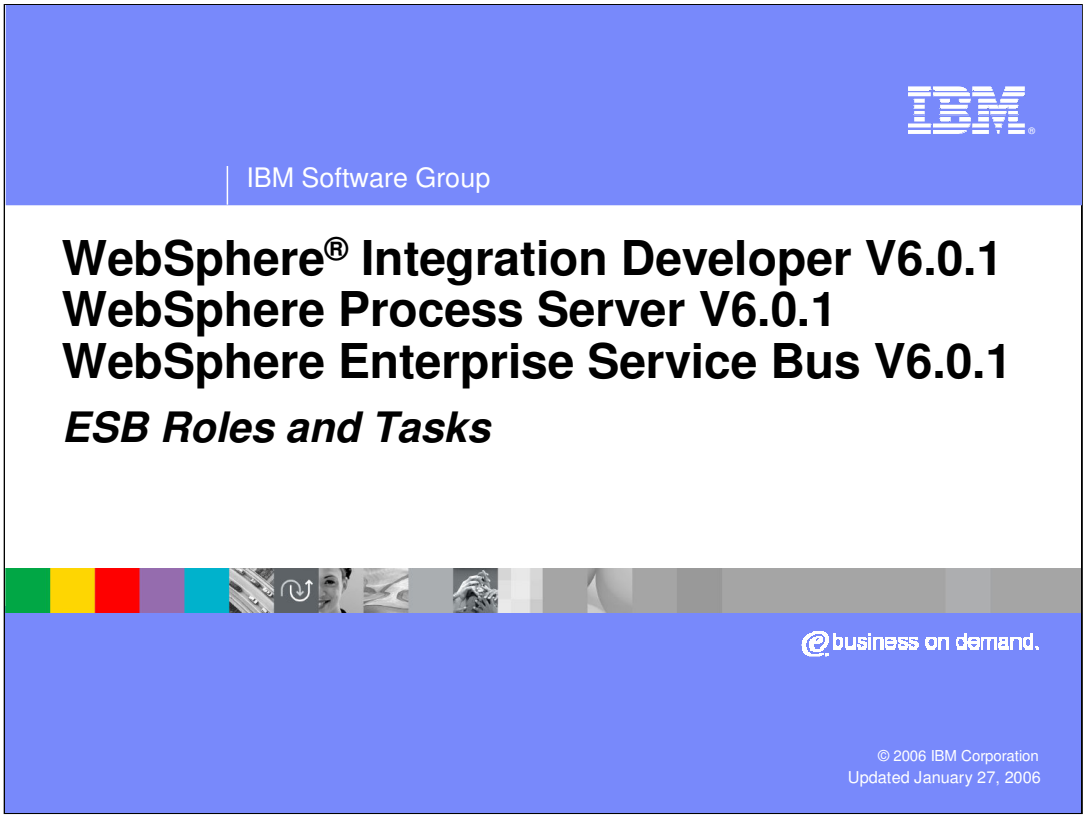
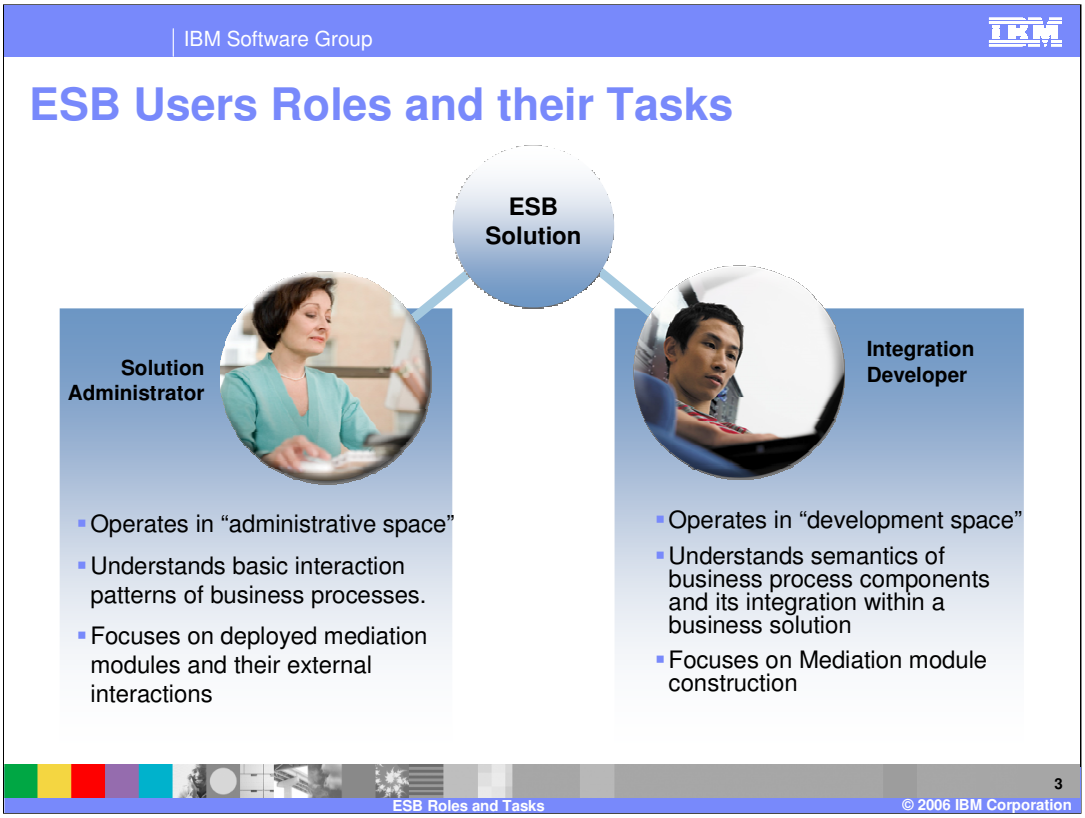


