

# Planning your Migration From IBM WESB to IBM Integration Bus

Callum Jackson

IBM Software Services for WebSphere

IBM Hursley, UK

## Please Note

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including 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 results similar to those stated here.

# Agenda

- Approaches to conversion
- WESB Convert Tool
  - What's new in the open beta
  - WESB Convert Tool Demo
  - Extending the conversion tool

# Approaches to conversion

# Possible approaches to conversion

There are **two** key parts to the conversion:

- Future integration solutions
- Existing integration solutions

# Possible approaches to conversion

There are two key parts to the conversion:

- **Future integration solutions**
- Existing integration solutions

Look to move future development of integration solutions to IBM Integration Bus.

Limiting future development on the WebSphere ESB platform will minimise the potential conversion work. If future development is required on WebSphere ESB, then structuring this in a IIB friendly manner is sensible to allow straight forward conversion.

# Possible approaches to conversion

There are two key parts to the conversion:

- Future integration solutions
- Existing integration solutions

Understand the existing WebSphere ESB estate and determine the best approach for conversion to IIB.

Conversion of existing integrations is normally handled using a combination of the following strategies:

- Run in parallel
- Gradual migration toward IIB
- Immediate migration towards IIB

# Possible approaches to conversion

- **Run in parallel – Wait**
  - Move future development of integration solutions to IIB.
  - Start training activities to understand IIB as the target platform for future applications
  - Wait to convert existing WebSphere ESB estate.
  - Build long term plan for conversion to IIB.
  - Consider the infrastructure and license requirements for running in parallel
- **Gradual migration toward IIB**
  - Initiate conversion pilot to build skills and learn lessons.
  - Phased conversion of integration solutions.
  - Run in parallel until conversion complete.
  - Consider the infrastructure and license requirements for running in parallel
- **Immediate migration towards IIB**
  - Determine the sizing of the overall migration and the associated risk.
  - Determine if a fall back strategy is required
  - Generally only recommended for customers with a limited WebSphere ESB deployment or at the early stages of deployment



# Considerations for conversion

- **Topology**
  - From WESB Golden Topology to..
  - The choice of topology depends on the customers specific needs.
  - Starting point would an active – active topology with multiple parallel IIB nodes. More later..
- **Monitoring**
  - Similar granularity of monitoring.
  - IBM or 3<sup>rd</sup> party external monitoring solutions.
  - IIB built-in monitoring and statistics support.
- **Security**
  - WESB based on WAS security model.
  - IIB supports LDAP, SSL, User Token, SAML etc.
  - Security gateway..
- **Administration and operations**
  - WESB based on WAS admin model
  - IIB supports scripting for administration and operations.
- **Applications**
  - Analyze your WESB applications and categorize according to the ease and approach of conversion. ....
- **Architectural approach**
  - Consider your WESB architectural approach and plan an architectural approach in IIB.
  - Similarity of approach will effect the conversion strategy for applications.

# Application Conversion Categories

- **Category 1: Tool accelerated**
  - The conversion tool is a sensible approach and the resulting applications will require limited customization.
  - Extensive use of the tool as-is for conversion.
- **Category 2: Tool assisted**
  - A standard template is used across multiple integration solutions and customization of the conversion tool or pattern templates represent a sensible approach to accelerate the conversion.
  - Extensive use of the tool with customizations.
- **Category 3: Manual conversion**
  - The core functionality is available within the product.
  - The integration solution may contain extensive custom use of custom code.
  - Due to the complexity of the solution a literal mapping of primitive to corresponding nodes would provide a sub-standard solution.
  - The customer may wish to combine conversion with a change of architectural approach.
  - Some use of the tool to kick start conversions.
- **Category 4: Custom solution –**
  - Similar to category 3, custom coding need in additional to core functionality in the product.

# Conversion Offerings from IBM SW Services and Partners

## ISSW Summary offerings

### Conversion Introduction (CI)

GOAL: Client wishes to understand the IBM Integration software strategy in more detail and what they might need to consider to plan for a successful conversion from WESB.

FORMAT: Remote presentations and conference calls (<= 4 hours duration in total)

### Conversion Planning Workshop (CPW)

GOAL: Client wishes to understand more detail on the implications of conversion as it specifically relates to their own deployment of WESB in order to build an initial conversion roadmap or plan.

FORMAT: Conducted on site in IBM consultant led workshop (3-5 days)

### Conversion Quick-Start (CQS)

GOAL: Client participated in a CPW (or equivalent) which has resulted in selection of this offering, which is a typical "Quick Start" style engagement to accelerate client adoption.

FORMAT: This activity will be conducted at client location by ISSW in IBM consultant led workshops and hands-on mentoring. (typically 2-4 weeks).

### Conversion Quick-Win Pilot (CQWP)

GOAL: Client participated in a CPW (or equivalent) which has resulted in selection of this offering, for delivery of a well-defined pilot project to accelerate conversion and build confidence

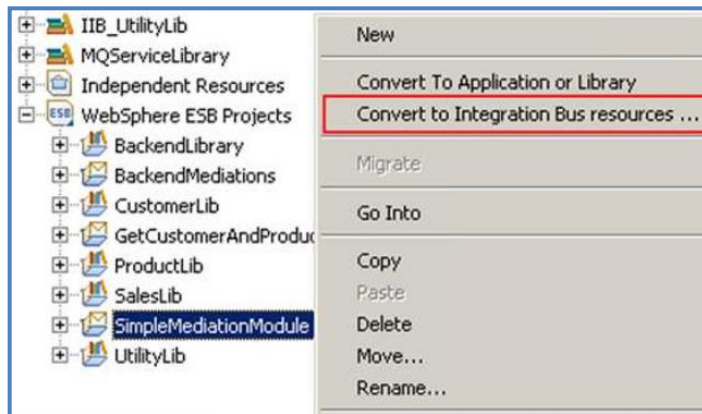
FORMAT: On site pilot project within an agreed time frame

# WESB to IIB Conversion tooling

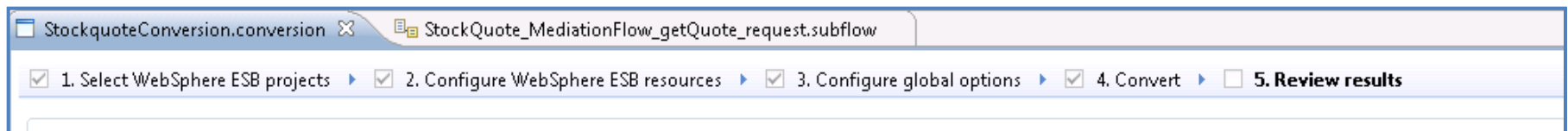
Accelerate conversion of WebSphere  
ESB source artefacts to IIB.

# Conversion from WebSphere Enterprise Service Bus

- Built-in conversion tools for WESB source assets
  - Accelerate conversion of WESB source assets to IIB source assets.
  - Tool improvements increase breadth and depth on conversion. Reducing further work.
  - Open framework for user and partner extensions to allow customization of the tool.

