Workflow Management Systems and ERP Systems: Differences, Commonalities, and Applications

Jorge Cardoso[1], Robert P. Bostrom[2] and Amit Sheth[3]

[1]Departamento de Matemática e Engenharia
Universidade da Madeira
9050-078 Funchal – Portugal
jcardoso@dme.uma.pt

[2]MIS Department
Terry College of Business
University of Georgia,
Athens, GA 30602, USA
BBostrom@terry.uga.edu

[3]LSDIS Lab, Computer Science Department
University of Georgia
Athens, GA 30602, USA
amit@cs.uga.edu

Contact information: Amit Sheth

LSDIS Lab, Computer Science Department
University of Georgia, USA
amit@cs.uga.edu
Tel:(706) 542-2310, Fax: -4771

Abstract. Two important classes of information systems, Workflow Management Systems (WfMSs) and Enterprise Resource Planning (ERP) systems, have been used to support e-business process redesign, integration, and management. While both technologies can help with business process automation, data transfer, and information sharing, the technological approach and features of solutions provided by WfMS and ERP are different. Currently, there is a lack of understanding of these two classes of information systems in the industry and academia, thus hindering their effective applications. In this paper, we present a comprehensive comparison between these two classes of systems. We discuss how the two types of systems can be used independently or together to develop intra- and inter-organizational application solutions. In particular, we also explore the roles of WfMS and ERP in the next generation of IT architecture based on web services. Our findings should help businesses make better decisions in the adoption of both WfMS and ERP in their e-business strategies.

Introduction

Today's enterprise infrastructure and organization require the support of critical business processes. Since business processes are the fundamental building blocks of an organization's success, information technologies that focus on process management and improvement have been good candidates to help organizations to fulfill their corporate visions and to improve their competitive positions. In the past two decades, special interest has been given to two distinct solutions that improve business processes: Workflow Management and Enterprise Resource Planning (ERP) systems.
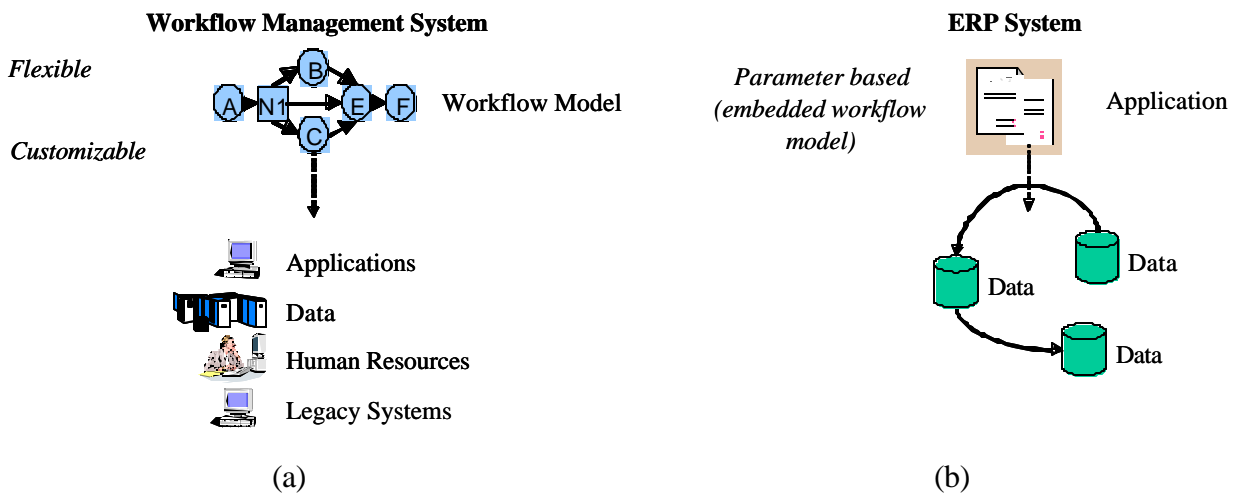

Figure 1. WfMS and ERP systems

Both classes of systems focus on business processes, but the approaches taken by them are different. A WfMS is implemented based on a process specification and execution paradigm (Figure 1 (a)). Under a WfMS, a workflow model is first created to specify organizational business processes, and then workflow instances are created to carry out the actual steps described in the workflow model. During the workflow execution, the workflow instances can access legacy systems, databases, applications, and can interact with users.

ERP systems are implemented around the idea of prefabricated applications as shown in Figure 1 (b). To achieve better "fit" between the prefabricated applications and the needs of the organization, ERP systems must be configured by setting various application parameters. The more parameters an ERP application has, the more flexibility in configuring the business process. However, the workflow model in conventional ERP systems is not explicitly specified because it is embedded in the applications and the parameter tables.

Figure 1 represents one of the key differences between WfMS and ERP systems. One way to better understand these differences is to distinguish between flow logic and function logic. Function logic deals with a specific task, such as updating a customer record or calculating order discounts, while flow logic deals with combining many functions in some sequence to solve more complex problems such as processing an order. In ERP systems, flow logic and function logic are both embedded in applications and parameter tables. In contrast, a WfMS separates the two explicitly. Flow logic is captured in a workflow model, usually graphically represented, and

function logic is captured in the applications, data, and people the model invokes. Thus, a WfMS enable developers to separate the flows among a system's components (applications, data, people) from the workflow model. Workflow systems are process-centric, focusing on the management of flow logic. On the other hand, ERP systems are data-centric, focusing on managing function logic via a common homogeneous data infrastructure across the organization to support multiple applications.

We can use the analogy with programming languages to illustrate this difference. Working with a WfMS is similar to programming in a non-procedural high-level language where developers are working directly with a graphical representation of the business workflow model. The WfMS development platform then generates necessary application components, database linkages, *etc.* to execute a workflow process. Working with ERP systems is like working with a 3rd generation procedural language, where it is necessary to deal directly with applications and data.

The recent trend from ERP vendors is to integrate a WfMS into ERP systems. For example, in order to facilitate the customization and deployment of ERP systems, some vendors are making their systems workflow-driven with the integration of workflow components. For example, Baan bought COSA [1] and integrated the COSA stand-alone workflow system into its ERP solution. Oracle has added a WfMS that allows any of its self-service ERP applications to be integrated in various workflows that automatically process and route information, support personalized business rules, and model and maintain business processes using a graphical workflow builder. These integrative approaches take advantage of the strengths of both technologies.

This paper analyzes the similarities and differences between WfMS and ERP systems, which is important because both technologies have a major role to play in the current and future organizational capabilities to manage business processes within and across organizations. As more and more companies are switching to a process-centric approach to application management, these technologies will become increasingly indispensable.

