Inter-enterprise System and Application Integration: A Reality Check

Jorge Cardoso[1], Wil van der Aalst[2], Christoph Bussler[3], Amit Sheth[4], and Kurt Sandkuhl[5]

[1] SAP Research, Dresden, Germany
[2] Eindhoven University of Technology, The Netherlands
[3] Merced Systems, Inc., CA, U.S.A.
[4] Kno.e.sis Center, Wright State University, Ohio, U.S.A.
[5] Jönköping University, Sweden

Abstract. This paper structures the summary of the panel held at the 9th International Conference on Enterprise Information Systems, Funchal, Madeira, 12-16 June 2007 that addressed the following question: "Are you still working on Inter-Enterprise System and Application Integration?" The panel aggregated distinguished experts from the areas of process management, workflow, Web services, SOA, and Semantic Web.

Keywords: Inter-enterprise integration, business process management, workflow, Web service, SOA, semantic Web.

Wil van der Aalst

Eindhoven University of Technology, The Netherlands

We Are Creating Our Own Problems, e.g., Current Standardization Efforts Are Only Increasing Complexity!

The problem of building enterprise systems (and the glue between these systems) is the overwhelming complexity. To set up a contemporary middleware platform is a complex task which is further complicated if software from different vendors needs to work together. The fundamental idea of web services is sound and promising. However, the abundance of overlapping standards for web services composition is overwhelming. In fact, the collection of competing and complementary web services standards is creating a new Tower of Babel. Standards are proposed without clear semantics and before a standard is adopted there are already new standards which build on it. The acronym WSAH (Web Services Acronym Hell) coined in [1] nicely illustrates the problem. Focus should be on a few good standards; otherwise we are creating our own problems. The "Not Invented Here" syndrome reflects the unwillingness to adapt standards and products to best practices and sound theories. This phenomenon results in people intentionally differentiating things from one

J. Filipe, J. Cordeiro, and J. Cardoso (Eds.): ICEIS 2007, LNBIP 12, pp. 3–15, 2008.

another rather than using solid, scientifically proven, foundations. For example, new process modeling languages are proposed on a daily basis by people unaware of foundational concepts such as bismulation, true concurrency, Turing completeness, Petri nets, etc.

Find Out What Is Really Going on Before (re)Designing a System!

Reality is often very different from what is modeled or what people think. Whatever representation is used (Petri nets, BPMN, UML or any other modeling language), the model is an abstraction (i.e., things are left out) and may not reflect reality. The first observation (abstraction) is unavoidable and should be accepted as a fact of life. However, the second observation is more problematic and should be addressed urgently.

As long as managers and system designer take a "PowerPoint" reality as starting point, information systems will remain to have serious alignment problems. Therefore, it is vital that more efforts are invested into finding out what is actually happening. If it is not possible to "catch" reality in a model, then it does not make any sense to develop or try to improve enterprise systems. Process mining [4] can be a valuable tool here. The omnipresence of event logs is an important enabler of process mining, i.e., analysis of run-time behavior is only possible if events are recorded. Already today systems ranging from cross-organizational systems to embedded systems provide detailed event logs.

Do Not Specify More Than Strictly Needed!

Both organizations and people are autonomous entities that in general behave in a reasonable way. However, many systems and standards do not acknowledge this. There is a tendency to over-specify things, i.e., to describe in a detailed and procedural way what should happen. A nice example is the role of BPEL. Whilst being a powerful language, BPEL is of a procedural nature and not very different from classical workflow languages e.g., the languages used by systems such as Staffware, COSA, SAP Workflow, and IBM WebSphere MQ Workflow. Hence, it is not clear why organizations need to agree on the use of BPEL. When an organization subcontracts part of it work to other organizations, it seems odd to require that the other parties need to enact specific BPEL models for their internal processes. Yet most of the languages proposed for orchestration are of a procedural nature, while it seems obvious that a more declarative language is needed [3]. Thrust in people and organizations implies an under-specification rather than an over-specification of processes.

Let's Make Fuzziness Explicit!

