Increasing Experimental Cooperation in Bioinformatics using Semantic Web Services

1. Introduction

As pointed out by Licoln Stein (REF), access to bioinformatics resources (e.g., databases and analysis software) needs to be made more universal through the use of Web services. A Web service is a universally accessible software component deployed on the Web. Such a software component is described by an interface listing the collection of operations that can be performed on it. Web services (unlike earlier distributed computing models) are suitable for integrating bioinformatics resources for the following reasons. They are XML based, allowing them to address data encoding problems that existed in earlier models. They support XML based distributed computing using SOAP (Simple Object Access Protocol) and can be accessed using ubiquitous transport protocols like HTTP and SMTP.

While commercial development of Web services is progressing rapidly, significant research is ongoing to make Web services more useful. Currently, Web services are described in registries (e.g., UDDI registries) using keywords. Languages and techniques are being developed to more precisely define the meaning and purpose of a Web service. Not only will help humans find a more appropriate Web service, it will allow automated tools to do so thus facilitating automated Web service composition. These efforts to bring more meaning to the description of Web services parallels current work on the Semantic Web and are referred to as semantic Web services (REF).

The University of Georgia's Genetics and Computer Science Departments propose to develop a prototype system to demonstrate how both Web services and semantic Web services can be used to provide universal access and increased cooperation between bioinformatics laboratories. Currently, Web services are based mainly on commercial efforts. Languages and protocols include SOAP, WSDL, WSFL and XLANG (REFS). Semantic Web services promise to make the search for and use of information even easier. This will help researchers more quickly and reliably find information relevant to their research. Semantic Web service technology is largely being developed by the academic community (REF) and includes languages and protocols such as DAML+OIL, OWL and DAML+S (REFS).

Web services hallmark is their universal accessibility which is superior to that of early distributed component technologies for the following reasons: The components are available on the Web, service descriptions are advertised and service requests are made in a language independent form (*i.e.*, some dialect of XML such as SOAP). Bionformatics computational services can be made more accessible by turning them into Web services. Examples of such services include <...>. The same can be done for information stored in databases such as GenBank, <...>. Rather than encoding messages in special formats for each information source, a more uniform style access based on XML request/query messages can be used. Responses will also be in form of XML messages. Using XML as a common substrate opens the door for greater automation. Less custom coding of format convertors will be needed. More importantly, Web service composition will permit the creation of Web processes to accomplish larger experimental goals. Web processes provide at least three advantages over the current day practice of writing Perl scripts: (i) they can be developed automatically or at least in computer−aided way, (ii) they are more maintainable and (iii) they can work on a larger scale since they are Web based.

In general, the emerging Semantic Web and Web services are seen as the new infrastructure for tomorrow's e−businesses and scientific applications. The type of interactions between today's organizations is dramatically and rapidly changing. With the advent of global economies and dynamic markets, organizations started to rely heavily on the Internet as a medium of communication to carry their businesses, increasingly using e−commerce and Business−to−Business (B2B) activities. The coordination and management of interactions become especially important when organizations establish link of cooperation to form *dynamic trading processes* [Sheth et al., 99], *dynamic value chains* [Lee and Whang, 01], *virtual organizations and virtual web organizations* [Ulrich, 01]. In dynamic markets and environments, selection of most appropriate Web services and composition of a global process utilizing the selected Web services becomes a challenge, since the Web services need to be dynamically identified, selected and integrated to form a process that meets its objectives. These same advantages can apply to the scientific community as well.

The objective of this proposed work is to investigate and develop techniques and technologies for making bioinformatics research more cooperative and productive. Web services will be used for accessing databases and carrying out computations. A novel aspect of this work is that it will adapt workflow technology to the problem of composing Web services into Web processes. Using semantics and quality of service metrics to compose Web service allows for more effective, meaningful and efficient composition. This will allow organizations to gain the benefit of semantically integrating a Web service into dynamic processes, characterize and quantify processes based on their quality of service, dynamically adapt processes to reflect organizational strategies and goals.

2. Bioinformatics Web Services

2.1.Issues in Web Services

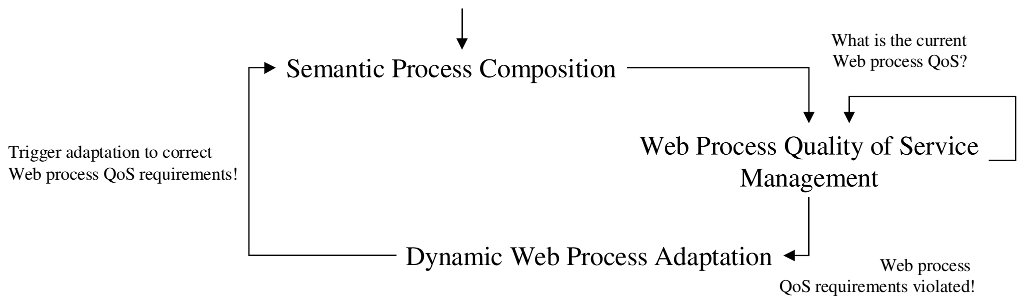
- **Discovery.** Web services must be advertised or registered in order to be used. Both service providers as well as service requesters need to access a registry/repository to register/lookup Web services. These registries/repositories, which will be located all over the Web, may be located either using a known Web address or through a discovery protocol.

- **Advertisement.** Web service providers will advertise their services in one or more registries/repositories. The information contained will describe the service, as well as how to invoke it. The greater the semantic content of this information, the more useful it will be.
- **Search.** A client will attempt to find a suitable Web service by searching one or more registries/repositories. Although searching based on types and keywords is useful, as Web services proliferate, greater semantic content, for example, based on ontologies, will be needed. Getting thousands of hits on a service request is hardly desirable.
- **Composition.** A critical issue that needs to be addressed is how to compose a Web process. Different laboratories operate in different ways, have different structures, use different information models and may use different domain specific vocabularies. The heterogeneity between laboratories/organizations makes the design and composition of Web processes a complex task. In order to establish the set of interactions to be taken among organizations their interface needs to be not only syntactically understood, but also semantically.
- **Scheduling/Monitoring.** If Web services are chained together into a Web process, many other issues naturally arise. The enactment or execution of the process needs to be controlled, *e.g.*, when should a service be invoked, what happens if it fails, how is information appropriately routed to services, are transaction semantics required, *etc.* In addition, if a large–scale Web process is not monitored, how can one possibly know whether it has executed correctly? In essence, the problems (and solutions) faced by the workflow community during the last decade will be faced by Web processes.
- **Adaptation.** A challenging issue that needs to be addressed is how can a Web process be dynamically adapted or evolved. Processes operating in dynamic environments need to be managed to account for changes that occur in goals, the environment, technology, or participating organizations. When changes are significant and impact a process, it is advantageous to identify adaptive procedures that can be applied to process enactment in order to realign their realization with new environments.
- **Quality of Service (QoS).** Quality metrics (*e.g.*, service time, cost, reliability, fidelity of results [Cardoso02]) will be useful in many regards. A Web service can be selected based on its a service profile. Is it fast enough, inexpensive enough, likely to produce an acceptable result and not prone to failure? Furthermore quality metrics will serve as the basis for adaptation. During process execution the QoS needs to be monitored and checked against predefined and expected values. When deviations are identified, dynamic adaptation mechanisms may be triggered.
- **Verification and Simulation.** Under the old adage of "try before you buy", it is important to check/test a Web process or adaptation before it is actually invoked or applied. Once a process (or adaptation) is specified, its structure and execution properties will be analyzed. In addition, estimates of quality metrics will be provided through simulation.
- **Dynamic Composition.** Traditionally, processes are well–defined and represented using a process/workflow schema, *i.e.,* a blueprint of the process. However, with the