One of the problems is that ERP and WfMS technologies are often managed by different groups within an organization. There is often little understanding of the other group's technology and little communication between groups. This problem is propagated by the lack of integration in academic institutions. ERP teaching and research is mainly done in Information Systems Departments within Colleges of Business, while WfMS teaching and research is mostly done in Computer Science Departments within Colleges of Arts and Science. There is a need for these two technology camps in both business and academic organizations to have a holistic view of these two technologies.

This paper is organized as follows. Section 1 presents the technological evolution of the systems. In section 2, the two systems are compared on three dimensions: domain scope, technological scope, and system implementation. Section 3 discusses how the two types of systems can be used independently and together to address intra- and inter-organizational application integration. Section 4 addresses implications of our analysis on research and education. Finally, section 5 presents our conclusions and summarizes the key findings.

1 Systems Evolution and Maturity

The evolution of WfMS and ERP systems has taken two distinct trajectories, and further, both are presently at different levels of maturity. For instance, ERP systems have found stronger acceptance and deployment around the world than WfMS.

Workflow management systems appeared in the 80's, but there is some consensus that the office information systems (OIS) field, an important field in the 70's, is the predecessor of workflow systems [2]. The first OIS prototypes were developed in the late seventies. The pioneer systems included the *SCOOP* project [3], which was oriented to the automation of office procedures, and *Officetalk* [4], which provided a visual electronic desktop metaphor, a set of personal productivity tools for manipulating information, and a network environment for sharing information.

In the 80's, due to several failures in office automation projects and installations [5], interest in office information systems declined, and research efforts were redirected towards flexible groupware systems and models [6]. WfMSs were initially introduced during this time period.

Advances in transaction processing and integrated office systems made workflow systems popular in the 90's. While Alonso *et al.* [7] pointed out that they were innovative and had gained a high level of popularity, WfMS had not yet matured into a well-proven and stable technology. Research prototypes include METEOR [8], MOBILE [9], ADEPT [10], and Exotica [11]; commercial products include IBM MQSeries Workflow, Staffware, TIBCO InConcert, and COSA Workflow. General information on WfMSs can be found at the web sites of the Workflow and Reengineering International Association [12] and the Workflow Management Coalition [13].

Historically, the origin of ERP systems can be traced back to the 60's when the focus of organizational information systems was mainly on handling traditional inventories. In the 70's, the systems focused on material requirements planning (MRP) [14]. These systems helped translate master production schedules into the planning of raw material requirements. In the 80's came the concept of MRP-II, which involved optimizing an entire plant's production processes.

New technological advances facilitated the development of software systems to manage functional areas such as finance and human resources, and made possible the implementation of the conceptual model proposed by Blumenthal [15]. This model described an integrated architecture and framework for organizational information systems, which can be seen as the seed of ERP systems. Important progress has been made since then, and currently various ERP systems are available, including Baan, CSC, JD Edwards, Oracle, PeopleSoft, and SAP. General information on ERP systems can be found at the TechRepublic's web site [16].

By analyzing the historical evolution and maturity of the two systems, we can observe that ERP systems emerged from a steady progression that can be traced from the 60's to a high point in the mid-late-90's, with major success stories around the world. This progressive evolution has lead to a trusted and well-known product in the industry, with high credibility, a good position in the market, and a strong advantage over competitive systems. Furthermore, ERP systems were in a mature state, and WfMS were not, at the time organizations needed to solve critical problems, such as the Y2K problem and the replacement of outdated legacy systems.

The stronger acceptance of ERP systems is reflected in industry buying patterns. According to Aberdeen Group's estimates, the spending in the Business Process Management software sector (which includes workflow systems) has reached $2.26 billion in 2001 [17]. In comparison, ERP systems revenue was $21.5 billion in 2000, according to the research company International Data Corp. (IDC) [18]. The cycles of popularity of OIS and WfMS – in the 70's, 80's, and 90's – has certainly generated confusion among managers and end-users and provoked a certain reluctance to adopt workflow systems. However, it appears that WfMS will have a key role in future efforts, especially in the systems integration area. This paper will help clear up this confusion by comparing these systems and discussing their future roles.

2 WfMS and ERP systems

Keeping in mind the evolution and maturity paths of the systems under study, we now turn our attention to the conceptual differences between the two systems. Our goal is not to do a comprehensive analysis of all factors involved in selecting and implementing WfMS and ERP systems, which is clearly beyond the scope of a single paper. Instead, we focus on the major differences in the technologies themselves and what they can do for a business. Thus, we do not discuss factors such as costs, human issues, *etc.*

We compare the two technologies along three main dimensions: domain scope, technological scope, and system implementation. Each dimension helps compare the two systems from a specific point of view. We picked these three dimensions because they highlight the key differences in the systems. The major differences are summarized in Table 1.

	WfMS	**ERP systems**
Domain Scope	• Customized processes • Domain independence • *Ad-hoc* and dynamic domains • No international settings	• Embedded processes with some customization • Domain specific • Static domains • International settings
Technological Scope	• Process-centric • Supports workflows involving humans, IT applications, and transactional workflows • Heterogeneous and autonomous environments	• Data-centric • Transactional workflows • Homogeneous environments with common data infrastructures
System Implementation	• Workflows are manually designed and the corresponding code is automatically generated • May require data conversions	• Based on pre-written "off-the-shelf" components • Require data conversions

Table 1 – WfMS vs. ERP Systems

2.1 Domain Scope

The domain scope defines the suitability of a system for a specific type of application or organization. This characterization is important since organizations have different needs and characteristics. For example, a multinational organization obviously has different needs compared to an organization that has only a regional base, and a financial organization has different requirements from a marketing organization.

Workflow systems have been installed and deployed successfully in a wide spectrum of organizations. Muth *et al.* [19] observe, "most workflow management systems, both products and research prototypes, are rather monolithic and aim at providing full-fledged support for the

widest possible application spectrum." The same workflow infrastructure (*e.g.*, METEOR [20]) can be deployed in various domains, such as bio-informatics [21], healthcare [22], telecommunications [23], military [24], and school administration [25]. Leymann and Roller [26] discuss the application of workflow to other areas, such as mobile computing, systems management, multi-databases, the Internet, application development, object technology, operating systems, and transaction management.