Related to the above point is the gap between high-level modeling languages and executable languages. As an illustration, let us look at the SAP reference model. Of the more than 600 process models in the SAP reference model, about 20 percent is incorrect in the sense that there are possible deadlocks, livelocks, etc. These models intend to describe how the SAP system works and how it should be used. However, the number

of errors shows that there is a complete disconnect between the actual software and these models. This nicely illustrates the general problem of going from "PowerPoint diagrams" to executable models/software. One of the problems is that today's languages are either informal or have some executable interpretation. Therefore, the modeling process of going from an imprecise model to an executable model cannot be supported adequately. It would be good to have single language to specify both things that are precise and executable and things that are still vague and left open. Making fuzziness explicit in models is important to avoid costly misinterpretations.

Christoph Bussler

Merced Systems, Inc., CA, U.S.A.

If you are still working on Inter-Enterprise System and Application Integration, then you should re-evaluate and re-assess your research agenda, because …

The World Moved on

One current development in the software industry is that the notion of 'application' is changing in the sense that an application (or software system) does not necessarily have to be licensed and locally installed at an organization any more for the sole purpose of this one organization. The notion of application service provider (ASPs), who installed a software system and made its interfaces available over the Web, also changes as software vendors provide the hosting themselves, opening up a very different model of software as services (SaaS), see below.

A second current development integrates existing Web interfaces in new forms providing new functionality that has not been available before. The term 'mash-up' is used for this development and gains significant momentum at this point, see the discussion below. The wider development is captured in the term 'Web 2.0' that clearly departs from the notion of simply providing the user interfaces for a single software system, but instead becomes the integration point for several types of functions from different software systems available remotely over the Web infrastructure.

From an academic research viewpoint the topic of 'integration' (application integration or inter-enterprise integration) completely failed from the angle of a clear conceptual and working model that is agreed upon by the research community (and industry for that matter). Unlike in the database research community with the relational database management system success, the research community around integration did never arrive at all to a sound foundation in terms of an appropriate conceptual model as basis for further research and industrial development.

SaaS (Software as a Service)

SaaS is very distinct from application hosting or making functionality available as Web Services. In a first characterization, SaaS is a specific engineering aspect for software systems that allows multiple tenants be present in the same installation of a software system. In fundamental terms, the software system is aware of several

organizations that have their functionality and data implemented in a single installation of the software system. Traditionally this meant that the functionality is the same across tenants, only their particular data is different. Examples of this are www.ebay.com where each customer shares the same functionality, but their data (e.g. bids, offers, financial power, etc.) is different. In addition to end user functionality, middleware services are available in this mode, too, in the meanwhile, e.g. Amazon S3 or Amazon SQS by www.amazon.com.

This multi-tenant awareness starts extending into the design-time domain where the software system configuration can be different for each tenant, but still in the same installation of the software system. An example is www.salesforce.com that allows customers (tenants) to modify the design-time configuration. So the same software system in a single installation supports tenant-specific modification of design-time data.

However, the industry does not stop here. The notion of community is coming into the picture in different ways. One is the community of tenants. For example, www.xeequa.com allows tenants to form a community, i.e., they are aware of each other for mutual benefit. While all share the same functionality, they cooperate at the same time. Xeequa pushes further by providing a clear model of 'data ownership' that goes beyond corporate boundaries following the insight that each person has a corporate life as well as a professional and private life (and data associated with those roles).

A second way of community, the developer community, is cultivated by www.salesforce.com where the development of the software system functionality is not limited to their own employees any more, but opened up to any developer who wishes to contribute. The developer goes to www.salesforce.com to develop, test and offer to the tenants. This clearly shows that the classical distinction of develop – install –use starts changing in fundamental ways.

In my mind, the community aspect will make all the difference, analogous to the effects that can be seen in other Web applications like www.amazon.com or www.hotels.com where the community provides ratings or other Web sites where the community starts adding meta-data for the community's mutual benefit.