emergence of Internet technologies such as e–commerce, B2B, and Web services, there is the need to dynamically compose process schema based on the available resources present in the Internet. Processes are no longer pre–constructed based on organizational knowledge, but are dynamically composed on the Internet from available services.

3. Research Challenges

The proposed research seeks to investigate both fundamental and applicative issues that



arise from the need to support open, multi–organizational, Web–based, and highly dynamic processes. In the future, laboratories/organizations will require the enhanced support for information retrieval as well as laboratory processes that can be provided by the Web services. In this section, research challenges for each of the issues presented in section 2 will be addressed.

Figure 1 – Fundamental issues in Semantic Process Composition

4.1. Discovery of Registeries/Repositories

How can semantic *discovery* of registries and repositories be achieved using ontologies as a discovery criterion?

4.2. Advertisement of Web Services

4.3. Searching for Appropriate Web Services

How can registries and repositories be *searched* in order to find Web services/tasks candidates? How can results from a search be *interpreted and ranked*? Can the ranking of potential Web services be made based on *syntactic and semantic match*? Semantic match is an important aspect of Web service selection. But new mechanisms new to be developed to compare ontologies and find mapping between ontologies.

4.4.Composition of Web Services

The creation of dynamic Web processes reflects a new type of business process structures required by new emerging organizations. The composition of Web processes becomes a challenge when heterogeneous, autonomous and distributed sub–processes are involved. The need to embody emergent technologies, such as Web services, into organization's processes raise fundamental issues that need to be investigated and create a new set of requirements. Coordination systems must be able to model, represent, and manage Web services interactions. The challenges are not only related with data and syntactic integration, but especially with the semantics of services composition. Key issues that must be addressed include:

- *Specification*. Which new structures need to be developed to allow the integration and interoperation of Web services? We propose to study issues that go beyond what is part of current relevant standards activity (*e.g.*, WSDL [<REF>]) to study specification of process semantics as part of service request documents. How can conversation protocols (*e.g.*, WSCL [<REF>]) be used and extended to increase the degree of interoperation between Web services? How can current resource/process description frameworks and ontology languages such as DAML–S [<REF>] be used to increase process specification richness? Will this additional richness enhance the degree of automation when semantic process composition is carried out?
- *Composition*. Composition techniques and methodologies need to applied at two levels. At the first level, two Web services need to be composed so that the output of the first feeds the second. This requires some semantic relationship between the two as well as a means transforming the data input (*e.g.*, SOAP message [<REF>]) to the second Web service. At the second level, entire processes must be designed or otherwise created. The two primary ways to achieve this are graphical design tools or planning techniques used in artificial intelligence [McIl01]. Because of the LSDIS Lab's extensive experience with the former in the context of workflow technology, this will be main focus of this proposed work. Ontologies will be used to facilitate the resolution of data and schematic–level conflicts among disparate Web services. During the composition how can the process QoS be checked for acceptable metrics and how can alternatives be generated and presented to the user?

We plan to base our research in this area on our experience in specifying executable specifications of processes for distributed execution environment [Kochut et al., 99], in multi–organizational processes [Kang et al., 99][Luo, 01], and our work on specifying semantic information requests with user defined functions and complex inter–ontological relationships [Thacker et al, 01], and recent work on distributed knowledge sharing environment with support for searching of relevant ontologies [Arumugam et al, 01] and personalization [Kim, 01].

4.5. Scheduling and Monitoring of Web Processes

Theoretically, once a Web process is completely specified, the rest is just developing execution strategies, mechanisms and environments for actually enacting the process. Again, there are two major ways to do this, multi–agent systems and enactment systems evolved from workflow technology. Because of the LSDIS Lab's experience with the latter and the desire to create effective and efficient systems as a near term goal, this proposal will focus on that approach. Enactment issues include control flow, data flow, scheduling options, exception handling, fault tolerance, transactional requirements, and monitoring.

4.6. Web Process Adaptation

In modern organizations, process change has become ever present [Hammer and Campy, 93]. There is a constant need for continuous process design improvement. Dynamic trading Web processes, dynamic value chains and virtual organization have highly dynamic and evolving structures. The development of methods and mechanisms to support these organizational models require research in the following areas.

§ *Detection*. When should Web Services processes be dynamically adapted? As a result of thrown exception? When quality of service falls below bottom threshold? When anomalies are observed from the monitor? When data mining agent identifies invalid data patterns (mining the monitors' log file)?

§ *Change Analysis*. When detection mechanisms trigger the need for adaptation, how can alternatives be generated? How can adaptive patterns be use to dynamically change the structure of Web process schema and Web process instances? Are adaptation patterns a good solution to modify processes structurally? How to filter only well–defined changes? How to rank alternative candidates? Can the selection be fully automated, or user involvement is ultimately needed?

§ *Change Introduction*. What are the requirements for adaptive Web processes beyond static processes? When can the adaptation be made to the currently running Web process without violating its consistency? What mechanisms, features and functionality the runtime enactment needs to support for carrying out the adaptation of Web processes? What are the side effects of process adaptation on resource states and data consistency? How Web processes adaptation can be dynamically monitored? What modeling features and functionality need to be provided by coordination systems to support adaptive Web processes?

Process adoption has been recognized to be a fundamental and complex task. Our work in this area has been divided in two sections. In one front we approach the problem from the theoretical point of view [Aalst, 99], targeting issues such as correctness and liveness. On the other front we approach the problem from the system design point of view [Chen, 00]. We are also interested in and conducting work on how adaptation mechanisms can be coupled with simulation [Miller et al, 02] and excepting handling [Kochut et al, 99][Luo, 00].

4.7. Process Quality of Service

Hammer and Campy [93] identify intense competition has one factor that characterizes modern businesses. One important factor that determines competitive advantage is the quality of services organizations render to customers. Web Process quality of service is indispensable for organizations striving to achieve a higher degree of competitiveness. In contrast to the QoS measurements of centralized systems, the QoS metrics of heterogeneous, autonomous and distributed coordination systems and components have new requirements.

