NSF GRANT # 0601167
NSF PROGRAM NAME: Information Technology & Infrastructure Systems

A Portal-based Web Service Framework for Construction Supply Chain Integration and Collaboration

Jack C.P. Cheng
Engineering Informatics Group
Department of Civil & Environmental Engineering
Stanford University
Stanford, CA 94305-4020

Kincho H. Law
Engineering Informatics Group
Department of Civil & Environmental Engineering
Stanford University
Stanford, CA 94305-4020

Abstract: The benefits of integrating and collaborating with supply chain partners have been well identified in many industries. In the construction industry, however, supply chain integration and collaboration is challenging due to the high fragmentation and the temporary project-based nature of the industry. Leveraging web service and web portal technologies, we have developed a prototype web service system to facilitate communication, integration and collaboration among project participants in construction supply chains. The system SC Collaborator implements a service oriented portal framework and provides a secure, modular and flexible solution for managing construction supply chains. This paper presents the service oriented portal framework, the orchestration of services in the framework, and the system architecture of SC Collaborator. A project rescheduling scenario is employed to demonstrate the potential value of SC Collaboration in construction industry applications.

1. Introduction: A supply chain consists of a network of key business processes and facilities, involving end users and suppliers that provide products, services and information [23]. Traditionally, departments (e.g. design, planning, and purchasing) and organizations (e.g. contractors and subcontractors) along a supply chain operate independently. The benefits of integrating and coordinating members along supply chains have been studied and identified in many industries [31, 41]. Integrating supply chain members and allowing them to collaborate help increase service level, reduce cost, facilitate decision making, better utilize resources, and enhance responsiveness to changes. To facilitate collaboration among individuals and organizations, information needs to be shared and become available for integration within and across organizations. This enhances supply chain visibility and therefore reduces members' vulnerability to problems and risks from business partners. Integrating information and services also avoids information delay and distortion, a major cause to the increase in demand signal variation along the supply chain upstream, a phenomenon called the bullwhip effect [24]. Therefore, integrating and collaborating with supply chain members are crucial to effective supply chain management.

New [32] and Cox [16] have suggested that supply chain research in construction should focus on the development of interactive and inter-organizational relationships, which requires integration and collaboration. Unfortunately, the construction industry is arguably the least integrated of all the major industrial sectors [18]. The lack of integration among project partners due to interoperability problems leads to significant economical costs in the construction industry. According to a study by the National Institute of Standards and Technology (NIST) in 2002, imperfect interoperability costs the capital facilities industry 15.8 billion dollars in a single year [20]. Obviously there is a strong need to facilitate integration and collaboration among project participants in construction supply chains.

Supply chain integration in the construction industry is a challenging task due to the high fragmentation and temporary project-based nature of the industry. The construction industry is fragmented among participants including general contractors, subcontractors, architects, engineers, laborers, and developers [22]. According to a study on the construction industry in the United States

[28], the top eight architectural, engineering and construction (AEC) companies control less than twenty percent of the market share while by contrast the top companies in the aerospace industry control over seventy-five percent of all trades within the industry. This is probably due to the fact that the construction industry is comprised of countless companies from many different trades, most of which are small to medium in size. These AEC companies tend to use a wide range of hardware platforms and software applications for their own operations, leading to the technical challenges in integrating the construction supply chains. The temporary project-based nature of construction projects also hinders integration and collaboration in construction supply chains. Since construction supply chains are highly dynamic, the organizational structure changes frequently. It is unlikely for project participants to work together long enough on any one project to build enough trust, which is necessary for organizations to become willing to share information and systems. Therefore, the system to facilitate integration and collaboration in construction supply chains should be widely accessible and scalable for geographically distributed users, secure to help establish trust, and customizable and flexible to allow quick reconfiguration upon changes in supply chains.

We have developed a flexible, scalable and customizable prototype web-based system for facilitating integration and collaboration in AEC supply chains. The SC Collaborator system framework implements web service technologies and service oriented architecture to provide modular development and flexible reconfiguration of system functionalities. Leveraging web portal technology, the framework offers customizable system layout, secure access control, and a single point of access to multiple information sources and systems. Utilizing open source technologies, SC Collaborator provides an economical solution for AEC companies, usually reluctant or unaffordable to huge IT investment, to manage their supply chains.

The paper is organized as follows: Section 2 describes the shortcomings of the enterprise resource planning systems for supply chain integration, and how they can be solved using web service technology. Section 3 presents the underlying service oriented portal framework, mechanisms to orchestrate services, and system architecture of the SC Collaborator system. Section 4 shows a scenario which illustrates the potential of the SC Collaborator system to integrate loosely coupled information and applications among members in construction supply chains.

2. Integration of Information, Applications and Services

2.1. Enterprise Resource Planning (ERP) Systems: An increasing number of AEC companies have implemented ERP systems to integrate loosely scattered information and applications among stakeholders. An ERP system is typically employed in corporations to integrate information including finances, accounting, human resources, supply chain, and customer information [17]. They can potentially enhance transparency along the supply chain by eliminating information distortions and increase information velocity by reducing information delays [3].

However, ERP systems were not designed and are often not suitable for the construction industry [46]. There are research studies and efforts on selection and implementation of ‘generic’ ERP systems in the construction industry [2, 15, 35, 40, 46]. Companies that use a generic ERP system often need to modify and customize it to support their own needs, which is a costly process in terms of time, efforts and cost. As most AEC companies are small and medium businesses, they are reluctant to the huge implementation investment in information systems such as ERP systems. In addition, the temporary and project-based nature of construction supply chains requires quick system deployments and frequent changes of trading partners. The inflexibility of ERP systems to accommodate changes of supply chain structures hinders the usability and applications of those systems in the dynamic, unstable construction supply chains.

2.2. Web Service Technology: The Internet provides a promising channel to integration and collaboration among geographically distributed individuals and organizations. With the rapid development of web technologies, the Internet has become ubiquitous and instantaneously accessible. The proliferation of the Internet makes it the most cost effective means of driving supply chain integration and information sharing [25]. Today, AEC companies increasingly take advantage of the Internet and information technology for design and learning [12, 19, 34, 42], for document and knowledge management [27, 48], and for project monitoring and management [8, 11, 14, 33].