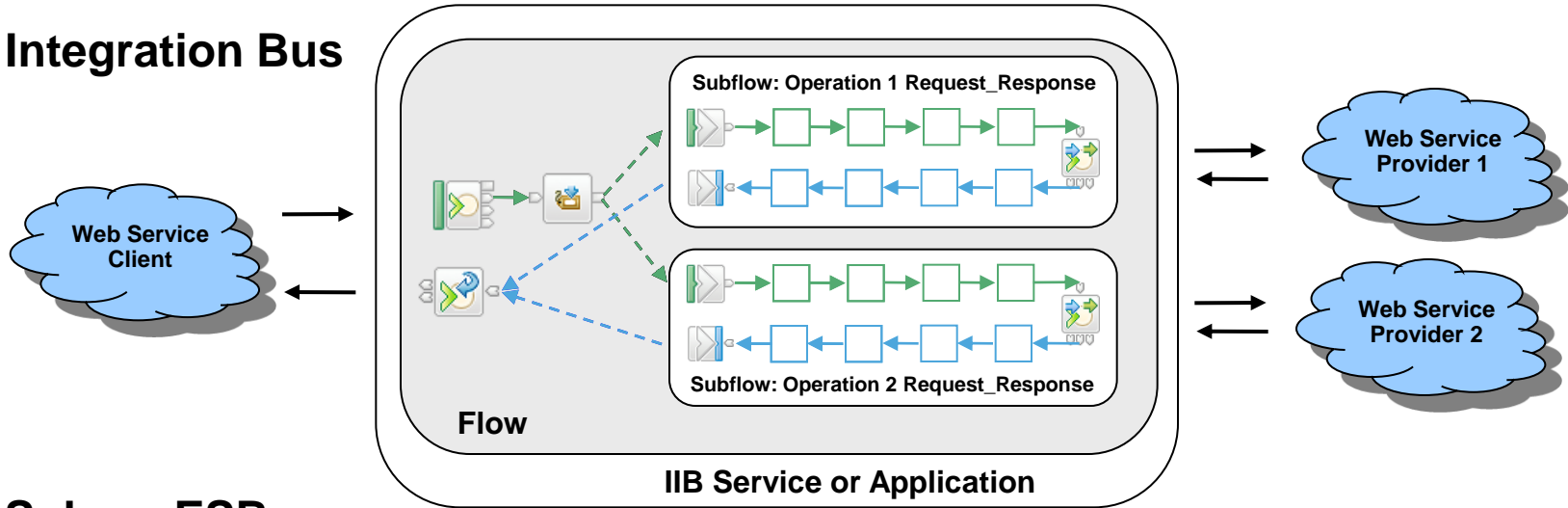


- Simple workflow creates IIB resources
  1. Export WESB PI from IID
  2. Import mediations into Eclipse Toolkit
  3. Right-click “convert” task to start conversion
  4. Follow guided editor to generate resources
  5. Task List will identify remaining manual steps
  6. Iterate as necessary

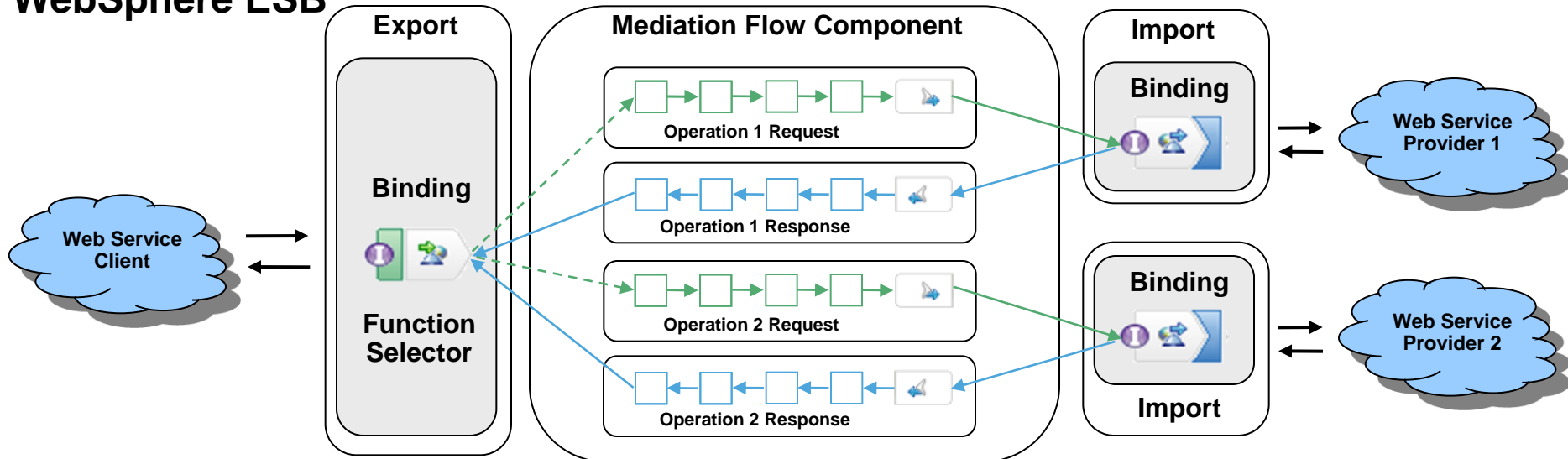


# Architectural Concepts

## IBM Integration Bus



## WebSphere ESB



# What's new in the tool ..

- At IIB v9 the tool offered some capability to convert web services based integrations.
  - Single export, single mediation component, single import.
  - Web services binding only.
  - Built-in converters for few mid-flow primitives
- **New capabilities in IIB open-beta (current) expand the breadth and depth of conversion**
  - Convert multiple exports with any binding
  - Convert multiple connected mediation components with multiple interfaces
  - Built-in converters for most mid flow primitives
  - WESB style encapsulation of logic
  - Enhanced Documentation
- What next ?
  - Look out for further enhancements as they appear in the IIB open-beta :

<https://ibm.biz/iibopenbeta>

<https://ibm.biz/iibwesbconvert>

*IBM's plans, directions,  
and intent are subject to  
change or withdrawal*

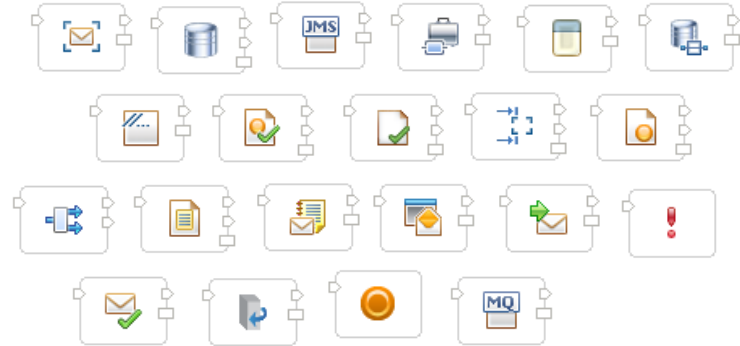
# What's new in the tool ..

Some built-in primitive converters



*IBM's plans, directions, and intent are subject to change or withdrawal*

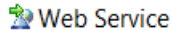
**New built-in primitive converters**



Single Export Single MFC Single Import



**Built-in converter for WS binding only**

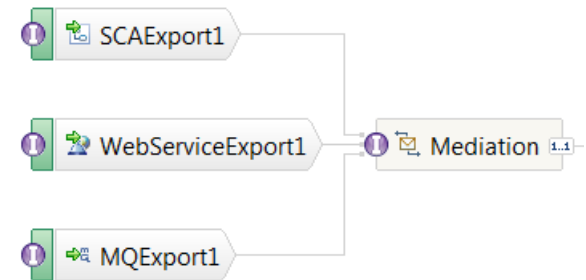


**New built-in binding converters**

**IIB v9(GA)**

- |             |                          |
|-------------|--------------------------|
| HTTP        | E-mail                   |
| Generic JMS | Flat File                |
| JMS         | JDBC                     |
| MQ          | JD Edwards EnterpriseOne |
| MQ JMS      | PeopleSoft               |
| SCA         | SAP                      |
|             | Siebel                   |