§ *Specification*. How can quality of service be specified? What are the dimensions that need to be include in a process quality of service (pQoS) model? How should such a quality model be represented so Web Services could use it? Since the integration of different Web processes requires the integration of possibly different QoS models, are translations mechanisms needed? If yes, how can global models be constructed?

§ *Computation*. What methods can be used to predict process quality of service? How can simulation be used to accurately predict pQoS? Which analytical methods can be used? And iterative methods?

§ *Monitoring*. What kind of monitoring tools needs to be developed? Do the distributed nature, the heterogeneity, and the autonomy of Web services affect the architectures of monitoring tools?

§ *Control*. What mechanisms need to be developed to control Web processes, in response to unsatisfactory quality of service metrics? Can dynamic process adaptation mechanisms and exception handling technology be used?

Our work on designing and modeling various business processes [Hall et al., 01][Luo, 00][Kang et al., 99][CAPA] for distinct market segments made us realize of the importance of Quality of Service has an integral part of processes. We are currently conducting research in the specification and computation of process quality of service [Miller et al., 02][Cardoso et al., 01].

4.8. Verification and Simulation of Web Processes

Verification **...**

Once a process or specification is verified, further checking of the process can be obtained through simulation [Miller02]. The simulation will mimic the execution of the actual process and allow estimates of cost, time, reliability and fidelity to be produced which can be used in assessing the suitability of the process. Simulation will also be very useful in assessing the desirability of adaptive changes. In this project, Web process specifications will be converted into simulation specifications using XSLT. These specifications will be used to generate JSIM [Nair96, Miller97] simulation models to produce the estimates as well as animate the process.

5. Research Plan

It is clear from the rapidly emerging research efforts to automate the use of Web services that this is becoming a key future research direction. Selection of appropriate Web services as well as composition of Web services to form Web processes will become more and more important as the number of Web services available on the Internet proliferate and the need for composite services increases.

The LSDIS Lab is well–positioned to make contributions in this area. The challenges and objective proposed in this document constitute a continuation of our past and current research. We will build on our previous work on Adaptive Workflow, Simulation and the Semantic Web.

In the area of Adaptive Workflow, we plan to base our research on our experience in specifying executable specifications of processes for distributed execution environment [Kochut99], in multi–organizational processes [Kang99, Luo01]. The foundation of this work is represented by OrbWork, an enactment service for the METEOR WfMS, which supports several types of well–defined dynamic changes [Chen00].

Several research papers on or extending this prototype include flexible specification driven exception handling [Luo01], quality of service contracts [Cardoso02], multi–organizational workflows [Kang99, Sheth00], and use of repositories for maintaining meta–data about workflows and tasks [Budak01].

More specifically with respect to adaptation, we have worked on the effects of process exceptions and how it can triggers dynamic adaptation [Luo, 01]. Since adaptation mechanisms rely on dynamic changes to actually change processes, we have developed a taxonomy and an API to dynamically change processes at runtime [Chen, 00]. To successfully apply adaptation operations it is important to guarantee that the changes made do not put the process in an inconsistent state. Therefore, we have investigated safety and correctness considerations on dynamic changes [Yan98, Aalst et al., 99]. Currently, we are developing techniques for adaptation feasibility analysis via analytic models [Cardoso02] as well as simulation models [Miller95, Miller99, Miller02].

This body of work has given us considerable experience in composing processes/workflows. The scope and dynamicity of the problem is clearly greater with Web services than traditional workflows, but our work, especially in the areas of adaptive workflow and multi−organization workflow, has been pushing us this direction already.

As the scale and scope of the problem increases with Web service providers and registries cropping up all over the Internet, Web services will face the same problem now encountered by the document/information side of the Internet. Mr. Burns−Lee's proposed solution, the Semantic Web, will become equally important for emerging Web services. The LSDIS Lab has considerable experience with using semantics for searching the Web [REFS??]. Recent work involves using DAML ontologies in order to much more precisely search the Web [Madhan02]. Along these same lines we propose to utilize DAML−S [REF], which is a Web Service Markup Language built on top of DAML and OIL.

Our specific research plan is as follows:

- Upgrade our current METEOR XML repository to store Web service specifications in WSDL/DAML−S. WSDL is for compatibility with current practice so our repository can serve as UDDI registry. Comparison of this baseline with the proposed approach of using DAML−S will allow quantifiable differences to be studied.
- The current METEOR workflow designer will be modified for the domain of Web service composition. It will be based on the composition constructs in the DAML−S process ontology.
- The METEOR Web process designer will interoperate with a Web service search engine that finds Web services in registries or repositories. Semantics will be used to make this more effective. In response to a "Request for Service" An ordered list of Web services will be returned. In addition, a "Conversation Matrix" can be returned to show how these Web services could potentially be linked together.
- Using the METEOR Web process designer and the METEOR Web service search engine, one may quickly build or adapt an Web process. Before these changes are allowed to affect the runtime, they need to be thoroughly checked. First, the structural correctness of the Web process will be checked. Then, using quality of service (QoS) models for each Web service, composite QoS metrics will be calculated to determine if user requirements can be met.
- As a optional final step in assessing the suitability of the proposed change, a simulation of the Web process can be carried out. To facilitate this, the JSIM simulation system [Nair96, Miller97, Miller98, Miller00] will be modified to accept simulation model specifications in DAML−S so that Web process designs can be used as the basis for simulation and animation.
- Once a change has been selected, it needs to be manifest in the runtime system. This will require enhancements to the OrbWork runtime system. <...>
- Since adaptation is an integral part of our proposed work, it is important to have something that triggers adaptation. The principle triggers under our approach include

thrown exceptions as well as unsatisfactory conditions observable from the Web process monitor. The observer could be a human or an agent. To support this will require upgrading the current METEOR monitor.

6. Past and Ongoing Research

The challenges and objective proposed in this document constitute a continuation of our past and current research. In the area of *semantics* we have just concluded the development a Peer–to–Peer (P2P) infrastructure with capabilities to find relevant sets of ontologies, facilitating reuse of existing ontologies to create additional ontologies, and advertising the resulting ontologies [Madhan et al., 01]. In the area of *process composition* we have developed a comprehensive model – MLS workflow – suitable for specifying multi–organizational workflows [Kang et al 99]. We also have developed a repository for reuse of process components [ref]. With respect to automatic *adaptation*, we have worked on the effects of process exceptions and how it can triggers dynamic adaptation [Luo, 01]. Since adaptation mechanisms rely on dynamic changes to actually change processes, we have developed a taxonomy and an API to dynamically change processes at runtime [Chen, 00]. To successfully apply adaptation operations it is indispensable to guarantee that the changes made do not put the process in an inconsistent state. Therefore, we have investigated safety and correctness consideration on dynamic changes [Aalst et al., 99]. Currently, we have investigating the relationship of *quality of service* (QoS) violations and adaptation [Cardoso et al., 01]. Concurrently, we also focus on the major role simulation modeling can play to assist process adaptation [Miller et al 99, Miller et al 02]

7. Technical Approach

Our project, involves the development of new algorithms, models, and a proof–of–concept prototype to make possible the semantics–based selection, composition and coordination of heterogeneous, autonomous and distributed Web services, the specification of quality of services, and dynamic adaptation of processes. To define the the architecture and demonstrate how it can be implemented we exploit and significantly extend Internet standards, use generic models to define quality of service, and employ process patterns to describe adaptive procedures. We believe that these new techniques introduced provide an indispensable basis for the development of new functionality valuable for modern and future organization. We will show how processes can be designed to include Web services, and how they can be dynamically adapted, based on quality of service specifications. The key components of our work are depicted in Figure 2.