This presentation focuses on the roles and tasks of Enterprise Service Bus user.

Agenda

- **Identify ESB User Roles**
- Describe Tasks Specific to ESB User Roles

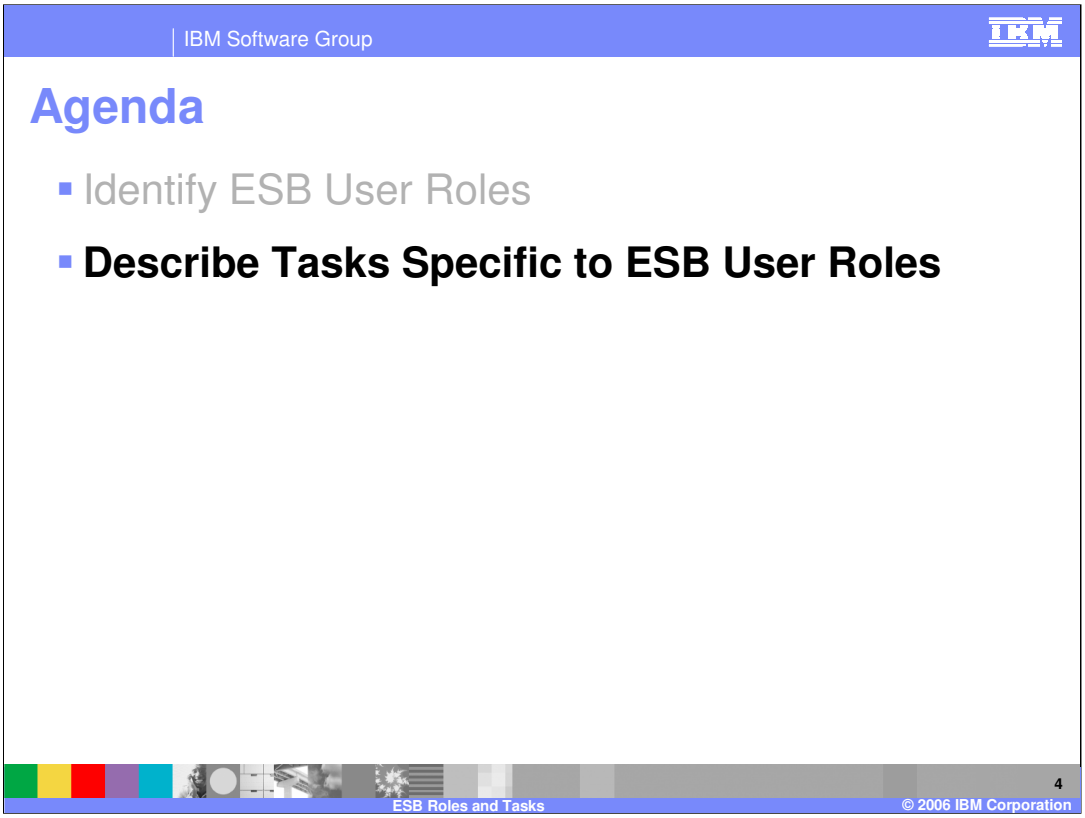
The agenda for this presentation is to specify the ESB user roles and then to describe the tasks for the ESB user roles.