In figure 2, a workflow process from the field of genomics [27] exemplifies how workflow systems can be used to design processes for a broad spectrum of domains.
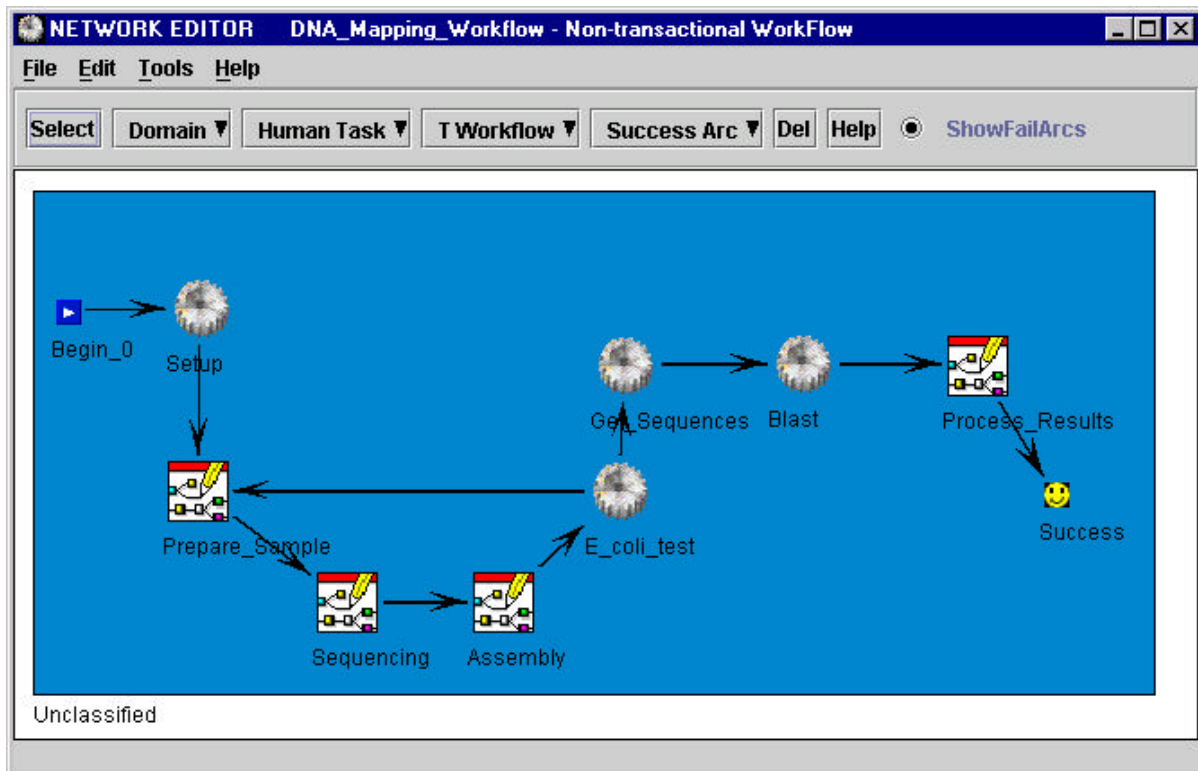

Figure 2. Genomic Workflow Example

A major task in genomics is determining the complete set of instructions for making an organism. Genome projects are very demanding, and incur high costs of skilled manpower. There are many different types of tasks that must be performed, such as sequencing, sequence finishing, sequence processing, data annotation, and data submission. A single genomic workflow may be spread across multiple research centers, and the individual tasks in a workflow may be carried out at one or more of the participating centers. Many of the challenges of building an information system to manage a physically distributed genome project can be addressed by a workflow system.

The workflow model for such a workflow graphically specifies the control and data flow among tasks. For example, the workflow model in Figure 2 is composed of several tasks and sub-workflows. The tasks illustrated with machine gears represent automatic tasks, while the ones illustrated with boxes represent sub-workflows.

At runtime, the workflow system reads the model specifications and transparently schedules task executions, providing the right data at the right time to the right worker. It manages

distributed genomic tasks located at different research centers, such as DNA sequencing machines, matching algorithms, and human resources. Further, the workflow system provides a framework to easily reengineer a genomic workflow when new technological, biological, and chemical advances are made.

The ability to separate flow logic from function logic makes workflow technology suitable in a large number of domains. Thus, WfMS constitute a generic tool that can be used to integrate different types of data, applications and people in a broad spectrum of contexts.

While some workflow systems are generic, others are more oriented to particular domains. The overall high degree of domain independence allows for customization, specialization, and a high level of uniqueness for the workflows created. While it has been said that workflow systems can be applied to many domains, no concrete solution has yet been proposed for workflow deployment in an international setting. This is because WfMS do not yet include some indispensable features, such as internationalization, multi-currencies, and multi-languages, which are valuable when deploying workflows in worldwide markets.

On the other hand, ERP systems are domain specific due to the adoption of reference models or process templates that embody the best practices in various business domains. ERP systems include libraries of predefined business processes for various functional areas. Reference models supposedly reflect preferred business models, including underlying data and process models. As a result, ERP systems, via supplying a broad spectrum of dedicated applications, provide a solution that satisfies many organizations.

ERP systems have developed industry best-practice solutions for most major industries, such as aerospace and defense, automotive, consumer products, chemicals, engineering and construction, retail, and health care. Implementing these ERP systems requires the setting of thousands of parameters in order to customize applications to individual organizational contexts.

While the adoption of best-practices is a seductive approach, Kumar and Hillegersberg [28] note considerable mismatches between the actual country, industry, and company specific business practices and the reference models embedded in ERP systems. These mismatches have led to many implementation problems and failures. One of the most notable failures is FoxMeyer Drug, a $5 billion company that filed for bankruptcy in 1996, arguing that the primary cause of problems was a failed ERP implementation [29, 30]. Mismatches between ERP systems and organizational processes can be resolved through the configuration of parameters, adding additional programs, or through the company adjusting to ERP processes. WfMS have had their problems and failures as well [31]. However, these problems seem to be related to culture and organizational dynamics associated with the implementation, not to domain specific process models, which is the case in ERP failures.

Compared to WfMS, ERP systems do not supply an effective framework for dynamic domains in which a process topology can constantly change due, for example, to new technological advances. Additionally, *ad hoc* and heterogeneous processes are better managed using a WfMS, mainly because they do not rely on predefined reference models. On the other hand, ERP systems are well suited for multinational applications, since they offer features such as multi-language support and multi-currency support. Additionally, ERPs provide dedicated solutions for specific industries, allowing organizations not striving for differentiation to be more efficient and competitive.

