

Deploying Intellix Solutions on the J2EE platform

Using BEA Web Logic and Apache Tomcat

Intellix A/S
H.C.Ørsteds Vej 4
DK-1879 Frederiksberg C
Denmark

Tel: + 45 70 23 37 00
Fax: + 45 70 23 27 00
www.intellix.com
info@intellix.com

© 2002 Intellix A/S

Contents

1. Introduction

1.1 Prerequisites

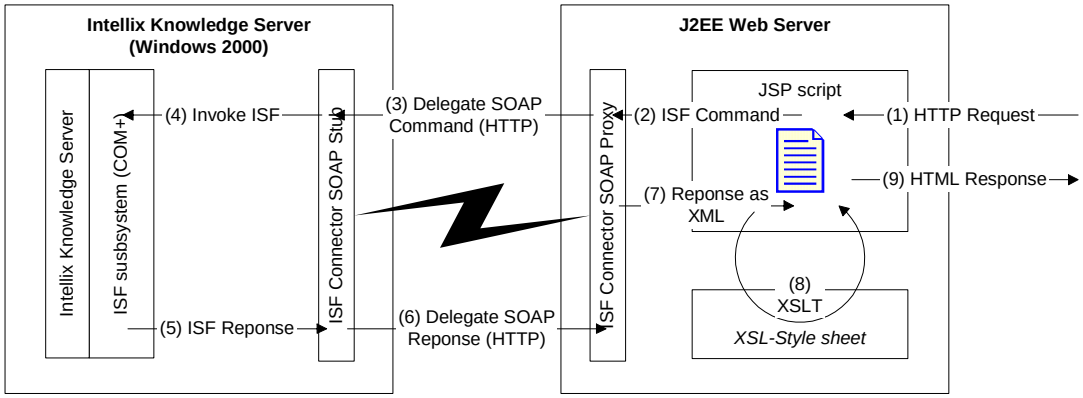
The reader of this document should be familiar with Intellix' Knowledge Server products [13], and have a general awareness of the following software technologies: J2EE [6], COM [7], SOAP [1], XML [11] and HTTP [12].

1.2 SOAP and HTTP

SOAP [1] and other related standards such as WSDL and UDDI have emerged to enable a new generation of "web services" that allow systems to communicate with other systems over open Internet protocols such as HTTP. This reliance on HTTP [12] is a great advantage because most firewalls are configured to let HTTP traffic pass fairly easily. See e.g. www.w3c.org for further background information.

1.3 J2EE implementation of Knowledge Server web layer

The Intellix Knowledge Server [13] was designed to publish Intellix Knowledge Domains as a set of dynamic web pages. Intellix has now furnished the Intellix Knowledge Server with a SOAP interface making it readily available to a variety of software platforms previously not supported.
A J2EE [6] compliant implementation of the web layer encapsulated by the ISF (Intellix Solution Framework) subsystem of the Intellix Knowledge Server has been developed and tested and is the main focus of this document. Figure 1 depicts the architecture of the system we have tested.



• Figure 1 System Architecture.

The first thing to note about the architecture is that the core business logic is located on the Intellix Knowledge Server (IKS) running on a Windows 2000 machine. However, this is neatly packaged into a web service and only accessible using HTTP. This has two implications. Firstly, the IKS machine can be placed anywhere within a corporate intranet and still be accessible from the web server. Secondly, it can be deployed with virtually no risk of compromising security since communication can be confined to three well-defined HTTP requests.

On the web server node java classes within the *ISF Connector SOAP Proxy* make the services of the ISF subsystem of the Intellix Knowledge Server available to the java environment. Behind the scenes these classes uses the SOAP protocol to delegate processing to the Intellix Knowledge Server node.
The Java Server Pages (JSP) [5] that tie the solution together on the web server rely heavily on XML transformations [12] to generate the dynamic content of the web pages. This has the advantage of reducing the JSP code to a trivial matter of delegating requests to the Intellix Knowledge Server and transforming responses. Another advantage is that the burden of performing transformations is easily transferred to clients that support XSL transformation, like e.g. MS Internet Explorer 6.0.

1.4 Setup for J2EE performance tests

The solution depicted in figure 1 was deployed and tested on the following two web servers:

Apache Tomcat 4.0.3
This version of the popular open source web server supports Java Server Pages. The Apache web server is incorporated into or supported by many commercially available J2EE application servers (e.g. IBM Web Sphere [9], BEA Web Logic [8]). It is estimated that over 60% of the web sites in the world are powered by Apache. Tomcat 4.0 [2,3] implements the three Java specifications: Java Servlets 2.3 [4], Java Server Pages 1.2 [5], and J2EE 1.3 [6]. For further information see www.apache.org.