A web service can be described as a specific function that is delivered over the Internet to provide information or services to users through application-to-application interaction. Leveraging well established Internet protocols and commonly used machine readable representations, web services can be located, invoked, and combined to provide complex business services. The implementation of web services are encapsulated and not exposed to the users. Therefore, changing the implementation of one web service function does not alter the way that the users invoke the function. This enables clean and robust deployment and maintenance of web services. Web services can be reused by multiple applications or other services residing on a network, avoiding duplicated developments of service units with similar functionalities.

Web services are the building block of service oriented architecture (SOA). SOA is a system development model in which information sources and software functionalities are delivered as individual service units over a network. SOA allows a large complicated system infrastructure to be built in a scalable manner. Modular system development and maintenance is enabled as the system is divided into web service components which can be managed separately. Quick reconfiguration and deployment is also allowed in service oriented systems due to their plug-and-play capability. This modularity and flexibility is crucial for information systems used in fast changing environments such as the construction supply chains.

3. Prototype Web Service Framework – Supply Chain Collaborator: Leveraging the web service technologies, we have developed a prototype web-based collaborative system, called SC Collaborator (Supply Chain Collaborator). SC Collaborator is a system designed for supporting AEC activities and integrating loosely coupled information and services. Based on a service oriented portal framework which will be described in Section 3.1, SC Collaborator implements the SOA approach and provides scalability, flexibility, customizability, single point of access, and extendable functionality. The service orchestration and system architecture of the SC Collaborator system will be discussed in the following sections. Figure 1 shows the homepage of the SC Collaborator system for users to log in.

Figure 1: Homepage of the SC Collaborator system

3.1. Service Oriented Portal Framework: Web portal technology provides a means to aggregate distributed web services. A web portal is a web-based system that acts as a gateway to a larger system or a network of web applications. It provides a single point of access to information sources and application functionalities regardless of their physical location or storage mechanism. The basic operational units of a portal system are web portlets, which are sub-programs that encapsulate a single or a number of web applications. Portal systems are necessary for web portlets to become visible and accessible. Portal systems enable multiple information sources and web applications to be retrieved and integrated into a workflow or a supply chain.

Web portals are commonly used as an internal repository of information and documents for data storage, publication and retrieval within organizations [30]. Due to their customizability and security, web portals allow authorized users to access sensitive personal information and enable system administrators to manage a huge amount of information in a centralized way. There is also a trend in the construction industry to establish project-specific web space using portal systems for cross-organizational collaboration. However, there is little, if any, rigorous research on portal design, development, maintenance, and updating for facilitating supply chain management decisions [45]. The framework presented in this section is a service oriented approach for portal design and implementation.

A service oriented portal framework is a system development framework that leverages web portal technology to provide a secure and customizable user interface and implements SOA to integrate information, applications and services in a flexible and reusable manner. In a service oriented portal framework, as illustrated in Figure 2, information sources, application functionalities and system operations are wrapped into individual web services, which can be located and invoked by application portlet units via standardized protocol. These web services are integrated and orchestrated into different workflows for various business processes in the application portlet units. These web services can be reused in different workflows or reused multiple times in a single workflow. Therefore, development of repeated system operations is avoided. In addition, modification of system functionalities becomes flexible and quick as every business process is decomposed into separate atomic web service components.

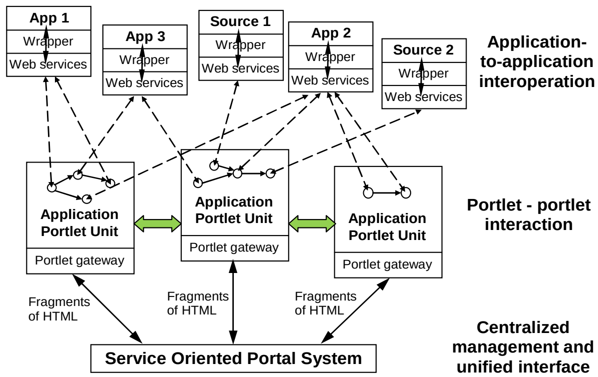


Figure 2: Conceptual framework for service oriented portal system

3.2. Service Orchestration: In a service oriented portal framework, information, applications and internal system operations are deployed and delivered as web services. These web services usually are not sufficient to perform a business process individually. These web services are often needed to aggregate with each other into a workflow. For instance, multiple cross-application activities are required to implement a business process "add purchase order." These activities may include adding a purchase order to the production plan, sending confirmation to the customer, changing the status of the order and the corresponding items, and allocating materials and resources to fulfill the order. Each of these activity components could be separated and deployed as individual web services. A mechanism to combine these activity component services is necessary to complete a business process. There are several research efforts on the mechanisms to invoke, terminate and combine web-based services [13, 21]. In SC Collaborator, the composition and orchestration of the web service units are performed using a model driven approach with the aid of process models residing on the business applications layer.

Model driven approach allows system developers to understand and to check the systems easily through the model representations. Model driven approach also enables system developers to transfer the models to another system and to regenerate the implementation codes. Therefore, it enhances the robustness and portability of system development and maintenance. Model driven approach has already been used for software development for several years, but its application on service orchestration is still under hot discussions and research in recent years [29, 38, 47]. There are two major approaches for model driven service orchestration: (1) a top-down approach which starts with high-level semantic models such as Unified Modeling Language (UML) [36] and Business Process Modeling Notation (BPMN) [37] to executable code, and (2) a bottom-up approach in which proprietary modeling is integrated within the IDE such as Oracle BPEL Designer. In SC Collaborator, the top-down approach is used because the high-level models are platform independent and portable across various systems and applications.