2.2   Technological Scope

The second dimension that we use to compare WfMS and ERP systems is the technological scope. This dimension characterizes the systems based on their technological capabilities. Both systems are similar in that their architectures have moved from mainframes to client server architectures, and more recently to the Web. Although both systems manage business processes, each one focuses on different types of business processes.

As an example, let us consider two business processes. The first one, a trading process, is used to update customer orders, inventory, and financial databases in response to commercial transactions between suppliers and customers. The workflow reflects the changes that are made to the order database, to the inventory, and to the financial database. The second business process manages the genetic sequencing procedure previously described (see Figure 2). The workflow is responsible for coordinating the tasks of the lab assistants, controlling sequencing equipment, and executing DNA matching algorithms against genetic databases. The two processes have a different set of technological requirements; in the first case, the support is targeted toward database access, data synchronization, and database interoperability, while the second process requires human coordination, equipment control, and application execution.

WfMS and ERP systems have been developed with distinct sets of technological capabilities. We can highlight these different capabilities by examining the different types of applications each system supports. Business process technology focuses its attention and effort on supporting three different types of applications [32]:

    (1) Workflows involving humans (see the genome sequencing example above),
    (2) Workflows involving systems and applications (see examples provided in Section 3), and
    (3) Transactional workflows (see the trading process example above).

In Type 1 workflow systems, the workflow involves humans. The WfMS is responsible for controlling and coordinating the human tasks. Such settings increase the complexity of WfMS implementation because the system has to share responsibility to ensure the consistency of documents and workflow data among its users.

In Type 2 systems, a WfMS is responsible for the control, coordination, and execution of computation-intensive operations and specialized software tasks, with little or no human intervention. In addition to being highly automated, this type of workflow system may require access to distributed information systems (for example, relational databases, application servers, and XML-repositories). This type of system must provide good mechanisms for integrating applications.

Finally, Type 3 systems involve human intervention and system orientation that is transaction-based. Such systems involve the coordinated execution of multiple tasks that may involve humans, require access to heterogeneous, autonomous and distributed systems, and support selective use of transactional properties (*e.g.*, atomicity, consistency, isolation, and durability) for individual tasks or for entire workflows. The support of such properties requires sophisticated concurrency control and recovery techniques in order to ensure the consistency and reliability of the system. The development of software that supports transactional workflows is a complex task. Nevertheless, solutions have been investigated in the context of extended transaction models by Rusinkiewicz and Sheth [33], Georgakopoulos *et al.* [32], Eder and Liebhart [34], Alonso *et al.* [7], and Worah and Sheth [35].

ERP systems constitute applications that focus on the integration of data. The objective of ERP applications is to provide an integrated solution to all business areas (financial, sales, human resource, *etc.*). The underpinning of shared data structures across many applications eliminates the need to pass data step-by-step among applications by accessing data from a common structure. The focus of ERP systems is mainly on structured data transactions, i.e., Type 3 systems. ERP modules operate directly with common interoperable databases to ensure consistent information for all purposes. This makes the manipulation of data easy.

The ERP concept makes the strong assumption that data infrastructures are homogeneous across the organization, that is, the data is stored in interoperable databases, and in some cases, the databases used are all from the same vendor. Some ERP systems (for example, Oracle 11i) only support specific database management systems. Other ERP systems are more versatile, supporting the most well-known database platforms. Such a strong assumption forces organizations to migrate from existing systems to a standardized data environment. During implementation, only data integration from interoperable databases needs to be considered. ERP systems are data-centric, and thus they are well suited for modeling transactional processes for which only data integration is needed as in the case of the first example above.

WfMSs can address all three types of workflows outlined above, however, they are most suitable for modeling workflows involving humans and software systems (Types 1 and 2), especially if the systems are autonomous and heterogeneous. On the other side, ERP systems are more appropriate to model transactional workflows, which are data oriented. Nevertheless, when transactional workflows involve heterogeneous systems, a more appropriate solution may be the adoption of a WfMS.

For small organizations with heterogeneous infrastructures, the adoption of a WfMS to integrate their systems may be a more adequate solution, since it does not require the time and monetary investments associated with ERP implementations.

2.3   System Implementation

The third dimension of our analysis answers a question often raised by managers: "What are the main differences between the implementation of a WfMS and that of an ERP system?" There are many system implementation factors discussed in the literature (see Scott and Vessey [30] for a summary.) Our focus is on technical differences that affect implementation. WfMS and ERP systems differ in two main technical aspects that influence implementation: code generation and data conversion.

2.3.1 Code Generation

Business information systems can be designed as custom applications, or they can be purchased as standard "off-the-shelf" solutions (*e.g.,* SAP and PeopleSoft applications). Since the development of custom applications is generally expensive and is often plagued by uncertainties, the second approach is often preferred when implementing information systems. ERP systems are composed of prewritten software modules available "off-the-shelf", often supplying sufficient flexibility to match many organizations' needs by configuring thousands of parameters. When an ERP module is acquired, it is fully deployed for a functional area. In order to link different functional areas, it is necessary to acquire different modules. For example, two ERP modules need to be deployed to link human resource and financial departments, one for

each department. The infrastructure created allows for an automatic flow of information between the two departments.

WfMS, on the other hand, are not module-oriented. There is no need to acquire and deploy special modules for coordinating departments. Since the WfMS is usually generic (see Section 2.1), workflows can be designed with pan-Web cross-enterprise [36], cross-organizational [23] and cross-departmental settings. The system controls the information flow from each department and transfers it to the appropriate task to be processed according to a workflow map. Workflow administrators or consultants define the workflows. Once workflows are designed, the deployment of applications is accomplished with little programming, the system automatically generates the necessary code for each application, but the code for individual tasks still has to be written or acquired. A few workflow systems, such as METEOR [8], can automatically generate the code for specific tasks.

In some cases, system integration expertise may be needed to manually code and integrate with the WfMS special features such as to link the workflow engine to legacy applications, set transactional properties, define recovery procedures, *etc.* Additionally, the integration of access control and user rights on both the workflow as well as the applications requires additional effort in a heterogeneous environment [37].

When business processes are represented as hard- or semi-hard-coded applications, as is the case with ERP systems, an inherent flexibility is missing. The only flexibility in an ERP system comes from the parameters that can be configured. In a WfMS, the idea is to be able to model processes, typically by using visual tools, and then delegate the responsibility of "designing" the behavior of the software to the workflow system.

