



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Composer Help

Using eServices Blocks

Using eServices Blocks

Contents

- **1 Using eServices Blocks**
 - 1.1 Example Multimedia Workflow
 - 1.2 Example Diagram
 - 1.3 Associated IPD
 - 1.4 eService Preparation
 - 1.5 Working with Returned Data
 - 1.6 Mandatory User Data for UCS Blocks

This page contains general information on working with eServices blocks.

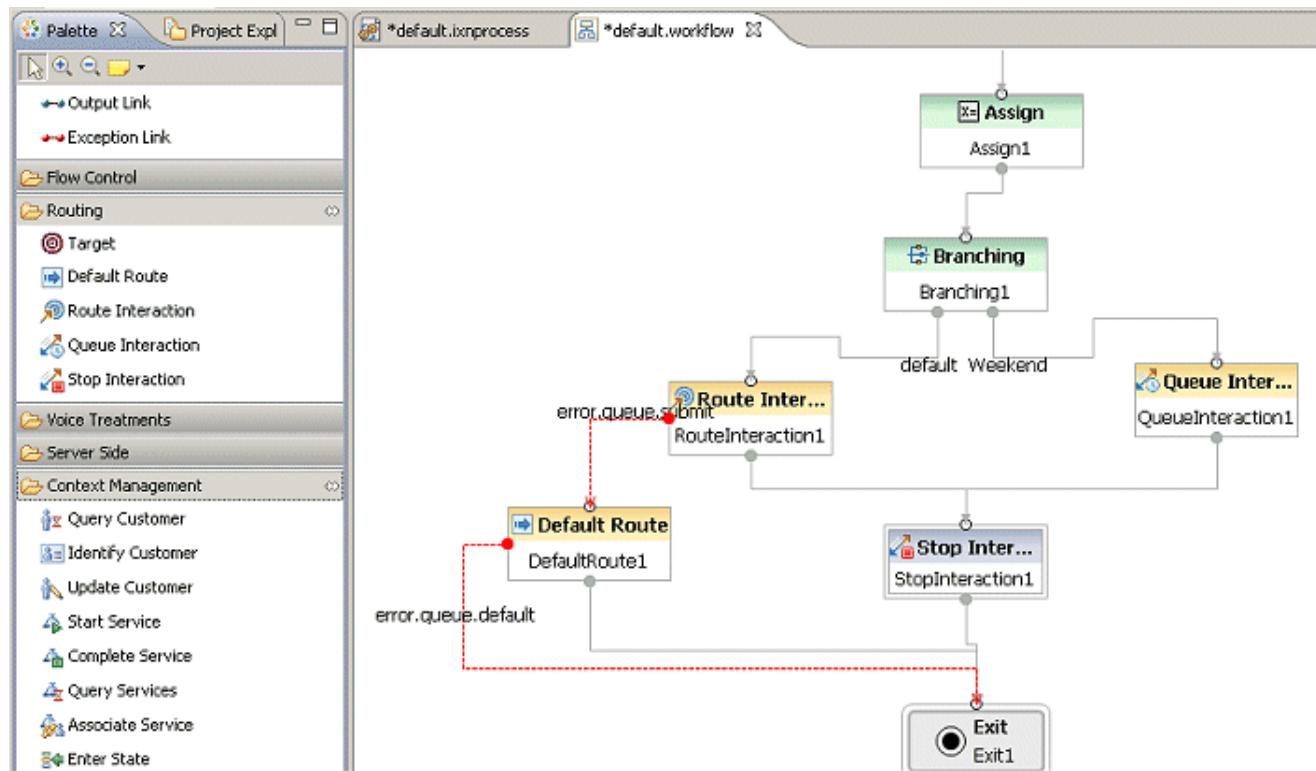
Example Multimedia Workflow

In general, an interaction process diagram (IPD) for multimedia interactions works like this:

1. A **media server** (such as E-mail Server Java) directs Interaction Server to place an interaction into an inbound interaction queue.
2. Using the **Interaction Queue View** property, the interaction is then taken out of the queue and submitted to a routing **workflow**.
3. The workflow performs specialized processing and eventually routes the interaction to a target, but not necessarily the final target. For example, an e-mail interaction may be placed in an agent queue for construction of a response.
4. The target processes the interaction and places it into another queue where another workflow may process it. For example, a workflow may send an agent's draft e-mail response to a queue for Quality Assurance checking.
5. The cycle of going from queue/view/workflow continues until processing is stopped or the interaction reaches some final (usually an outbound) queue.

Example Diagram

The figure below shows a sample multimedia workflow diagram in Composer Design **perspective**.