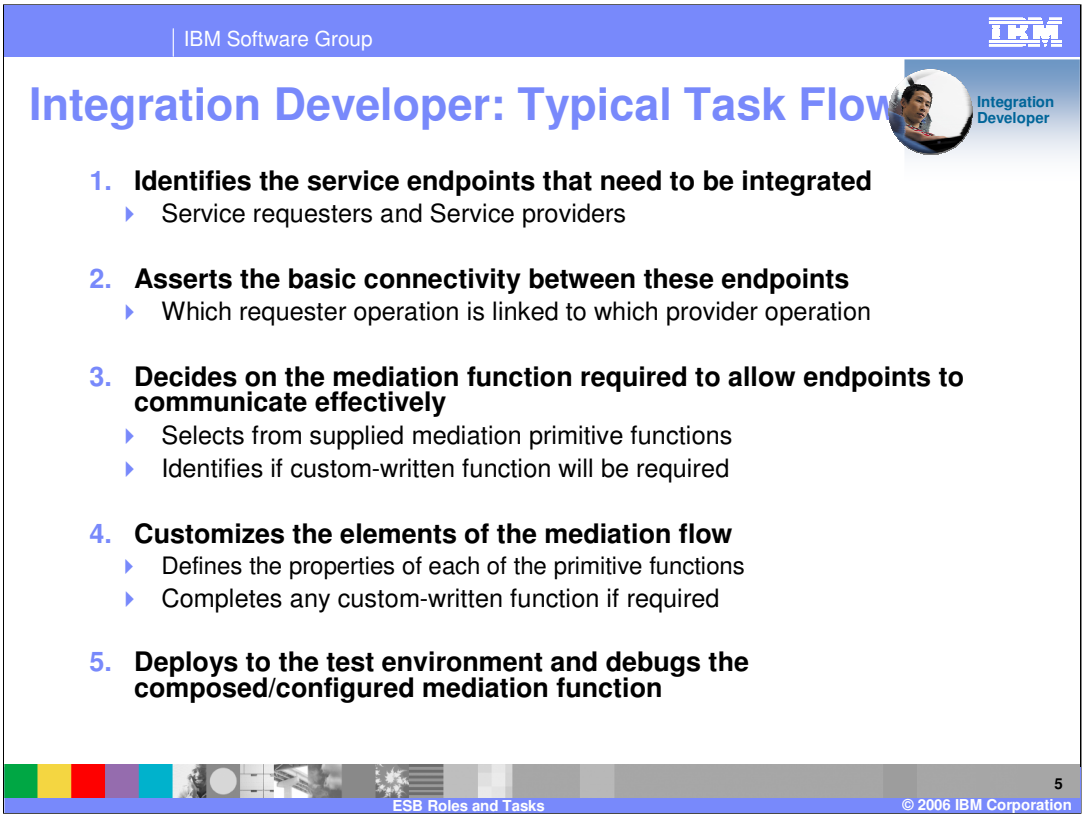
There are two main user roles in the WebSphere ESB space. One is the Integration developer, and the other one is the solution administrator.

The Integration developer uses WebSphere Integration Developer and focuses on creating the mediation module. Integration developers need to understand the integration of the business components that make the business process solution.

The Solution Administrator uses WebSphere ESB or WebSphere Process Server to manage the mediation module within the server. In some situations, the Solution Administrator may need to understand the basic interaction patterns of the business processes to change the routing of the business process if needed.



The next section focuses on the tasks by the two ESB user roles, the Integration developer and the solution administrator.



The basic task flow of the Integration Developer is listed on this slide.

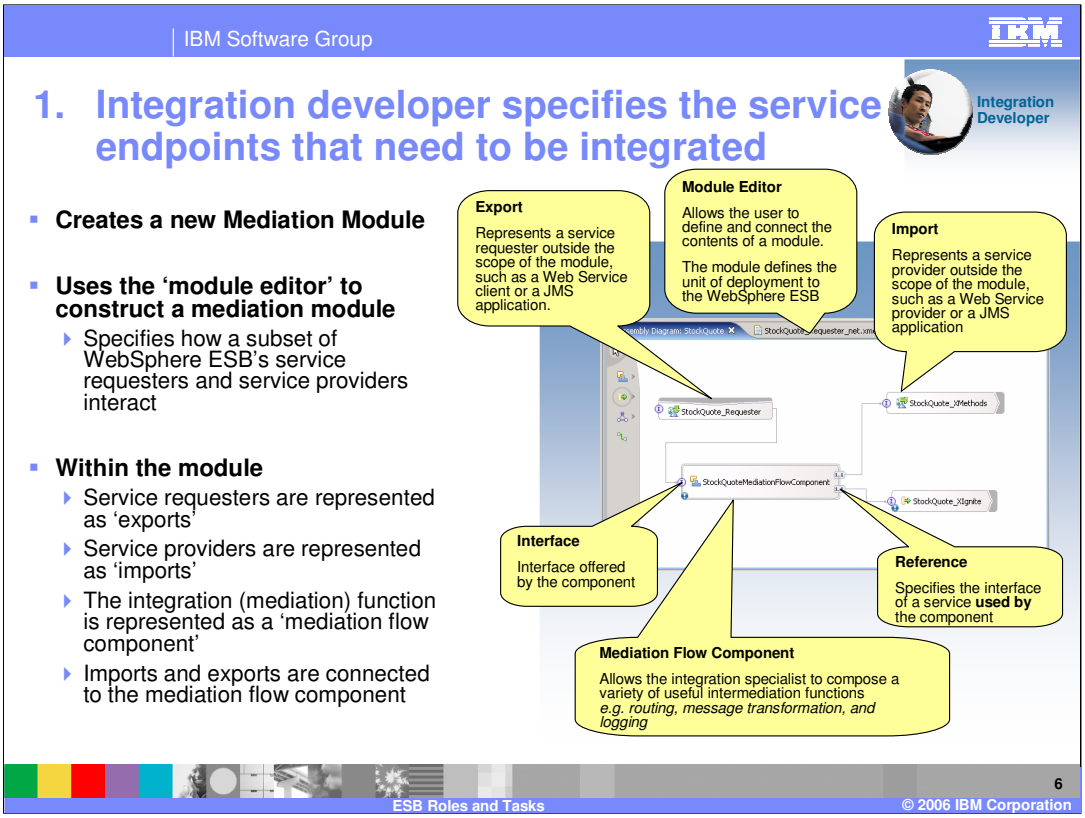
The developer needs to identify the service requestors and providers interacting with the bus. They need to understand the interface and the protocols used by the requestors and providers. This is done using the exports and imports within the mediation module.