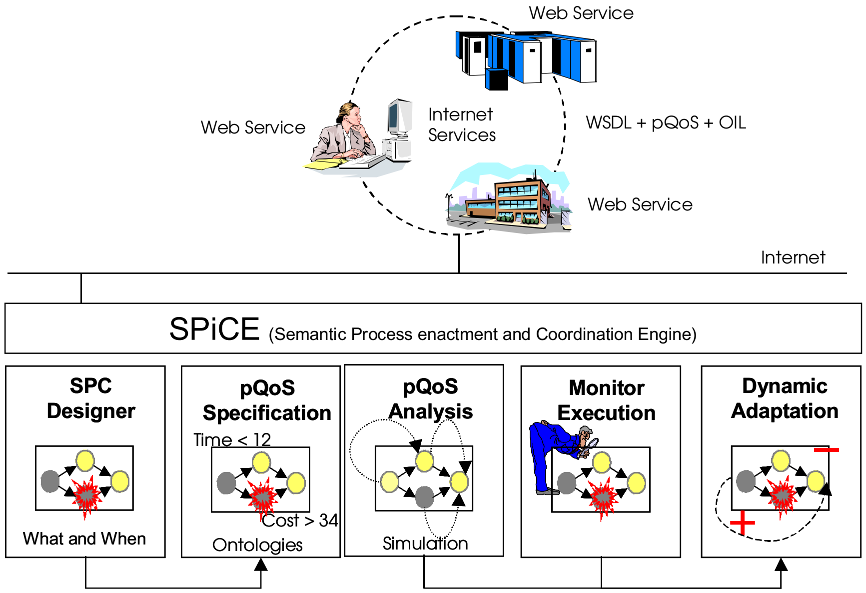


Figure 2 – System Architecture

The runtime system consists of a trading middleware (SPiCE), a SPC designer, process quality of service specification and analysis tools, a monitor, and a dynamic adaptation component.

This trading middleware coordinates and manages the Web Services described in a process. At run time, and based on semantic properties, SPiCE discovers, lookups (using Syntactic and Semantic Locators), and binds Web Services to the process. Additionally, the trading middleware acts as a *brokering mechanism* between pQoS requirements of trading applications and pQoS capabilities of individual services. The SPC design tool allows application designers to construct dynamic Web trading process. In other words, the designer can specify what services are required to achieve higher–level goals and how they work together (i.e., control logic and data flow among nodes). Also pQoS requirements can be expressed through the design tool. Analytical and simulation tools [Miller et al., 01] for testing critical properties will be included in the system. We will devise a suitable specification language that will describe Web Services semantically. We will base the description syntax on extended WSDL (Web Services Description Language). However, the WSDL, which has an XML format for describing network services, provides only syntactic information about the components. In order to further specify the type of service offered we plan to adapt the DARPA Agent Markup Language and Ontology Inference Layer (OIL) specification (DAML+OIL) [DAML+OIL]. Ontologies form important aspects of a specification to support semantic interoperability and knowledge sharing. RDF, DAML+OIL and such W3C advocate and provide basic facility to define and associate ontologies with resources (or components and services in our case). An example of our extensive research in developing multi–ontology system appears in [Kashyap and Sheth 98, Mena et al. 00].

The semantics–based selection, composition, and integration and coordination of Web Services cannot be successfully accomplished without a deep study of the Semantic Web [Berners–Lee, 99] phenomena. We have seen a steady progress from syntax, to representation and structure, and to semantics [Sheth, 98], in the ways we approach and solve the challenges of finding, integrating and using information of diverse types and from diverse sources. Integration and coordination are related to two fundamental issues. The first issue concerns the semantic dynamic identification and reconfiguration of process components and the subsequent resolution of the schematic differences between the service to be integrated and the system integrator to create dynamic trading processes. The second important issue relates to the access and use of Web services without prior knowledge of their behavior, operation or function. Achieving these goals in the areas of Web services is much more complex than in traditional information systems due to the high autonomy, high distribution, and high heterogeneity of the services involved, and due to the semi–structured and unstructured nature of the data implicate. The mechanism of service description is one of the key elements for Web Services. The use of ontologies and semantic mediators/agents facilitate the interoperability among Web service. One of the important research objectives is how to automatically identify and dynamically resolve various data and schematic–level conflicts among disparate Web services.

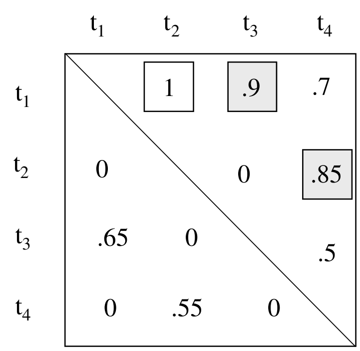


Figure 3 – Conversation Matrix

Figure 3 represents a conversation matrix used to assist a designer to semantically compose a process. When a particular activity or task (t_i) needs to be added to a process, the SPC engine search through thousands of Web services. The services found are then filtered and ranked, based on syntactic and semantic criteria. Only the most adequate Web services, from the point of view of quality of service, are shown to the designer using the conversation matrix. The matrix indicates for each task, which can be selected and added to the process, a fitness index based on syntactic and semantic properties.

The direction of current work on defining Web services and semantic standards (WSDL (Web Services Description Language) [WSDL], RDF (Resource Description Framework) [RDF], RDFS (RDF Schema) [Brickley and Guha, 00], DAML (DARPA Agent Markup Language) and OIL (Ontology Inference Layer) [DAML] needs to be studied to identify possible extensions needed. Web Process integration and coordination often require conversation protocols. Current protocols, such as WSCL (Web Services Conversation Language) [WSCL], only model syntactic interactions between services. Here again, semantics is our main concern. We recognize the importance of enhancing and extending such protocols to support semantic conversations, which reflect more accurately the reality. Additionally, we believe that conversations can be represented using workflow technology.

To archive Web process coordination we choose to adopt the most advanced technological progresses in this area: workflow management technology. Workflow systems have been used to successfully support and manage business processes for more than a decade now. Current workflow systems provide a generic solution to the coordination of activities. Enhancements and extension of workflow technologies constitute the best candidate to develop a reference architecture for interoperable e–commerce applications [Yang and Papazoglou, 138] and to provide answers to Web service coordination requirements. The development of new workflow technologies needs to be in the direction of "networked economy" [Sheth et al., 99], and "internet–mediated workflow, which will most likely be the single most important technology of the early 21st century" [Petrie and Sarin, 00].