Each business task in SC Collaborator is associated with a single process flow model, represented in UML. The model describes the sequence of service units and the logic involved during the transitions between the service units. For example, interactions among the activity components to complete the aforementioned business process "add purchase order" can be represented in a UML model as illustrated in Figure 3. For implementation, the process flow model is converted into Business Process Execution Language (BPEL) [7], an emerging service orchestration standard for describing the behavior of web services at different levels of abstraction. BPEL is a layer on top of Web Service Definition Language (WSDL) [9] and XML Schema, with WSDL and XML Schema defining the structural aspects of service interactions, and BPEL defining the behavioral aspects. Figure 4 shows an excerpt of the BPEL file converted from the UML model in Figure 3.

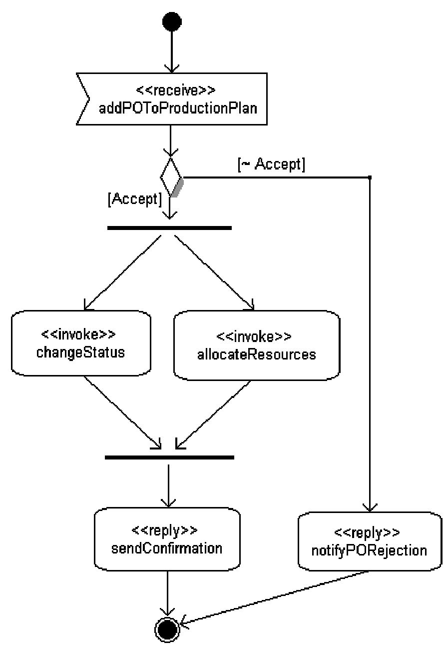


Figure 3: UML activity diagram of the business process "add purchase order"

```
<bpws:process exitOnStandardFault="yes" name="addPurchaseOrder" suppressJoinFailure="yes"
  targetNamespace="http://eig.stanford.edu/test" xmlns:bpws="http://docs.oasis-
  open.org/wsbpel/2.0/process/executable" xmlns:ns="http://eig.stanford.edu/testArtifacts"
  xmlns:tns="http://eig.stanford.edu/test">
  <bpws:import importType="http://schemas.xmlsoap.org/wsdl/" location="addPurchaseOrder.wsdl"
    namespace="http://eig.stanford.edu/test" />
  <bpws:import importType="http://schemas.xmlsoap.org/wsdl/"
    namespace="http://eig.stanford.edu/testArtifacts" />
  <bpws:import importType="http://schemas.xmlsoap.org/wsdl/"
    location="addPurchaseOrderArtifacts.wsdl"
    namespace="http://eig.stanford.edu/testArtifacts" />
- <bpws:partnerLinks>
    <bpws:partnerLink myRole="addPurchaseOrderRequester" name="client"
      partnerLinkType="tns:addPurchaseOrder" partnerRole="addPurchaseOrderRequester" />
    <bpws:partnerLink myRole="processPO" name="server" partnerLinkType="ns:server"
      partnerRole="processPO" />
  </bpws:partnerLinks>
+ <bpws:variables>
- <bpws:sequence name="main">
    <bpws:receive createInstance="yes" name="addPOToProductionPlan"
      operation="addPOToProductionPlan" partnerLink="server" portType="tns:addPurchaseOrder"
      variable="serverResponse" />
  - <bpws:assign name="Assign" validate="no">
    - <bpws:copy>
        <bpws:from />
      - <bpws:to variable="Accept">
        - <bpws:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">
            <![CDATA[ /tns:input ]]>
          </bpws:query>
        </bpws:to>
      </bpws:copy>
    </bpws:assign>
  - <bpws:if name="Accept">
    - <bpws:sequence name="Sequence">
      - <bpws:flow name="Flow">
          <bpws:invoke inputVariable="serverRequest" name="changeStatus"
            operation="changeStatus" outputVariable="serverResponse" partnerLink="server"
            portType="tns:processPurchaseOrder" />
          <bpws:invoke inputVariable="serverRequest" name="allocatingResources"
            operation="allocateResources" outputVariable="serverResponse"
            partnerLink="server" portType="tns:processPurchaseOrder" />
        </bpws:flow>
        <bpws:reply name="sendConfirmation" operation="onResult" partnerLink="client"
          portType="tns:addPurchaseOrderCallback" variable="outputMessage" />
      </bpws:sequence>
    - <bpws:condition>
        <![CDATA[ Accept ]]>
      </bpws:condition>
    - <bpws:else>
        <bpws:reply name="notifyPORejection" operation="onResult" partnerLink="client"
          portType="tns:addPurchaseOrderCallback" variable="outputMessage" />
      </bpws:else>
    </bpws:if>
  </bpws:sequence>
```

Figure 4: Excerpt of the BPEL file for the business process "add purchase order"

3.3. System Architecture of SC Collaborator: Figure 5 shows the system architecture of the SC Collaborator framework. The framework consists of an access control engine, a database support, and four layers of integrated functionalities – a communication layer, a portal interface layer, a business application layer, and an extensible computing layer. The communication layer provides a communication channel for users to access the system. The portal interface layer serves as a unified and customizable platform to support interactions between users and the system. The business applications layer provides an environment for executing various business processes such as decision making and connecting to external data sources, applications and services. The extensible computing layer is potentially comprised of numerous databases, software applications and web services that the business applications layer can integrate to support high-level or computationally intensive business functions.

Open source technologies are leveraged to minimize implementation costs which hinder the usability in AEC companies, which are usually small and medium in size and reluctant to huge investment in IT. In specific, open source software Apache Tomcat [5], Liferay Portal [26] and MySQL [44] are used to support the communication layer, the user interface, and the database support. Open source packages, such as Apache Struts [6], Apache Axis [4] and Hibernate [39], are also utilized to support the communication channels and data mapping. In the following sections, the components of SC Collaborator are presented in detail.

3.3.1. Communication Layer: A user-friendly and readily accessible communication channel is essential to the usability of a system. The SC Collaborator system uses an open source platform – Apache Tomcat [5] – to enable the connectivity and access to the system. Apache Tomcat serves as a web servlet container for the communication servlets, Apache Struts [6] and Apache Axis [4]. The Struts servlet allows users to access the SC Collaborator system using web browsers, which are commonly available on every computer. The Struts servlet also enables remote users to access the system using wireless devices via WAP protocol.