The requestors and providers may have different operations. The developer connects the appropriate service requestor operation to the appropriate service provider operation.

Based on what mediation is needed between the request/response message of the requestor and the provider, the developer applies mediation primitives, either built-in or custom, to the request and the response message flow. Any customization of the mediation primitives is also done by the developer.

Finally, the developer can use the test servers to test and debug the message flow through the mediation module.

More details on each of these steps are shown in the next few slides.



The Integration Developer specifies how service requesters and service providers interact by creating a 'Mediation Module'. This is a special kind of 'SCA' module.

Business Module represents the business application; the mediation module represents the mediation of the service message

What makes a mediation module different from any other kind of module is that it MUST contain a Mediation Flow Component.

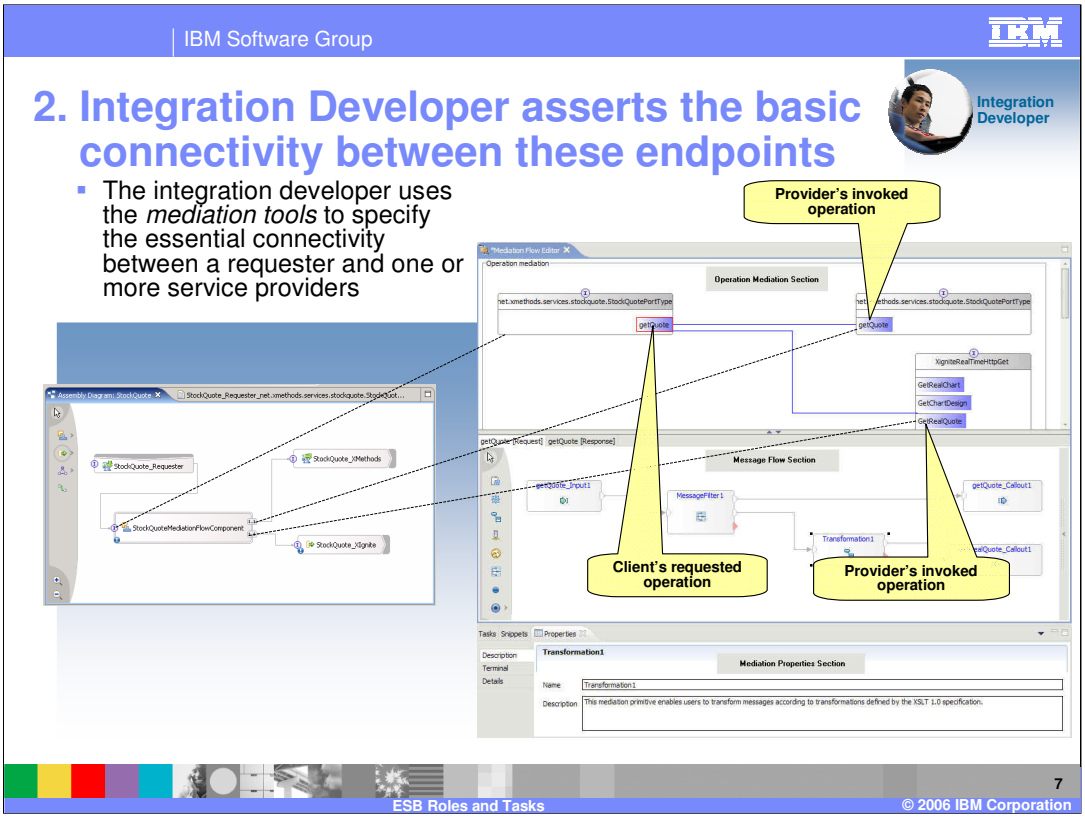
It may also contain:

- imports, which are the standard way that the Service Component Architecture describes 'service requesters'

- exports, which are the standard way the Service component architecture expresses the existence of service providers

- wires, which connect the component and the imports and exports

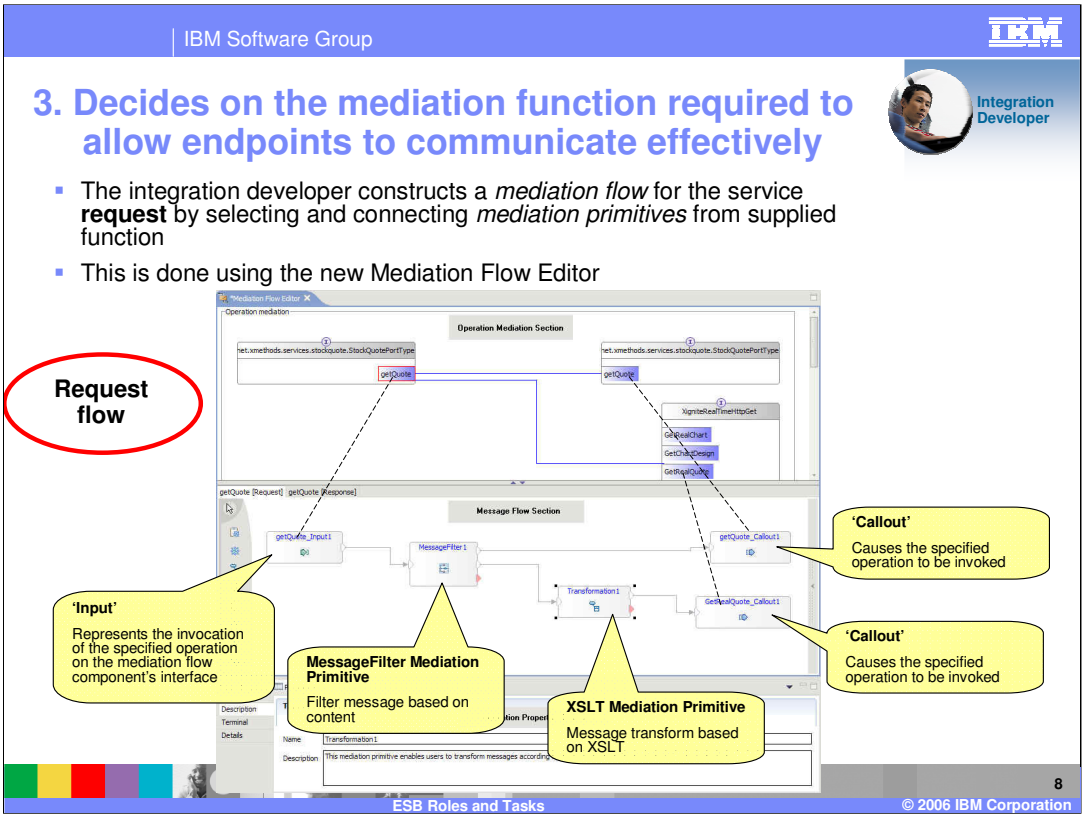
- Other components, which are used to create custom mediation primitives



The Integration Developer now defines the behavior of the Mediation Flow component.

Initially, the basic context, defined by **interfaces** and **references** of the Mediation Flow Component, is present.

Each blue line in the 'Operation Mediation' section (the upper pane) specifies the connectivity between an operation on the component's interface and an operation on one of the mediation flow component's references



The Integration Developer decides on the mediation functions required between the message flow of the service end points. The mediation functions are specified by using mediation primitives and wiring them together for the message flow.

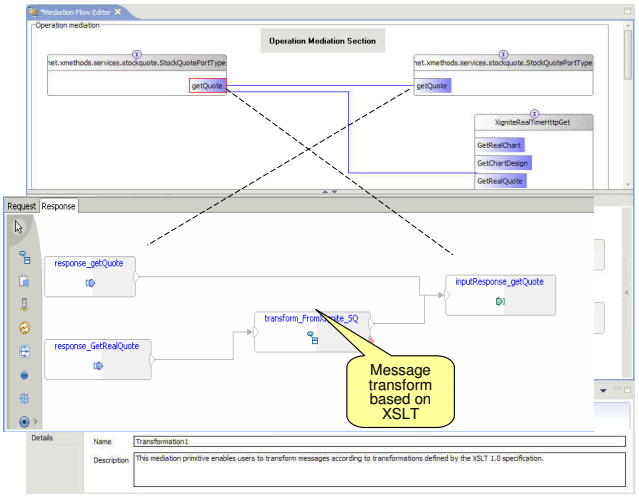
Different mediation primitives perform different functions on the message like routing, transforming, logging, or any custom processing of the message. There are several built-in mediation primitives available in the WebSphere Integration Developer. In addition, the developer can create custom mediation primitives to handle any custom processing of the message.