**New Multiple Export**



**New Multiple Mediation Component**



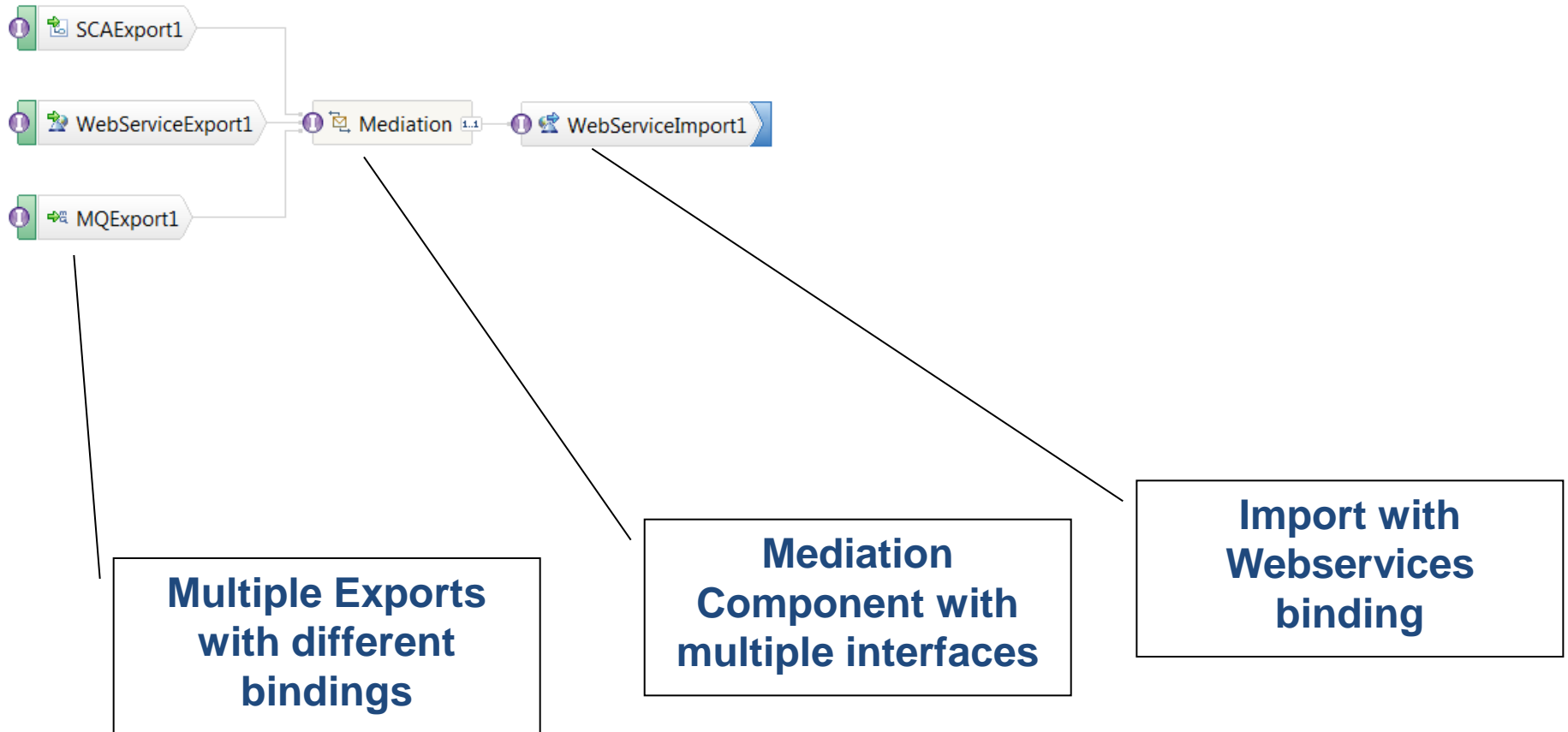
**IIB open-beta (current)**

Look out for further updates at : <https://ibm.biz/iibopenbeta>

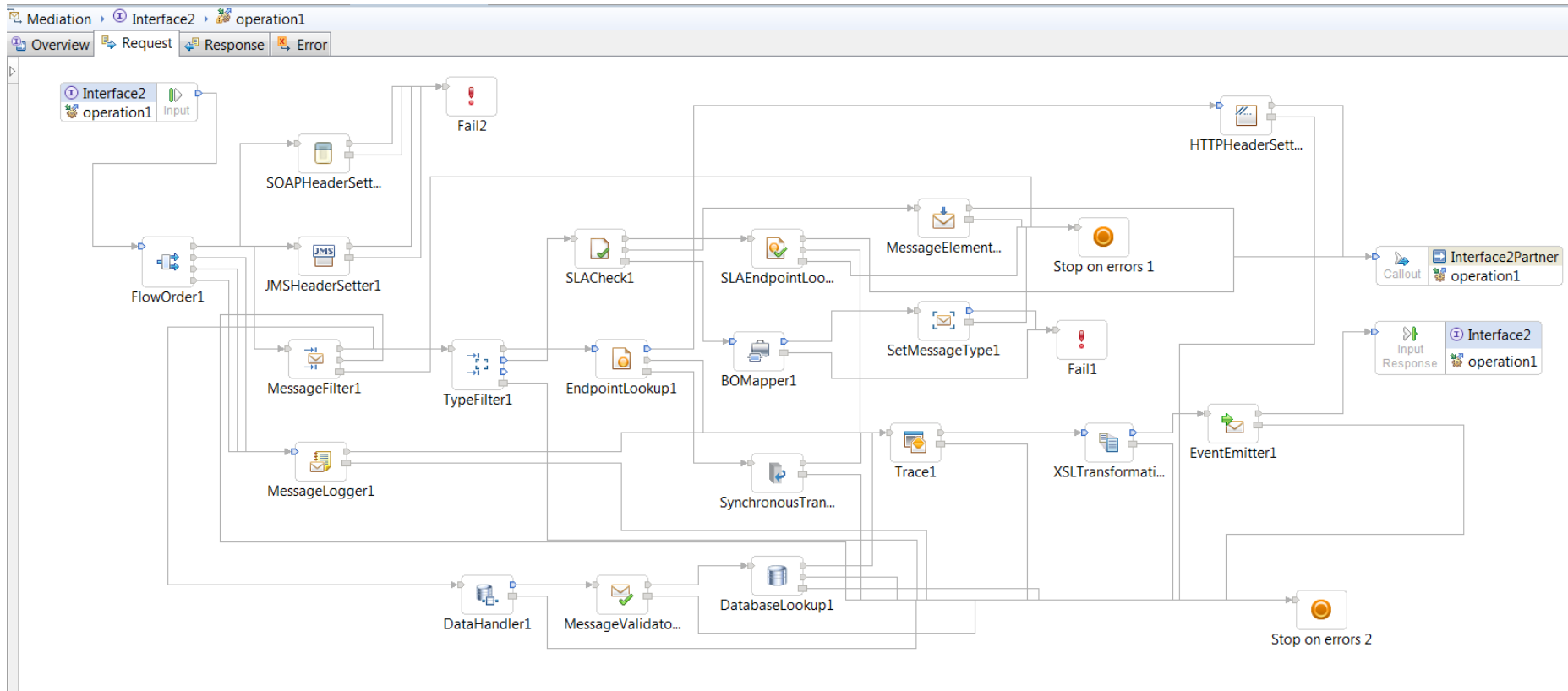


# WESB Convert Tool

# Example Mediation Module - IID

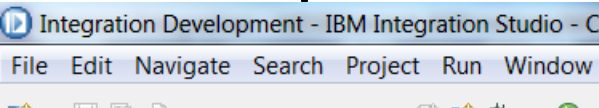


# Example Mediation Module - IID



**Message Flow with wide variety of primitives and complex wiring**

# Import WESB source projects into IIB Studio



Application Development

Application Development

[New Application...](#)

[New Integration Service...](#)

[New Library...](#)