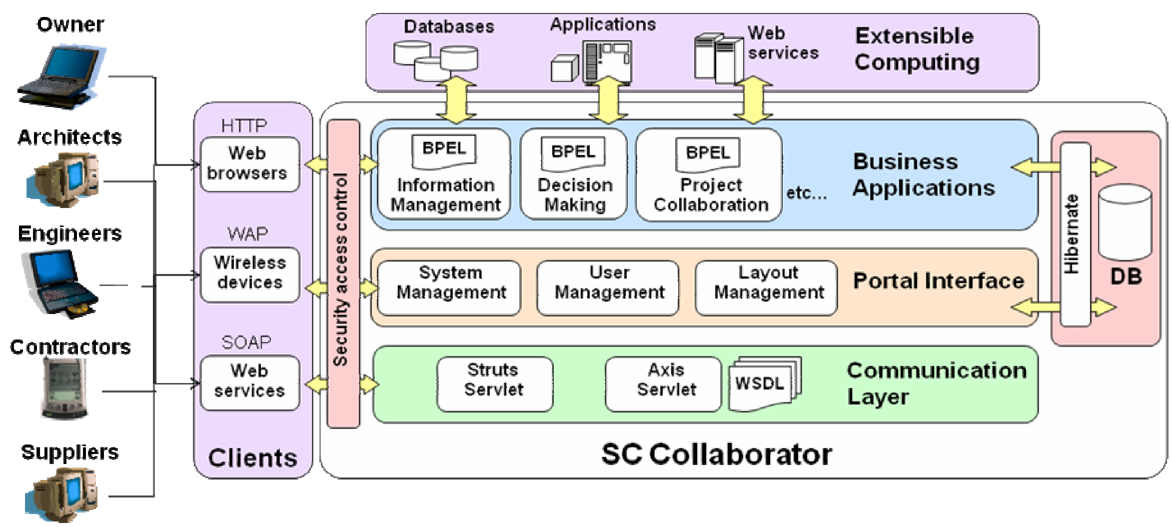


Figure 5: System architecture of the SC Collaborator system

The Axis servlet enables system operations of the SC Collaborator system to be exposed as standard web services, which can be accessed via the Simple Object Access Protocol (SOAP) [10]. The deployed web services are described in standardized WSDL [9] files for service discovery, description and invocation. Figure 4 shows the WSDL file of a simple system operation which sends purchase orders to suppliers. As internal system operations can be exposed to external systems via standardized web service protocol, information and applications that reside in the SC Collaborator can be integrated in external software applications. This greatly extends the accessibility of the SC Collaborator system.

```xml
  <?xml version="1.0" encoding="UTF-8" ?>
- <wsdl:definitions targetNamespace="urn:http.service.material_buyer.portlet.ext.com"
   xmlns:apachesoap="http://xml.apache.org/xml-soap"
   xmlns:impl="urn:http.service.material_buyer.portlet.ext.com"
   xmlns:intf="urn:http.service.material_buyer.portlet.ext.com"
   xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
   xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
   xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  + <!--   -->
  - <wsdl:types>
    - <schema
        targetNamespace="urn:http.service.material_buyer.portlet.ext.com"
        xmlns="http://www.w3.org/2001/XMLSchema">
        <import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
      + <complexType name="ArrayOf_xsd_string">
      + <complexType name="ArrayOf_xsd_int">
      + <complexType name="ArrayOf_xsd_double">
      </schema>
    </wsdl:types>
    <wsdl:message name="addPurchaseOrderResponse" />
  + <wsdl:message name="addPurchaseOrderRequest">
  - <wsdl:portType name="MaterialBuyerServiceSoap">
    - <wsdl:operation name="addPurchaseOrder" parameterOrder="orderId
        orderNumber fromCompany itemId productCode product modelNumber
        material color quantity unitPrice totalCost">
        <wsdl:input message="impl:addPurchaseOrderRequest"
          name="addPurchaseOrderRequest" />
        <wsdl:output message="impl:addPurchaseOrderResponse"
          name="addPurchaseOrderResponse" />
      </wsdl:operation>
    </wsdl:portType>
  - <wsdl:binding name="Portlet_MaterialBuyer_MaterialBuyerServiceSoapBinding"
      type="impl:MaterialBuyerServiceSoap">
      <wsdlsoap:binding style="rpc"
        transport="http://schemas.xmlsoap.org/soap/http" />
    - <wsdl:operation name="addPurchaseOrder">
        <wsdlsoap:operation soapAction="" />
      - <wsdl:input name="addPurchaseOrderRequest">
          <wsdlsoap:body
            encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:http.service.material_buyer.portlet.ext.com"
            use="encoded" />
        </wsdl:input>
      + <wsdl:output name="addPurchaseOrderResponse">
      </wsdl:operation>
    </wsdl:binding>
  - <wsdl:service name="MaterialBuyerServiceSoapService">
    + <wsdl:port
        binding="impl:Portlet_MaterialBuyer_MaterialBuyerServiceSoapBinding"
        name="Portlet_MaterialBuyer_MaterialBuyerService">
    </wsdl:service>
  </wsdl:definitions>
```

Figure 6: Example WSDL file deployed in the SC Collaborator system

3.3.2. Portal Interface Layer: Web portal technology is leveraged to provide a flexible and customizable user interface in the system. The portal user interface of the SC Collaborator system is managed in separate modules. Every module represents a project, an organization, or a group of similar business functionalities. A single module contains a number of submodules, each of which can integrate multiple application portlet units. Configuration, permissions and layout can be configured for each module, submodule and portlet.

User management can be performed at the levels of individual users, organizations, user groups and roles. A user is an individual who performs tasks in the system. An organization represents a corporate hierarchy. A user group is a grouping of users. A user can be associated with any number of user groups, but only one organization. Every member inherits the roles and permissions that are assigned to the organization or user group that the member belongs to. SC Collaborator is a role-based system. A role is a collection of permissions. The types of roles in the SC Collaborator system are system administrator, module administrator, advanced user, module member, normal user and guest. Each role has its predefined set of permissions to the system, layout, modules, submodules and portlets.