2.3.2 Data Conversion

Since both systems provide solutions for applications integration, it is expected that during system implementation some sort of data manipulation or data conversion will occur. Since workflow systems do not require a uniform and interoperable data infrastructure, legacy database systems can be integrated without any substantial change. For a WfMS, data conversion is not mandatory; nevertheless, for optimization and organizational purposes, data can be converted into a more compatible format. On the other hand, ERP systems usually require data conversions and define an architecture for data storage. As a consequence, legacy databases need to be replaced with ERP-compatible databases. The conversion task involves standardizing, transferring, and "cleaning" existing data elements.

3 Application Integration

System interoperability and application integration are critical areas of concern and the cause of many failures. Recently, Gateway wrote off $140 million from its failed effort to run their on-line store with a purchased software system [38]. The software did not work well with Gateway's existing systems. Another example is the candy maker Hershey Foods. They installed three software application packages, part of a $112 million system, with disastrous results due to incompatibilities with other application programs [38].

Both WfMS and ERP systems are promoted based on their application integration capabilities. It should be clear from previous sections that these two systems have distinct advantages that could be combined to produce enhanced application integration capabilities. In this section, we

discuss how WfMS and ERP systems can be used independently or together to address intra- and inter-organizational application integration needs. We also discuss the impact of Web Services in this regard.

3.1   Enterprise Integration

ERP vendors are trying to provide organizations with a total set of integrated applications that meet all the organizational needs. Although organizations might find it advantageous to interact with only one vendor, this is not happening because ERP vendors cannot meet the needs in all the areas with quality software. Even in key ERP domains, companies may buy and integrate different vendors' ERP components. For example, the US Navy is moving towards using PeopleSoft in the human resource area and SAP in the supply area.

One of the key problems developers have with ERP systems is to understand and change the process flow model embedded in the applications and parameter tables. To address this problem, a recent trend has been to incorporate workflow components into existing ERP systems. The first strategy was to bring in a WfMS simply as a stand-alone tool to implement workflows and to use the WfMS as a documentation aid to document ERP flows.

This strategy has changed to make ERP systems more workflow-driven or workflow-enabled with the integration of workflow components [39] as indicated by Baan's purchase of COSA[1] and Oracle's efforts towards integration of ERP into a WfMS mentioned earlier. In order to do this, ERP vendors need to replace the flow logic embedded in their applications and parameter tables with a WfMS. The role of the WfMS is to manage the flow logic and invoke applications when necessary.

Although it appears the full integration of workflow into ERP suites is just getting started, we see this pattern continuing. The movement to the two-level programming model utilized by WfMS (i.e., separation of flow logic from function logic) enables the creation and integration of applications with a higher degree of agility to respond to business needs.

WfMSs can provide an important enterprise integration function by themselves. Workflow systems can orchestrate and start other applications such as spreadsheets, legacy systems, ERP systems, *etc.* This capability makes them ideal for implementing workflows involving systems and applications, described previously in Section 2.2 as Type 2 systems. Thus, a WfMS acting in this mode could be viewed as a type of "middleware" platform serving to integrate diverse applications such as legacy applications [35] and ERP applications [2].

In the practitioner literature, the systems used to develop this type of workflow applications are referred to as "Enterprise Application Integration" (EAI) [40] and "Business Process Management" (BPM) [41] tools or systems. These systems are often labeled the second generation WfMSs because they provide much richer integration capabilities than traditional WfMSs. Slater [46] cites an EAI example from Bose where two legacy call center applications and an e-Commerce Web application are linked to an underlying database that is connected to an ERP system. They are linked by an EAI toolset such as the one available from SeeBeyond [42].

As with most workflow systems, EAI/BPM systems implementers use a graphical tool to model processes (see Figure 2). A process model can be constructed, changed, or adapted through a GUI interface. When alterations are made, the EAI/BPM software reflects the changes correctly in its code for routing or transforming data. In these types of applications, the nodes in a process model represent different external systems or applications. Using this approach, it is possible to continue using existing applications, including ERP systems, underneath the

workflow layer and use EAI/BPM to manage and change processes. Workflow systems, such as EAI/BPM, provide an important enterprise integration function.

3.2   E-Commerce: Value/Supply Chain Integration

Interoperability is a key issue in e-Commerce because more and more companies are creating B2B (Business-to-Business) links to better manage their value chain. In order for these B2B links to be successful, heterogeneous systems from multiple companies need to interoperate seamlessly. Automating inter-organizational processes across supply chains presents significant challenges [2]. These processes are often complex and involve more varieties of systems than enterprise integration.

  ERP vendors have moved into the supply chain management area. The integration of ERP systems into a supply chain is a complex task. ERP modules are designed to reflect a particular way of doing business. Organizations must usually adapt to the ERP system and not vice versa. This philosophy makes the integration of two or more different businesses in the supply chain difficult. One of the questions that arises is which ERP business model to select. Consider the case of integrating two ERP systems in a supply chain. If the two systems are from the same vendor, the integration can be achieved after the two application models involved are modified and connected. But, if the two ERP systems have a different architecture because they are from different vendors, the integration can be very difficult to achieve.

  However, using a workflow system to integrate a supply chain may be simpler than using an ERP system. A WfMS can work as a bridge between two or more organizations. The approach we outlined above for enterprise integration can be applied to e-Commerce as well. Remember that a WfMS does not require a radical change to basic applications and data infrastructures of organizations. Instead, a workflow process that spans organizations is created on top of the supply chain topology. This workflow interoperation capability can be applied to link and manage the control and data flows between two or more ERP systems.

  There are ongoing efforts in the workflow area to develop systems specifically designed for e-Commerce workflows [2, 36]. The EAI/BPM tools discussed above are also used to provide real-time management of B2B processes. However, the future development in this area will come from the implementation of Web Services.

3.3   Web Services: the Future of Application Integration

The biggest providers of computer software and services are making large investments to support and promote something call Web Services [43]. What they are promoting is a whole new approach to information systems (IS) and IS architecture. The approach goes by different names: IBM calls it "Web Services," Microsoft refers to it as ".Net," Oracle calls it "Network Services," and Sun talks about an "Open Network Environment." Web Services promise to enable organizations to integrate and reuse software already built and reduce the hassle and expense of systems integration. In this section, we briefly explain the Web Services architecture, its role in enterprise and supply chain integration, and the role of workflow and ERP technology within it.

  At the foundation of Web Services architecture are software standards and communication protocols, such as XML and SOAP [44], which allow information to be accessed and exchanged easily among different programs. These tools allow applications to communicate with each other

regardless of the programming languages they were written in or the platform they were developed for. Web Services are not used to build monolithic systems; they are a set of tools used to stitch together existing applications to create new distributed systems.