The developer specifies the required mediation functions for the request flow, shown on this slide, and the response flow shown in the next slide.

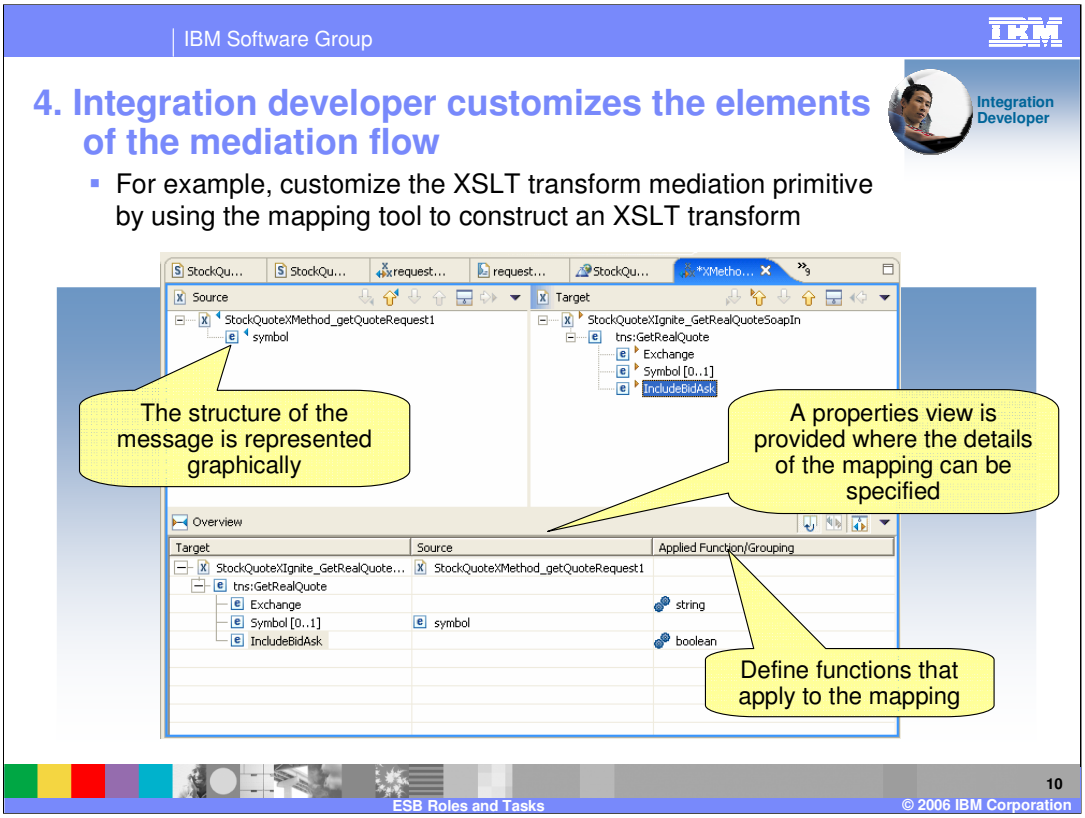
3. Decides on the mediation function required to allow endpoints to communicate effectively

- The integration developer constructs a *mediation flow* for the service **response** by selecting and connecting *mediation primitives* from supplied function

Response Flow



This slide shows the response flow between the service end points. As indicated before, the Integration developer has the ability to provide different mediation services to the request flow and the response flow of the message.

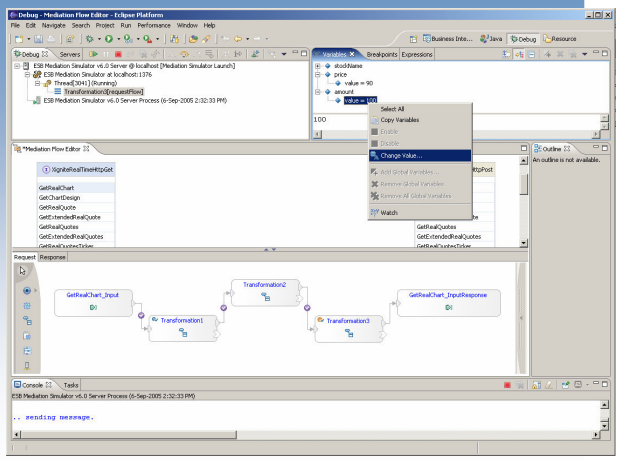


The Integration developer will need to customize the mediation primitives. For example, in case of the XSL transform mediation primitive, the developer can create the XSLT map to transform the part of the message body to the appropriate output interface.