The user interface for web browsers and wireless devices can be configured through the layout management portlet unit. The portlet unit allows users with either a system administrator role or a module administrator role to add and delete submodules, to set up the permissions of submodules, and to configure the submodule style. On each submodule, the administrative users can add, delete and allocate application portlet units. The administrative users can also grant individual users the permissions to view, modify and configure a specific module, submodule and portlet. Therefore the system layout can be highly customizable so that some modules or portlet units are available only to the designated users, organizations or user groups. This ensures that the right information is delivered to the right person at the right time.

3.3.3. Business Implementations Layer: Each application portlet unit is an independent unit which performs a specific task or business process. Based on the Java framework, a portlet unit can perform computations, execute other applications, connect to databases, and invoke web services. Therefore, multiple services can be integrated in a single portlet unit to implement various business processes. For instance, the application portlet unit that helps retailers manage the purchase orders they have submitted integrates three different services: (1) service that submits purchase orders to manufacturers, (2) service that monitors the status of each purchase order, and (3) service that triggers warning notifications when a problem is encountered. A portlet unit in SC Collaborator can also interact with other portlets to solve complicated business problems. The application portlet units in SC Collaborator are compliant with JSR 168 standard [1], a specification that defines a standard programming model for portlet development. Consequently, the portlet units can be packaged and reused by other portal systems, allowing high portability across platforms.

3.3.4. Database Support: In the database tier, an open source database – MySQL – is used to store the application data as well as the system information including user information, layout configurations, and user and system settings. The SC Collaborator system is not bounded to a particular database system. The system can be installed with any Java Database Connectivity (JDBC) [43] compliant database without any complicated configuration and modification of codes due to the use of the Hibernate framework [39]. The Hibernate framework maps the objects in a relational database into object-oriented Java classes. If a user has already installed other databases such as PostgreSQL and Oracle database, SC Collaborator can integrate with the existing database with little effort. The user does not need to install and execute MySQL in order for SC Collaborator to run.

4. Demonstrating Project Rescheduling Scenarios: To illustrate the SC Collaborator system for the construction industry applications, a demonstrating scenario is described in the following sections. The scenario is based on data collected from a recently completed construction project of a supermarket in Boars, Sweden (Figure 7). The project started in April 2007 and finished in April 2008. In this project, the main contractor employed 21 subcontractors for various tasks such as excavation, piling, and pouring concrete. Turnkey-type contracts were used between the main contractor and the subcontractors. In other words, material procurement, delivery and installation were performed by the subcontractors individually. The main contractor was not involved in any of these activities. As a result, efficient communication and collaboration among the main contractor and subcontractors are essential to the success of the project.

According to the main contractor, the project was seriously affected by the schedule delay from the subcontractors, which caused the project manager to reschedule almost every day. Information such as material delivery and activity start time is crucial for project rescheduling. However, this information was loosely distributed among the main contractor, subcontractors, and suppliers. Poor communication among them could prevent the project manager from collecting all the necessary information for making the right decisions in schedule change. The SC Collaborator system provides a scalable platform that could potentially facilitate the communication and integration across the organizational boundary.

The flows of information and interactions in the rescheduling scenario are shown in Figure 8. In the scenario, the decisions made by project participants were interrelated to the information provided by the participants. Production status information and expected delivery time information were reported to corresponding subcontractors. This information may lead to changes of the scheduled activity start, finish and the scheduled material delivery time by the subcontractors. The change information was sent to the main contractor and the suppliers, possibly affecting the overall schedule and the suppliers' production plan. Therefore, information transparency is important and value-adding for the rescheduling process, as well as other operations in a construction supply chain. This can be shown in the results when different cases of collaboration and information transparency were tested using the SC Collaborator system.

Figure 7: Floor plan and finished layout of the supermarket in Boras, Sweden

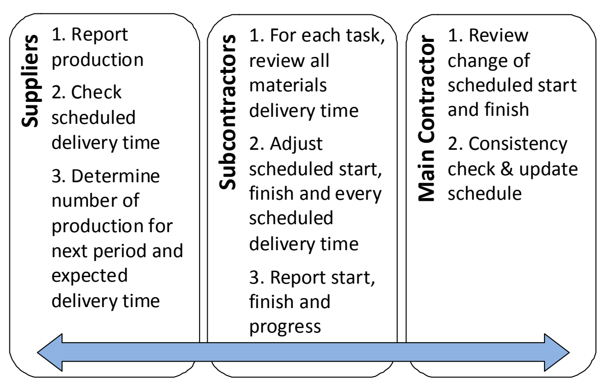


Figure 8: Information flow and interaction in the rescheduling scenario

As an example, consider the activity "7.1.1 foundation works – concrete surrounding beam – major part 1." The activity required three materials: 2,388 m^2 of form material (wood), 1,121 m^3 of concrete and 430 m^2 of sandwich concrete element called Siroc (Figure 10). There was a one week (five working days) production and delivery delay for Siroc starting from Day 1 of Week 20 (May 21). Therefore, the Siroc supplier notified the subcontractor *Muniak* and the form material (wood) supplier *Pacific Plastics* of the material delay (Figure 9). Figure 11 shows the message instantaneously received by *Muniak* and *Pacific Plastics* when they logged on the SC Collaborator system. There were several constraints that had to be satisfied: every delivery must be confirmed at least three working days before the delivery time. Moreover, product type, configuration, amount, and delivery time cannot be changed after confirmation.

Figure 12 plots the inventory on site of the form material over time. The area under graph multiplied by per-unit per-day holding cost represents the total inventory holding cost of the form material due to the material delay of Siroc. If the Siroc supplier notified *Muniak* and *Pacific Plastics* of the delay at least three days earlier, the activity 7.1.1 as well as the delivery of the corresponding materials could be postponed, avoiding the unnecessary inventory on site. In addition, the subcontractor could save more than half of the inventory holding cost if *Pacific Plastics* knew the delay one day earlier than delivery time instead of two days. It shows that instantaneous information sharing, which can be facilitated by SC Collaborator, can add significant value to each supply chain member although it looks simple.