In terms of integration, Web Services provide a core architecture that allows any two Web applications to talk to each other. Thus, it offers an important solution to enterprise and e-Commerce application integration. Several researchers have identified workflows as the computing model that enables a standard method of building Web Services applications and processes to connect and exchange information over the Web [45-47]. We briefly addressed this application of workflow technology when discussing EAI/BPM systems in the Enterprise Integration section. Leymann *et al.* [48] provide a comprehensive overview of the role of workflow technology in Web Services and how it is being implemented in IBM's approach.

Once the applications or Web services are identified, a process flow model (similar in format to the one in Figure 2) depicting the business process is constructed. Based on this model, the WfMS automatically generates the appropriate code to coordinate the flow of data and messaging between the Web applications using standards defined in the foundational level of the Web Service architecture. At runtime, the WfMS reads the flow model specification and transparently schedules Web services for execution.

An additional advantage of workflow technology is that once the workflow model is implemented, suitable tools are available to manage the execution of business processes. This can be accomplished by defining events in the business flow model that can be used to report on progress of individual processes or overall aggregate process execution. In the IBM Web Services architecture these progress reports are collected by a dashboard infrastructure [48]. The dashboard provides a high-level summary of process status and also provides drill down capability to perform a detailed analysis of specific applications and activities.

Currently, Web Services implementations only support limited integration capabilities as an alternative to the EAI/BPM tools discussed earlier. However, most current Web Service implementations are not using workflow technologies, which are available in some EAI/BPM tools. Integration of applications in current Web Services is accomplished at the programming level, not through a flow model.

In late 2002, we are starting to see Web Services architectures that integrate workflow components. IBM released the latest version of its application server environment for implementing Web Services, WebSphere 5.0. For the first time, we find at the core of WebSphere 5.0 a workflow engine built around an emerging specification language called Business Process Execution Language for Web Services (BPEL4WS). The key to developing the workflow component of Web Services is a standard workflow or Web Services specification language. This area is still under development and no definitive standards exist today. However, there are proposals currently being considered. The two more prominent process specifications include BPEL4WS (mentioned above) and DAML-S. (A comprehensive review of specification languages and standards issues is presented in [49].)

At the heart of current EAI/BPM software and future Web Services is a process-centric approach to integration, which represents a movement away from the data-centric approach of ERP systems. In the past, to get the integration desired, customers had to rely on a particular suite of applications – based on proprietary interfaces – from a single ERP vendor. However, the trade-off for the customers was that not all applications in a suite were best-of-breed. Many companies are starting to seek out best-of-breed applications and move away from ERP suites because of greater flexibility, deeper functionality, lower cost, and ease of implementation with specific applications [50]. Although ERP systems have solved many integration problems, most

large companies still have a hodgepodge of non-integrated systems. Thus, many companies will continue to face major integration problems. Many companies see Web Services, not ERP suites, as a way to handle these integration problems [50].

ERP vendors argue that the best-of-breed approach will result in higher costs of integration, lots of data integration, and having to deal with multiple vendors [50]. The latter assertion is true. However, early evidence shows that Web Services will be a cheaper approach to integration [50]. As for data integration, this is the strength of the ERP approach. The question to explore is: *Is data integration really needed at the transactional level?* Many of the data integration needs are to support data mining of historical transactions and other data. An alternative data integration approach, which many companies are exploring, is to handle these retrieval needs with data warehousing technology.

ERP systems will not go away! They provide a solid core infrastructure in many organizations. However, their role, especially in terms of integration, is changing. As discussed in previous section, it appears that ERP vendors are moving to workflow-enable ERP applications to make them available as Web services. Thus, ERP applications will be potential components in component-based systems made possible by Web Services. WfMSs will also be potential components. However, WfMS technology will also provide a core technology around which future system integration efforts will be based. Thus, WfMS and ERP technologies will be combined in new ways using the Web Services framework to enhance application integration capabilities.

4 Implications for Research and Education

Stohr and Zhao [2] and Basu and Kumar [39] outlined key research issues in the workflow area. Although the integration issues previously discussed were cited as critical research areas in these articles, the key emerging role of workflow technology in EAI and Web Services and the relationship between WfMS and ERP systems were not mentioned with the exception that Basu and Kumar briefly mentioning the need to research how to workflow-enable existing ERP systems. Thus, we would add these items to these research agendas.

Stohr and Zhao note that most academic research in workflow area has been done in Europe and the research appears mostly in the Computer Science literature. It is interesting to note that most ERP research has taken place in the United States and appears in IS journals, while ERP research is relatively unknown in Computer Science. However, both WfMS and ERP researchers have a common interest in information technology that focuses on process improvement and management. Thus, there is a need to see integrated research teams, from Computer Science and IS, attack some of the key workflow issues and challenges. The authors of this paper, two computer scientists (a researcher and a Professor/Entrepreneur) and an IS Professor, are a good example of such an integrated team.

Stohr and Zhao [2] list three major WfMS research areas: technical issues, management and organizational issues, and market, economic and social issues. Technical issues have received the most focus since they are the primary interest in Computer Science. ERP researchers in IS bring important expertise, relevant research models, and empirical data in the latter two research areas. Since these issues are also applicable to WfMSs, better integration of research in these two areas, via joint research teams or other mechanisms, is needed. This is especially true in the Web Services area where we are looking at a totally different paradigm for acquiring, developing, and maintaining information systems.

A similar argument can be made for education. Although one will find some process mapping in system development classes, it is limited. Workflow technology and business process definition are not covered in depth in IS required classes in most universities. Workflow technology may be covered briefly in Groupware or Decision Support Systems electives. On the other hand, ERP systems are discussed in most IS required classes and many schools have a one or two complete courses devoted to ERP systems. Given the importance of workflow technology, especially in the Web Services area, more attention to business process definition and workflow technology is needed in the IS curriculum. More attention to ERP systems and their relationship to WfMSs is needed in Computer Science.

5 Conclusions

We have conceptualized three different types of business process applications: (1) Workflows involving humans, (2) Workflows involving systems and applications, and (3) Transactional workflows, which can be used to illustrate the differences between WfMS and ERP systems.

Most WfMSs address all the three types of workflows outlined above, whereas ERP systems mainly address the third type, transactional workflows. WfMSs are more suitable to model workflows involving humans and software systems (Types 1 and 2), especially if the systems are autonomous and heterogeneous. On the other hand, ERP systems are more appropriate to model transactional workflows. Nevertheless, when transactional workflows involve heterogeneous systems a more appropriate solution might be the adoption of a WfMS. This is because ERP systems generally rely on a common, homogeneous, and interoperable data infrastructure.

