns", *Workflow Handbook*, Future Strategies, Lighthouse Point, FL, pp. 265-94.

Winograd, T. and Flores, F. (1986), *Understanding Computers and Cognition: A New Foundation for Design*, Ablex, Norwood, NJ.

Wohed, P. (2004), *Pattern-based Analysis of UML Activity Diagrams*, Beta, Research School for Operations Management and Logistics, Eindhoven.

Wohed, P., van der Aalst, W.M.P., Dumas, M. and ter Hofstede, A.H.M. (2003a), "Analysis of web services composition languages: the case of BPEL4WS", *Conceptual Modeling-Er 2003: Proceedings of the 22nd International Conference on Conceptual Modeling, Chicago, IL, USA, 13-16 October*.

Wohed, P., van der Aalst, W.M.P., Dumas, M. and ter Hofstede, A.H.M. (2003b), *Pattern-based Analysis of BPEL4WS*, Department of Computer and Systems Sciences, The Royal Institute of Technology, Stockholm University, Stockholm.

Wohed, P., van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M. and Russell, N. (2006), *On the Suitability of BPMN for Business Process Modelling*, Business Process Management, Vienna, pp. 161-76.

Wombacher, A., Fankhauser, P., Mahleko, B. and Neuhold, E. (2003), "Matchmaking for business processes", *Proceedings of the IEEE Comput. Soc., Newport Beach, CA, USA*.

Woodley, T. and Gagnon, S. (2005), "BPM and SOA: synergies and challenges", in Ngu, A.H.N. (Ed.), *Proceedings of the 6th International Conference on Web Information Systems Engineering (WISE), New York, NY, USA, 20-25 November*, pp. 679-88.

Woods, D. and Word, J. (2004), *SAP NetWeaver for Dummies*, Wiley, Hoboken, NJ.

XPDL (2008), "XML process description language", available at: www.wfmc.org/standards/xpdl. htm (accessed 20 January 2008).

YAWL (2007), *Yet Another Workflow Language (YAWL) Homepage*, available at: www.yawl-system.com/ (accessed 2 February 2008).

Zdun, U. and Dustdar, S. (2007), "Model-driven and pattern-based integration of process-driven SOA models", *International Journal of Business Process Integration and Management*, Vol. 2 No. 2, pp. 109-19.

zur Muehlen, M. (2007), "Tutorial – business process management standards", *Proceedings of the 5th International Conference on Business Process Management* (*BPM 2007*), Brisbane, p. 13.

About the authors

Ryan K.L. Ko is a PhD candidate in the Manufacturing Engineering Division in the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. Prior to his current research, he was a Systems Engineer from a Singapore MNC specializing in SAP Enterprise Portal, NetWeaver, ERP and BPM systems. In the course of his previous employment, he designed, implemented and managed more than 50 business processes for his manufacturing plant. He holds a Bachelor of Engineering (Computer Engineering) (Hons) from the School of Computer Engineering, Nanyang Technological University. His research interests are business management, dynamic enterprise collaboration and natural language processing. Ryan K.L. Ko is the corresponding author and can be contacted at: l060044@ntu.edu.sg

Stephen S.G. Lee is an Associate Professor and Head of the Manufacturing Engineering Division in the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. He holds a Master's degree from the University of Manchester, UK, and a PhD from Nanyang Technological University, Singapore. His research interests are in design methodology, knowledge-based design and manufacturing, product life cycle management and dynamic enterprise collaboration. He is a registered professional engineer and a Fellow of the Society of Manufacturing Engineers. In 1992, the Society of Manufacturing Engineers conferred on him its Award of Merit and in 1998, he was elected to its College of Fellows.

Eng Wah Lee is a Senior Scientist at Singapore Institute of Manufacturing Technology and is currently heading the Planning and Operations Management Group. His current research interests have extended from computer-integrated manufacturing to manufacturing enterprise integration, web service-enabled supply chain business process integration and secure EPC-global network for supply chain management, and broadly involved in a number of technological areas like XML, interoperability and integration, web services for information exchange. He is an active member of OASIS, and set up the Information Exchange Technical Committee for IT Standards Committee, Singapore; looking into XML, Unicode, EDI and Product Information Interchange Standards Development in October 2001. His more recent publication is in information exchange, web services approach in business process integration and manufacturing enterprise integration.