SERVICE ORIENTED ARCHITECTURE FOR HETEROGENEOUS INFORMATION SYSTEMS INTEGRATION

Leonardo de Carvalho Vidal (CEFET/RJ)  
leo.vidal@au1.ibm.com
Marina Rodrigues Brochado (CEFET/RJ)  
marinabrochado@gmail.com
Rui Manuel Alves da Silva e Sousa (UM)  
rms@dps.uminho.pt
Antonio José Caulliraux Pithon (CEFET/RJ)  
apithon@unisys.com.br

Constant business process changes and fast technology evolution requires high flexibility on the organizations, to quickly create and change business processes, in order to gain a competitive advantage. However, the traditional information systems of the enterprises are mostly rigid, monolithic, centralized, not based on open standards and non up-to-date with new methodologies, unable thus to support this new reality. Such business requirements are creating new challenges in the process of information systems integration, demanding a new flexible and adaptable software environment in order to achieve such requirements. This work presents a study on the implementation of Service Oriented Architecture (SOA) and the use of Web Services in order to achieve seamless integration and reutilization of existing software, increase collaboration with business partners and suppliers, and propose the alignment between information technology strategy and business strategy. A partial implementation of this architecture is also presented.

Palavras-chaves: SOA, Web Services, Information Systems
1. Introduction

The computer programming languages have made considerable developments. First electronic computers were custom programmed using machine code, the basic binary “1” and “0” language that all digital computers use at their core. Over the past two decades, we have seen a broad range of third and fourth-generation programming languages rising. The concept of software architect comes from how programs and lines of code are structured, definition of data types, how data will be stored, how the modules will connect each other, how the user interaction will happen and many other aspects, as described by (Hurwitz J. et al., 2006).

Over the past 40 years, it is clearly possible to identify four distinct software architectural paradigms. The earliest paradigm being the monolithic architecture, implemented on early mainframe systems, followed by the client-server architecture, and then the distributed or component architecture. The fourth architectural paradigm (SOA) is now emerging. Most organizations have developed, internally or externally, business systems based on the first three referred architectures. Such systems were not originally designed to work together, neither to connect to other business applications, making hard all integration efforts. The middle 1990’s, beyond custom development of applications, have seen the emergence and near dominance of packaged software solutions such as Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) and Supply-Chain Management (SCM) systems. The majority of mid to large sized organizations use enterprise applications from SAP, Oracle, Siebel, Peoplesoft, and others to support and manage significant aspects of their business, but these enterprise systems need to co-exist with custom applications and other legacy systems performing multiple enterprise functions. As described by (Laurindo, 2000), the rigid and proprietary interfaces of these enterprise systems removed the business flexibility.

In this context, the implementation of SOA using Web services helps organizations to extend CRM, ERP, and other large monolithic software applications, adding new business functions or capabilities in response to changing business needs, while also ensure that they
have the needed flexibility to adapt to rapidly changing business needs and market dynamics in today’s business environment.

This paper describes previous integrations attempts involving enterprise systems. Then it describes the new architectural paradigm, SOA, and the Web services. A model for SOA deployment and management is proposed, and applied for a partial implementation of SOA in two retail companies, merged after the acquisition of Company B by Company A. The SOA implementation process has provided a seamless business systems integration.

2. Service Oriented Architecture

SOA is an architectural style that allows Web services applications to interoperate dynamically with one another, even when these applications were not originally designed to work together. According to (Bieberstein et al., 2006), SOA provides the flexibility to treat business processes and the underlying IT infrastructure as components that can be reused and recombined to address changing business priorities. The key principles of Web services are enabled by agreement on standards across a broad group of hardware and software organizations. These five defining principles, defined by (Marks, 2006), can be combined into a succinct definition of a Web service:

“Web Services are loosely coupled, self-describing services that are accessed programmatically across a distributed network, and exchange data using vendor, platform, and language-neutral protocols.”

Loosely Coupled. This concept is related to the modular structure of a system, as described by (Page-Jones, 1980). One module may be described as a procedure or function. The coupled level means the level of dependency between two (or more) modules. Loosely coupled systems are more flexible and more easily reconfigured, enabling components of a system to be replaced or exchanged easily (Sampaio, 2005).

Self-Describing. The Web Services Description Language (WSDL) is an XML document that describes a Web service’s inputs and outputs in a structured manner. The
WSDL document enables other software to determine how to invoke the service and determine what results the service will return.

