

# **Aligning Service Oriented Architecture and Business Process Management Systems to Achieve Business Agility**

**Haitham Abdel Monem El-Ghareeb**

Department of Information Systems  
Faculty of Computers and Information Sciences  
Mansoura University, EGYPT  
helghareeb@mans.edu.eg

**Abstract:** Business agility is a new paradigm that is a solution for maintaining enterprise competitive advantage. Business improvement approaches, such as Total Quality Management, Business Process Reengineering, and Workflow Management Systems attempted to satisfy business agility concepts and requirements, but suffered from fatal limitations like lack of concepts definition and measurement, lack of practical implementation, and failure to support ongoing change. The three business improvement approaches realized the need for Business Process Management (BPM) that is the key to business agility. Business Process Management Systems (BPMS) are information systems that enable implementation of BPM. Current IT infrastructure and information systems architecture do not satisfy BPMS objectives. Service Oriented Architecture (SOA) as a design pattern addresses technical agility that satisfies BPMS objectives in order to achieve business agility. This paper presents a coupling model of Business Process Management System and Service Oriented Architecture in order to satisfy process and technical agility aspects of business agility. Proposed model utilizes standards available for mapping BPM concepts via BPMS into SOA, and consists of three layers: Business, Business Services, and Application Services. Business layer enables business executives to handle business processes as information, BPMS resides in this layer. Business services layer is the layer that maps BPM concepts and requirements addressed by BPMS into SOA based IT infrastructure and enterprise information systems. Application Services layer holds the core services ready to be consumed by different BPMSs and shared among enterprises.

**Keywords:** Business Agility, Business Process Management, Business Process Management System, and Service Oriented Architecture.

## **1. Introduction**

Business Agility is a relatively new paradigm painted as a solution for maintaining competitive advantage during times of uncertainty and turbulence in the business environment [1]. Business improvement approaches, such as Total Quality Management (TQM), Business Process Reengineering (BPR), and Workflow Management Systems (WfMS) failed to satisfy business agility requirements due to different reasons. Three approaches realized the need for BPM [2]. Business process is a series of inter-related activities that cross functional enterprise boundaries with individual inputs and outputs [3,4]. BPM is a systematic, structured approach to analyze, improve, control, and manage processes [2]. BPMS is an enterprise information system that supports designing, administrating, and improving the business processes [5]. Enterprise information systems suffer from lack of agility and inefficiency [6]. Lack of agility refers to enterprises inability to match business requirements, especially new ones, onto current IT

infrastructure and information systems architecture. Inefficiency refers to the information systems development process that costs too much when compared to the actual output. Service Oriented Architecture (SOA) as a design pattern presents IT infrastructure and information systems architecture as loosely coupled, fine granular services that can address business requirements once they are presented either by adding new services or modifying existing ones. SOA also addresses enterprises information systems' inefficiency by enhancing reusability, thus theoretically, shortening information systems development time and effort required. Two main approaches for designing and implementing SOA based information systems are software agent based SOA, and Web services based SOA. Both approaches are presented in this paper.

This paper goes as follows: section two presents business agility principles, and framework. Section three presents business process aspect of business agility, and the need for BPMS. Section four presents SOA as the solution to technical agility as one of business agility aspects. Section five presents the proposed model aligning BPMS and SOA highlighting potential enterprises' benefits and advantages of this align regarding business agility.

## **2. Business Agility**

Business agility is a concept that extends adaptability and flexibility to include speed and scalability [7]. Adaptability is the opportunity for enterprise information system adaptation to different specified environments without applying other actions or means than provided for this purpose. Flexibility is the ability of an enterprise information system to respond to potential internal or external changes affecting its value delivery, in a timely and cost effective manner [8]. Scalability refers to the ease with which an enterprise information system can be modified to fit the problem area. Anarchic scalability defines the need for architectural elements to continue operating when they are subjected to an anticipated load, or when given malformed or maliciously constructed data, since they may be communicating with elements outside enterprise's control [9]. Business Agility is the term applied to an organization's ability to react to unanticipated market change, defining informed rapid transformation of organizational process and product [10]. Business Agility is the ability to sense change and opportunity in the marketplace, respond quickly, and execute successfully. Business agility is the sustained ability to sense and respond to change after change, executing well, occasionally inflicting change on competitors, yielding market leading returns [11]. Agile enterprise is the enterprise capable of responding quickly to changes [12].

Business agility is a promising strategy in times of change and uncertainty [7]. Enterprises must increasingly be organized, managed, and executed in ways that allow them to effectively sense and respond to unpredictable events in their environment. Enterprises must demonstrate agility in order to stay competitive [13]. Business change is inevitable, and whatever organizational framework is developed, it needs to be flexible to change [14,15].