Thus, organizations may need ERP or WfMS systems, or both. Workflow management systems are more directed toward process management, involving application and data integration of heterogeneous, autonomous, and distributed systems. Most of the WfMSs show domain independence, in the sense they can be implemented in any business sector.

ERP systems are data-centric, and therefore, they are more focused on information management and data integration. This type of system is also domain-dependent. Business templates are provided to be used in specific functional and market sectors. ERP systems are very suitable for a departmental, organizational, and cross-organizational scope operating on a national or international scale, where there is a good fit between desired organizational processes and those embedded in ERP applications. The system is built from prefabricated applications.

Both WfMS and ERP systems will play a major role in organizations' application integration. We presented in Section 3 how WfMS and ERP systems can be used independently or together to address intra- and inter-organizational application integration. The current directions in application integration point to the architectures shown in Figure 3. In both intra- and inter-organizational integration, flow logic is being separated out and captured in WfMS with function logic found in ERP systems, other applications or other WfMS. In intra-organizational area, we see this represented in the moves by ERP vendors to integrate WfMS into their ERP architectures and emergence of EAI/BMP workflow based tools.
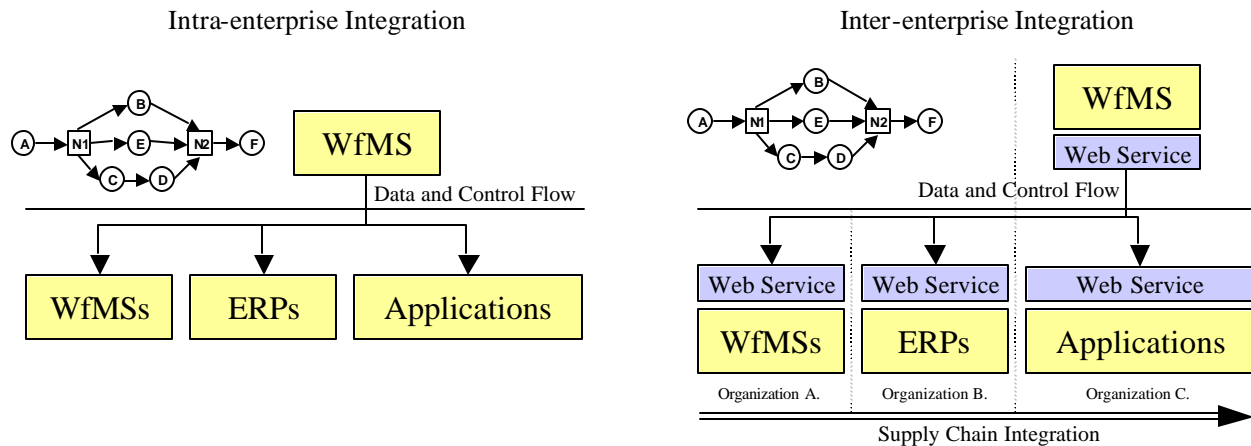
Figure 3. Intra and inter-enterprise integration using WfMS and ERP systems

As noted in this paper, a solution to enterprise (intra-organization) and e-Commerce (inter-organization) application integration is the use of workflow technology in conjunction with the architecture provided by Web Services. The use of workflow technology and Web Services architecture provides for the orchestration of Web Services and allows any two Web applications to talk to each other using standard interfaces. ERP customers, in the past, have been locked into a particular suite of applications based on propriety interfaces. The emerging trend from ERP vendors of decoupling the flow logic of their applications in order to allow them to be used as a Web services will continue in the future.

This paper provided a holistic view of ERP and WfMS systems by outlining key differences and similarities, describing which domains they are most applicable in, and explaining how these technologies are being combined to address business process improvement and integration. The view presented was limited by the fact that empirical research data sources were inadequate. Thus, we had to rely on cases, stories and analyses found in the practitioner literature. To address this limitation, we did an extensive literature review to provide the most complete and accurate comparison of these systems.

This paper started with the observation that there were different camps in both business and academic organizations that deal with ERP and WfMS systems. We hope this paper will stimulate dialog and integration between these two technology camps.

6 References

1.   COSA, *COSA Workflow*, 2002. http://www.ley.de/de/news/vortraege/BaanWorld.pdf
2.   Stohr, E.A. and J.L. Zhao, *Workflow Automation: Overview and Research Issues.* Information Systems Frontiers, 2001. **3**(3): p. 281-196.
3.   Zisman, M., *Representation, Specification and Automation of Office Procedures*, in *Department of Business Administration, Wharton School*. PhD Dissertation, 1977, University of Pennsylvania: Philadelphia, PA.
4.   Ellis, C.A. *Information Control Nets: A Mathematical Model of Office Information Flow*. in *Conference on Simulation, Measurement and Modelling of Computer Systems.* 1979: ACM, New York. p. 225-239
5.   Hammer, M., *The OA Mirage*, in *Datamation*. 1984. p. 36-46.

6. Ellis, C.A. and G.J. Nutt. *Workflow: The Process Spectrum*. in *NSF Workshop on Workflow and Process Automation in Information Systems.* 1996. Athens, Georgia. p. 140-145

7. Alonso, G. *Advanced Transaction Models in Workflow Contexts.* in *Proceedings of the International Conference on Data Engineering.* 1996. p. 574-581

8. Kochut, K., A. Sheth, and J.A. Miller, *Optimizing Workflow.* Component Strategies, 1999. **1**(9): p. 45-57.

9. Jablonski, S. *MOBILE: A Modular Workflow Model and Architecture*. in *Proceedings of the 4th International Working Conference on Dynamic Modelling and Information Systems.* 1994. Noordwijkerhout, Netherlands

10. Reichert, M. and P. Dadam, *ADEPTflex - Supporting Dynamic Changes of Workflows Without Losing Control.* Journal of Intelligent Information Systems - Special Issue on Workflow Managament, 1998. **10**(2): p. 93-129.

11. Mohan, C., et al., *Exotica: A Research Perspective ob Workflow Management Systems.* Data Engineering Bulletin, 1995. **18**(1): p. 19-26.

12. WARIA, *Workflow and Reengineering International Association*, 2002. http://www.waria.com/

13. WfMC, *Workflow Management Coalition*, 2002. http://www.wfmc.org/

14. Lunn, T. and S.A. Neff, *Material Requirements Plannning: Integrating Material Requirement Planning and Modern Business*: McGraw-Hill Professional Publishing, 1992.

15. Blumenthal, S.C., *Management Information Systems: A Framework for Planning and Development*. NJ: Prentice Hall, 1969.