New

Copy

Paste

Delete

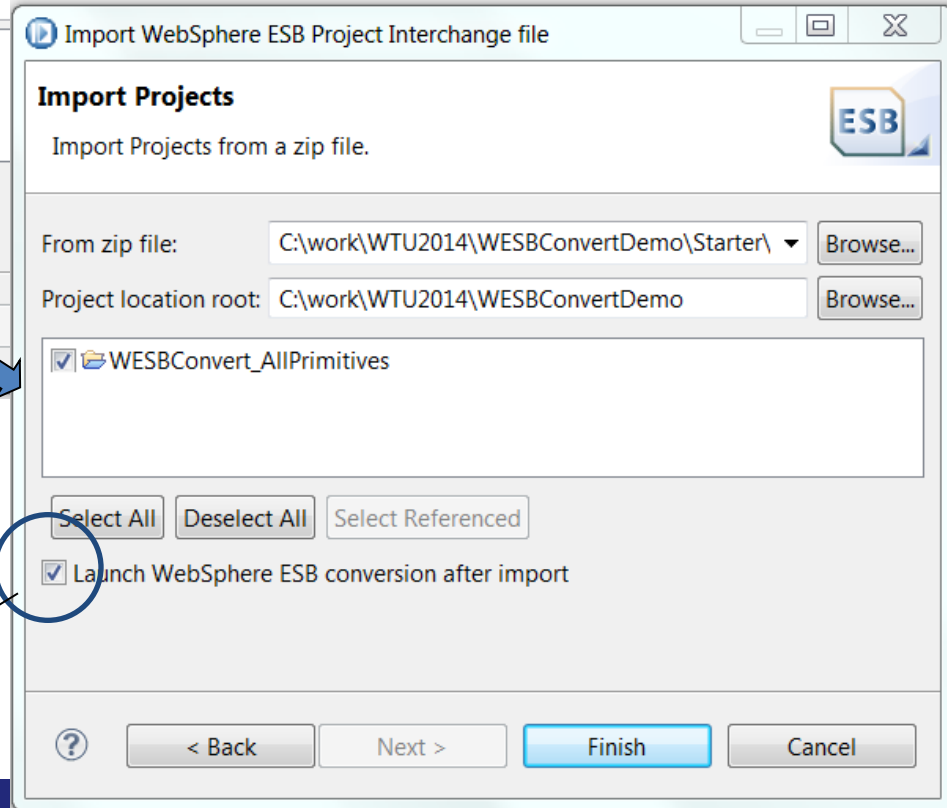
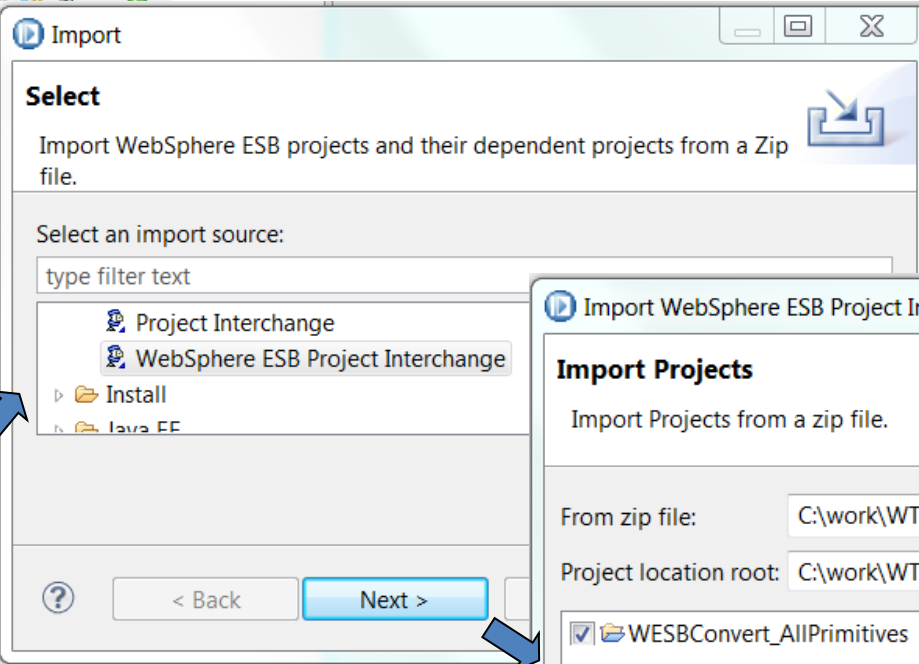
Move...

Rename...

**Import...**

Export...

Refresh



**Import WebSphere ESB source projects into IBM Integration Studio**

**Launch Conversion tool directly from import**



# Create WESB Conversion tool session

**New WebSphere ESB Conversion Session**

To create a new WebSphere ESB conversion session file, select a project and specify a conversion session filename.

Project: WESB\_Conversions

Conversion session filename: wtudemo

The conversion session file contains information about the WebSphere ESB projects that you want to convert, your WebSphere ESB conversion configuration choices, and your conversion results.

**Create a new conversion session.**

**This will persist any state associated with the conversion, such as TODO tasks generated by the conversion.**



# WESB Conversion tool in IIB Studio



\*wtudemo.conversion

1. Select WebSphere ESB projects
2. **Configure WebSphere ESB resource options**
3. Configure global conversion options
4. Convert WebSphere ESB resources
5. Review results

Configure WebSphere ESB resource options. Filter resources by name or by using a wildcard character.

WebSphere ESB Resources:

type filter text

- WESBConvert\_AllPrimitives
  - Module
    - Export 'MQExport1'
    - Export 'SCAExport1'
    - Export 'WebServiceExport1'
    - Component 'Mediation'
    - Import 'WebServiceImport1'
    - XSD/WSDL
    - XML Maps (some maps are selected for conversion)

Select the XML maps that you want to convert. Double-click on one map to see information on its usage analysis.

XML Map	Usage
<input checked="" type="checkbox"/> XSLTransformation1_req_1.map - xslt/XSLTransformation1_req_1.map	Mediation.component
<input type="checkbox"/> XSLTransformation1_req_2.map - xslt/XSLTransformation1_req_2.map	
<input type="checkbox"/> XSLTransformation1_req_4.map - xslt/XSLTransformation1_req_4.map	
<input checked="" type="checkbox"/> XSLTransformation1_req_5.map - xslt/XSLTransformation1_req_5.map	Mediation.component
<input type="checkbox"/> XSLTransformation2_req_1.map - xslt/XSLTransformation2_req_1.map	

Select all Deselect all

Conversion notes

All selected WebSphere ESB XML maps will be converted to IBM Integration Bus XML maps.

[< Previous](#)

[Next >](#)

**Choose resources to convert.  
eg. By default only maps  
referenced in the module will be  
converted.**

# WESB Conversion tool in IIB Studio



\*wtudemo.conversion

1. Select WebSphere ESB projects | 2. Configure WebSphere ESB resource options | **3. Configure global conversion options** | 4. Convert WebSphere ESB resources | 5. Review results

Configure global conversion options. Add extensions for those resources for which you want to use your own conversion code.

## Conversion Result

Specify how the conversion result should be recorded.

Merge new conversion results with the results from previous runs of this conversion session

## Mediation Primitive Converters

Each mediation primitive will be converted to a message flow node or subflow. You can supply your own converter to convert mediation primitives. Double-click on the 'Usage' column of a mediation primitive to see information on its usage analysis.

Mediation ...	Convert to	Usage	Converter class
BOMapper	Map	Mediation.component	Built-in converter
Callout	Output	Mediation.component	Built-in converter
CalloutResp...	Input	Mediation.component	Built-in converter
DataHandler	JavaCompute	Mediation.component	Built-in converter
DatabaseLo...	Map	Mediation.component	Built-in converter

## Export and Import Binding Converters

Each export or import binding will be converted to a message flow node or subflow. You can supply your own converter to convert an export or import binding. Double-click on the 'Usage' column of an export or import binding to see information on its usage analysis.

Binding	Convert to	Usage	Converter class
SCA Export	MQInput	SCAExport1.export	Built-in converter
Jax/Ws Exp...	SOAPInput	WebServiceExport1.export	Built-in converter
Jax/Ws Imp...	SOAPRequest	WebServiceImport1.import	Built-in converter
MQ Export	MQInput	MQExport1.export	Built-in converter

[< Previous](#)

**Choose conversion options.  
Optionally replace Built-in converters  
for primitives and bindings with  
custom converters.**



- 1. Select WebSphere ESB projects
- 2. Configure WebSphere ESB resource options
- 3. **Configure global conversion options**

Configure global conversion options. Add extensions for those resources for which you want to use your own conversion code.

**Conversion Result**

Specify how the conversion result should be recorded.

Merge new conversion results with the results from previous runs of this conversion session

**Mediation Primitive Converters**

Each mediation primitive will be converted to a message flow node or subflow. You can supply your own converter for a primitive to see information on its usage analysis.