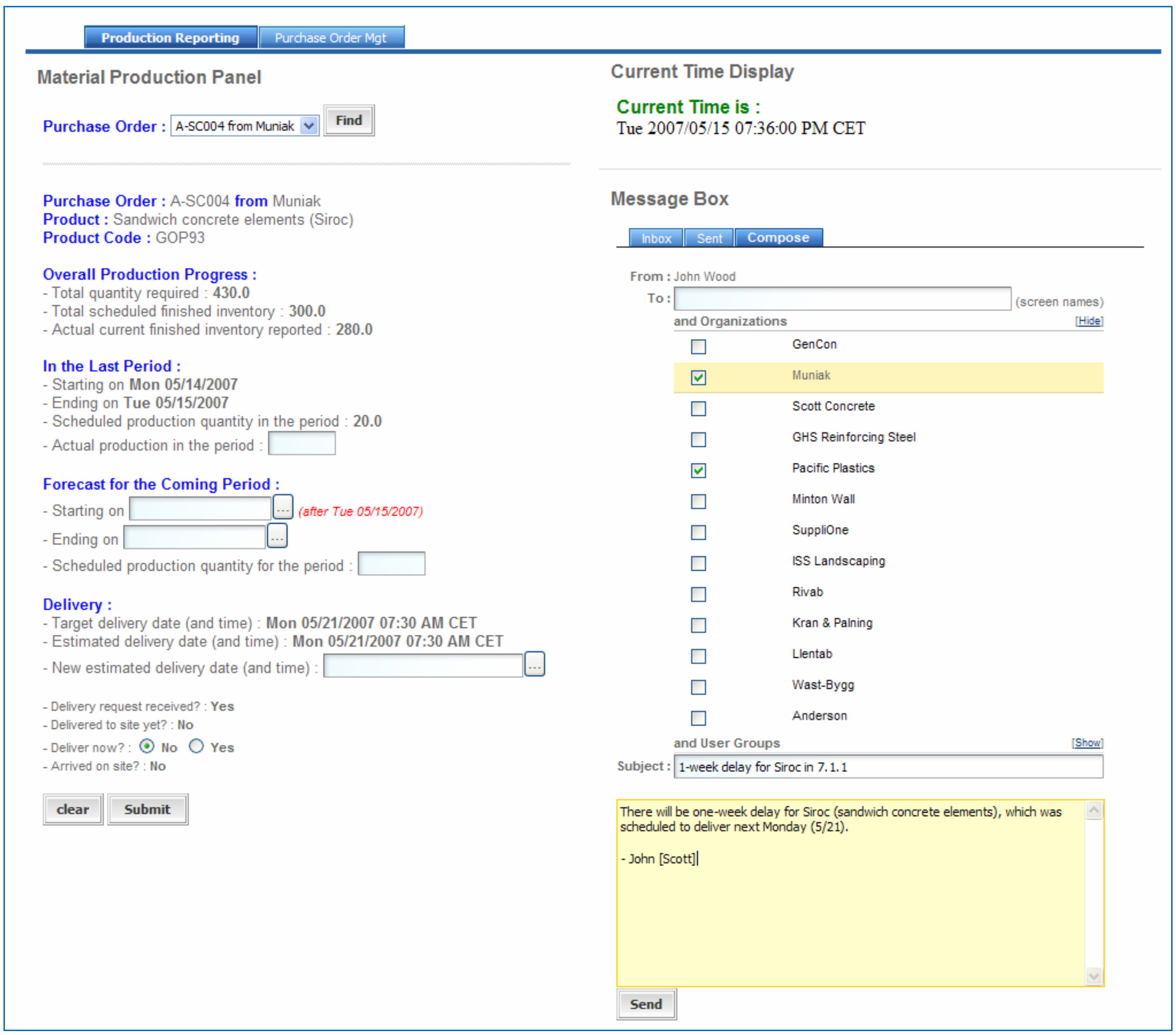


Figure 9: Siroc supplier's view in SC Collaborator

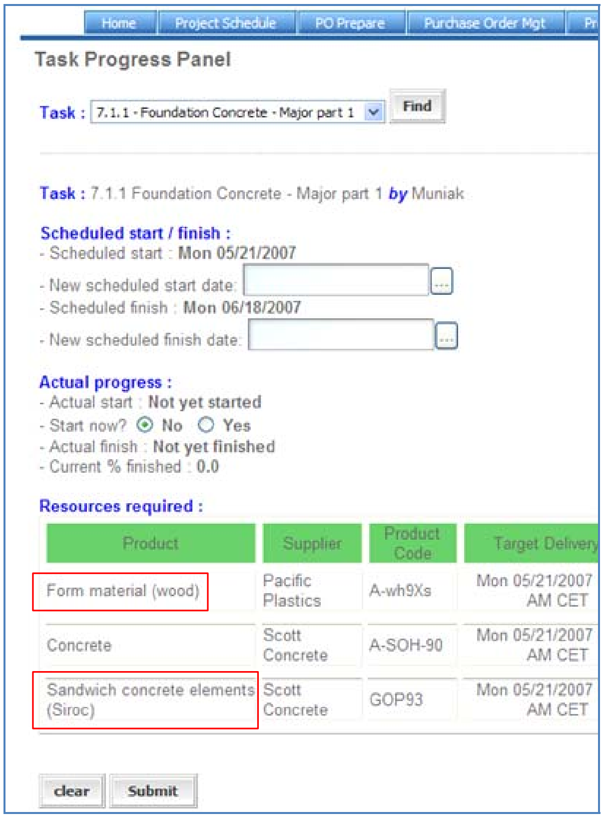
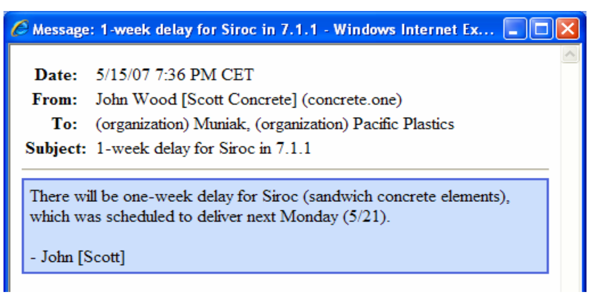


Figure 10: Subcontractor's view of the activity 7.1.1 in SC Collaborator


Message: 1-week delay for Siroc in 7.1.1 - Windows Internet Ex...