7.1 Process Quality of Service

We plan to develop a new language to specify both Web process Quality of Service (pQoS) capabilities of Web Services and pQoS requirements of applications. Web Services will be described in WSDL (Web Services Description Language) [WSDL] and RDF (Resource Description Framework) [RDF] in conjunction with pQoS measures. We intend to adapt the DARPA Agent Markup Language and Ontology Inference Layer (OIL) specification (DAML+OIL) [DAML+OIL]. Together, the WSDL and DAML+OIL specifications would provide the *functional description* of the service. pQoS will provide extra characteristics of Web Services.

The use of ontologies to describe pQoS allows the inference of knowledge relative to quality of service metrics. Relationships between QoS terms permit a better description and better understanding of Web Service. Additionally, the Web process Quality of Service model will include:

- Development of quality of service models to allow the *specification* of Web process quality of service (pQoS) metrics. A pQoS model permits the specification of services and products to be delivered, deadlines, quality of products, cost of service, etc. Processes can be quantifiably characterized according to their QoS. The extension of new standard, such as the WSDL and RDF languages can be used.
- Development of algorithms to *predict* the quality of service of a Web process schema before execution. Before the celebration of a contract, suppliers must ensure that the processes to be executed indeed exhibit the quality of service, which is requested by the customer. Therefore, it is indispensable to compute before the actual realization of a process its pQoS. Algorithms must compute directed cyclic graphs attributes (corresponding to QoS metrics), which have special node with input and output logic. Algorithm must account for conditional paths, as well as for parallel paths.
- Adequate tools need to be developed to *monitor* the quality of service associated with Web processes process execution. Monitoring Agents can monitor pQoS and react to undesired process execution behaviors. The observation of undesired pQoS metrics triggers a dynamic process adaptation. Developing Web service–monitoring tool requires a new set of expertise, since the component to monitor are dynamically selected, high distributed, heterogeneous and autonomous.

- Development of mechanisms to *control* the quality of service of Web processes. When pQoS exhibits undesired measurements, dynamic mechanisms must be activated to trigger process adaptation. Let's consider the following example; process instances are running correctly when a task fails. The Web Service "A" stops it's processing because of an algorithmic problem. In consequence, pQoS specifications are no longer satisfied and the process coordination system raises an alert. The faulty Web service has to be replaced by an equivalent Web service to restore the soundness of the system.
- Development of translation mechanisms to integrate different QoS models. And development of tools to carry analytical analysis and simulation of pQoS metrics [Miller at al., 02].
- Development of Process Analysis Agents that carry pQoS analysis and perform "what–if" analysis. The use of analysis agents allows users to indicate specific analysis to be carried out in order to better understand the status of current executing process.

7.2 Dynamic Process Adaptation

The trading middleware manages the Web Services in Web processes and dynamically adapt processes, executing operations such as the reassignment, modification, deletion and addition of services if necessary.

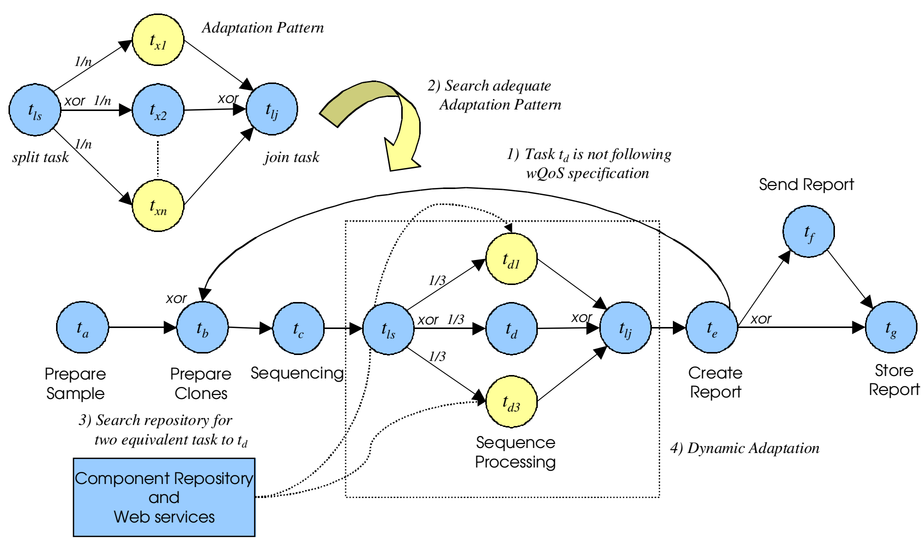


Figure 4 – Dynamic Process Adaptation

These operations are triggered in response to pQoS violations. Dynamic Process Adaptation techniques include:

- Interfaces and mechanisms need to be developed to dynamically change the structure of Web process schema and instances. Process structure properties such as likeness, soundness and correctness need to be carefully addressed, to ensure that adaptation pattern do not create inconsistent processes. Especially adaptation patterns developed by the user.

- A process adaptation pattern library needs to be developed to store and share adaptation patterns. This library needs to include algorithms and mechanisms to automatically search and select adaptation patterns based on pQoS metrics and semantics.
- Adaptation patterns that are retrieved from the pattern library need to be classified categorized, and ranked according to the type of structural and functional changes they make to a process.
- Adaptation of a process can be classified into structural and functional. Structural adaptation comprises all changes of the process topology. Functional adaptation comprises the modification of component's functionality. The use of graph theory is relevant to identify the impact on process structures based on the type of adaptation taken.
- One aspect of dynamic adaptation that has received little attention is the participation that is required of the individual components when a dynamic adaptation change is needed. Component participation includes divulging syntactic and semantic information, and using information from the system to change internal state. In this context ontologies can be used to represent not only components interfaces, but also infrastructure interfaces. The use of ontologies will facilitate the aggregation of components with system's infrastructure.

8. Background

Our past research gave us a strong set of core competencies in process specification, verification, enactment, monitoring, applications deployment, integration and interoperability of processes. Our core competencies have been gain with the development of two enactment services for workflow management. Our infrastructure, the METEOR system, and specifically its fully distributed enactment service – OrbWork [Kochut et al., 99] – and Web based enactment service WebWork [Miller et al., 98] [Miller et al., 97][Miller et al., 96] – has been deployed to various domains, such as bio–informatics [Hall et al., 01], healthcare [Anyanwu et al., 00], telecommunications [Luo, 00], military [Kang et al., 99], and school administration [CAPA]. Our work has allowed us to identify a set of major problems that need to be answered to allow modern organizations to increase their competitiveness in global markets. Additionally we have carried extensive research on the correctness of adaptation [Aalst, 99] and on simulation [Miller et al., 99]. We are now interested in how workflow systems, quality of service and simulation can be coupled together [Miller et al, 02].