5. Debugs the composed or configured mediation function: Mediation Visual Debug

Use the visual debugging tools to debug a solution

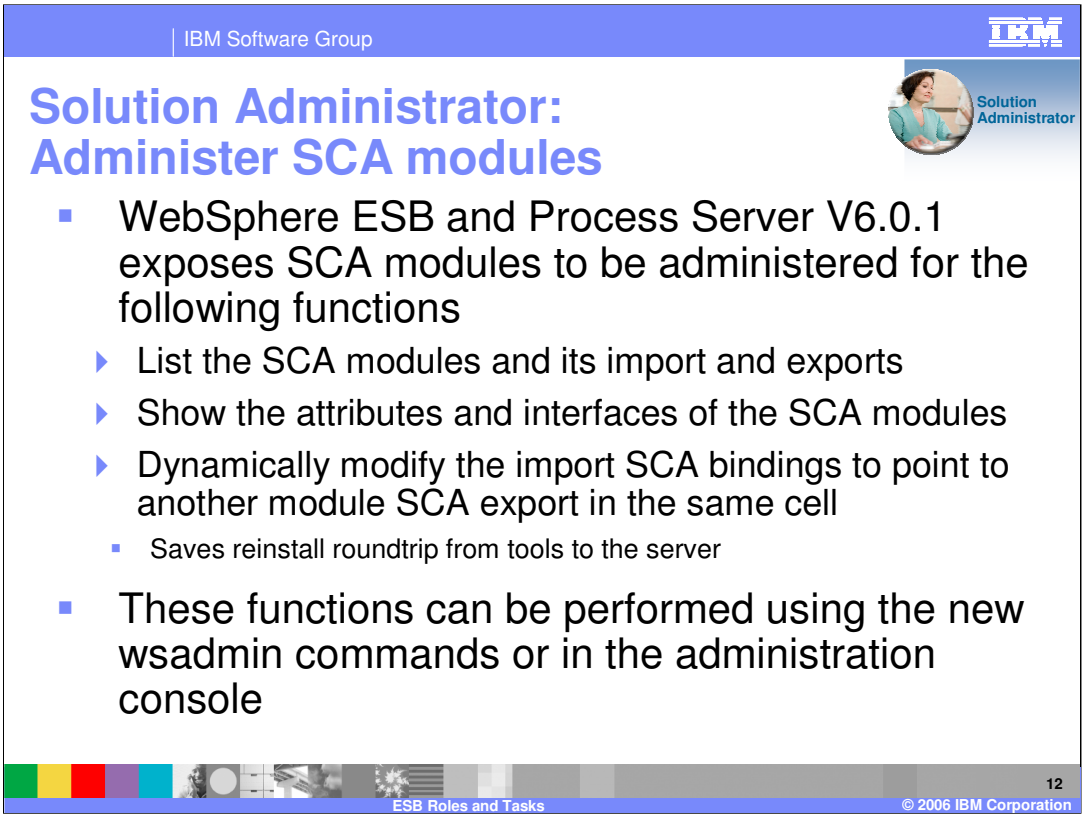
- Debug mediation flows using an in-place visual debugger
- Breakpoints can be added, step into, through, or over areas of interest while inspecting the values of the messages