### **2.1 Business Agility Principles**

Business agility principles can be extracted from different approaches of adaptive organizations [16]. Adaptive organizations are organizations that get affected by the social characteristics of its participants as well as the varied pressures imposed by its environment [17]. There are many

approaches for adaptive organizations, like Haeckel's approach. Business agility principles include [16,18]:

1. Agile enterprise entities must address four main agility characteristics:
  - Cooperate to enhance competitiveness.
  - Enrich the customer.
  - Master change and uncertainty.
  - Leverage the impact of people and information.
2. Traditional enterprise information systems that work well in stable predictable environment is replaced with adapted enterprise information system with specific governance mechanism.
3. Translation of enterprise mission and practices into information that can be communicated and interpreted.
4. Each business process should have dimensions that manage its relationship with other different enterprise entities.
5. Enterprise needs to implement business process lifecycle on different levels. Business process lifecycle is one of BPM promises about business process processing. Business process lifecycle phases include: model, design, simulate, execute, monitor, and optimize [19].
6. Enterprise needs to implement mechanisms for business process learning on different levels. Everyone in the enterprise should have access to the same business processes repository. The repository is used to support quality, definition of information requirements, and BPM modeling [20].

## 2.2 Business Agility Framework

Business agility means fast reaction to change and the ability to rapidly implement changes. Business agility needs to be holistic in scope [14]. Figure 1 presents business agility framework core components. Business agility consists of three interoperable components: Human, Business Process, and Technical agility. Humans are assumed to be agile in management and operations for the enterprise to be agile. Human agility is the main enabler of business agility. Business process agility has gone long road till it reached BPM [19]. BPMS is the enterprise information system that realizes BPM. Technical agility, that addresses IT infrastructure and information systems architecture, can make use of SOA. Aligning BPMS and SOA can enhance business agility.



Figure 1: Business Agility Framework Components

## 3. Business Process Agility

Business Process Management (BPM) is the key to business agility [19]. Business process is a series of inter-related activities that cross functional enterprise boundaries with individual inputs and outputs [3,4]. Business processes are either operational or supporting [21]. Operational business processes are associated with the way enterprise develop strategies, invent, market and sell products or services. Support processes include the provision of Human Resource Management (HRM) activities, information systems infrastructure, and finance and asset management. BPM, as one of business improvement approaches, has a number of predecessors that include TQM, BPR, and WfMS. BPM predecessors have realized the need for process management [2].

### **3.1 Total Quality Management**

TQM wave as a business improvement approach has risen at the late 1980s [19]. TQM is a management approach for an organization, centered on quality, based on the participation of all its members and aiming at long-term success through customer satisfaction, and benefits to all members of the organization and to society [22]. Unfortunately, quality is neither easily defined nor measured. TQM does not have a firm measure by which progress can be monitored. Quality must be converted into something more tangible to be measured, but this does not take place with TQM. Further, TQM does not introduce a clear path to follow.

### **3.2 Business Process Reengineering**

BPR as a business improvement approach has emerged in the early 1990s [6]. BPR promised deliver of dramatic business improvements in a relatively short period of time by completely reinventing existing business processes that is, starting from scratch with new, optimized ways of doing business throwing out the old, encrusted, inefficient procedures of the past [6,19]. Unfortunately BPR involved lot of manual work, failed to provide agility or support ongoing change, could not adequately represent the full complexity of processes due to dependence on computer systems, and provided process discovery and design only in team meetings.

### **3.3 Workflow Management Systems and Enterprise Resource Planning**

WfMS and Enterprise Resource Planning (ERP) promised Chief Executive Officer (CEO) the most flexible IT and providing enterprises requirements via software packages. Unfortunately, software packages were not flexible, installation could take months/years, subsequent applications couldn't be configured to meet all business requirements, business users were enforced to add components. Business process mapping Tools were added to ERPs, and WfMSs promising to produce documentation about the way company works, and to assist review and refinement of business process descriptions, but business process mapping tools could not carry business process models to execution, used proprietary formats and incompatible notations, focused mainly on modeling input-output activities and dataflow, with no focusing on collaborative activity aspect or processes complexity.

### **3.4 Business Process Management**

BPM is the art of understanding, codifying, automating, and improving the way a company does business [4]. BPM is a systematic, structured approach to analyze, improve, control, and manage processes [2]. BPM promises that Processes viewed by human as information and machines as executable code, Process design precedes both from top down - at the level of business strategy and business process design - and bottom up - at the level of leveraging existing IT systems,

Process design is reflected directly in the IT infrastructure, leverage existing investments by connecting databases, legacy systems and package solutions into flexible end-to-end business processes. BPM introduces the concept of 'process processing' and stresses that this concept is not limited to the automatic execution of digital process models, but 'encompasses the discovery, design, and deployment of business processes, as well as the executive, administrative, and supervisory control over them to ensure that they remain compliant with business objectives' [6,23].

### **3.5 Business Process Management Systems**

BPMS enables enterprises to realize BPM initiatives [5]. BPMS is a revolutionary information system that supports designing, administrating, and improving the business processes systematically. BPMS is applied to create, execute, and optimize the business process model that powers the life cycle of each business process instance [24]. BPMS can be used to manage people-to-people, machine-to-machine, machine-to-people, and people-to-machine interactions [25].