For other sample diagrams, see the [Sample Applications](#) topic. The default.workflow shown above works as follows:

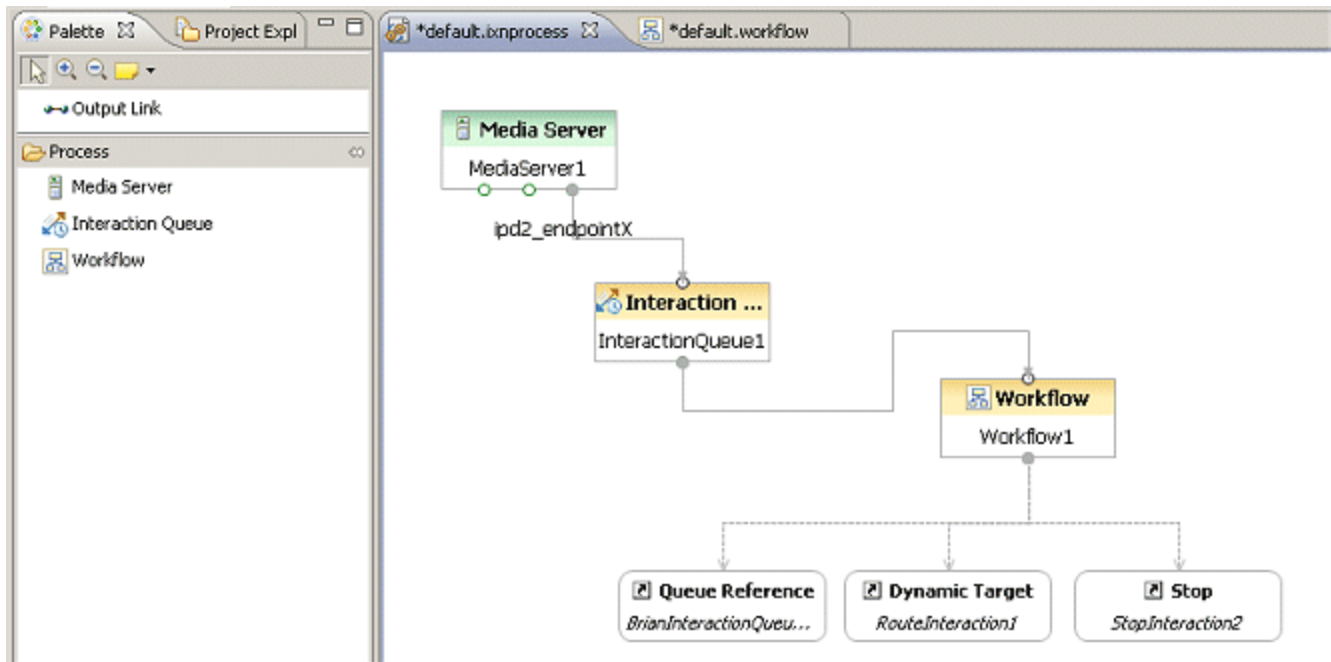
- The Entry block defines a variable called Today.
- The Assign block gives a value to the Today variable. In Expression Builder this is defined as:

```
data.Today==(_genesys.session.day.Saturday) || (data.Today==_genesys.session.day.Sunday)
```

- The Branching block Conditions property contains an expression used for segmenting interactions based on whether today is a week day or the weekend. The expression determines whether an interaction goes to a queue for the weekend crew or whether it is routed to a target.
- The Stop block notifies Interaction Server to stop processing and whether to notify Universal Contact Server about the interaction.

Associated IPD

The figure below shows the IPD containing the Workflow block that points to the workflow diagram above. Three **work-flow generated blocks** are automatically generated in this example.



eService Preparation

Genesys eService/Multimedia lets you process non-voice interactions in your contact center. At the center of the collection of components are:

- Interaction Server: Works with Universal Routing Server and Stat Server to process non-voice interactions by executing **interaction process diagrams**.
- Universal Contact Server (UCS): Works with its database to deliver customer contact history and information to the agent desktop.
- In addition, there are a number of other Multimedia servers that facilitate the handling of non-voice media, including E-mail Server Java, SMS Server, and Chat Server.

This help system assumes you have already installed and configured the eService/Multimedia components as described in the *eService/Multimedia 8.1 Deployment Guide*.

Working with Returned Data

A few Composer Route blocks will return data back to the application:

- Email Response (**Output Result**)
- Create Email (**Output Result**)
- Create SMS (**Output Result**)

- Identify Contact (**Result Property**)
- Render Message (**Result Property**)
- Query Customer (**Result Property**)

Each qualifying block will expose a output result property (or equivalent) that will specify an application variable to store the results. These results will then be available in other blocks in the application for further processing. The format of returned data is usually JSON. Any post-processing work to be done on returned results can be done in the existing Assign block. It provides access to ECMAScript functions and supports writing simple or complex expressions to extract values out of JSON strings and arrays.

Mandatory User Data for UCS Blocks

When working with the Update Contact and Render Message blocks (which map to Universal Contact Server services), certain properties must exist in the interaction **User Data**. These properties are:

Update Contact Block

- ContactId

Render Message Block

- ContactId (if some contact-related Field Codes (as described in the *eServices 8.1 User's Guide*) are used in the text to render)
- InteractionId (if some interaction-related Field Codes are used in the text to render)
- OwnerEmployeeId (if some agent-related Field Codes are used in the text to render)

As is the case with IRD, these properties are not set in the blocks themselves. Instead, the properties are assumed to be put in the interaction's User Data by some other block earlier in the workflow, such as the **Identify Contact block** or **Create Interaction block** with the Update User Data property set to true. In case no other block does this, the **User Data block** may be used for this purpose. Note: If those properties are not available, an explicit UCS error message (missing parameter) shows in the Orchestration Server log.