**Accessed Programmatically.** Web services are not designed to be accessed directly through human interaction. Instead, they are invoked by and exchange data with other software applications programmatically using the Simple Object Access Protocol (SOAP). This programmatic access enables a Web service to be incorporated into other software applications, Web sites, or even other Web services.

**Network Distributed.** Web services are accessed using open Internet protocols and data formats such as TCP/IP, HTTP, and XML. Using these existing protocols and data formats, Web services comply with current company security measures and policies (for example, corporate firewalls). This feature makes it possible for Web services to be deployed and accessed across corporate intranets or the Internet.

**Exchange Data Using Vendor, Platform and Language-Neutral Protocols.** The ability for a Web service to exchange data in a vendor, platform and language-neutral format is facilitated through broad industry agreement on open standards. Through Web services, it is possible that two systems developed using Microsoft .Net technology on the same machine exchange data each other. In the same way, one JAVA application running on a RISC computer in Brazil may exchange data with a C++ system running on a CISC computer in China, both using different operating systems. This is perhaps the most important and compelling aspect of Web services.

### 3. Proposed Model for SOA Deployment and Management

The organizations are applying innovative business models and solutions to improve performance, grow the business and simultaneously decrease costs. In this context, SOA is a key enabler of innovation. According to (Freire, 2002), innovation is mandatory for any organization development. Also describes that each innovation project comprises a six phases cycle: (1) opportunity, (2) idea, (3) development, (4) test, (5) introduction, and, (6) diffusion. In order to manage this cycle, it is necessary to adopt a global model for innovation management, that binds the innovation cycle and four management basic competences: (1) strategic management: the innovation initiative must be aligned with the company’s strategic
orientation, thus contributing for its competitiveness, (2) project management: the projects must be conducted through models and techniques, at planning, execution and control levels, (3) functional management: innovation activities must be supported by the companies policies, and, (4) change management: the organizational culture must be open to changes. Expertise on management competences is a necessary condition, but not enough to ensure the success of an innovation project. That success arises from the interaction between the innovation cycle and management competences. Beyond the adoption of a global model for innovation management, that binds the innovation cycle and management competences, there are still four specific and mandatory phases for SOA deployment and management in any organizations, as described by (Ganci, 2006):

**Model phase.** This is the process of capturing the business design from an understanding of business requirements and objectives. The business requirements are translated into a specification of business processes, goals, and assumptions for creating a model of the business. In some cases, businesses design is modeled using primitive techniques such as drawing the design or using text documents. In other cases the capture of business design is made using a sophisticated approach that includes the use of specialized tooling that performs what-if scenarios with various parameters the business may experience. The model will also capture key performance indicators, such as business metrics, that are important business measurements. These key performance indicators are input to the assembly of the application. In addition, the indicators can be monitored in production to capture the critical data to measure whether the objectives are being met.

**Assemble phase.** The business design is used to communicate the business objectives to the IT organization that will assemble the information system components that implement the design. The IT architect and business analyst work together, to design the services. During the process of resolving the design and implementation of the modeled business processes and services, a search should be performed of existing artifacts and applications in an effort to find components that meet the needs of the design. Some applications will fit perfectly, some will have to be re-factored, and some will have to be developed to meet the requirements of
the design. These existing assets should be rendered as services for assembly into composite applications. Any new services required by the business design will need to be created.

**Deploy phase.** The deploy phase includes a combination of creating the hosting environment for the applications and the deployment tasks of those applications. This includes resolving the application’s resource dependencies, operational conditions, capacity requirements, and integrity and access constraints. A number of concerns are relevant to construct the hosting environment including the presence of the already existing hosting infrastructure supporting applications and pre-existing services.

**Manage phase.** The manage phase includes the tasks, technology, and software used to manage and monitor the application assets such as services and business processes that are deployed to the production runtime environment. Monitoring is a critical element for ensuring that the underlying IT systems and application are up and running to maintain the service availability requirements of the business. Monitoring also deals with monitoring performance of service requests and timeliness of service responses. In addition, monitoring includes maintaining problem logs to detect failures in various services and system components, as well as localizing failures and restoring the operational state of the system. The manage phase also comprises managing the business model, tuning the operational environment to meet the business objectives expressed in the business design, and measuring the success or failure of meeting those objectives. SOA is distinguished from other styles of enterprise architecture by its correlation between the business design and the software that implements that design, and its use of policy to express the operational requirements of the business services and processes that codify the business design.

