

SOA Applications for Insurance data provider

The Client

The client is a leading source of information about risk. The company supplies data, analytics, and decision-support services for professionals in many fields - insurance, finance, real estate, health services, government, and human resources. Their products help customers measure, manage, and reduce risk.

The Challenge

Provide simple and secured access of data to the external customers who request the data that is critical to the day-to-day operations of their financial applications.

The Objectives

- To develop a set of applications based on Service Oriented Architecture (SOA).
- The customer applications communicate with these applications using Web Services.
- The communication should be secured.

The Solution

The proposed application is an EJB application developed using EJB that allows insurance companies to place orders for a large variety of business reports for determining insurance risk from both internal and remote data sources through a common interface. Reports can be ordered in fixed or XML formats via HTTP, SOAP (Web services) using secured socket layer (SSL) model.

Our employees provided the following services to the client:

- Gathering system requirements, defining use cases, business rules and developing project plans
- Development of JMS point-to-point messaging and Message Driven EJBs to order multiple reports simultaneously
- Developed the GUI using JSP, XSLT, HTML and Apache Struts
- Used J2EE Connection Architecture (JCA) for connecting the WebSphere App Server to the CICS transaction server running on OS/390
- Performed extensive test driven development-using JUnit for unit testing.
- Organized corporate J2EE infrastructure (coding standards/best practices, source management, and common component development)

Tools and Technologies used

- WebSphere Server (Application Server)
- DB2 (Database)
- Java, XML/XSLT, HTML (Languages)
- Unix (Operating System)
- Java Servlets, JSP (Web)
- J2EE, EJB, JMS, Apache Struts (Architecture)