BEA Web Logic 6.1
This is one of the world's leading application server platforms, providing more than 12,500 customers worldwide with a complete J2EE [6] compliant platform for building, integrating, and extending enterprise applications. The BEA Web Logic [8] runs on a wide range of operating systems, and can harness the immense hardware resources of mainframe systems. For further information see www.bea.com.

2. Test Results

2.1 Test preparation

The tests were performed using Mercury Interactive's Load Runner. A complete user session (using predefined valid input) was recorded in a test script and replayed using Load Runner. Two parameters "Number of Virtual Users" and "Think time" were adjusted to produce as high a load as possible measured in Hits/sec. These parameters are used by Load Runner to simulate a certain number of users who go through the recorded session, pausing to think at each page.
The tests were executed with 210 effective concurrent users, each using a think time at 30 seconds. This corresponds to 30 seconds for each user to think about a reply to a question during the dialog.

2.2 Performance results

The performance of various system configurations was then assessed by comparing factors such as load, response time and throughput.

Table 1 below summarises the results for the chosen J2EE web servers and includes a reference column for an ASP solution based on MS Internet Information Server.

Table 1. Performance test results using two J2EE platforms and a Windows 2000 platform.

	Apache Tomcat 4.0.3	BEA Web logic	IIS (ASP solution)
Load (Hits/sec)	25.6	20	18
Response Time (sec) (95% quantile)	5.3	5.0	5.7
Throughput (KB/sec)	230	212	190

Comparing Apache Tomcat and BEA Web Logic we see that the former handles a 25% higher load and also a higher throughput at approximately identical response times. This is in accordance with benchmark results obtained with a standard Intellix web server setup based on a Microsoft Internet Information Server, where figures at the same order of magnitude have been obtained. However, it should be noted that in the standard Intellix setup the web server node and the Knowledge Server node were located on the same machine, which explains the slightly lower load seen in Table 1. For further information on performance tests please refer to the performance test report [14] and the benchmark whitepaper [15] for the Intellix Knowledge Server.

3. Conclusion

The performance results demonstrate that companies running J2EE environments can get the same top notch performance, as well as all the other customer benefits, as those running in pure Windows environments. If your company has an IT strategy based on the Java platform it is now possible to deploy Intellix solutions through a Java based web server. This will usually be done in a classic deployment architecture where the Intellix Knowledge Server is installed on a Windows 2000 Server machine independently of the existing IT infrastructure. This way, no conflicts with Java based strategies occur, and security issues are handled within the existing framework of your company's IT infrastructure.

4. References

1. SOAP 1.1 submission to W3C, URL:http://www.w3.org/TR/SOAP/
2. Apache Soap follow-on project Axis, URL:http://xml.apache.org/axis/
3. Apache Tomcat 4.0, URL: http://jakarta.apache.org
4. Java Servlets 2.3, URL: http://java.sun.com/products/servlet
5. Java Server Pages 1.2, URL: http://java.sun.com/products/jsp/

6. J2EE 1.3 specification, URL: http://java.sun.com/j2ee/docs.html
7. Microsoft COM specification, URL:http://www.microsoft.com/com/resources/specs.asp
8. BEA Web logic specification of J2EE compliance, URL:http://www.bea.com/products/weblogic/server/j2ee_13_compliance.shtml
9. IBM Websphere - Supporting open standards for Web services and J2EE, URL:http://www-109.ibm.com/cgi-bin/click.pl?url=http://www.ibm.com/software/webservers/appserv/whitepapers/openstds_j2ee.pdf&qry=j2ee%20websphere
10. MI Load Runner - Optimizing IBM WebSphere Performance With Mercury Interactive Solutions, URL:http://www-svca.mercuryinteractive.com/accelerate/may-june02/tips/
11. XML specification, URL: http://www.w3.org/XML/
12. HTTP specification, URL: http://www.w3.org/Protocols/
13. Intellix Knowledge Server, URL:http://www.intellix.com/products/kserver/kserver.html
14. Intellix Knowledge Server 2.1 - Performance Test Report, URL:http://www.intellix.com/pdf/uk/KS21%20Test%20Report.pdf
15. Intellix Knowledge Server 2.1 - Benchmarks, URL:http://www.intellix.lan/pdf/uk/KS21%20Benchmark%20Whitepaper.pdf