#### **3.5.1 BPMS Key Features**

BPMS will bring all the benefits of integration, flexibility, end-to-end visibility and control to the whole extended enterprise. BPMS key features include:

- BPMS integrates and orchestrates other IT systems: BPMS will connect existing communication systems together, convey the required messages from one business entity to another, and ensure that the supply chain constituents stay aligned, with the system putting all of these interactions in the context of a whole process.
- BPMS connects and integrates existing databases, legacy systems, and package solutions into flexible end-to-end business processes.
- Allows processes to be shared across BPMS and across business boundaries: the use of a standard language for describing all aspects of a process will allow the same model to be deployed on several different systems.

### **4. Technical Agility**

Technical agility refers to the ability to quickly change the type and flow of information within an organization within enterprise. Technical agility parameters are IT infrastructure, and information system architecture. IT advance has not yet satisfied business requirements due to improper information systems architectures. SOA addresses technical agility requirements by presenting composability, modularity, and loose coupling concepts as services that wrap underlying IT infrastructure, databases, and legacy systems and present them via standard interface. There is a need to stabilize IT infrastructure rather than developing new ones [7] and SOA enables this stabilization. Enterprises should balance IT to become better positioned and more agile [26]. Services are the building Blocks of an agile enterprise [27].

W3C defines Service as 'A Component capable of performing a task'. Service is 'A vehicle by which a consumer's need or want is satisfied according to a negotiated contract (implied or explicit) which includes Service Agreement, Function Offered and so on' [28]. SOA is the

design pattern that utilizes services concept to achieve architectural advantages. W3C defines SOA as 'A set of components which can be invoked, and whose interface descriptions can be published and discovered'. This definition can be expanded to include the science, art and practice of building applications [29]. SOA is defined as 'The policies, practices, frameworks that enable application functionality to be provided and consumed as sets of services published at a granularity relevant to the service consumer. Services can be invoked, published and discovered, and are abstracted away from the implementation using a single, standards based form of interface' [28].

IT Benefits of adopting SOA include enterprise information systems capabilities of [30,31]:

- Avoid effect(s) of service provider change of implementation as a result of using standard interface.
- Choose alternative instance of same service type without modifying requesting information system.
- Implement different technologies than partners. Implementation of standard interface enables integration and interoperability.
- Increase services exposed in a loosely coupled manner, so information systems can easily combine existing services, and new ones based on business needs. This means less duplication of resources and more potential for reuse.
- Leverage existing legacy applications by wrapping them with standard interface.

#### **4.1 Software Agents as SOA implementation**

Different SOA implementations using different software agents and mobile agents were presented [13, 32-35]. One or more agent can perform a certain task; tasks can be thought as services that compose SOA. Unfortunately, certain limitations inhibited software agents from being the widely accepted SOA implementation. Software Agent is a computer system that is situated in some environment and is capable of autonomous actions in this environment in order to meet its design objectives [36, 37].

Software Agents have characteristics that make them suitable to perform complex functionalities. Characteristics include: Autonomy, Interactivity, Reactivity, Proactivity, Intelligence, and Mobility [38]. Agent is autonomous; it is capable of acting on its own. An agent is goal oriented, collaborative, and flexible, so, it must be autonomous. Agents are designed to interact with other agents, humans, or software programs (Interactivity). Instead of making a single agent conduct several tasks, additional agents can be created to handle un-delegated task. Agents perceive environment via preceptors [39] and respond to changes (Reactivity). Agents do not just act in response to their environment, but agents are able to exhibit goal-directed behavior by taking an initiative (proactive). Agent may need mobility to work on different machines. An agent with this capability is called mobile agent, it can transport itself across different system architectures and platforms, and is far more flexible than those that cannot. Many electronic commerce agents are mobile [38,40]. Mobile agent is an executing program that can migrate during execution from machine to machine in a heterogeneous network [37].

Multi-Agent Systems (MASs) are becoming increasingly important: as a scientific discipline, as a software engineering paradigm, and as a commercially viable and innovative technology [41]. A Multi Agent is any system that contains [42]:

- Two or more agents;
- At least one autonomous agent; and
- At least one relationship between two agents where one satisfies goal of the other.

Some of Multi-Agent frameworks proposed from 2005 till now include work presented in [43-48]. Multi-Agent architecture standards were attempted in order to force MAS global integration [49]. Knowledge Query and Manipulation Language (KQML) was presented in order to support knowledge sharing among agents [50]. Knowledge Interchange Format (KIF) is a computer-oriented language for the interchange of knowledge among disparate programs developed by the Advanced Research Projects Agency (ARPA) sponsored Knowledge Sharing Effort [51]. The OMG group proposed a reference model as an attempt to standardize the development of agent technologies [52]. Knowledgeable Agent-oriented System (KAoS) is described as "an open distributed architecture for software agents." The KAoS architecture describes agent implementations (starting from the notion of a simple generic agent, to role-oriented agents such as mediators and matchmakers), and elaborates on the interactive dynamics of agent-to-agent messaging communication by using conversation policies [53]. The Foundation for Intelligent Physical Agents (FIPA) is a multi-disciplinary IEEE standardizing group pursuing the standardization of agent technology. FIPA's approach to MAS development is based on a "minimal framework for the management of agents in an open environment." [54]. Unfortunately as a result for all the standardization effort, there were no universally accepted commercially supported standard yet. Software agents are not the main SOA enabler because:

- Problems not easily defined into MAS organizations
- Absence of unified framework
- Security is a very big concern, specially with mobile agents
- Lack of commercial support
- Semantic web and Web services presentation (availability) as alternative technologies

#### **4.2 Web services as SOA enabler**

Web services are applications that use standard transports, encodings, and protocols to exchange information [55]. A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. W3C defines Web service as 'A software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a format that machines can process (specifically WSDL), Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with XML serialization in conjunction with other Web-related standards'. Web service can also be defined as 'A programmatic interface to a capability that is in conformance with Wsnn protocols' [28]. Wsnn protocols are present efforts in the W3C and more recently in OASIS to reach a Web service maturity model. Wsnn protocols include WSDL, SOAP, and XML [56]. Web services is a general framework that expedites the sharing of heterogeneous data and software resources dispersed on the internet. The standard-based resource sharing and platform-neutral characteristics of web services have motivated many organizations to apply the technology in diverse areas, such as supply chain management, virtual enterprise, homeland defense, e-government, and e-business [57].

Within SOA context, Web services are categorized into utility and application services [58]. Utility services address and map business process workflow logic to application services. Web services are stateless, so business workflow logic should be maintained explicitly via utility services. Business process workflow logic can include Web services consumed, Web services will be consumed, and how to handle exceptions. Application services represent services that contain logic derived from a solution or technology platform. Application services present abstraction of system functionality.

Web service that maps business process workflow is called process service, and it is a subcategory of utility service. Process services reside in the orchestration or choreography service layer that is a superior layer to the application services layer. Business process workflow logic is extracted, abstracted, and presented as process services. Business process workflow is easily maintained when presented explicitly as process service than embedded within individual solution components. Orchestration layer expresses a business process that is typically owned by the organization. Choreography layer addresses the realm of collaboration between multiple services from different enterprises.

## **5. Utilizing SOA for BPMS**

BPM and BPMS are nothing new, but the use of technology to manage and improve the execution of business processes is more recent [4]. BPMS needs to address new enterprises requirements to support business agility. BPMS should enable enterprises to present and manage business processes as a new information type. BPMS is a necessity for agile enterprises [19].

Technically, BPMS is not Web services [24]. Utilizing SOA for BPMS is not about adopting Web services in BPMS. Though BPMS architecture doesn't have to be SOA based, the tremendous advantages of SOA direct BPMS architecture towards it. BPMS interoperability requirements is extremely different than other enterprise information systems, because BPMS does not just require interactions with other BPMSs; in order to share business processes; but BPMS is aimed to interact with different IT infrastructures and information systems exposed by different enterprises in order to support business processes. IEEE defines Interoperability as 'the ability of two or more systems or components to exchange information and to use the information that has been exchanged'. According to ISO/IEC 2382-01, interoperability is defined as "The capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units". BPMS, either SOA based or not, lies above SOA exposed IT infrastructures and enterprise information systems that include databases, legacy systems, packaged solutions. BPMS tends to connect and integrate different enterprise information systems horizontally rather vertically. Horizontal integration is achieved via integrating business processes. In order to give enterprises flexibility to integrate business processes, their IT infrastructure and information systems should be exposed as loosely coupled stateless services that is accessible via standard interface. Business processes based integration presents maximum integration benefits, and is enabled mainly by SOA [59].

Figure 2 presents the proposed model for aligning SOA and BPMS. Proposed model utilizes standards available for mapping BPM concepts via BPMS into SOA, and consists of three layers: Business, Business Services, and Application Services layer.