Web 2.0

Still outside the corporate software system world, but for sure coming into it sooner or later, the Web 2.0 development addresses the integration of existing functionality available of the Web. For example, www.trulia.com or www.zillow.com integrates various sources like real estate offerings, spatial maps and financial information into one Web site. This combination is termed mash-up following the insight that the sum is greater then the individual parts.

www.programmableweb.org is tracking mash-up developments and the number is constantly increasing, making this a very important 'movement'.

Furthermore, the explicit notion of social networks like www.linkedin.com start to appear in this space as they can offer new type of functionality because of their explicit knowledge of social relationships. www.linkedin.com, for example, does not only have static forward links (i.e., who is related to me and whom can I reach?), but also dynamic backward links (i.e., who looked at my social network profile)?

The World Really Moved on

In summary, the world really moved on from various viewpoints. In terms of integration this means that the entities that require integration changed in their nature quite a bit (as in fundamentally) requiring new thoughts and developments around it. The idea that an entity represents one organization is left behind completely as well as the notion that there is one location for integration for a given company or organization.

Of course, like all developments in IT and computer science, it takes time for all software systems to follow the SaaS and Web 2.0 model. And some of those software systems will never embark on this paradigm. However, the fact that this paradigm is picked up in all domains like sales, real estate, human resources, finance, and so on makes me believe that it will continue to grow and become main stream.

Meaning of 'Integration'

So, if your heart is still in the world of integration, which is not necessarily a bad place to be, you need to ask yourself, what does 'Inter-enterprise' and 'application' integration mean in this changing world and how the notion of 'integration' will change going foward? And you need to ask yourself if maybe this time around it might be worth-while to put effort into a common conceptual model in the academic research community to achieve a similar success as the database research community achieved with relational database management systems.

Amit Sheth

Kno.e.sis Center, Wright State University, Ohio, U.S.A.

New World Order for Interactions Across Enterprise Information Systems in the Flat World

As the world has become flat ("The World Is Flat" [5]), the dominance of agriculture and then of manufacturing has given way to the dominance of services. Businesses have changed correspondingly and nowhere more than in their interdependence. A company may, for example, outsource not only the non-critical back office operations, the production of the components, or even a subassembly, but also the logistics of its entire supply chain, which is critical to its operations. Increasing interdependence has also led to myriad ways a company's operations, competitive-ness, and profitability are tied to those of its suppliers and partners. In terms of business models, many have changed from selling products to providing services, even when underlying materials and intellectual property are about the same.

In spite of these drastic changes in businesses, entrenched legacy enterprise application systems have evolved slowly, whereas newly developed enterprise applications have leapt ahead. We feel that interactions among the applications have changed significantly and that the underlying reliance on and realization of processes has changed the most and will continue the do so. In 1999 ("Processes Driving the

Networked Economy" [6]), we noted "*So far, most of the attention in Information Systems has gone to data. We believe that this attention will increasingly shift to information and knowledge on one hand and processes on the other. The first deals with service and product, the second deals with how to effectively support or render it.*"

In the area of data and data interoperability, we often need to revisit the same challenges we addressed a long time ago. For example, recently we revisited work on data mapping [7] more than a decade old to address data mediation in Web Services [8]. And, interestingly, we found that while using SAWSDL, the newly developed Semantic Web Services standard, we were able to reuse the earlier work on data mapping with little or no fundamental advances. We can reduce unnecessary revising and rehashing of earlier work if we understand the four aspects or levels of interoperability: system, syntax, structure, and semantics [9]. Although these levels have been well discussed (and we hope well understood) in the context of data and data interoperability, two difference can be noted: (a) data mediation in the context of more dynamic processes (such as dynamic trading processes or adaptive web processes referred to in this note) present new challenges not encountered above and require increasingly sophisticated use of semantics; and (b) these levels of interoperability also apply to processes and process interoperability issues. Our view is that Web Services is NOT "old wine in new bottles." Clearly some of the old problems, such as those related to data mediation, resurface in the context of newer infrastructure for supporting interoperability when adapting Service Oriented Architecture and Web Services. However, as we argued in [10], low initial complexity, its use of XML, and support for an intrinsic loose coupling architecture, etc. provide just the right incremental advances in software componentry to make them practically useful, that has now resulted in wide adoption.