We have also performed extensive past research in semantic interoperability [Sheth 98], various layers of semantic networking (c.f. Figure 1 of [Kashyap 01]), and Semantic Web (Berners–Lee 99]. In particular, we have discussed issues of semantic meta–model and object interoperability in [Kashyap and Sheth, 98, Kashyap and Sheth, 00], ontology interoperation [Mena et al., 00a] and corresponding issues of information loss with multi–ontology query processing and ontology interoperation [Mena et al 00b], and information correlation and complex relationships in Semantic Web in [Sheth et al., 01, Thacker et al 01]. Recently, we have investigated issues in developing distributed knowledge sharing environment over a P2P type infrastructure, and associated services for semantic searching relevant ontologies and personalizing knowledge spaces

[Arumugan et al 01, Kim 01]. Nevertheless, ontology interoperation, information loss in multi−ontology query processing and applications of ontology interoperation techniques in business process modeling and integration remain areas of further research [Kashyap 01].

Recent work, much of it funded by DARPA−ITO through the Quorum program, is extending QoS concepts and mechanisms to higher semantic levels to allow the definition, measurement, and control of the quality of service delivered by services and complete applications. Additionally, qualities of service dimensions have been studies in several area of computing ([Gray and Reuter, 93], [Cristian, 91], [Reibman and Veeraraghavan, 91], [Birman, 86], [Maffeis, 95], [Littlewood, 91], [Frolund and Koistinen, 98], [Hiltunen et al., 00]). However, little work has been done in terms of QoS of Web Services that we proposed in this proposal.

The ADEPT project [Hensinger et al., 00] investigate new technologies that allow workflow instance deviation from the pre−modeled process template, such as skipping of steps, going back to previous steps, inserting new steps, etc., in a secure and safe way. Müller et al. [99] investigate the potential automation of knowledge bases to dynamically adapt the control and data flow of running instances. In their work workflow plans are fully constructed at run time using knowledge bases. Placeholder nodes are used to represents under specified sub−workflows. When a sub−workflow has to be determined at run−time to refine a placeholder node, a so−called refinement agent identifies an appropriate plan in the knowledge base and transforms it to an operational workflow notation. Han et al. [98] classifies various types of workflow adaptation, and discusses potential mechanisms for achieving adaptive workflow management. The developments are based on dynamic composition of workflow resources, including workflow models, as well as a strong association between exception handling and adaptive workflow management.

Recently, the need for constructing new applications across multiple domains using existing applications has received much attention in the context of business−to−business applications. Even though there are many efforts to facilitate such needs, such as enterprise application integration (EAI) and cross−organizational workflow management system [Grefen et al., 01], security and survivability of critical applications have been largely overlooked.

9. Standards and Technology Infrastructure

In this section, we briefly review recent commercial moves towards services based architecture. For brevity, we do not review the relevant specifications and standards (XML, RDF, RDFS, SOAP, UDDI, DAML+OIL, WSDL, WSFL, XLANG and DAML−S) that are the basis of component−based middleware upon which we build Darwin.

Web services are emerging as an important application for the Semantic Web, and a hotbed of activity in industry [Hendler and Mcllraith, 01]. Current, Web Services architectures give an indispensable infrastructure to deploy Web applications. Existing solutions provide universal communication using loosely coupled connections, and vendor−, platform−, and language−independent protocols. Web Services can be developed using any programming language and can be deployed on any platform. Web Services can communicate because they all use the same protocol (XML) to describe their interfaces and to encode their messages. But XML by itself does not ensure effortless communication. The applications need standard formats and protocols that allow them to properly interpret the XML. Hence new XML–based technologies are emerging for Web Services, namely SOAP, WSDL and UDDI. These three technologies are complementary and fundamental to support the Web Service concept. While, WSDL defines a standard mechanism to describe a Web Service, UDDI provides a standard mechanism to register and discover Web Services, and finally SOAP defines a standard communications protocol for Web Services.

In this area, our work target the research on how Internet protocols, such as SOAP, WSDL and UDDI, can be extend to allow the support of applications and system that better meet real world requirements, i.e. Semantic Process Integration and Coordination, Process Quality of Service and Dynamic Process Adaptation. We do not target the development of architectures or infrastructure to support Web Services. Since basic Web Service architectures and infrastructure already exist, our goal is to we wish to extend them.

The commercial world is moving to service–based infrastructure and solutions, which the proposed work seeks to extend with Semantics, Dynamic Adaptation and Quality of Service properties. Below we provide a very brief discussion of some commercial systems as well as research work based on service paradigm:

- Propel Platform Development Team [Carey, 01] focuses on a scalable infrastructure for advanced e–services.
- The Propel team and Anil Nori et al. [01] discuss requirements for a comprehensive e–service platform and how the developed system architectures meet them.
- Tavant team [Narayanan 01] studies the specific aspect of integrating distribution channels of manufacturers from the viewpoint of an application service provider.
- Fabio Casati and Ming–Chien Shan from HP Labs [Casati, 01] and BizTalk Server 2000 by Bimal Mehta et al. [Mehta, 01] focus on the process–oriented dimension of e–services and discuss how workflow technology contributes to current solutions.
- Vassilis Christophides [Christophides et al., 01] from Bell Labs is studying on workflow mediation using XML–based vortex architecture.
- Bell Labs work and the WISE approach by Amaia Lazcano et al. [01] emphasize process based e–commerce. In their papers, they review ongoing research efforts towards more flexible, interoperable, and highly dependable workflows in an e–service environment.
- The CrossFlow [Grefen et al., 01] is a multi–national research project on cooperation in virtual enterprises. They use a contract mechanism for the service outsourcing and integration.
- Kraiss et al. [01] discusses the importance of performance guarantees in a banking environment and a mathematical approach for appropriate system configuration.
- ObjectGlobe [Braumandl et al., 01] is an open distributed query processing system, and it extends the notion of e–services into Internet–based infrastructure for highly distributed, global querying.

- Sheth et al. [99] discuss three evolving architectures for multi–organizational processes in the increasingly networked economy.

Additional commercial efforts in service–based architecture include .NET and Bowstreet.

10.Reference

[Aalst, 99] W.M.P. van der Aalst, Generic Workflow Models: How to Handle Dynamic Change and Capture Management Information, *Proceedings of the Fourth IFCIS International Conference on Cooperative Information Systems (CoopIS'99)*, Edinburgh, Scotland, September 1999: 115–126. (Technical Report UGA–CS–TR–99–01, Department of Computer Science, Athens, University of Georgia, USA, 1999.)

[Anyanwu et al., 00] Kemafor Anyanwu, Amit P. Sheth, John A. Miller, Krys J. Kochut and Ketan Bhukhanwala. "Healthcare Enterprise Process Development and Integration." Journal of Systems Integration: An International Journal (JSI) Kluwer Academic Publishers, 2000.