Date: 5/15/07 7:36 PM CET
From: John Wood [Scott Concrete] (concrete.one)
To: (organization) Muniak, (organization) Pacific Plastics
Subject: 1-week delay for Siroc in 7.1.1

There will be one-week delay for Siroc (sandwich concrete elements), which was scheduled to deliver next Monday (5/21).

- John [Scott]

Figure 11: Message received by the main contractor *Muniak* and the form material (wood) supplier *Pacific Plastics*

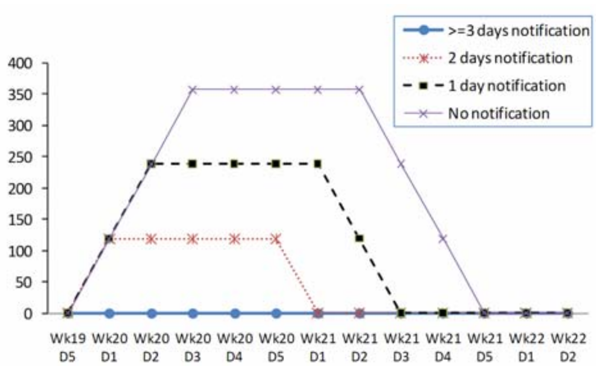


Figure 12: Inventory (in m^2) of form material (wood) under different supply delay notification conditions

5. Summary: Value of supply chain integration and collaboration has been shown in many industry sectors. Some AEC companies employ ERP systems to support integration of organizations. However, these tools are not suitable for managing construction supply chains, due to the unstable project-based nature of construction supply chains and the small to medium size of companies in the construction industry. An alternative approach to link individuals across organizational boundaries and to integrate scattered applications is offered by the Internet technology and web service technology.

This paper describes a prototype web service system, SC Collaborator, designed for construction supply chain integration and collaboration. The service oriented approach and the portal technologies have been leveraged. SC Collaborator system provides a single point of access to distributed information, applications and services among scattered supply chain members. It is modular, flexible, secure, and easy to install and reconfigure, which make the SC Collaborator system a desirable means for companies in the construction industry. SC Collaborator implements service oriented portal framework and coordinates discrete service units using a top-down model driven approach. The system consists of a security access control engine, a communication layer, a portal user interface, a layer of business application units, and a database support component. A project rescheduling scenario has been presented to illustrate the potential of the SC Collaborator system to integrate partners and facilitate their collaboration in construction projects.

6. Acknowledgements: The authors would like to acknowledge the supports by the US National Science Foundation, Grant No. CMS-0601167, the Center for Integrated Facility Engineering (CIFE) at Stanford University, the Enterprise Systems Group at the National Institute of Standards and Technology (NIST) and Wast-Bygg, AB, Sweden. The authors would like to thank Prof. Hans Bjornsson of Chalmers University of Technology, Sweden for his collaboration in this project and for valuable comments and suggestions throughout. Any opinions and findings are those of the authors, and do not necessarily reflect the views of NSF, CIFE, NIST or Wast-Bygg, AB. No approval or endorsement of any commercial product by NIST, NSF or Stanford University is intended or implied.

7. References:
[1] A. Abdelnur, E. Chien and S. Hepper, "JSR 168: Portlet Specification," Java Specification Requests, Java Community Process, Sun Microsystems and IBM, 2003.

[2] U. Acikalin, M. Kuruoglu, U. Isikdag and J. Underwood, "Evaluating the Integrative Function of ERP Systems Used Within the Construction Industry," eWork and eBusiness in Architecture, Engineering and Construction, Taylor & Francis Group, London, 2009.

[3] H.A. Akkermans, P. Bogerd, E. Yücesan and L.N. van Wassenhove, "The Impact of ERP on Supply Chain Management: Exploratory Findings from a European Delphi Study," European Journal of Operational Research, vol. 146(2), pp. 284-301, 2003.

[4] Apache, Apache Axis, 2006.

[5] Apache, Apache Tomcat 5.5, 2007.

[6] Apache, Apache Struts, 2008.

[7] BEA Systems, Microsoft, IBM and SAP, Business Process Execution Language for Web Services (BPEL4WS), 2003.

[8] B. Becerik, "A review on past, present and future of web based project management & collaboration tools and their adoption by the US AEC Industry," International Journal of IT in Architecture Engineering and Construction, vol. 2(3), pp. 233-248, 2004.

[9] D. Booth and C.K. Liu, "Web Services Description Language (WSDL) Version 2.0 Part 0: Primer," W3C Working Draft 21, 2004.

[10] D. Box, D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H.F. Nielsen, S. Thatte and D. Winer, Simple Object Access Protocol (SOAP) 1.1, 2000.

[11] S.L. Chan and N.N. Leung, "Prototype Web-Based Construction Project Management System," Journal of Construction Engineering and Management, vol. 130, pp. 935, 2004.

[12] H.M. Chen and H.C. Tien, "Application of Peer-to-Peer Network for Real-Time Online Collaborative Computer-Aided Design," Journal of Computing in Civil Engineering, vol. 21, pp. 112, 2007.

[13] J. Cheng, "A Simulation Access Language and Framework with Applications to Project Management," Ph.D. thesis, Stanford University, Stanford, CA, 2004.

[14] S.O. Cheung, H.C.H. Suen and K.K.W. Cheung, "PPMS: a Web-based construction Project Performance Monitoring System," Automation in Construction, vol. 13(3), pp. 361-376, 2004.