In [6], we also outlined three types of inter-organizational workflows (processes)—process portal, process vortex, and dynamic trading processes. For non-technical reasons there are few examples of process portals and process vortexes, but with the highly interdependent and dynamic nature of businesses and their interactions with global partners and suppliers, we see the increasing importance and relevance of and research in dynamic trading processes. This has involved recognizing the events that affect a process [11] and how optimally to adapt a process once an event relevant to it has been identified ("Adaptive Web Processes" [12]).

Let us now turn our attention to the role of humans in enterprise information systems and inter-organizational processes. Although automation and process technologies have reduced the role of humans in repetitive and mundane tasks, humans as well as organizations play integral and increasingly sophisticated roles in managing processes. Consider a supply chain process that has technical capabilities to adapt to relevant events such as currency fluctuation or a fire at a supplier's fabrication plant. In addition to the challenges of adapting to optimize cost and time factors, the business also needs to keep a keen eye on risks associated with different choices. This requires decision makers to be integral to the inter-organizational process. And for a process in service business, allocation of human resources needs an integral model of the organizational structures of both the enterprise and its partners. IBM has recently outlined the notion of Services Science, which emphasizes the need to model not only technical but also human and organizational aspects of systems (including enterprise information systems and processes) that support services.

In the context of Services Science, we have outlined a Semantic Services Science (3S) model [13], which seeks to demonstrate the essential benefits of semantics to the broader vision of Services Science, with service descriptions that capture technical, human, organizational and business-value aspects. We asserted that ontology-based semantic modeling and descriptions can be used to energize services across the broad service spectrum. In this article, we demonstrate how the 3S approach could be used along three points in this spectrum: (1) semantic descriptions of standard Web services with the help of SAWSDL, semantic policies, and agreements; (2) semantic descriptions of lightweight Web services using Web 2.0 technologies (e.g., REST, AJAX); and (3) ontology-based profiling of people and organizational aspects of the assets associated with the knowledge services.

The use of semantics for data interoperability and integration was discussed in the 1980s. We now see the emergence of an era in which the use of semantics will be much more pervasive, spanning interoperability related to middleware, data, services, and process within and across enterprises. This is coupled with advanced technical capabilities associated with Semantic Web and Semantic Web Services, as well as a better understanding of how to apply both weaker forms of semantics (also called "little semantics"), incorporating folkonomies and limited agreements in social communities, and deep semantics encapsulated as domain ontologies involving domain knowledge and agreement across scientific and business communities that are captured in formal languages.

Kurt Sandkuhl

Jönköping University, Sweden

It Is an Illusion to Believe We Will Ever Solve All Interoperability Problems!

The main challenges when creating integration and interoperability between enterprise systems is to overcome heterogeneity. On all levels between enterprises, we see various and incompatible elements, including the IT infrastructure, applications, business processes, or information and knowledge models. Integration and interoperability require an understanding of objectives to reach and technical basis.

Despite all progress made during the last decades, integration and interoperability will continue to be challenges. It is an illusion to believe that we will ever solve all interoperability problems. Some reasons for this are:

- The providers of enterprise systems and IT infrastructure components are market actors like all other "for profit" companies. These companies will continue developing innovative products. If they are commercially successful, others will offer similar solutions for the same problem, and – if commercially successful - create the next interoperability challenge. Interoperability and integration are only a strategic issue for enterprise system developers, if their market position and competitiveness is improved. As long as the own products have unique selling points and functionality, why open them too early for interoperation?

- Standards only partially solve the problem. Many official standards are based on de-facto industrial standards implemented much earlier than the official standardization takes place. Furthermore, the knowledge about existing standards and their implementation is often insufficient. Standards are like other products: only if they meet their "market window" they will be successful.
- Legacy systems increase complexity. Old enterprise systems in many cases are highly optimized and highly productive. To replace these "legacy systems" with up-to-date solutions often can not be motivated, neither from an economic nor from a technical perspective. But inter-enterprise integration and interoperability has to include these legacy systems leading to increased complexity. And we are creating new legacy every day. Today's highly sophisticated SOA systems will in 15 years from know be considered legacy.