Figure 1 presents the proposed model for SOA deployment and management.
4. Case Study

This section presents a partial implementation of SOA involving two retail companies, merged after the acquisition of Company B by Company A, and how the SOA implementation process has provided a seamless business systems integration. Company A is a huge retail company, with more than 200 stores located across 70 Brazilian cities. Company B is an e-commerce company, which does business through its Internet site. Company A decided to expand their business over the Internet, through Company’s B acquisition. Company A strategic level required from it’s IT staff that could make possible to view and analyze sales reports from Company B from the same existing systems in Company A, in order to closely monitor the new business results. It was also requested that the products, suppliers and stock databases should be unified, expanding commercial benefits to company B.

Survey Phase. During this initial phase, a survey was made in all systems from both Companies A and B. Existing TI systems in Company A were classified in two groups: (1) store front-end systems, and, (2) store backend systems, comprised by Products Catalog, Customers and Suppliers Catalog and the Sales Information systems (Sales Flash). Tables 1 and 2 show the core backend systems identified in both companies.

<table>
<thead>
<tr>
<th>Application</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales</td>
<td>System developed using IBM Lotus Notes technology. Contains customers, products and suppliers database.</td>
</tr>
<tr>
<td>Catalog</td>
<td>System developed using IBM Lotus Notes technology, to consolidate sales information from all stores.</td>
</tr>
</tbody>
</table>
Table 1 - Company A business applications

<table>
<thead>
<tr>
<th>Application</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System B</td>
<td>Web systems developed using JAVA technology and relational database. It is a fully e-commerce application.</td>
</tr>
<tr>
<td>Sales Report</td>
<td>System developed using Oracle relational database, responsible for fiscal reports and sales reports.</td>
</tr>
<tr>
<td>Tool</td>
<td></td>
</tr>
</tbody>
</table>

Table 2 - Company B business applications.

Change and Functional Management. The team responsible for the project execution was structured as described in Table 3.

Project Management. The project was fully conducted according to project management models and techniques, as described in section 3, at planning, execution and controlling levels. The Microsoft Office Project 2003 tool was used on this phase.

<table>
<thead>
<tr>
<th>Rule</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Analyst</td>
<td>The role of the Business Analyst is capture the business design using a sophisticated approach that includes the use of specialized tooling to perform what-if scenarios with various parameters the business may experience.</td>
</tr>
</tbody>
</table>
IT Architect

The role of the IT architect is to evaluate business requirements and build solutions to solve them. The architect begins by gathering input about the problem, developing an outline of the desired solution, and considering any special requirement that need to be factored into that solution. The architect then takes this input and designs the solution, which can include one or more computer applications that address the business problems by supplying the necessary business functions. The IT Architect and Business Analyst work together to flesh out the design of the services.

Project Manager (PM)

Responsible for the project leadership, the Project Manager coordinates activities, target dates and resources. The PM must ensure that the communication is perfect flowing between all team members. The PM is also responsible for the financial and project execution, and must have a clear vision of the project objectives.

Developers (Programmers)

First, Developers are responsible for the service implementation, that involves the creation of new services or enhance existing applications with new services in order for those applications to participate in the solution. Developers are also responsible for creating both high and detailed designs of the overall system, as well as interactions between the assembled components.

Table 3 - Functional management.

Strategic Management and Model Phase. As mentioned before, strategic management is one of the basic competences for any innovation project. The newly
implemented architecture can only reinforce company’s competitiveness if it is linked to the business goals, defined by the strategic levels of the organization. After business requirements definition, the business analyst and IT architect performed the business modeling, in order to reach defined objectives: (1) make possible to view and analyze sales reports of Company B from the same existing systems in Company A, in order to closely monitor the new business results, and (2) request the unification of products, suppliers and stock databases, expanding commercial benefits to company B. During the modeling phase, the IBM Websphere Business Modeler tool was used to capture business requirements.