Mediation Primitive	Convert to	Usage	Converter class
JMSHeaderSetter	JMSHeader node or Mapping node	Mediation.component	Built-in converter
MessageElementSetter	JavaCompute	Mediation.component	Built-in converter
MessageFilter	Route	Mediation.component	Built-in converter
MessageLogger	Map or Trace	Mediation.component	Built-in converter
MessageValidator	Validate	Mediation.component	Built-in converter

Use the built-in converters  
**And optionally**  
 Extend the tool with custom converters for specific primitives or Export/Import bindings

**Export and Import**

Each export or import...  
 as an export or import...

Select Converter Class

Choose a Java class which extends AbstractMediationPrimitiveConverter. For an example of how to create a user-defined converter class, see [WebSphere ESB user-defined converters sample](#).

Converter Java class

# WESB Conversion tool in IIB Studio

\*wtudemo.conversion

1. Select WebSphere ESB projects ▶ 2. Configure WebSphere ESB resource options ▶ 3. Configure global conversion options ▶ **4. Convert WebSphere ESB resources** ▶ 5. Review r

Review the configuration details for this conversion session. Start the conversion of your WebSphere ESB resources. Double-click on a map entry to verify the list of maps that are selected for conversion.

Summary of the conversion configuration:

WebSphere ESB resource	Type	Option	Value
WESBConvert_AllPrimitives	Project	XML Maps to convert	xslt/XSLTransformation1_req_1.map,xslt/XSLTransformation1_req_5.map
	Global option	Merge conversion result	false

[Start conversion...](#)

[< Previous](#) [Next >](#)

**Final summary of conversion**

**Start the  
conversion process**

# Conversion Tool – IIB open-beta Example

1. Select WebSphere ESB projects | 2. Configure WebSphere ESB resource options | 3. Configure global conversion options | **4. Convert WebSphere ESB resources** | 5. Review results

Review the configuration of selected for conversion.

Summary of the conversion

WebSphere ESB resource

WESBConvert\_AllPrimitive

Start conversion...

< Previous

Review the conversion results. Complete all the to-do tasks to finish converting WebSphere ESB resources.

1. Select WebSphere ESB projects | 2. Configure WebSphere ESB resource options | 3. Configure global conversion options | **4. Convert WebSphere ESB resources** | 5. Review results

Show all | Show resources with incomplete tasks

Conversion summary

- All to-do tasks
- All problems in the converted projects
- Detailed conversion results

Messages and tasks

- Complete conversion of the SLAEndpointLookup primitive 'SLAEndpointLookup1' to Regi
- Consider if correct HTTP Header type used in conversion of the HTTPHeaderSetter primi
- Complete conversion of the DatabaseLookup primitive 'DatabaseLookup1' to Message

CWWOC6041I: Consider if correct HTTP Header type used in conversion of the HTTPHeaderSetter primitive 'HTTPHeaderSetter1' to HTTPHeader node 'HTTPHeaderSetter1'.

Actions from the HTTP Header Setter primitive 'HTTPHeaderSetter1' were converted to HTTPRequest header actions in HTTPHeader node 'HTTPHeaderSetter1'. In IBM Integration bus the HTTPHeader node can operate on four different HTTP headers in the message assembly. HTTPRequest, HTTPInput, HTTPResponse and HTTPReply headers.

header is appropriate and change to HTTPInput, if necessary.

completing a to-do task for HTTP Header Setter primitives.

Integration Development - IIB\_WESBConvert\_AllPrimitives/IIB\_WESBConvert\_AllPrimitives.msgflow - IBM Integration Studio

File Edit Flow View Palette Navigate Search Project Abbot Run Window Help

75%

Test: [Icons]

Application Development

- IIB\_WESBConvert\_
  - Schema Definit
  - WSDL Definitic
  - Flows
  - IIB\_WESBCo
  - Subflows
  - Maps
  - Java
  - Other Resource
- Independent Reso
  - WESB\_Conversi
  - WebSphere ESB P
  - WESBConvert\_

WebSphere MQ

MQTT

JMS

HTTP

Web Services

SaaS Providers

SCA

WebSphere A...

Routing

.NET

Transformation

Construction

Database

File

Email

TCPIP

CORBA

MQExport1

SCAExport1

WebServiceExport1

RoutesToOperation

Mediation\_Interface2\_operation1

Mediation\_Interface2\_operation1\_Request

WebServiceImport1

WebServiceImport1\_RouteToRespo

Mediation\_Interface1\_operation2

Mediation\_Interface1\_operation2\_Request

Convert tool produces IIB services and applications and a list of documented follow-on tasks for the user.

# WESB Conversion tool in IIB Studio

\*wtudemo.conversion

1. Select WebSphere ESB projects | 2. Configure WebSphere ESB resource options | 3. Configure global conversion options | 4. Convert WebSphere ESB resources | 5. Review result

Review the conversion results. Complete all the to-do tasks to finish converting WebSphere ESB resources.

Show all  Show resources with incomplete tasks

- Conversion summary
  - All to-do tasks
  - All problems in the converted projects
- Detailed conversion results
  - Source: WESBConvert\_AllPrimitives
    - Converted to:IIB\_WESBConvert\_AllPrimitives
    - Converted to:IIB\_WESBConvert\_AllPrimitivesJava
      - Mediation\_Interface2\_operation1\_Request\_MessageElementSetter2.java - /IIB\_W

WebSphere ESB files	Integration Bus files
Project WESBConvert_AllPrimitives	Project IIB_WESBConvert_AllPrimitives
FTPEXport.wsdl	FTPEXport.wsdl FTPEXport_InlineSchema1.xsd
FlatFileExport2.wsdl	FlatFileExport2.wsdl FlatFileExport2_InlineSchema1.xsd
Interface1.wsdl	Interface1.wsdl Interface1_InlineSchema1.xsd
Interface2.wsdl	Interface2.wsdl Interface2_InlineSchema1.xsd
MQExport1.export	gen/exports/MQExport1.subflow gen/exports/MQExport1_reply.subfl
Mediation.component	Mediation_Interface2_operation1_F Mediation_Interface2_operation1_F Mediation_Interface2_operation1_F gen/mediationflows/SetCalloutOper gen/mediationflows/SetCalloutResp
SCAExport1.export	gen/exports/SCAExport1.subflow gen/exports/SCAExport1_reply.subfl
WESBConvert_AllPrimitives.mfc	Mediation_Interface2_operation1_F gen/Mediation_Interface1_operatio gen/Mediation_Interface2_operatio gen/Mediation_Interface2_operatio gen/exports/WebServiceExport1 su

[< Previous](#) [Save](#)

# WESB Conversion tool in IIB Studio

\*wtudemo.conversion

1. Select WebSphere ESB projects ▶ 2. Configure WebSphere ESB resource options ▶ 3. Configure global conversion options ▶ 4. Convert WebSphere ESB resources ▶ 5. Review result

Review the conversion results. Complete all the to-do tasks to finish converting WebSphere ESB resources.

Show all    Show resources with incomplete tasks

Conversion summary  
 ! All to-do tasks  
 x All problems in the converted projects  
 Detailed conversion results  
 Source: WESBConvert\_AllPrimitives  
 ▶ Converted to: IIB\_WESBConvert\_AllPrimitives  
 ▶ Converted to: IIB\_WESBConvert\_AllPrimitivesJava  
   Mediation\_Interface2\_operation1\_Request\_MessageElementSetter2.java - /IIB\_W

Messages and tasks

- ! Update map 'XSLTransformation1\_req\_1.map' to use XPath 2.0. For more informat
- ! Fix any validation errors and warnings for map '/IIB\_WESBConvert\_AllPrimitives/xs
- ! Configure the properties for the MQOutput node 'SCAExport1\_reply' to match the
- ! Trace root property '/' has been converted to '\${Root}'. Check that this is correct a
- ! Trace literal property '\${CURRENT\_TIMESTAMP}, {1}, {2}, {3}, {4}, {5}' includes a refe
- ! Complete conversion of the EventEmitter primitive 'EventEmitter1' to Passthrough
- ! Complete conversion of the primitive 'SLACheck1' to node 'SLACheck1\_RegistryLo
- ! Complete conversion of the SLACheck primitive 'SLACheck1' to RegistryLookup nc
- ! Complete conversion of the SLAEndpointLookup primitive 'SLAEndpointLookup1'
- ! Consider if correct HTTP Header type used in conversion of the HTTPHeaderSette

CWWOC6150I: Complete conversion of the SLACheck primitive 'SLACheck1' to RegistryLookup node 'SLACheck1\_RegistryLookup' and JavaCompute node 'SLACheck1\_SLACheck' by manually completing the Java code to process the endpoint extracted from WSRR.