Understand the Driving Forces Before Integrating Systems!

When discussing about the battle of interoperability, how successful this has been in the past and whether existing technologies are adequate for solving contemporary challenges, we have to agree on the perspective of the discussion. Driving forces typically found in industries and public authorities are:

- Automation of routine processes
- New laws and regulations requiring and integration of earlier separated applications
- Improvement of customer service
- Improvement of decision support on operative and management level
- Reduction of lead times and duration of business processes or production processes

For all these areas, there are lots of examples for successful integration of systems and applications or for creating interoperability. Understanding the driving force for integration and interoperability will help to reduce complexity and increase the probability for success.

Think in Long-Living Infrastructures!

What should we do differently when working on interoperability and integration in networked enterprises? Any successful contribution to inter-enterprise system and application integration will be long-living (otherwise it cannot be considered successful) and has to be designed for this purpose. Industry areas like banking, insurance, energy or automotive show that enterprise systems with an age of 20 years or more can still fulfill their purpose. A key feature of long-living systems is the design for maintainability, which often includes separation from business logic and implementation platform. Let us not design interoperability and integration solutions just for a single technology. Furthermore, inter-enterprise integration should be designed as infrastructures, not as solitary solutions for specific enterprises. Infrastructures have to be scalable, offer a high availability and provide means for managing service levels, including performance management, security management and configuration management.

This Was Just the Beginning!

Networking of systems and applications for inter-enterprise integration is just the beginning. Several industrial areas work on integration of knowledge structures, both process knowledge and product knowledge. To develop a new product or service usually requires various competences including traditional service and engineering fields (electrical, mechanical, computer, material, etc.) and contributions from financial and service sectors. Even multi-national enterprises often do not have all these competences in-house. The resulting need for cooperation leads in times of a global economy to collaboration and competition at the same time: In one project, a development partner might be a contributor to developing a new product or service, in another context this same partner might be allied with a competitor.

The integration of all stakeholders into a flexible and dynamic organization covering the complete lifecycle of products or services and of the associated business services will create new challenges for inter-enterprise integration.

Jorge Cardoso

SAP Research, Dresden, Germany

Devise Conceptual "Nodes" Instead of "Leaves" Solutions!

Given a problem, such as the inter-enterprise integration of systems and applications, the generation of a solution can be categorized, contextualized and explained using a tree structure. Let us assume that a tree structure can represent the solution space for a particular problem. The root node and the nodes represent conceptual solutions. On the other hand, leaves symbolize the actual implementation of a system to solve a given problem. Nodes describe generic solutions that can be applied to solve recurring problems, within a defined context. A generic solution means that a node does not define a specific solution. Rather, it identifies the set of problems that can be solved with a specific conceptual approach. Its influence is derived from the fact that it is an abstraction that can be re-used transversely in a large number of situations. Nodes are useful to solve recurring problems when the problem is not unique, and are most useful when the problem occurs often. The defined context means that it is necessary to put bounds on a generic solution represented by nodes since there are no universally true solutions. It is necessary to understand the circumstances in which this generic solution is suitable. It is also important to elaborate on it to create specific designs that will be represented by leaves. We believe that sound and generic solutions have not been found yet to devise intra- and inter-EAI systems since vendors and academia has been focused on "leaves" instead of focusing on "nodes."

Current EAI, WFM, and ERP Systems Are "Leaves"!

Having illustrated the relationships between generic and specific solutions we can analyze the development and adoption of inter-enterprise system and application integration. EAI (Enterprise Application Integration), Workflow Management (WFM),

EAI Vendor	Software Revenues (million $)	Market Share (%)
IBM	66.5	12.0
NEON	59.6	10.8
Mercator	55.5	10.0
TIBCO	50.4	9.1
SUN	3.4	6.4
BEA Systems	28.0	5.1
STC	24.5	4.4
Vitria	21.8	3.9
Active	20.5	3.7
Extricity	16.6	3.0

