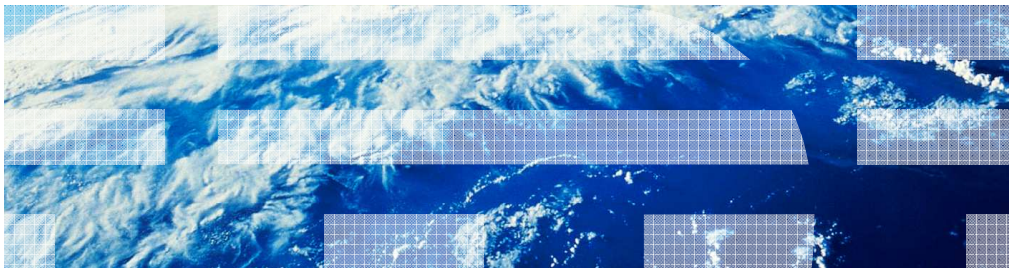


IBM WebSphere Application Server

Application routing



This presentation covers the new application routing model introduced in the SIP servlet 1.1 specification, JSR 289.

Agenda

- Application routing overview
- Application router example

The first section of this presentation provides an overview of the new application routing component. The second section walks through an example of how SIP requests are passed through the application router.

Application routing overview

The SIP application router is a new component introduced in the JSR 289 SIP servlet 1.1 specification. The application router provides a flexible mechanism for creating a composite application offering end-to-end SIP services.

Application routing overview

- Application servers that support the SIP servlet specification often rely on many applications to provide a complete SIP-based service
- Application routing enables deployers to build complex services out of modular components
- The application router:
 - Determines which application to invoke based on an incoming request
 - Can access external information (database, subscriber service) to help choose the appropriate application
 - Is only responsible for routing and does not implement any application logic

SIP servlet application servers are typically provisioned with many different applications. Each application provides specific functionality, but, by invoking multiple applications to service a call, the deployer can build a complex and complete service. This modular and compositional approach makes it easier for application developers to develop new applications and for the deployer to combine applications from different sources and manage feature interaction. A typical example from traditional telephony is a call-screening application and a call-forwarding application. If the application server receives an incoming INVITE destined to a callee who subscribes to both services, both applications should be invoked.

The application router is a separate component, outside of the SIP container. The container receives initial requests, calls the application router to determine which application to invoke, and then the container calls that application. Once the container has called into an application, that application calls into the appropriate servlet to handle the request based on the application's configuration; for example, using mappings defined in the application's deployment descriptor. By default, WebSphere Application Server uses application startup weights to define the routing order. The JSR 289 specification also defines a Default Application Router (DAR) properties file format and a custom application router application format to describe application routing.

Benefits of using application routing

- Simplifies the process of integrating applications to provide rich services
 - Without needing to develop customized wrappers for application components
- Gives the deployer control over application composition
 - Deployer is responsible for end-to-end services
 - Deployer maintains subscriber information that the application router can access

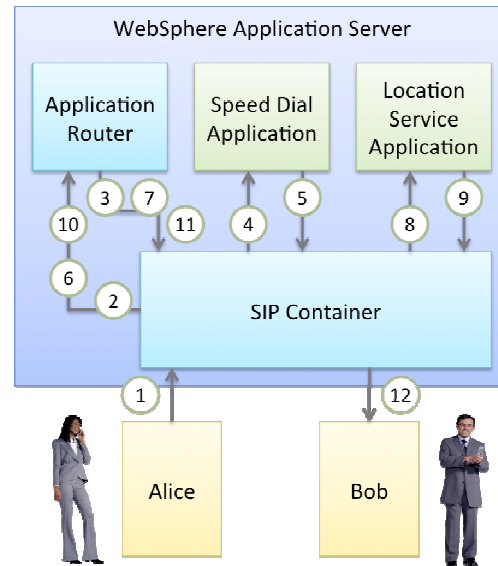
The application router makes it easier to buy a vendor application and invoke its services, without having to write custom wrapper code. This gives the deployer control over how the services behave, rather than leaving integration decisions in the hands of the application developer. Say, for example, you provide telephone service to a large number of subscribers, and a law enforcement agency comes to you with a call tracing and monitoring application that you need to run on a specific subset of your subscribers. Previously, this application was invoked for all subscribers and had to include logic to run only on the required subscribers, or you needed to write an application wrapper to determine whether to invoke the application for a particular user. Now, under the SIP servlet 1.1 specification, all of the logic for determining which users require which application services can be moved outside the scope of the application itself and into the application router.

Application router example

This section contains a sample call flow showing how the application router works with the SIP container to create the correct call chain for a set of applications.