The SLA Check primitive 'SLACheck1' was converted to a RegistryLookup node 'SLACheck1\_RegistryLookup' and JavaCompute node 'SLACheck1\_SLACheck'. The RegistryLookup node retrieves the Capability Version and associated documents from WebSphere Service Registry and Repository (WSRR). The JavaCompute node is needed to find the SLA and validate that the endpoint is defined in the SLA and is active .

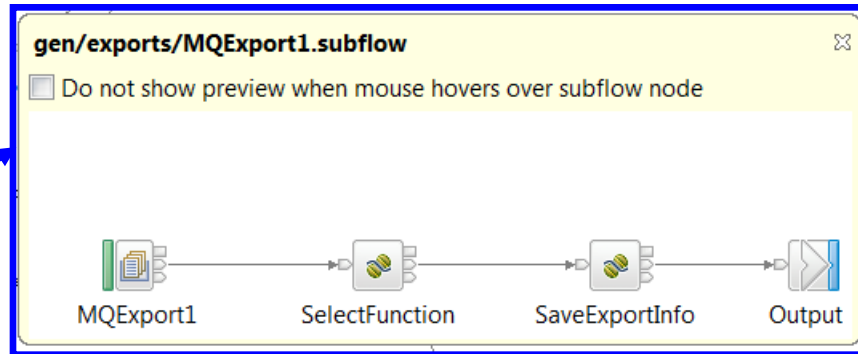
Configure the JavaCompute node 'SLACheck1\_SLACheck' to check that the endpoint exists and is active for the SLA retrieved from WSRR. Also in the

[Previous](#) [Save](#)

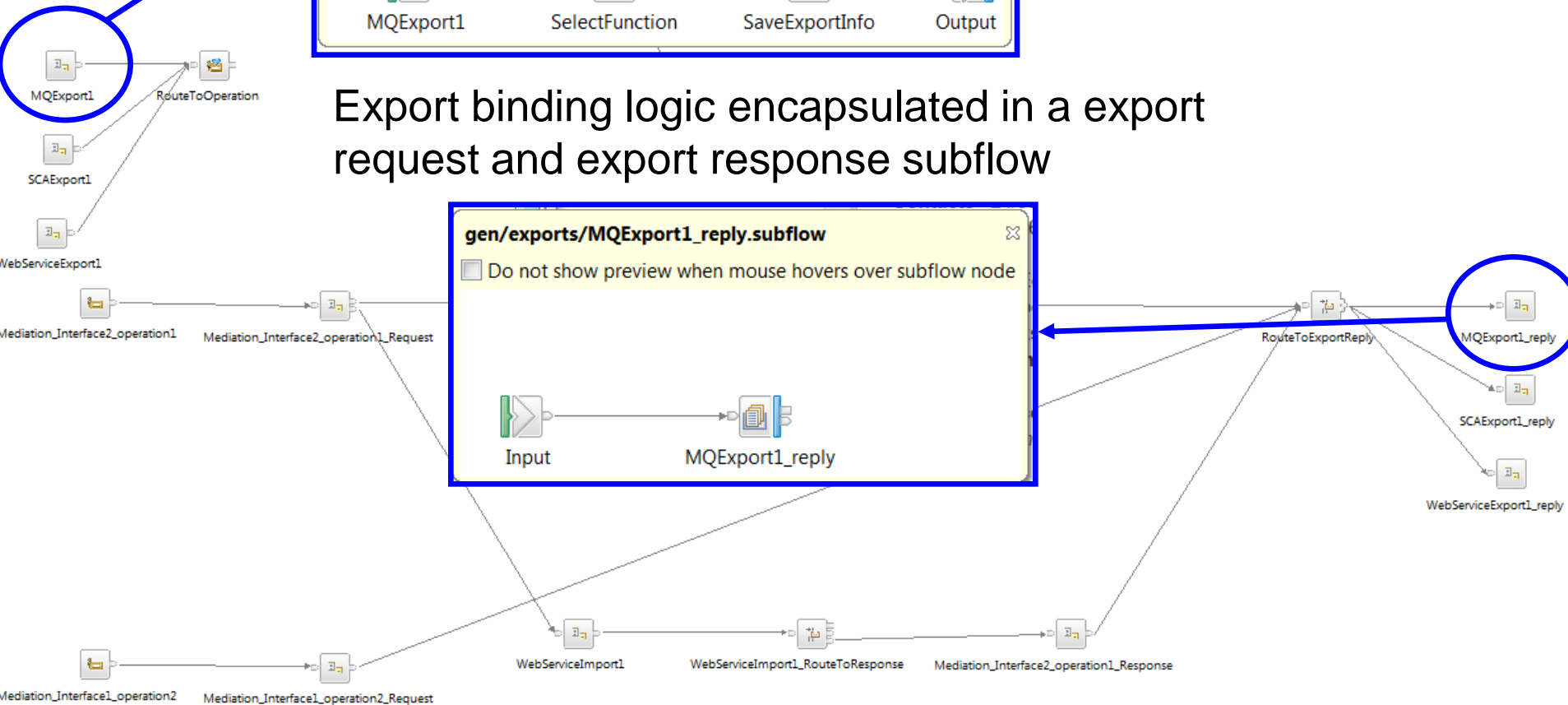
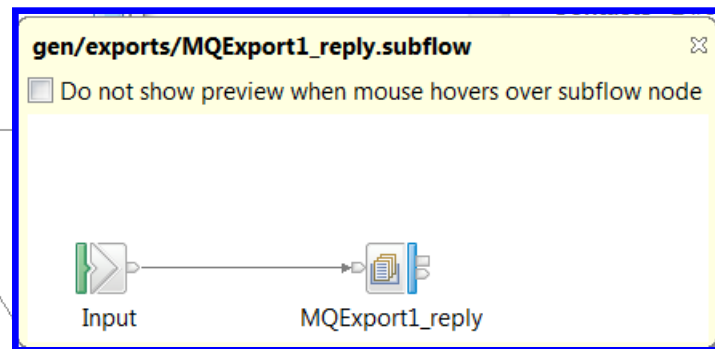
**Task list containing specific post tool conversion tasks.**