Fig. 1. Top 10 EAI vendors [1]

and ERP (Enterprise Resource Planning) solutions have typically been developed as "leaves" implementations without first investing on the development of conceptual models for "nodes". This means that the systems, languages and tools that were developed lacked conceptual and theoretical foundations. Typically, no generic solutions to solve recurring problems within a defined context were developed. As a result, we are daily faced with an endless number of incompatible integration management systems (e.g. IBM MQSeries Family, BEA eLink, SUN Forte Fusion, TIBCO, Vitria BusinessWare, etc), a vast variety of languages to describe processes models (e.g. BPEL, BPML, WSFL, XLANG, XPDL, BPMN, etc), and a broad diversity of isolated non-interoperable tools specific to integration systems (tools for design, administration, simulation, analysis, planning, scheduling, etc).

Figure 1 shows the plethora of EAI systems available in the market. Each solution uses internal models and architectures, and very often the languages to define processes are proprietary. This situation gives rise to the worldwide promotion of islands of intra-enterprise integration solutions without supporting the inter-enterprise interoperability of systems. EAI suffered and suffers from a lack of standardized practices during early implementation that lead to reduced outcomes and strong disappointments.

Is There Any Bright Future for EAI Solutions?

While the set of EAI solutions available in the market suffers from a lack of strong foundations and conceptual models, the EAI software license market is expected to reach $4.9 billion by 2012 [1]. This represents a growth of more than 300% in 7 years, since in 2005 the market was evaluated in $1.4 billion. The driving force for this growth has been linked with the adoption of the Internet as a channel to conduct business (i.e. e-commerce and B2B). Therefore, while there are some concerns with respect to the soundness of the theoretical foundations of EAI, WFMS, and ERP systems, from an academic perspective, it is clear that these systems will continue to find a strong acceptance in the industry.

EAI Solutions Will Eventually Adopt and Use Semantics

Semantic inter-enterprise interoperability is the key for the implementation of the idea of a knowledge based economy where networks of enterprises (and SME – small to medium sized enterprises – in particular) can gain advantages from the peculiarities of the productive fabric (many small companies, highly specialized, closely connected with each other, highly flexible thanks to the diffusion of knowledge and very skilled people in the whole fabric). World-class competitiveness of enterprises strongly depends, in the future, on their ability to rapidly build dynamic networked enterprises. Novel technologies for interoperability within and between enterprises need to emerge to radically solve a problem that has not been addressed by the research community before.

As integration becomes multifaceted, more complex, and crosses organizations' boundaries, there will be an increasing need to make sense of hundreds of data stores, tables and attributes. Nowadays, the data managed by organizations is stored in various types of schemas and using different terminologies for data assets. As a result, the creation of a common terminology for information is fundamental for establishing a strategic integration infrastructure. Let us consider the following example. Intel Corporation has 12 factories, assembly and test facilities worldwide. To take advantage of the latest technological developments, Intel processed electronically more than 60% of materials transactions and 85 % of customer orders. The critical goal for Intel is to timely and accurately manage the production flow throughout the manufacturing network. In practice this is a hard task to achieve since the various information systems spread among the manufacturing network have strong semantic differences. For example, depending on the country, materials are categorized using different taxonomies. As a result, information systems have difficulties in communicating. These difficulties can be alleviated by using a semantic approach that we will be analyzing in the following sections.

Today, integration is a top priority for many European and worldwide enterprises and most organizations have already realized that the use of semantic Web technologies is the best solution to support cross-organizational cooperation for SME that operate in dynamically changing work environments. Semantic Web technologies are already viewed as a key technology to resolve the problems of interoperability and integration within the heterogeneous world of ubiquitously interconnected systems with respect to the nature of components, standards, data formats, protocols, etc. Moreover, we also believe that semantic Web technologies can facilitate not only the discovery of heterogeneous components and data integration, but also the communication, coordination and collaboration behavioral of employees and individuals.

