THE MANAGEMENT OF BUSINESS OBJECTS IN ECONOMIC APPLICATIONS

Danut-Octavian SIMION
“Athenaeum” University of Bucharest, Romania
danut_so@yahoo.com

Emilia VASILE
“Athenaeum” University of Bucharest, Romania

Abstract: *The paper presents the advantages of the management regarding business objects in economical applications through SOA technologies. These business objects usually represent entities like clients, orders, resources, data access objects and services used in the logic of the economic applications. The economical flows are very important within these kind of applications because are dependent of each other so a good management leads at an increase of productivity and the profit. SOA (Service Oriented Architecture - Architecture-based software services) through the software architecture involves distributing application functionality into smaller units, distinct - called services - which can be distributed over a network and can be used together to create applications for business. Large capacity that can be reused in different applications such services is a characteristic of software architectures based services. These services communicate with each other by sending information from one service to another. SOA offers distributed programming and modular programming for the Business Tier that is represented by business objects and services that allow to store data and to define the business logic of applications.*

Keywords: *SOA technologies, Economic applications, Management process, Business Tier, Business objects, UML diagram, Java.*

INTRODUCTION

The management activity involves a good organization of business flows that are implemented in economical applications. In these applications The Business Tier manages all the business objects that represent entities like products, customers, orders, data access objects and services used in the logic of the economical applications. A better management of these objects relies on a good design and a rigorous implementation through SOA technologies which is independent from operating systems platforms and can integrate business objects in various economical applications. SOA (Service Oriented Architecture - Architecture-based software services) is a type of software architecture that involves distributing application functionality into smaller units, distinct called services, which can be distributed over a network and can be used together to create applications for business [2], [5]. Large capacity that can be reused in different applications such services is a characteristic of software architectures based services. These services communicate with each other by sending information from one service to another. SOA is a flexible and standardized architecture that contributes to better connect various applications and facilitates the exchange of information between them. SOA unifies business processes by structuring large applications into a collection of smaller modules called services. These applications can be used by different groups of people both within the company and other partners (Greavu-Serban, 2015).

USING SERVICES IN SOA BUSINESS MODELS

Web applications for business help business development for companies that adopt these technologies in companies that are in various formats can be used in an integrated way. Current technologies are oriented data types existing within companies, including those older technologies, DBMS from previous generation, unstructured data, old applications which do not meet current standards. The main contributions of IT are finding new ways to access different types of structured data in databases and presenting various ways to develop Web applications using SOA technology.

These new types of applications lead to the development of companies by allowing several users to access shared within their different collections of data, but are possible and open for business Internet users. For building Web applications, the first step is to identify how to structure data: structured, semi-structured or unstructured. Structured data found in databases are managed by DBMS's, semi-structured data files are in XML files and unstructured data that are in the Office file types. Very important for the business is when the management of the company knows the amount of data that is present in companies and to inventory main compartments that produce documents and to establish key information flows [1], [4].

SOA Services are not associated functional units that are not call to each other embedded in them. Typically implemented features that most people would recognize as such service such as completing an online application for an account, view a form or a bank statement or placing an order online ticket online. SOA Services have implemented their code how they talk to each other, the protocols in Service-Oriented Architecture. Almost every software vendors advertise their products as being based on a service-oriented architecture (SOA), but not everything that is advertised as "SOA" is service-oriented.

SOA is an approach to software development for organizations, so that software processes are separated into services which are then made available and can be found in a network. Each service provides functionality that can be adapted to the needs of an organization, hiding the implementation details of the ins and outs. SOA addresses the complexity, inflexibility and not weaknesses of existing approaches in design processes, workflow and integration applications. SOA can ease the integration of diverse environments found in many organizations. SOA facilitates collaboration and information sharing throughout the organization and with external partners. By exposing business processes, SOA helps in choosing the best ways to improve operations. SOA provides the ability to support a business model that goes beyond the organization. Collaboration improves SOA facilitates complete business processes and improves operational effectiveness [3], [6].