**Verbose description of the task with links to more detailed documentation**

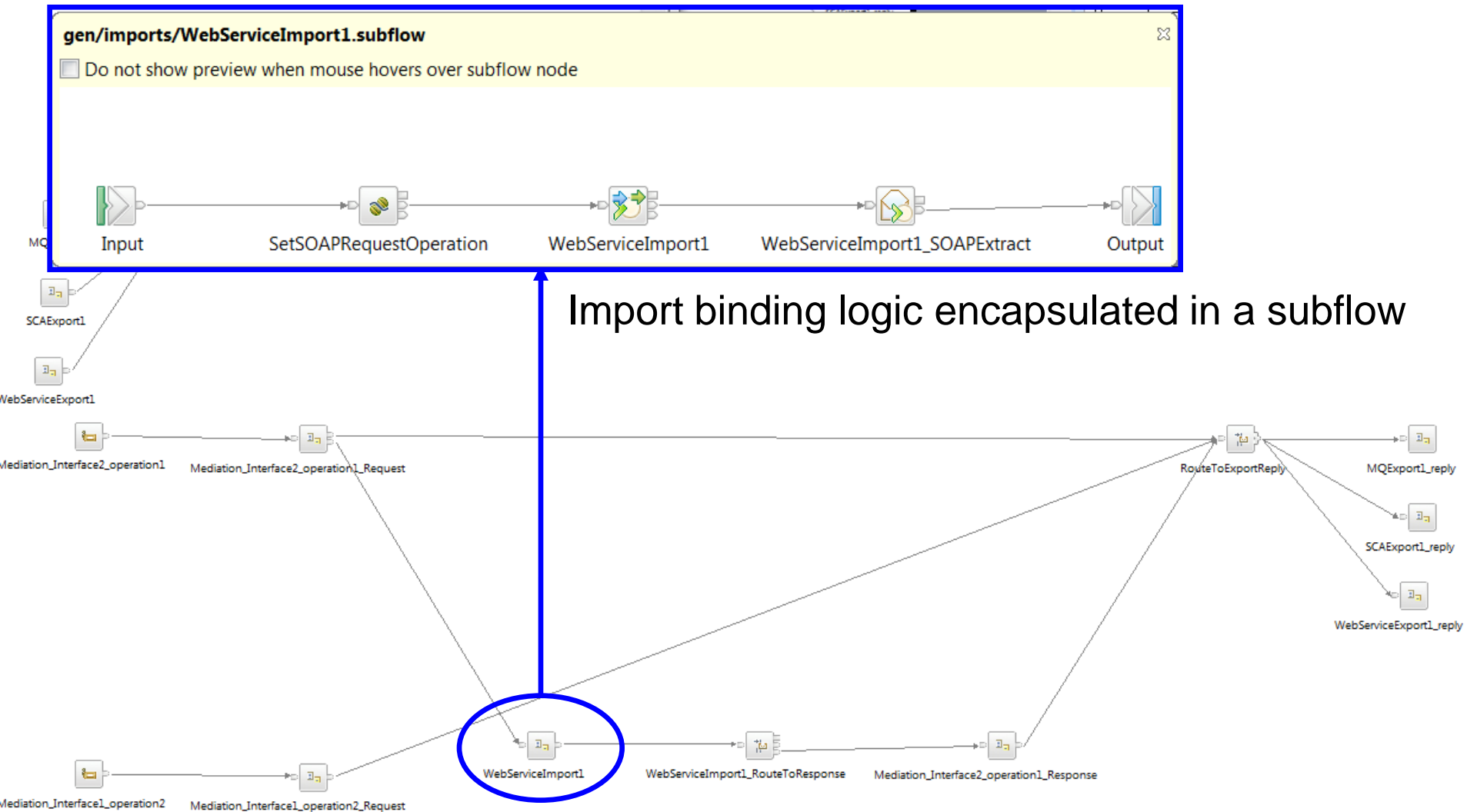
# WESB Conversion tool in IIB Studio



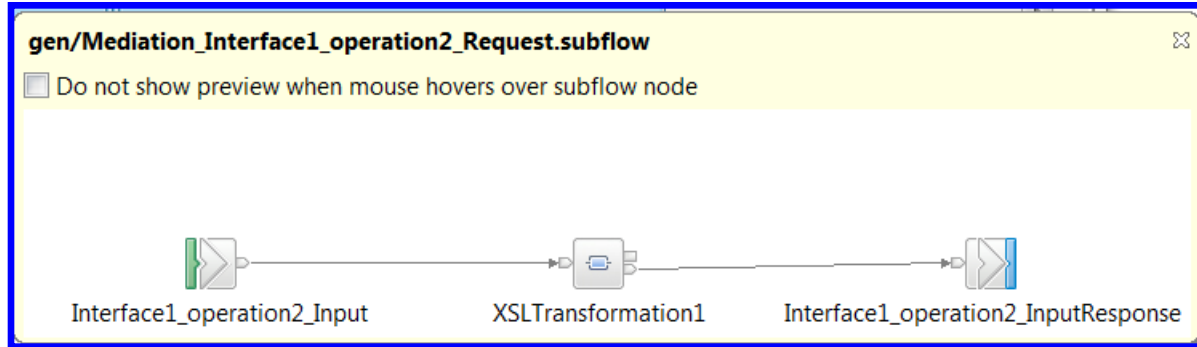
Export binding logic encapsulated in a export request and export response subflow