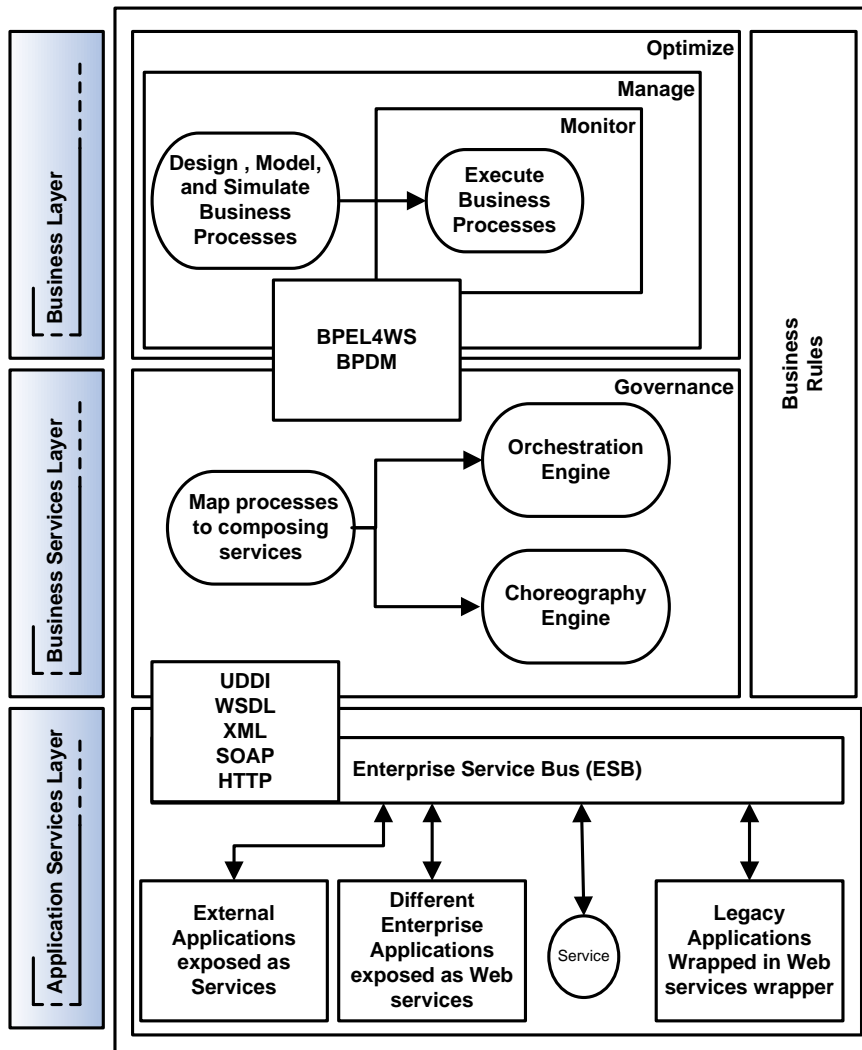


Figure 2: Proposed SOA and BPMS alignment model

Business layer enables business executives to handle business processes as information, BPMS resides in this layer. Business services layer is the layer that maps BPM concepts and requirements addressed by BPMS into SOA based IT infrastructure and enterprise information systems. Application Services layer holds the core services ready to be consumed by different BPMSs and shared among enterprises.

### 5.1 Business Layer

Business layer is responsible for supporting business process life cycle. Business process lifecycle consists of five stages: design, model, simulate, monitor, manage, and optimize business processes. Business layer users are either business analysts, or business managers [60]. Business analysts create the initial drafts of the business processes, and business managers will manage and monitor those business processes.

Business Process Modeling Notation (BPMN) is the standard developed by Business Process Management Initiative (BPMI) to provide a notation that is readily understandable by all

business layer users. Utilizing UML to model business processes were presented in [61-64]. BPMN provides a number of advantages to modeling business processes over the UML; it offers a more conducive process flow modeling technique, and a solid mathematical foundation to map to business execution languages, whereas UML is not. BPMN can map to UML [65]. BPMN has been intended as a human readable layer that hides the complexity of designing business processes in executable XML languages.

Business rules describe the enterprise's constraints, policies, and rules that apply to in achieving its goals. Business contracts are the constraints, policies, and rules orientation among enterprises. Agile enterprises need to present business rules repositories that can be managed, maintained, and checked against current or new executing business processes and / or services.

For business layer to achieve required functionalities, a suitable IT architecture should be presented. IT architecture that utilizes Web services in order to obtain Web services advantages within organizations is presented at the Application Services layer. The connection between business layer and application layer is presented at Business Services Layer.

## **5.2 Business Services Layer**

Business services layer holds orchestration and choreography engines under governance mechanisms to map business processes to composing Web services. Orchestration and choreography engines are the mapping enablers of business processes into executable services. Web services are stateless services that can not maintain business logic, operation flow, or user state; so, the need of an orchestration layer to include business logic is addressed. Orchestration and choreography engines maintain business process workflow logic, performance requirements, and system/user state. Business services layer has access to business rules repository.

BPMN maps to BPEL4WS in order to translate designed business processes into executable orchestration and choreography Web services actions [65]. BPMN maps to BPEL either directly or via XML Process Definition Language (XPDL). XPDL is a format standardized by the Workflow Management Coalition (WfMC) to interchange Business Process definitions among different BPMSs. The goal of XPDL is to store and exchange the process diagram, to allow one tool to model a process diagram, and another to read the diagram and edit, another to "run" the process model on an XPDL-compliant BPM engine. XPDL is not an executable programming language, but specifically a process design format that literally represents the "drawing" of the process definition. XPDL is effectively the serialization of BPMN, as well as any non-BPMN design method or process model which use in their underlying definition the XPDL meta-model. XPDL maps to BPEL4WS.

BPEL4WS allows BPMS to transform business processes into executable complex processes by creating and wiring together different activities that can perform Web services invocations, manipulate data, throw faults, or terminate a process. These activities may be nested within structured activities that define how they may be run, such as in sequence, or in parallel, or depending on certain conditions [66]. BPEL is an "execution language" designed to provide a definition of Web services orchestration, specifically the underlying sequence of interactions, and flow of data from point-to-point. For this reason, BPEL is best suited for straight-through

processing or data-flow application integration. New BPEL extensions that include BPEL4People and BPEL Extension for Sub Processes (BPEL-SPE) are proposed.