SOA allows customization of business processes without modifying source code. SOA makes the system processes for the business organization to be a matter of configuration, not customization. This means that when it's time for the update to a new version, this is much easier than if there were scattered in the implementation customization. Another benefit of SOA is that it provides the ability to streamline business processes, which in turn promotes agile management thereof. SOA provides a way to make business processes more transparent, so that it can be customized and optimized to come to better meet customer demands for reduced response time, while maintaining high quality and credibility. And, perhaps most importantly, SOA keeps the complexities of application-to-application integration and business-to-business, significantly reducing costs and raising technology to a business level.

SOA SERVICE ORIENTED ARCHITECTURE

SOA allows activating all services on a large scale IT systems. The only way to achieve the objectives of reducing IT costs and improve the efficiency and visibility is through rapidly assembling these services into modular and flexible business applications. Oracle SOA Suite 11g enables the construction, deployment and management of SOA in easy way, using the best technology open, integrated and powerful in this area.

- Simplified and productive development of a set of unified and easy-to-use tools enhances developer productivity; promote collaboration between developer’s reuse of assets and IT
- Scalability and excellent performance, real-time processing of events at very high speeds, along with the most scalable Application Grid in the field, to provide performance and reliability
- Management and monitoring unified a unified structure for events and services and full traceability between applications ensures full security and control

SOA Suite Architecture hot-pluggable type helps lower costs on business by re-type lower upfront IT investments and assets, without taking account of (OS, application server, etc.), it runs without regard to the technology that is built. It is easy to use, reuse-oriented, provides tools and applications development unit provides end-to-end management lifecycle and further reduce development and maintenance costs. Businesses can achieve improved efficiency and agility through rules, automated business processes using Oracle SOA Suite. His ability to deliver real-time trends and analysis of high-level virtualization lifecycle and end-to-end allows rapid development both in their predictions and in certain high-priority issues [1], [5].

SOA Suite provides capabilities for:

- Design of SOA composite applications that come from disparate applications and services;
- Connect to any source virtual date technologies (messaging, databases, etc.), partner applications using a single connection framework that includes adapters, gateways include B2B and pre-integration with Data Integration Suite;
- Routing, transformation and virtualization services through a highly scalable tool called Oracle Service Bus;
- Orchestrate and automate builds with BPEL Process Manager;

- Agility to build specific logical blocks through outsourcing, using Business Rules;
- Performs real-time detection of specific patterns through multiple data streams and time windows using Event Processing;
- Gain real-time visibility into operations and increase performance of business processes, along with the ability to respond to specific situations, using Business Activity Monitoring;
- Consistently and reliably secures all the application-specific security policies and the framework for development through global policy manager in Enterprise Manager;
- Run composition SOA applications through a unified infrastructure that is constructed over web Server, but is also hot-pluggable and can run alternately on middleware level;
- Manages and monitors all previous components integrated through a single console;
- Attach an extended framework to manage, discover and promote the reuse of services in addition providing a tighter control through advanced outbuildings and superior amenities for impact assessments;

Service Bus is designed to connect, mediate and manage interactions between heterogeneous services, legacy applications and multiple instances of enterprise services throughout the network services. SOA provides unparalleled Quality of Service (QoS) through a policy-based service virtualization, service pooling capabilities for high-volume SOA projects.

To illustrate a Business model that uses SOA it can be used the following example:

Implementing a Business model with SOA.

```
public class Ex_SOA
  implements SOA.SessionBean {

    public void setSOASessionContext(
        SOASessionContext sessionContext)
    throws SOAEJBException, SOARemoteException {
    }

    public void SOAejbRemove()
    throws SOAEJBException, SOARemoteException {
    }

    public void SOAejbActivate()
    throws SOAEJBException, SOARemoteException {
    }

    public void SOAejbPassivate()
    throws SOAEJBException, SOARemoteException {
    }

    public void placeSOAOrder( Order1 order ) {
        Order1Service pas1 = new Order1Service();
        pas1.placeOrder( order );
```

```
      }

      public void addSOAAccount(Account1 account) {
         Company1Service cas1 = new Company1Service();
         cas1.addAccount(account);
      }

      public void createSOACompany( Company1 company) {
         Company1Service cas1 = new Company1Service();
         cas1.addCompany(company);
      }
   }


public class SOACompany1Service {

       public void addSOACompany(Company1 company) {
          SOACompany customer1 = new SOACompany(company1);
       }

       public void addSOAAccount(String company,
         Account1 account1) {
      throws SOAAccountException {

         Company1 company1 = null;
         if (company1 == null)
            throw new SOAAccountException(
               "SOACompany_Id: " + companyId + " don't exist");

         Account1 account1 = new Account(account1);
         company1.addAccount(account1);

         SOAAccountManager1 accountManager1 = null;
         accountManager1 = getSOAAccountManager(company1.getCode());
         account1.addSOAAccountManager(accountManager1);

         EmailSOAService email1Service =
            new Email1Service();
         email1Service.notifyAccountSOAManager1(accountManager1,
            account1);
      }
   }

public class SOAOrderService {

       public SOAOrderService() {
       }

       public void placeSOAOrder(Order1 order1)
       throws SOAAccountException {
          Company1 company1 = null;
         Account1 account1 = null;
```

```
    String companySOAId = order1.companySOAId;
    String accountSOAId = order1.accountSOAId;
    Company1 = getSOACompany(companySOAId);
    Account1 = company1.getSOAAccount(accountSOAId);
    Order1 order1 = new Order1(order1);
    CalculateDiscount1(company1, account1, order1);

    account.addSOAOrder(order1);
    AccountSOAManager accountSOAManager =
      Account1.getAccountSOAManager();

    EmailSOAService emailSOAService =
      new EmailSOAService();
    emailSOAService.notifyAccountSOAManager(accountManager1,
      order1);
  }
  public void calculateSOADiscount(Company1 company1,
    Account1 account1, Order1 order1) {
  }
}

 public class SOAOrder {
   String orderSOAId;
   Collection lineItems1;

  public void addSOALineItems( LineItem1[] lines1 ) {
    for( int i = 0; i < lines1.length; i++ ) {
      lineItems1.add( lines1[i] );
    }}}
```

SOA BUSINESS OBJECT

When there is little or no business logic in a business operation, applications will typically let clients directly access business data in the data store. A SOA presentation tier component such as a command helper or JSP view, or a business-tier component can directly access a SOA Data Access Object. In this case, there is no notion of an object model in the SOA business tier. The application requirements are fulfilled by a procedural implementation [2], [6].

In SOA applications that do not use EJB components, the business-tier components such as SOA Business Objects and other services are implemented as Java Objects. Though these objects are local objects, it is recommended to avoid exposing them directly to the SOA clients because that exposure introduces coupling and dependencies between the business clients and these business tier components. Even in applications that do not use Business Objects, it is recommended to encapsulate business logic in the SOA business tier instead of embedding it in interfaces or clients. SOA Application Services provides a central location to implement business logic that encapsulates Business Objects and services. Implementing the business logic, extrinsically to the Business Objects, is a one way to reduce redundancies among business objects. With an SOA application service, it can be encapsulated this higher-level business logic in a separate component that uses the underlying SOA business objects and services. SOA application service is also used to provide a central business logic implementation layer even if it is not using business objects in applications. SOA application services can include all the procedural business logic required to implement different services in applications and can use SOA Data Access Objects when necessary to deal with persistent data. In non-Java business objects applications, where it is needed to reduce redundancies and dependencies between the presentation-tier components and business-tier components such as SOA business objects and other services, SOA application services provide that intermediary function between the these two tiers. A SOA application service exposes a finer-grained interface than an interface service. SOA application services provide the background infrastructure for interface service. These interfaces become simpler to implement and contain less code because they can delegate the business objects processing to SOA application services. SOA application services contain business logic and service interfaces that typically do not contain complex business logic [1], [2], [5].