[Arumugam et al 2001] M. Arumugam, A. Sheth and I.B. Arpinar, "The Peer–to–Peer Semantic Web: A Distributed Environment for Sharing Semantic Knowledge on the Web," Technical Report, LSDIS Lab, University of Georgia (submitted for publication), November 2001.

[Arumugam et al, 01] Madhan Arumugam, Amit Sheth, and I. Budak Arpinar "The Peer–to–Peer Semantic Web: A Distributed Environment for Sharing Semantic Knowledge on the Web." (Submitted), 2001.

[Attie et al 96] P. Attie, M. Singh, E. Emerson, A. Sheth, and M. Rusinkiewicz, "Scheduling Workflows by Enforcing Intertask Dependencies", Distributed Systems Engineering Journal, Vol. 3 (4), December 1996, pp. 222–238.

[Berners–Lee, 99] Tim Berners–Lee, Weaving the Web, Harper, 1999.

[Birman, 86] K. P. Birman. ISIS: A System for Fault-Tolerant Distributed Computing. Department of Computer Science, Cornell University. TR86-744, April, 1986.

[Braumandl et al., 01] R. Braumandl, M. Keidl, A. Kemper, D. Kossmann, S. Seltzsam, and K. Stocker, ObjectGlobe: Open Distributed Query Processing Services on the Internet, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E–Services, 24 (1), March 2001. http://research.microsoft.com/research/db/debull/A01mar/issue.htm

[Brickley and Guha, 00] Brickley, D. and Guha, R. (2000). Resource Description Framework (RDF) Schema Specification 1.0. Candidate recommendation, World Wide Web Consortium. See http://www.w3.org/TR/2000/CR–rdf–schema–20000327.

[CAPA] Course Approval Process Automation (CAPA). Project realized an the LSDIS lab, Department of Computer Science, funded by the Vice President for Academic Affairs, University of Georgia, July 1, 1996 – June 30, 1997.

[Cardoso et al., 01] Jorge Cardoso, John A. Miller, and Amit P. Sheth. "Workflow Quality of Service: its Specification and Computation". (working paper), 2001.

[Carey, 01] M. Carey and the Propel Platform Team, Towards a Scalable Infrastructure for Advanced E–Services, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E–Services, 24 (1), March 2001.

[Casati 01] F. Casati, and M. C. Shan, Definition, Execution, Analysis, and Optimization of Composite E–Services, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E–Services, 24 (1), March 2001.

[Chen et al., 01] Jian Chen, Shih–Ming Lee, Qiang Su. "Quality management system's design for virtual organizations". Electronic Journal of Organizational Virtualness. Vol. 3 No. 5 Page: 65 – 79. 2001.

[Chen, 00] Yufeng Chen. "Design and Implementation of Dynamic Process Definition Modifications in OrbWork Enactment System". M.Sc. thesis., University of Georgia., July, 2000.

[Christophides et al., 01] V. Christophides, R. Hull, A. Kumar, and J. Simeon, Workflow Mediation using VorteXML, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E−Services, 24 (1), March 2001.

[Cristian, 91] F. Cristian. Understanding Fault-Tolerant Distributed Systems. Communications of the ACM , Vol. 34, No. 2, February 1991.

[DAML] www.daml.org

[Frolund and Koistinen, 98] S. Frolund and J. Koistinen, "Quality−of−Service Specification in Distributed Object Systems," Distributed Systems Engineering Journal, December 1998.

[Gray and Reuter, 93] J. Gray and S. Reuter. Transaction Processing: Concepts and Techniques. Morgan Kaufmann. 1993.

[Grefen et al., 01] P. Grefen, K. Aberer, H. Ludwig, and Y. Hoffner, CrossFlow: Cross−Organizational Workflow Management for Service Outsourcing in Dynamic Virtual Enterprises, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E−Services, 24 (1), March 2001.

[Hall et al., 01] David Hall, John A. Miller, Jonathan Arnold, Krys J. Kochut, Amit P. Sheth and Michael J. Weise, "Using Workflow to Build an Information Management System for a Geographically Distributed Genome Sequence Initiative," Genomics of Plants and Fungi, R.A. Prade and H.J. Bohner, Editors (2001) pp. −. Marcel Dekker, Inc. New York, NY. (to appear).

[Hammer and Campy, 93] Michael Hammer and James Campy. Reengineering the Corporation−A Manifesto for Business Revolution. Harper Business, 1993.

[Han et al., 98] Yanbo Han, Amit Sheth and Christoph Bussler. "A Taxonomy of Adaptive Workflow Management". Workshop of the ACM 1998 Conference on Computer Supported Cooperative Work, Seattle, WA, USA, November 1998.

[Hendler and Mcllraith, 01] J. Hendler and S. Mcllraith, The Web Services and Web Applications Track, International Semantic Web Working Symposium (SWWS), Stanford University, California, USA, July 30 – August 1, 2001, www.cs.uic.edu/~ifc/SWWS/.

[Hensinger et al., 00] C. Hensinger, M. Reichert, Th. Bauer, Th. Strzeletz, P. Dadam. "ADEPTworkflow − Advanced Workflow Technology for the Efficient Support of Adaptive, Enterprise−wide Processes." VII. Conference on Extending Database Technology, Demotrack, Konstanz, Germany, March 2000.

[Hiltunen et al., 00] Matti A. Hiltunen, Richard D. Schlichting, Carlos A. Ugarte, and Gary T. Wong. "Survivability through Customization and Adaptability: The Cactus Approach", DARPA Information Survivability Conference and Exposition (DISCEX 2000), pp. 294−−307, Jan 2000.

[Kang et al., 99] Kang, M., Froscher, J., Sheth, A., Kochut, K., and Miller, J. (1999). A Multilevel Secure Workflow Management System. In Proc. of 11th Conference on Advanced Information Systems Engineering (CAiSE'99), Heidelberg, Germany, 1999.

[Kashyap and Sheth, 00] V. Kashyap and A. Sheth, Information Brokering Across Heterogeneous Digital Data, Kluwer Academic Publishers, August 2000.

[Kashyap and Sheth, 98] V. Kashyap and A. Sheth, "Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context and Ontologies", in Cooperative Information Systems: Current Trends and Directions, M. Papazoglou and G. Schlageter (eds.), Academic Press, 1998, pp. 139−178.

[Kashyap, 01] V. Kashyap, Report on the Integration, Interoperation and Composition Track, International Semantic Web Working Symposium (SWWS), Stanford University, California, USA, July 30 – August 1, 2001, www.cs.uic.edu/~ifc/SWWS/.

[Kim 01] W. Kim, "Knowledge−based Personalization," Thesis (Advisor: A. Sheth), Department of Computer Science, University of Georgia, November 2001.

[Kochut at al., 99] Krys J. Kochut, Amit P. Sheth, and John A. Miller. "ORBWork: A CORBA−Based Fully Distributed, Scalable and Dynamic Workflow Enactment Service for METEOR". Large Scale Distributed Information Systems Lab, Computer Science Department, University of Georgia, 1999.