A new specification that is expected to be adopted in 2007 is the Business Process Definition Metamodel (BPDM). BPDM is an XML-based proposal developed by the Object Management Group (OMG) that 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." The specification developed in response to this RFP is expected to achieve, among other things:

- The ability to integrate process models for workflow management processes, automated business processes, and collaborations between business units.
- Support for the specification of Web services choreography, describing the collaboration between participating entities and the ability to reconcile the choreography with supporting internal business processes.

Governance refers to putting a consistent process in place to make sure there are checks and balances that ensure that the expected results happen. In the case of SOA, governance refers to keeping checks and balances between business and IT, between the business and government regulations, and between service and performance [8]. Governance rules are maintained explicitly in the proposed model to be checked by executing orchestration and governance engines to present flexibility for executing engines and ease manageability of applied rules. The separation of orchestration and choreography engines from business processes and composing services increased enterprise agility because the workflow logic encapsulated by an orchestration and choreography engines can be modified or extended in a central location [58].

### **5.3 Application Services Layer**

Application services layer holds applications exposed as services, newly added services, and legacy applications wrapped by standard Web services interface. Services at Application Services layer are set of stateless Web services that perform certain task(s). Business process is the summation of tasks performed by one or more services of application services layer at the sequence maintained by orchestration and choreography engines. Services of Application Services layer are reusable among different business processes, can be integrated in new applications, and can be extended address new business processes.

Legacy systems are computer systems that have been in operation for a long time, and whose functions are too essential to be disrupted by upgrading or integration with another system despite its poor competitiveness. Legacy systems compatibility with modern equivalents has been facilitated via wrapper services. Wrapper service is a type of integration service that encapsulates and exposes logic residing within a legacy system via standard Web services interface to be integrated in the new SOA based systems. Utilizing SOA for building new applications within enterprises exposes the ease of integration capabilities between new adapted/developed applications and existing applications.

In SOA, messages are critical to delivering end-to-end services. Messages must be guaranteed a quick and correct delivery. To enhance messages transportation between services, SOA can use

an enterprise service bus (ESB). ESB is the Infrastructure software that makes reusable business services widely available to users, applications, business processes, and other services [67]. ESB is a special layer that runs on top of the network that provides a guaranteed messaging service for the most important messages on the network, including the messages that the components of SOA continuously exchange. Adopting ESB in SOA solutions is not a must, but a recommended best practice.

## 6. Conclusion

Business Agility is a relatively new paradigm painted as a solution for maintaining competitive advantage during times of uncertainty and turbulence in the business environment. Business agility consists of three interoperable components: Human, Business Process, and Technical agility. Human are the initiators and maintainers of enterprises agility. Business Process agility can be maintained by applying BPM concepts. BPM concepts are achieved via BPMS. Technical agility has gained new perspectives by adapting SOA for enterprise information systems. Service-oriented architecture based applications are being heralded as the only viable way to overcome the complexities involved in supporting agile enterprises. Business agility principles are satisfied via utilizing SOA for BPMS by providing the capability of:

- Translating enterprise mission and practices into information that can be communicated and interpreted. This translation is addressed by presenting the business rules as an external standalone repository that reflects enterprises constraints, policies, and practices. Business rules guides interactivity relations among enterprises.
- Replacing traditional enterprise information system with adapted one with specific governance mechanism. Governance mechanisms are exploited explicitly in the presented over executable service via orchestration and governance engines.
- Providing dimensions for each business process that manage its relationship with other different enterprise entities. Business processes are presented in the model as new information type that has an external repository and management, and monitoring activities.
- Implementing business process lifecycle on different levels. Business process lifecycle is one of BPM promises about business process processing. Business process lifecycle phases include: model, design, simulate, execute, monitor, and optimization.

## 7. References

1. **H. Sharifi, and Z. Zhang**, "Agile manufacturing in practice application of a methodology", International Journal of Operations & Production Management, Vol. 21, pp. pp.772-94.
2. **J. F. Chang**, "Business Process Management Systems: Strategy and Implementation", AUERBACH, 2005.
3. **M. MAY**, "Business Process Management: Integration in a Web-enabled environment", Prentice Hall, 2003.
4. **G. V. de Putte, T. Benedetti, D. Gagic, P. Gersak, K. Krutzler, M. Perry**, "Intra-Enterprise Business Process Management", IBM Redbooks, 2001.
5. **B. Ha, J. Bae, and S. Kang**, "Workload Balancing on Agents for Business Process Efficiency Based on Stochastic Model", Second International Conference on Business Process Management (BPM), Germany, 2004.
6. **K. Dirk, B. Karl, and S. Dirk**, "Enterprise SOA: Service-Oriented Architecture Best Practices", Prentice Hall PTR, 2004.