**Assemble Phase.** On this phase, a search within existing artifacts and applications was performed, in an effort to find components that met the needs of the design. These existing assets were rendered as services for assembly into composite applications. Any new services required by the business design would need to be created. The model phase made possible to identify each service that needed to be implemented and shared, in order to meet business requirements. Table 4 shows all selected functions from the model phase that needed to be implemented as services:

<table>
<thead>
<tr>
<th>Actual Function</th>
<th>Description</th>
<th>System</th>
<th>Type</th>
<th>Data Processed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Payment Handling</td>
<td>Record customer order in Sales Resume DB</td>
<td>System B</td>
<td>Consumer</td>
<td>Store_ID, Data</td>
</tr>
<tr>
<td>Product Check</td>
<td>Get product availability in stock</td>
<td>System A</td>
<td>Consumer</td>
<td>Prod_ID</td>
</tr>
<tr>
<td>Product Availability</td>
<td>Returns product availability in stock</td>
<td>Company A</td>
<td>Provider</td>
<td>Prod_ID, Avail (Boolean)</td>
</tr>
<tr>
<td>Sales Activity</td>
<td>Consolidate sales information by store</td>
<td>Sales Flash</td>
<td>Provider</td>
<td>Cod_Loja, Date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Company A</td>
<td></td>
<td>Ammount</td>
</tr>
</tbody>
</table>
Deploy Phase. The deploy phase includes a combination of creating the hosting environment for the applications and the deployment tasks of those applications. In order to illustrate the Web Services technology, and also to present one of the products of the deploy phase, one Web service, able to handle all products availability requests in Sales Catalog application, including requests from System B, was developed. This application has a native function named getProductAvail that receives any product code requests as input and returns the product availability as output. This native Lotus Script code is illustrated below:

```lss
%INCLUDE "lsxsd.lss"
Class ProductAvail
    Function getProductAvail(productNumber As String, Fault1 As WS_FAULT) As String
        Dim session As NotesSession
        Set session = New NotesSession
        Dim db As notesdatabase
        Dim view As notesview
        Dim doc As notesdocument
        Set db = session.GetDatabase("","catalog/products.nsf")
        Set view = db.GetView("vaPNumber")
        Set doc = view.GetDocumentByKey(Cdbl(productNumber),
            True)
        If doc Is Nothing Then
            Call Fault1.setFault(True)'required for fault activation
            Call Fault1.setFaultString("Product " & productnumber & 
                " doesn't exist")
        Else
            getProductAvail = doc.availability(0)
        End If
    End Function
End Class
```

A Web service named GETPRODUCTAVAIL was implemented in the Sales Catalog application, extending its functionality to other systems. Figure 2 shows the WSDL file used to provide GETPRODUCTAVAIL service.

<table>
<thead>
<tr>
<th>Product Registration</th>
<th>List all existing products</th>
<th>Sales Catalog</th>
<th>Provider Company A</th>
<th>Array of Products</th>
</tr>
</thead>
</table>

Table 4 - Interfaces needed to be implemented as services.
Figure 2 - WSDL file description.

The samples below illustrate a SOAP request and response for GETPRODUCTAVAIL method, through SOAP protocol:

Sample of SOAP request for a product code 45000:

```
<soapenv:Envelope
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"

```
Sample of SOAP response for a product code 45000:

```xml
<soapenv:Body>
  <ns1:GETPRODUCTAVAILResponse
    soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:ns1="urn:DefaultNamespace">
    <GETPRODUCTAVAILReturn
      xsi:type="xsd:string">No</GETPRODUCTAVAILReturn>
  </ns1:GETPRODUCTAVAILResponse>
</soapenv:Body>
</soapenv:Envelope>
```

5. Conclusion

Most organizations have heterogeneous business systems that utilize many elements of different systems architectures, which are extremely difficult and expensive to support and
maintain. Also, the organizations are applying innovative business models and solutions to improve performance, grow the business and simultaneously decrease costs. In this context, the emergence of SOA, implemented using Web services, is making significantly easier for the organizations to implement, maintain and change their core enterprise systems, ensuring that they have the necessary flexibility to adapt to rapidly changing business needs. In the presented case study, the proposed model for SOA deployment and management was applied, making possible the creation of standardized components (services) that can be reused and combined to address changing business, with greater efficiency, cost saving, and increased productivity. The use of existing technology enables to more closely align information technology (IT) with business goals.

Today, there is a previously unseen level of agreement and collaboration on the development of a core set of standards for system interoperability using Web services, from companies like IBM, Microsoft, Oracle and BEA. The standards that form the foundation upon which Web services are implemented are extremely important to the realization of business benefit from SOA.

References