Can use the built-in or remote WebSphere ESB or Process Server for testing

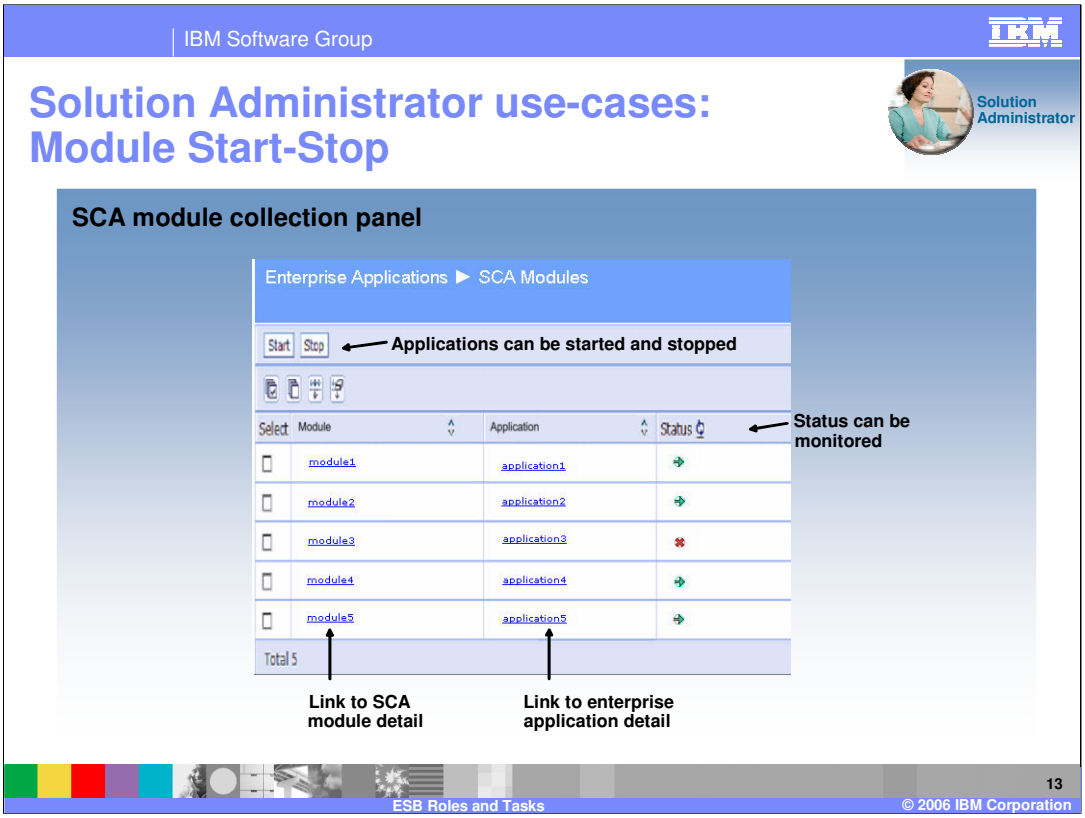
Finally, the integration developer can test or visually debug the message flow using the WebSphere ESB or WebSphere Process Server test servers, or any remote server.

Visual debugging allows the developer to insert breakpoints at different locations in the flow of the message to allow inspecting the contents of the message.



WebSphere Process Server and WebSphere ESB expose the SCA modules to solution administrators. Administrator can list and show the SCA modules within the server or the cell. More important is the ability to dynamically modify the import SCA bindings to point to another module SCA export in the same cell that has the same interface. This allows the administrator to change the routing of the message flow and the ability to replace the SCA module with a newer implementation, without the need to go to the development tool and change it there.

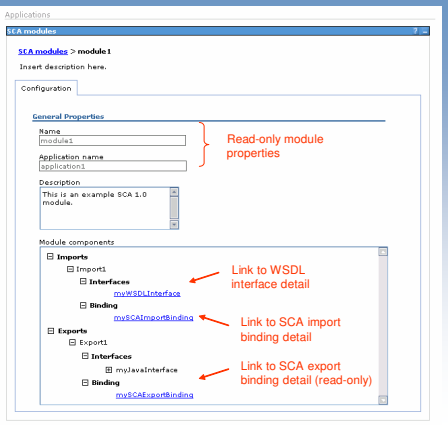
All the administration functions can be performed using the administration console or the wsadmin command line tool.

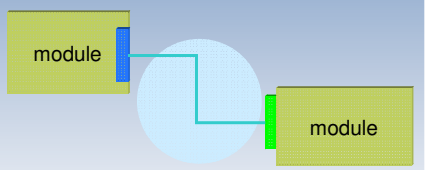


At the operational level, each module maps to a single WebSphere application. The administrative console allows you to easily navigate to these applications and to stop or start them.

Solution Administrator use cases: Module connectivity

- One module can be directed to interact with another by 'wiring' an import to an export
  - This is supported by the WebSphere ESB 'default bindings'
  - Service requests from the import will be fulfilled by the specified export
- The administrator can change this wiring post-deployment
  - This allows the administrator to change how modules are connected by altering the 'wiring' of an import





Wire

Configures the connection between a default module import binding and a default module export

The details of the module (including its imports and exports) can be explored, and for default SCA import bindings, the administrator can dynamically modify them to point to the export of another SCA module with the same interface.

Section

Summary and References

The next section covers the summary of this presentation.

Summary

- There are two ESB user roles:
  - Integration developer uses WebSphere Integration Developer to create mediation service applications
  - Solution administrator manages these applications in the WebSphere ESB and Process Server runtime
- WebSphere Integration Developer tools enhancements provide easy visual way to compose and test the mediation service application
- WebSphere ESB and Process Server provides new administration capabilities for the SCA modules

In summary, there are two ESB user roles, the Integration Developer and the solution administrator, each with their own set of tasks, as described in the presentation.

Template Revision: 11/22/2005 12:10 PM

Trademarks, Copyrights, and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM(logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

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

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.