7. **R. Baskerville**, *"Business Agility and Information Technology Diffusion"*, Springer, 2005.
8. **J. Hurwitz, R. Bloor, C. Baroudi, M. Kaufman**, *"Service Oriented Architecture For Dummies"*, Wiley, 2007.
9. **R. T. Fielding**, *"Architectural Styles and the Design of Network-based Software Architectures"*, University of California, 2001.
10. **R. Vokurka, and G. Fliedner**, *"The journey toward agility"*, MCB UP Ltd, 1998.
11. **R. Foster, and S. Kaplan**, *"Creative Destruction: Why Companies That Are Built to Last Underperform the Market – and How to Successfully Transform Them"*, McKinsey, 2001.
12. **R. Dove**, *"Response Ability-The Language, Structure, and Culture of the Agile Enterprise"*, Wiley, 2001.
13. **R. Qui**, *"Enterprise Service Computing: From Concept to Deployment"*, IGI Global, 2007.
14. **N. Evans**, *"Business Agility: Strategies for Gaining Competitive Advantage Through Mobile Business Solutions"*, Prentice Hall, 2001.
15. **G. Kappel, E. Michlmayr, B. Pröll, S. Reich, W. Retschitzegger**, *"Web Engineering - Old Wine in New Bottles?"*, Springer, 2004.
16. **I. Aaen, A. Börjesson, L. Mathiassen**, *"SPI agility: How to navigate improvement projects"*, John Wiley & Sons, 2006.
17. **D. P. Millar, R. L. Heath**, *"Responding to Crisis: A Rhetorical Approach to Crisis Communication"*, Lawrence Erlbaum Associates, 2003.
18. **L. Lau**, *"Managing Business with SAP: Planning, Implementation, and Evaluation"*, IGI Publishing, 2005.
19. **H. Smith, and P. Fingar**, *"Business Process Management: The Third Wave"*, Meghan Kiffer Pr, 2006.
20. **J. Coelho**, *"BPM and Continous Improvement"*, Meghan-Kiffer Press, 2005.
21. **N. Llewellyn, and C. Armistead**, *"Business Process Management: Exploring social capital with processes"*, International Journal of Service Industry Management, MCB University Press, 2000, Vol. 11, pp. 225-234.
22. **G. Tennant**, *"Six Sigma : SPC and TQM in Manufacturing and Services"*, Ashgate Publishing, 2001.
23. **M. McClellan**, *"In Search of BPM Excellence - Straight from the Thought Leaders"*, Meghan-Kiffer Press, 2005.
24. **L. Ferrara, F. Hayden, D. Neal, C. Poirier**, *"The Networked Supply Chain: Applying Breakthrough BPM Technology to Meet Relentless Customer Demands"*, J. Ross Publishing, 2003.
25. **R. Thierauf, and J. Hoctor**, *"Optimal Knowledge Management: Wisdom Management Systems Concepts And Applications"*, Idea Group Publishing, 2006.
26. **D. Leidner, R. Beatty, and J. Mackay**, *"How CIOs Manage IT During Economic Decline: Surviving and Thriving Amid Uncertainty"*, MIS Quarterly Executive, 2003, Vol. 2.
27. **N. Pal, and D. Pantaleo**, *"The Agile Enterprise: Reinventing Your Organization for Success in an on Demand World"*, Springer, 2005.
28. CBDI Service Oriented Architecture Practice Portal, *CBDI*, <http://www.cbdiforum.com/>.
29. **D. Sprott, and L. Wilkes**, *"Understanding Service Oriented Architecture"*, Microsoft Architect Journal, 2004, Vol. 1, pp. 10-17.
30. **L. Wilkes, and R. Veryard**, *"Service Oriented Architecture - Considerations for Agile Systems"*, Microsoft Architect Journal, 2004, Vol. 2, pp. 11-23.
31. **M. Endrei, J. Ang, A. Arsanjani, S. Chua, P. Comte, P. Krogdahl, M. Luo, T. Newling, Tony**, *"Patterns: Service Oriented Architecture and Web Services"*, IBM RedBooks, 2004.
32. **I. Zinnikus, G. Benguria, B. Elvesæter, J. Vayssi ère**, *"A Model Driven Approach to Agent-Based Service-Oriented-Architectures"*, ATHENA, European Integrated Project.
33. **J. Poozhikunnel**, *"Developing a Simple Service Oriented Architecture"*, internet.com, <http://15seconds.com/issue/050119.htm>.
34. **T. Zhang, and H. Jiang**, *"A Framework of Incorporating Software Agents into SOA"*, Benidorm, Spain, : IASTED, Artificial Intelligence and Soft Computing - 2005.
35. *Software Building Blocks - Mobile Agent Based Service Oriented Architecture (ASOA)*, Mobile Agent Technologies, 2005, <http://www.agentos.net/Software%20Building%20Blocks.pdf>.
36. **M. Wooldridge**, *"Introduction to MultiAgent Systems"*, Wiley, 2002.
37. **M. Klusch**, *"Intelligent Information Agents: Agent-Based Information Discovery and Management on the Internet"*, Springer, 1999.
38. **E. Turban, J. Aronson, and T. Liang**, *"Decision Support Systems and Intelligent Systems"*, Prentice Hall, 2004.
39. **S. Russell, and P. Norvig**, *"Artificial Intelligence: Modern Approach"*, Prentice Hall, 2003.
40. **A. M. Riad, and H. A. El-Ghareeb**, *"New Architecture for Mobile News Agent System"*, Egyptian Informatics Journal, 2007, Vol. 8.