Application routing example (1 of 3)

1. Container receives an INVITE request from Alice attempting to place a call to Bob
2. Container asks the application router which application should be used to process the message
3. Router tells the container that the speed dial application needs to serve this caller first
4. Container invokes the speed dial application

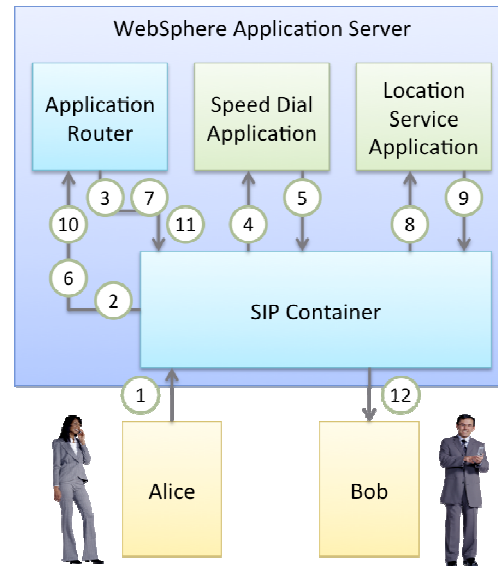


The next few slides walk through a scenario in which Alice wants to make a telephone call to Bob. There are two application services that help make this happen – a speed dial application that supports aliasing telephone numbers to simplify the process of making a telephone call, and a location service application that performs a database lookup to determine the destination of the callee.

When Alice places the call to Bob, the container receives an INVITE request. Since the INVITE does not belong to an existing SIP dialog, the container calls the application router with the request information to determine which application should be invoked to service the request. The router determines that the speed dial application is the first application that needs to serve Alice, so the application router returns the name of the application and some state information, including the originating region, to the container. The container then calls into the speed dial application in the context of the appropriate application session and SIP session.

Application routing example (2 of 3)

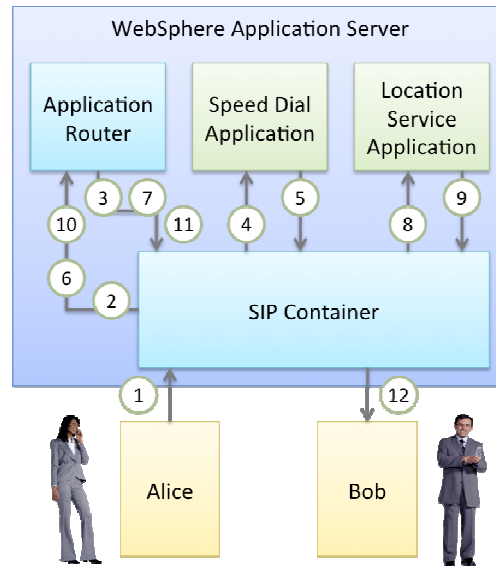
5. Speed dial application looks up the full address corresponding to the speed dial number, proxies the request to that address – the container receives the proxied request
6. Container calls the application router to get the next application
7. Router recognizes that all processing is done for the caller and returns the location service application and caller information to the container



Based on the caller's identity, the speed dial application performs a database lookup to determine Alice's speed dial settings, then maps the speed dial number to the full address of the intended recipient of the call (in this case, Bob), and proxies the request back to the container. The container receives the proxied request and again calls the application router with the request and state information. Based on the stated information, the router is able to determine that the speed dial application has already been invoked and that there are no other applications that need to be called for Alice in the originating region. Assuming that the neutral region does not contain any applications for Alice, the router then moves on to the terminating region and determines that the location service application is the next application that is needed to service the callee – in this case, Bob. The router returns the application name, Bob's identity, the terminating region, and other state information to the container.

Application routing example (3 of 3)

8. Container invokes the location service application, which finds Bob's external location
9. Location service application proxies the request to the container
10. Container calls the router to determine the next application
11. There are no other applications in the chain, so the router returns null
12. Container proxies the INVITE request to Bob



The container invokes the location service application, in context, and the application performs a lookup, based on Bob's identity, to find Bob's location settings. The location service proxies the request to Bob's destination, then the container receives the proxied request and calls the application router to determine which application needs to be invoked. Based on the destination information for Bob's destination, the router is able to determine that there are no other applications that need to be called, so the router returns null to the container. When the container receives null from the application router, it proxies the INVITE request outside of the application server, to Bob's destination.

Summary and reference

This section contains a summary and reference.

Summary

- Application router allows grouping related SIP components together to provide complex services
- Router works in conjunction with the SIP container provides a mechanism to guide SIP requests through the correct call flow

The JSR 289 application router provides a mechanism for removing application composition logic from the applications themselves, moving it to the administrative layer so that the application composition is defined at the configuration level. This simplifies the integration of application components. The application router exists outside the scope of the SIP container but communicated directly with the container to guide SIP requests through the correct call flow.

Reference

- JSR 289 specification
 - <http://icp.org/aboutJava/communityprocess/final/jsr289/index.html>

More details about the JSR 289 application router are available in the JSR 289 specification document shown here.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send email feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WASv8_SIP_ApplicationRouter.ppt

This module is also available in PDF format at: [../WASv8_SIP_ApplicationRouter.pdf](..WASv8_SIP_ApplicationRouter.pdf)

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. (or its affiliates), an IBM Company, in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2011. All rights reserved.