16. TechRespublic, *TechRespublic ERP supersite*, 2002. http://www.techrepublic.com/supersiterd.html

17. Cowley, S., *Study: BPM market primed for growth*, 2002. www.infoworld.com

18. Landergren, P., *Net Presents ERP Vendors with New Challenges*, 2002. www.itworld.com

19. Muth, P., et al. *Workflow history management in virtual enterprises using a light-weight workflow management system*. in *Proceedings of the 9th International Workshop on Research Issues in Data Engineering*. 1999. Sydney, Australia: Available at http://www-dbs.cs.uni-sb.de/~mlite/. p. 148-155

20. METEOR, *METEOR (Managing End-To-End OpeRations) Project Home Page*, 2002, LSDIS Lab. http://lsdis.cs.uga.edu/proj/meteor/meteor.html

21. Hall, R.D., et al., *Using Workflow to Build an Information Management System for a Geographically Distributed Genome Sequence Initiative*, in *Genomics of Plants and Fungi*. 2003, ed. R.A. Prade and H.J. Bohnert, Marcel Dekker, Inc.: New York, NY. p. 359-371.

22. Anyanwu, K., et al., *Healthcare Enterprise Process Development and Integration.* Journal of Research and Practice in Information Technology, Special Issue in Health Knowledge Management, 2003(accepted for publication.).

23. Luo, Z., *Knowledge Sharing, Coordinated Exception Handling, and Intelligent Problem Solving to Support Cross-Organizational Business Processes*, in *Department of Computer Science*. Ph.D. Dissertation, 2000, University of Georgia: Athens, GA. p 171.

24. Kang, M.H., et al. *A Multilevel Secure Workflow Management System*. in *Proceedings of the 11th Conference on Advanced Information Systems Engineering*. 1999. Heidelberg, Germany: Springer-Verlag. p. 271-285

25. CAPA, *Course Approval Process Automation (CAPA).* 1997, LSDIS Lab, Department of Computer Science, University of Georgia: Athens, GA.

26. Leymann, F. and D. Roller, *Production Workflow: Concepts and Techniques*. Upper Saddle River, New Jersey: Prentice-Hall, 2000.

27. Cardoso, J., *Quality of Service and Semantic Composition of Workflows*, in *Department of Computer Science*. Ph.D. Dissertation, 2002, University of Georgia: Athens, GA. p 215.

28. Kumar, K. and J.V. Hillegersberg, *ERP Experiences and Evolution.* Communications of the ACM, 2000. **43**(4): p. 23-26.

29. Iacovou, C.L., *Managing MIS Project Failures: A Crisis Management Perspective.* Ph.D. Dissertation, 1998, University of British Columbia: Vancouver, B.C., Canada.

30. Scott, J.E. and I. Vessey, *Managing Risks in Enterprise Systems Implementations*, in *Communications of the ACM.* 2002. p. 74-81.

31. Warren, J. and V. Lee, *Culture, Organizational Dynamics and Workflow Implementation: Lessons from a Case of Failure*. 2000, University of Illinois at Chicago: Chicago. p. 30.

32. Georgakopoulos, D., M. Hornick, and A. Sheth, *An Overview of Workflow Management: From Process Modeling to Infrastructure for Automation.* Distributed and Parallel Databases, An International Journal, 1995. **3**(2): p. 119-153.

33. Rusinkiewicz, M. and A.P. Sheth, *Specification and Execution of Transactional Workflows*, in *Modern Database Systems: The Object Model, Interoperability, and Beyond*. 1995, ed. W. Kim, Addison-Wesley. p. 592-620.

34. Eder, J. and W. Liebhart. *Workflow Recovery*. in *IFCIS Conference on Cooperative Information Systems.* 1996. Brussels, Belgium. p. 124-134

35. Worah, D. and A.P. Sheth, *Transactions in Transactional Workflows*, in *Advanced Transaction Models and Architectures*. 1997, ed. S. Jajodia and L. Kerschberg, Kluwer Kluwer Academic Publishers. p. 3-34.

36. Sheth, A.P., W.v.d. Aalst, and I.B. Arpinar, *Processes Driving the Networked Economy.* IEEE Concurrency, 1999. **7**(3): p. 18-31.

37. zur Muehlen, M. and R. Allen, *Embedded vs. Autonomous Workflow - Putting Paradigms into Perspective*, in *Excellence in Practice Volume IV: Innovation and Excellence in Workflow and Knowledge Management*. 2000, ed. L. Fischer, Future Startegies: Lighthouse Point, FL. p. 49-58.

38. Hopkins, J. and M. Kessler, *Companies Squander Billions on Tech*, in *USA TODAY.* 2002. p. 1A.

39. Basu, A. and A. Kumar, *Research Commentary: Workflow Management Issues in e-Business.* Information System Research, 2002. **13**(1): p. 1-14.

40. Slater, D., *Costly, Painful and Worth It*, in *CIO Magazine*. 2002.

41. Q-Link, *BPM2002: Market Milestone Report*. 2002, Q-Link Technologies. http://www.qlinktech.com.

42. SeeBeyond, 2002, SeeBeyond. http://www.seebeyond.com/ (see eBusiness integration demo for good overview of EAI type tools and applications)

43. Hagel, J. and J.S. Brown, *Your Next IT Strategy.* Harvard Business Review, 2001. **79**(9): p. 105-113.

44. SOAP, *Simple Object Access Protocol*, 2002. http://www.w3.org/TR/SOAP/

45. Fensel, D. and C. Bussler, *The Web Service Modeling Framework*, 2002, Vrije Universiteit Amsterdam (VU) and Oracle Corporation. http://www.cs.vu.nl/~dieter/ftp/paper/wsmf.pdf

46. German Shegalov, Michael Gillmann, and G. Weikum, *XML-enabled workflow management for e-services across heterogeneous platforms.* The VLDB Joumal, 2001. **10**: p. 91-103.
47. Chen, Q., et al. *Dynamic-Agents, Workflow and XML for E-Commerce Automation.* in *EC-Web.* 2000. p. 314-323
48. Leymann, F., D. Roller, and M.T. Schmidt, *Web Services and Business Process Management.* IBM Systems Journal, 2002. **41**(2): p. 198-211.
49. Cardoso, J., C. Bussler, and A. Sheth. *Tutorial: Semantic Web Services and Processes: Semantic Composition and Quality of Service.* in *International Federated Conferences: DOA/ODBASE/CooPIS 2002.* 2002. Irvine, CA
50. Overby, S., *This Could Be the Start of Something Small*, in *CIO Magazine.* 2003, http://www.cio.com/archive/021503/erp.html.