41. **K. Syeara**, "*Multi-Agent Systems and Applications*", 9th ECCAI Advanced Course, ACAI 2001 and Agent Link's 3rd European Agent Systems Summer School (EASSS), Prague, Czech Republic, 2001, pp. 17-45.
42. **M. d'Inverno, and M. Luck**, "*Understanding Agent Systems*", Springer, 2003.
43. **D. Moldt, and J. Ortmann**, "*A Conceptual and Practical Framework for Web-Based Processes in Multi-Agent Systems*", Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'04), 2004, pp. 1464-1465.
44. **H. LAU, L. ZHANG, and C. LIU**, "*Solving Generalized Open Constraint Optimization Problem Using Two-level Multi-agent Framework*", IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2005, pp. 558-564.
45. **J. Mostafa, W. Ke, and Y. Fu**, "*Automated text classification using a multi-agent framework*", Fifth ACM/IEEE-CS Joint conference on Digital libraries (JCDL '05), 2005, pp. 157 - 158.
46. **S. Roy, S. Dasgupta, and N. Mukherjee**, "*A Multi-agent Framework for Resource Brokering of Multiple Concurrent Jobs in Grid Environment*", Proceedings of The Fifth International Symposium on Parallel and Distributed Computing (ISPDC'06), 2006, pp. 329-336.
47. **H. EL Yamany, M. Capretz, and L. Capretz**, "*A Multi-Agent Framework for Testing Distributed Systems*", 30th Annual International Computer Software and Applications Conference (COMPSAC'06), 2006, pp. 151-156.
48. **W. Chen, and W. Tseng**, "*A Novel Multi-agent Framework for the Design of Home Automation*", International Conference on Information Technology (ITNG'07), 2007, pp. 277-281.
49. **R. Flores-Mendez**, "*Towards a standardization of multi-agent system framework*", Crossroads, ACM Press, 1999, Vol. 5, pp. 18 - 24.
50. **T. Finin, R. Fritzson, D. McKay, and R. McEntire**, "*KQML as an Agent Communication Language*", 3rd International Conference on Information and Knowledge Management (CIKM'94), 1994.
51. **M. Genesereth**, "*draft proposed American National Standard (dpANS)*", Knowledge Interchange Format, 2004.
52. **S. Virdhagriswaran, D. Osisek, and P. O'Connor**, "*Standardizing agent technology*", ACM Press, 1995, Vol. 3.
53. **J. Bradshaw**, "*KAoS: An Open Agent Architecture Supporting Reuse, Interoperability, and Extensibility*", <http://ksi.cpsc.ucalgary.ca/KAW/KAW96/bradshaw/KAW.html>.
54. The Foundation for Intelligent Physical Agents (FIPA), <http://www.FIPA.org>.
55. **P. Shodjai**, "*Web services and the Microsoft Platform*", Microsoft Software Developers Network (MSDN), 2006.
56. **D. Sprott**, "*A Web service Maturity Model*", CBDI, <http://roadmap.cbdiforum.com/reports/maturity/maturity2.php>.
57. **X. Zhou, J. Li, H. Shen, M. Kitsuregawa, Y. Zhang**, "*Frontiers of WWW Research and Development*", Springer, 2006.
58. **T. Erl**, "*Service-Oriented Architecture: Concepts, Technology, and Design*", Prentice Hall, 2005.
59. **A. M. Riad, and H. A. El-Ghareeb**, "*A Service Oriented Architecture to integrate Web services and Software Agents in Course Management Systems*", Egyptian Informatics Journal, 2007, Vol. 8.
60. **S. White**, "*Introduction to BPMN*", IBM, 2004.
61. **P. Liew, K. Kontogiannis, and T. Tong**, "*A Framework for Business Model Driven Development*", 12 International Workshop on Software Technology and Engineering Practice (STEP'04), 2004, pp. pp. 47-56.
62. **S. de Cesare, and A. Serrano**, "*Collaborative Modeling Using UML and Business Process Simulation*", 39th Annual Hawaii International Conference on System Sciences (HICSS'06), 2006.
63. **P. Jiang, Q. Mair, and J. Newman**, "*Using UML to Design Distributed Collaborative Workflows: from UML to XPDL*", Twelfth International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003.
64. **M. Cambronero, G. Diaz, J. Pardo, and V. Valero**, "*Using UML Diagrams to Model Real-Time Web Services*", Second International Conference on Internet and Web Applications and Services (ICIW'07), 2007.
65. **J. Owen and M. Raj**, "*BPMN and Business Process Management*", Popkin Software, 2003.
66. **S. Weerawarana, and F. Curbera**, "*Business Process with BPEL4WS: Understanding BPEL4WS, Part 1*", IBM, 2002.
67. **K. Vollmer, and M. Gilpin**, "*The Forrester Wave: Enterprise Service Bus*", Forrester, 2006.