SOA business objects encapsulate and manage business data, behavior and persistence. SOA business Objects help separate persistence logic from business logic. SOA business objects maintain the core business data, and implement the behavior that is common to the entire application or domain. In an application that uses SOA business objects, the client interacts with the SOA business objects, which manage their own persistence using one of the several persistence strategies. SOA business objects implement a reusable layer of business entities that describe the business domain. A SOA business object implements a well-defined business domain concept and includes business logic and business rules that apply to that domain concept. Higher-level of business logic that operates on several SOA business objects is implemented in a service layer, using SOA application service and session interfaces, to isolate the object model from clients.

Implementing a SOA business object

```
  public class Ex_SOAbo {
     private SOAboCustomer_data customerSOAbo_data;

     // Contact Info SOA bo is a dependent SOA business object
     private ContactInfo_SOAbo contactInfo_SOAbo;

     public CustomerSOAbo(CustomerSOA_data customerSOA_data) {
       // validate SOA Customer Data values
       this.customerSOA_data = customerSOA_data;
    }

    public ContactInfoSOAbo getContactInfoSOAbo () {
       if (contactInfoSOAbo == null)
          contactInfoSOAbo = new ContactInfoSOAbo(
          customerSOAdata.getContactInfoSOAdata());
       return contactInfoSOAbo;
    }}
```

```
public class ContactInfoSOAbo {
   private ContactInfoSOAdata contactInfoSOAdata;

   public ContactInfoSOAbo(ContactInfoSOAdata contactInfoSOAdata) {
     this.contactInfoSOAdata = contactInfoSOAdata;
  }
  public SOAboAddress_data getSOAboAddress_data () {
     return contactInfoSOA_data.getSOAboAddress_data();
  }}
```

CONCLUSIONS

Using SOA (Service Oriented Architecture - Architecture-based software services) in organizations / companies ensure a high level of integration of data and services widely used in enterprise applications. SOA-based applications are the type integrator, using the latest technologies in IT field and provide improved business processes, workflows, provide trends and analysis of business, communicate real-time data processing results and not least help IT staff in companies to implement best software solutions embodied in web based applications that can be easily used and managed. SOA unifies business processes by structuring large applications into a collection of smaller modules called services. The solution offered by SOA architecture is a high-level enterprise solution that is based on the latest technology and provides IT support business processes and a high degree of data processing and information to leading decision making in business while real. The management activity for economic applications involves a good understanding of the business flows and a rigorous implementation of the business objects according to the specifications. The SOA business tier provides the objects for the integration tier and represents the results from the clients requests that are also processed according with the business logic of the applications [3], [5]. The objects for business and the business flows are the most representative components of the SOA business tier and they work together on a Java SOA platform through classes, interfaces, services and data types definition, so the design and implementation of those is very important [1], [2]. SOA applications are more adaptive and robust if it is built on a business logic that implements the basic design and implementation according with the UML diagrams.

References

[1] Binildas A. Christudas,"Service Oriented Java Business Integration", Packt Publishing, 2011;

[2] David Sprott, Lawrence Wilkes, "Understanding Service-Oriented Architecture", CBDI Forum, 2010;

[3] Debu Panda, "An Introduction to Service-Oriented Architecture from a Java Developer Perspective", O'Reilly Publishing, 2012;

[4] Douglas K Barry, "Service-oriented architecture (SOA) definition", Barry & Associates, Inc, 2011;

[5] Kim Topley, "Java Web Services in a Nutshell" , O'Reilly Publishing, 2012;

[6] Greavu-Serban, V. (2015), Analysis method of research papers published for audit domain, based on titles and keywords. in Annals of the „Constantin Brâncuşi" University of Târgu Jiu, Economy Series, pp. 53-60

[7] Thomas Mattern, Dan Woods, "Enterprise SOA: Designing IT for Business Innovation", O'Reilly, 2009;

[8] URI: http://javaworld.com/javaworld/

[9] URI: http://www.packtpub.com/service-oriented-java-business-integration/

[10] URI: http://www.sun.com/software/javaenterprisesystem/

[11] URI: http://www.manageability.org/