[15] B.Y. Chung, M.J. Skibniewski, H.C. Lucas and Y.H. Kwak, "Analyzing Enterprise Resource Planning System Implementation Success Factors in the Engineering–Construction Industry," Journal of Computing in Civil Engineering, vol. 22(6), pp. 373-382, 2008.

[16] A. Cox, "A Research Agenda for Supply Chain and Business Management Thinking," Supply Chain Management: An International Journal, vol. 4(4), pp. 209-11, 1999.

[17] T.H. Davenport, "Putting the Enterprise into the Enterprise System," Harvard Business Review, vol. 76(4), pp. 121, 1998.

[18] A. Fearne and N. Fowler, "Efficiency versus Effectiveness in Construction Supply Chains: the Dangers of Lean Thinking in Isolation," Supply Chain Management: An International Journal, vol. 11(4), pp. 283-287, 2006.

[19] R. Fruchter, "A/E/C Teamwork: A Collaborative Design and Learning Space," Journal of Computing in Civil Engineering, vol. 13(4), pp. 261-269, 1999.

[20] M.P. Gallaher, A.C. O'Connor, J.L. Dettbarn and L.T. Gilday, "Cost Analysis of Inadequate Interoperability in the Capital Facilities Industry," Technical Report No. GCR 04-867, National Institute of Standards and Technology (NIST), 2004.

[21] D. Greenwood, M. Calisti, W.T. Ag and S. Zurich, "Engineering Web Service-Agent Integration," Proceedings of 2004 IEEE International Conference on Systems, Man and Cybernetics, pp. 1918 - 1925, 2004.

[22] R.R.A. Issa, I. Flood and G. Caglasin, "A Survey of E-business Implementation in the US Construction Industry," Journal of Information Technology in Construction, vol. 8, pp. 15-28, 2003.

[23] D.M. Lambert, M.C. Cooper and J.D. Pagh, "Supply Chain Management: Implementation Issues and Research Opportunities," The International Journal of Logistics Management, vol. 9(2), pp. 1-19, 1998.

[24] H.L. Lee, V. Padmanabhan and S. Whang, "Information Distortion in a Supply Chain: The Bullwhip Effect," Management Science, vol. 50(12 Supplement), pp. 1875-1886, 2004.

[25] H.L. Lee and S. Whang, "Supply Chain Integration over the Internet," Supply Chain Management: Models, Applications, and Research Directions, Springer US, 2005.

[26] Liferay Inc., Liferay Open Source Enterprise Portal System, 2008.

[27] P.E.D. Love, F. Edum-Fotwe and Z. Irani, "Management of knowledge in project environments," International Journal of Project Management, vol. 21(3), pp. 155-156, 2003.

[28] E. Luening, Can construction industry rise to online challenge?, 2000.

[29] P. Mayer, A. Schroeder and N. Koch, "A Model-Driven Approach to Service Orchestration," Proceedings of the 2008 IEEE International Conference on Services Computing (SCC 2008), Honolulu, Hawaii, USA, pp. 533-536, 2008.

[30] D. Michelinakis, "Open Source Content Management Systems: An Argumentative Approach," Ph.D. thesis, University of Warwick, 2004.

[31] E.A. Morash and S.R. Clinton, "Supply Chain Integration: Customer Value through Collaborative Closeness versus Operational Excellence," Journal of Marketing Theory and Practice, vol. 6(4), pp. 104-20, 1998.

[32] S.J. New, "The Scope of Supply Chain Management Research," Supply Chain Management: An International Journal, vol. 2(1), pp. 15-22, 1997.

[33] P. Nitithamyong and M.J. Skibniewski, "Web-based construction project management systems: how to make them successful?," Automation in Construction, vol. 13(4), pp. 491-506, 2004.

[34] W. O'Brien, L. Soibelman and G. Elvin, "Collaborative Design Processes: An Active-and Reflective-Learning Course in Multidisciplinary Collaboration," Journal of Construction Education, vol. 8(2), pp. 78-93, 2003.

[35] J.T. O'Connor and S.C. Dodd, "Achieving integration on capital projects with enterprise resource planning systems," Automation in Construction, vol. 9(5-6), pp. 515-524, 2000.

[36] Object Management Group, Unified Modeling Language (UML) Superstructure, version 2.0, 2005.

[37] Object Management Group (OMG), Business Process Modeling Notation (BPMN) Specification, 2006.

[38] K. Pfadenhauer, S. Dustdar and B. Kittl, "Comparison of two distinctive model driven Web service orchestration proposals," Proceedings of the 2005 Seventh IEEE International Conference on E-Commerce Technology Workshops (CECW'05), pp. 29-36, 2005.

[39] Red Hat, Hibernate framework, 2008.

[40] J.J. Shi and D.W. Halpin, "Enterprise Resource Planning for Construction Business Management," Journal of Construction Engineering and Management, vol. 129(2), pp. 214-221, 2003.

[41] T.M. Simatupang, A.C. Wright and R. Sridharan, "The Knowledge of Coordination for Supply Chain Integration," Business Process Management Journal, vol. 8(3), pp. 289-308, 2002.

[42] R.D. Sriram, Distributed and integrated collaborative engineering design, Sarven Publishers, 2002.

[43] Sun Microsystems, JDBC Data Access API, 2002.

[44] Sun Microsystems, MySQL 5.0, 2007.

[45] A.J. Vakharia, "e-Business and Supply Chain Management," Decision Sciences, vol. 33(4), pp. 495-504, 2002.

[46] J.-B. Yang, C.-T. Wu and C.-H. Tsai, "Selection of an ERP system for a construction firm in Taiwan: A case study," Automation in Construction, vol. 16(6), pp. 787-796, 2007.

[47] M. Zhang and Z. Duan, "From Business Process Models to Web Services Orchestration: The Case of UML 2.0 Activity Diagram to BPEL," Service-Oriented Computing – ICSOC 2008, Springer, Berlin-Heidelberg, 2008.

[48] Y. Zhu, R.R.A. Issa and R.F. Cox, "Web-Based Construction Document Processing via Malleable Frame," Journal of Computing in Civil Engineering, vol. 15(3), pp. 157-169, 2001.