Enterprise Application and Human Integration

The semantic Web relies on the theoretical research done in the context of ontologies as a formal support for treating the semantic-sharing and interoperability problems. Ontology-based human integration aims at reducing and eliminating terminological and conceptual confusion by defining a shared understanding, that is, a unifying framework enabling communication and cooperation amongst people in reaching

better inter-enterprise organization. Presently, one of the most important roles ontology plays in communication is that it provides unambiguous definitions for terms used in a software system, but semantic needs to be applied rapidly to human integration to enable communication, coordination, and cooperation. The use of ontologies for improving communication has already had been shown to work in practice. Interesting examples of successful ontologies include the Disease Ontology[1] (a hierarchical and controlled vocabulary for human disease representation), the FAO[2] (Food and Agriculture Organization of the United Nations) – which is committed to help information dissemination by providing consistent access to information for the community of people and organizations – and the Open EDI ontology[3] which defines the ontology for data management and interchange between enterprises.

References

1. van der Aalst, W.M.P.: Don't go with the flow: Web services composition standards exposed. IEEE Intelligent Systems 8(1), 72–76 (2003)
2. van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M.: Web Service Composition Languages: Old Wine in New Bottles? In: Chroust, G., Hofer, C. (eds.) Proceeding of the 29th EUROMICRO Conference: New Waves in System Architecture, pp. 298–305. IEEE Computer Society, Los Alamitos (2003)
3. van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M., Russell, N., Verbeek, H.M.W., Wohed, P.: Life After BPEL? In: Bravetti, M., Kloul, L., Zavattaro, G. (eds.) EPEW/WS-EM 2005. LNCS, vol. 3670, pp. 35–50. Springer, Heidelberg (2005)
4. van der Aalst, W.M.P.: Process Mining and Monitoring Processes and Services: Workshop Report. In: Leymann, F., Reisig, W., Thatte, S.R., van der Aalst, W.M.P. (eds.) The Role of Business Processes in Service Oriented Architectures, Schloss Dagstuhl, Germany, July 2006. Dagstuhl Seminar Proceedings. Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), vol. 6291 (2006)
5. Friedman, T.: The World Is Flat: A Brief History of the Twenty-first Century. Douglan and McIntyre, Ltd (2005)
6. Sheth, A., Aalst, W., Arpinar, I.: Processes Driving the Networked Economy. IEEE Concurrency 7(3), 18–31 (1999)
7. Kashyap, V., Sheth, A.: Schematic and Semantic Similarities between Database Objects: A Context-based Approach. Very Large Data Bases (VLDB) Journal 5(4), 276–304 (1996)
8. Nagarajan, M., Verma, K., Sheth, A., Miller, J., Lathem, J.: Semantic Interoperability of Web Services - Challenges and Experiences. In: Proc. of the 4th IEEE Intl. Conference on Web Services, Chicago, IL, September 2006, pp. 373–382 (2006)
9. Sheth, A.: Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics. In: Goodchild, M.F., Egenhofer, M.J., Fegeas, R., Kottman, C.A. (eds.) Interoperating Geographic Information Systems, pp. 5–30. Kluwer Academic Publishers, Dordrecht (1999)
10. Sheth, A., Miller, J.A.: Web Services: Technical Evolution yet Practical Revolution? In: Web Services: Been There, Done That? IEEE Intelligent Systems, Trends & Controversies, pp. 78–80 (January/February 2003)

[1] http://diseaseontology.sourceforge.net
[2] http://www.fao.org/agris/aos/
[3] http://www.jtc1sc32.org/, known as ISO/IEC JTC 1/SC 32

11. Gomadam, K., Ranabahu, A., Ramaswamy, L., Sheth, A., Verma, K.: A Semantic Framework for Identifying Events in SOA. In: Intl Conference On Web Services (ICWS) (2007)
12. Verma, K., Sheth, A.: Autonomic Web Processes. In: Benatallah, B., Casati, F., Traverso, P. (eds.) ICSOC 2005. LNCS, vol. 3826, pp. 1–11. Springer, Heidelberg (2005)
13. Sheth, A., Verma, K., Gomadam, K.: Semantics to energize the full Services Spectrum. Communications of the ACM (CACM) sp. issue on Services Science 49(7), 55–61 (2006)
14. WinterGreen Research Inc., http://www.wintergreenresearch.com/