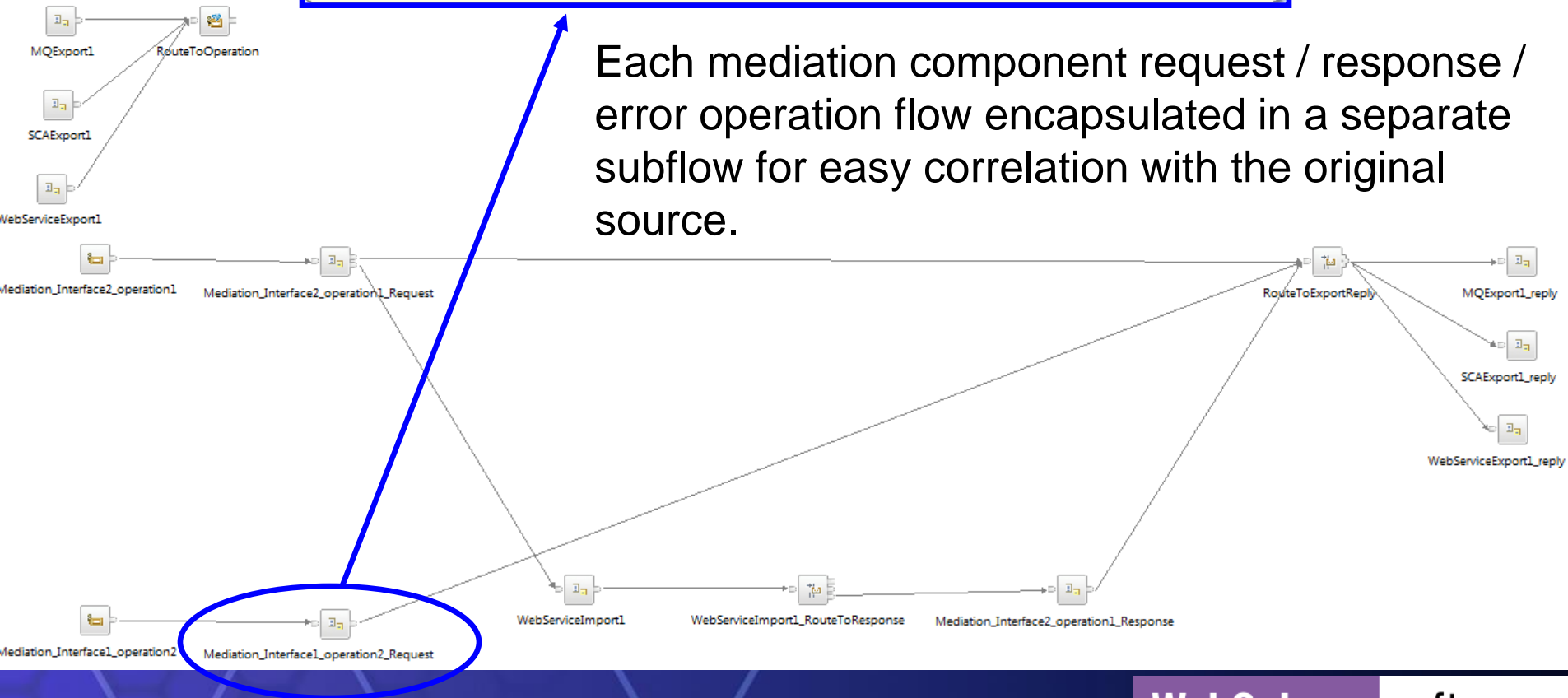
# WESB Conversion tool in IIB Studio



# WESB Conversion tool in IIB Studio



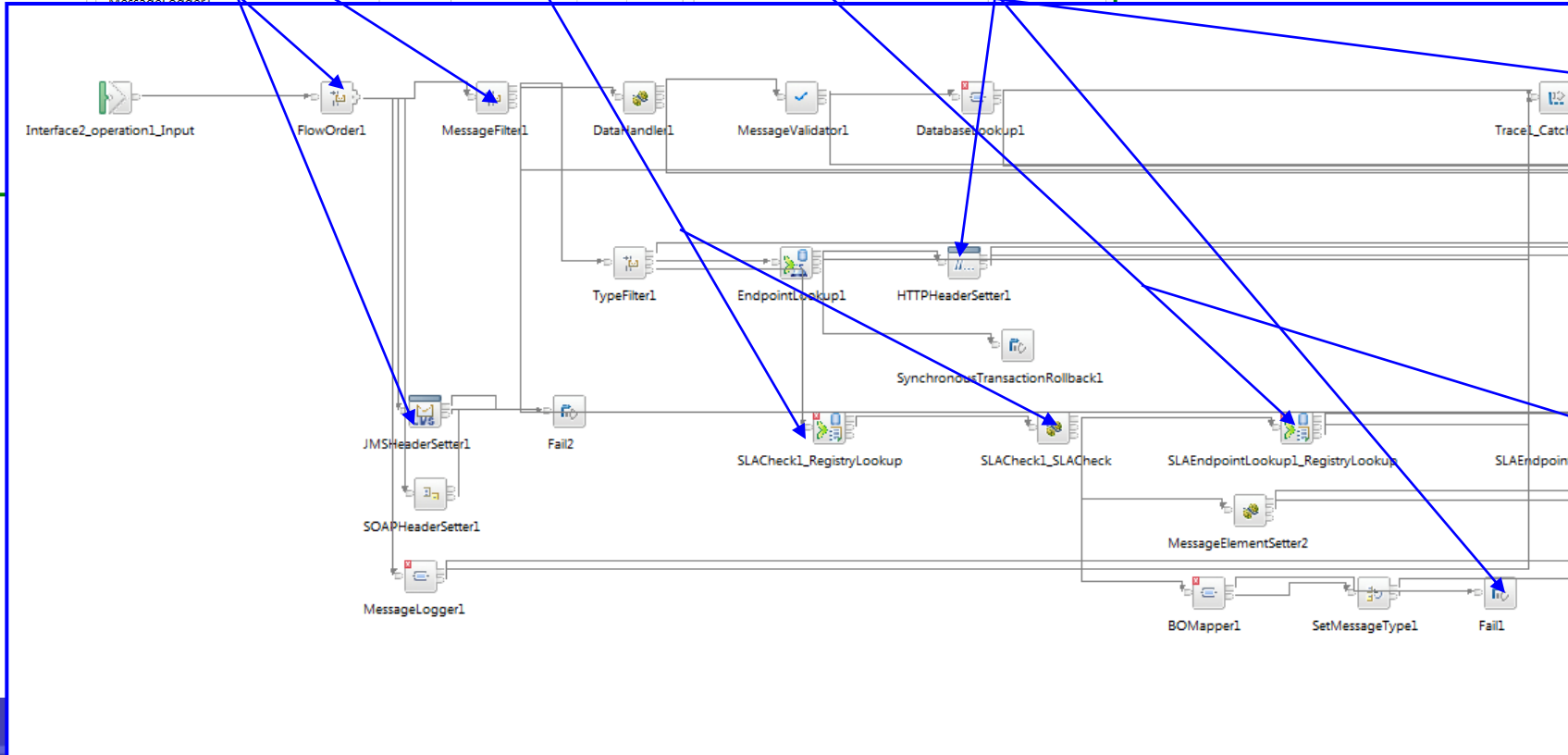
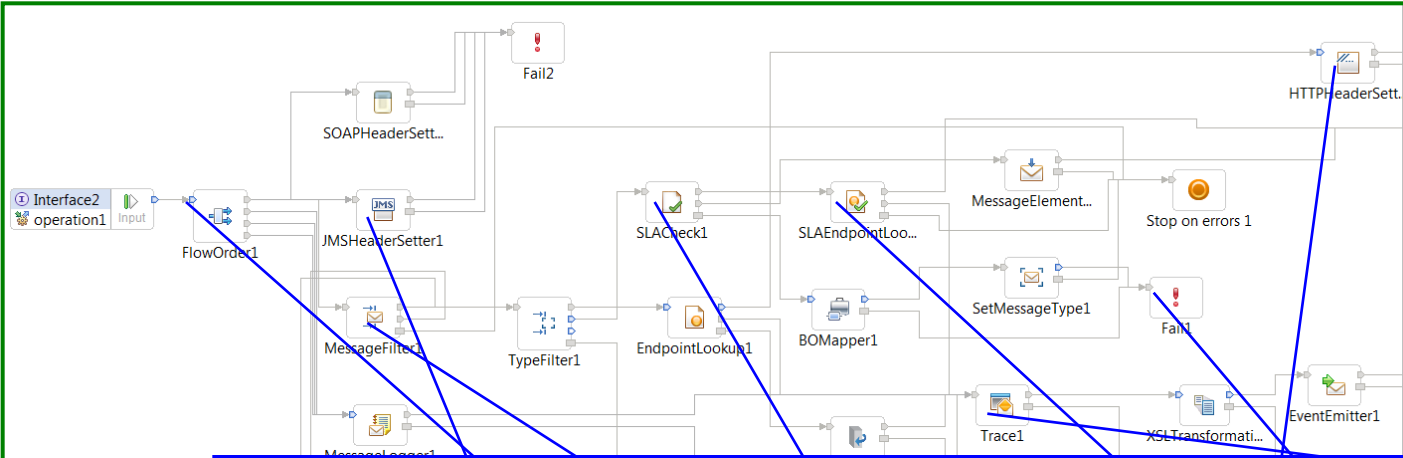
Each mediation component request / response / error operation flow encapsulated in a separate subflow for easy correlation with the original source.





Original flow wiring preserved.

Built-in converters convert to equivalent IIB node set.



# Customizing the Tool for your needs

- The conversion tool is designed to be extensible.
  - Users can author custom conversion logic for mediation primitives and import/export bindings whilst leveraging the capabilities of the tool for everything else.
- Conversion tool source code is open and available on [gitHub](#).
  - Users can modify the tool to for their specific needs.

# Extending the Conversion Tool

- Nearly all primitives and export/import bindings (as of current open-beta) are handled by a Built-in Converter class out of the box.
- However user-defined converters may help a specific customer where they have repeatable conversion logic they wish to include in the conversion.
- Extension points allow re-usable customer specific customizations of the tool.
- You can create a Java converter class which extends `AbstractMediationPrimitiveConverter`, which will provides a primitive conversion capability specific to a customers needs.
- You can create a Java converter class which extends `AbstractBindingConverter`, which will provides a export/import binding conversion capability specific to a customers needs.

# Modify the tool.

WESB to IIB Conversion Tool on GitHub open source community

<https://github.com/ot4i/open-convert>

The screenshot shows the GitHub repository page for `ot4i / open-convert`. The repository is public and has 7 commits, 1 branch, 0 releases, and 2 contributors. The current branch is `master`. The repository description is "Reuse existing assets when moving to a new technology base". The commit history shows a commit by `chenzhongming` on Nov 29, 2013, with the message "update GitHub repo URL". The commit includes files `doc` (Code base), `src` (Rollback the accidental changes by previous change set.), `INSTRUCTIONS.md` (update GitHub repo URL), and `README.md` (Fix Readme).

File	Commit Message
<code>doc</code>	Code base
<code>src</code>	Rollback the accidental changes by previous change set.
<code>INSTRUCTIONS.md</code>	update GitHub repo URL
<code>README.md</code>	Fix Readme

# For Additional Information

- IBM Integration Bus vNext open beta:
  - What's new: <https://ibm.biz/iibopenbetawhatsnew>
  - Download: <https://ibm.biz/iibopenbeta>
  - Documentation: <https://ibm.biz/iibopenbetadocs>
  - Discuss: <https://ibm.biz/iibopenbetaforum>
- IBM Integration Community:
  - IIB community <https://ibm.biz/iibcommunity>
  - WESB Convert Wiki <https://ibm.biz/iibwesbconvert>
  - WESB Convert Topologies [https://ibm.biz/iibwesbconvert\\_topologies](https://ibm.biz/iibwesbconvert_topologies)