[Kraiss et al., 01] A. Kraiss, F. Schoen, G. Weikum, and U. Deppisch, Towards Response Time Guarantees for e–Service Middleware, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E–Services, 24 (1), March 2001.

[Lazcano et al., 01] A. Lazcano, H. Schuldt, G. Alonso, and H. Schek, WISE: Process based E–Commerce, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E–Services, 24 (1), March 2001.

[Lee and Whang, 01] Hau L. Lee and Seungjin Whang. "E–Business and Supply Chain Integration". Stanford University. Stanford Global Supply Chain SGSCMF– W2–2001. Management Forum, November 2001.

[Littlewood, 91] B. Littlewood. Software Reliability Modelling. In Software Engineer's Reference book . Section 31, Butterworth-Heinemann Ltd.,1991.

[Luo et al 02] X. Luo, A. Sheth, J. Miller, I.B. Arpinar, "Exception Handling for Conflict Resolution in Cross–Organizational Workflows," Distributed and Parallel Database Systems Journal, 2002 (to appear).

[Luo, 001] Zongwei Luo. "Knowledge Sharing, Coordinated Exception Handling, and Intelligent Problem Solving to Support Cross–Organizational Business Processes." Ph.D. dissertation, Department of Computer Science, University of Georgia, December 20001.

[Maffeis, 95] S. Maffeis. Adding Group Communication and Fault-Tolerance to CORBA. in Proceedings of the USENIX Conference on Object-Oriented Technologies. June, 1995.

[Mehta, 01] B. Mehta, M. Levy, G. Meredith, T. Andrews, B. Beckman, J. Klein, and A. Mital, BizTalk Server 2000 Business Process Orchestration, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E–Services, 24 (1), March 2001.

[Mena et al 00a] E. Mena, A. Illarramendi, V. Kashyap and A. Sheth, "OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre–existing Ontologies", international journal Distributed and Parallel Databases (DAPD), Vol. 8, No.(2), pp. 223–271,, ISSN 0926–8782, April 2000: 223–271.

[Mena et al., 00b] E. Mena, V. Kashyap, A. Illarramendi, A. Sheth, "Imprecise Answers in Distributed Environments: Estimation of Information Loss for Multi––Ontology based Query Processing" the International Journal of Cooperative Information Systems (IJCIS), Vol. 9, No. 4 (2000): 403–425.

[Miller at al., 02] John A. Miller, Jorge Cardoso and Gregory Silver, "Using Simulation to Facilitate Effective Workflow Adaptation", Proceedings of the 35th Annual Simulation Symposium (ANSS'02), San Diego, California (April 2002) pp. (submitted)

[Miller at al., 96] John A. Miller, Amit P. Sheth, Krys J. Kochut and Xuzhong Wang, "CORBA–Based Run–Time Architectures for Workflow Management Systems," Journal of Database Management (JDM), Special Issue on Multidatabases, Vol. 7, No. 1 (Winter 1996) pp. 16–27. Idea Group Publishing.

[Miller at al., 98] John A. Miller, Devanand Palaniswami, Amit P. Sheth, Krys J. Kochut and Harvinder Singh, "WebWork: METEOR2's Web–Based Workflow Management System," Journal of Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies (JIIS), Special Issue on Workflow Management Systems, Vol. 10, No. 2 (March/April 1998) pp. 185–215. Kluwer Academic Publishers.

[Miller et al, 02] John A. Miller, Jorge Cardoso and Gregory Silver. "Using Simulation to Facilitate Effective Workflow Adaptation" Proceedings of the 35th Annual Simulation Symposium (ANSS'02), San Diego, California (April 2002) pp. (submitted).

[Miller et al., 99] J. Miller, A. Sheth and K. Kochut, "Perspectives in Modeling: Simulation, Database and Workflow," *Conceptual Modeling: Current Issues and Future Directions*, P. Chen, J. Akoka, H. Kangassalo, B. Thalheim, Ed, LNCS Vol. 1565, Springer Verlag, Berlin–Heidelberg, April 1999: 154–167.

[Müller et al., 99] Müller, R.; Rahm, E.: Rule–Based Dynamic Modification of Workflows in a Medical Domain. In: Buchmann, A.P. (ed.): Datenbanksysteme in Büro, Technik und Wissenschaft (BTW99), Freiburg im Breisgau, 1.–3. März 1999. Springer, Berlin 1999: 429–448.

[Narayanan et al., 01] S. Narayanan, S. Subramanian, and the Tavant Team, Sell−side Channel Integration − Tavant’s Approach, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E−Services, 24 (1), March 2001.

[Nori et al., 01] A. Nori, C. Venkatraman, and R. Jain, Defining the Next Generation e−Business Platform: A Discussion of the Asera eBusiness Platform, in Data Engineering Bulletin Special Issue on Infrastructure for Advanced E−Services, 24 (1), March 2001.

[Petrie and Sarin, 00] Charles Petrie and Sunil Sarin. Controlling the flow. *IEEE Internet Computin*g, 4(3):34−36, May−June 2000.

[RDF] "Resource Description Framework," http://www.w3.org/RDF/

[Reibman and Veeraraghavan, 91] A. L. Reibman and M. Veeraraghavan. Reliability Modeling: An Overview for System Designers. IEEE Computer , April, 1991.

[Sheth and Kochut 99] A. Sheth, K. Kochut, "Workflow Applications to Research Agenda: Scalable and Dynamic Work Coordination and Collaboration Systems, in Workflow Management and Interoperability, A. Dogac et al Eds., Springer Verlag, 1999, pp. 35−59.

[Sheth et al., 99] Amit P. Sheth, Wil van der Aalst, and Ismailcem B. Arpinar. "Processes driving the networked economy." IEEE Concurrency, pages 18−31, July−September 1999.

[Sheth, 98] A. Sheth, "Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics", in Interoperating Geographic Information Systems. M. F. Goodchild, M. J. Egenhofer, R. Fegeas, and C. A. Kottman (eds.), Kluwer, Academic Publishers, 1998, pp. 5−30.

[Thacker et al., 01] S. Thacker, A. Sheth and S. Patel, "Complex Relationships for the Semantic Web," in Creating the Semantic Web, D. Fensel, J. Hendler, H. Liebermann, and W. Wahlster (eds.), MIT Press, 2001 (to appear).

[Ulrich, 01] Franke, Ulrich. "The Concept of Virtual Web Organisations", Electronic Journal of Organizational Virtualness 3 (2001) 4, pp. 43−64.

[WSCL] "Web Services Conversation Language," HYPERLINK "http://www.w3.org/TR/wsdl" UDDI.org

§[WSDL] "Web Services Description Language," HYPERLINK "http://www.w3.org/TR/wsdl"http://www.w3.org/TR/wsdl

[Yang and Papazoglou, 00] Jian Yang and Mike P. Papazoglou. "Interoperation support for electronic business." Communications of the ACM, 43(6):39−47, June 2000.