



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.

# Orchestration Server Developer's Guide

Migration from IRD

4/3/2025

# Migration from IRD

The following table documents the mapping from URS/IRD function block functionality to SCXML and Functional Modules functionality for use in Orchestration. This mapping will result in the creation of an SCXML snippet or a specific <state> element definition which represents the functionality of the function block. The following is a <state> element template that could be used:

```
<state id="function_block_name">
  <datamodel>
    <!-- This defines the data needed for processing this function block -->
  </datamodel>
  <onentry>
    <!-- Do the necessary function block action set up and action execution -->
    <xxxx:yyyy requestid="_data.reqid" .../>
  </onentry>
  <transition event="xxxx.yyyy.done" cond="..." target="...">
    <!-- transition based on results (equal to function block output port) -->
  </transition>
  <transition event="xxxx.zzzz" cond="..." target="...">
    <!-- additional actions if necessary and transitions based on it. (equal to
function block output port ) -->
  </transition>
  <transition event="error.xxxx.yyyy">
    <!-- transition based on results - (equal to function block error port) -->
  </transition>
</state>
```

Function Block	SCXML and Functional Module equivalent
Acknowledgement	<msgbased:createmessage> for appropriate media
Add Record	<session:fetch... method = "'post'" srcexpr="'http://server.com...req=AddRecord'">
ANI	<script> if done in orchestration, <session:fetch> with application server code, if done outside.
Assign	<assign> using ECMAScript object properties and functions
Attach Categories	Use _genesys.ixn.interactions[].udata with category information (Business Attribute CME objects) from configuration server
Autoreponse	<msgbased:createmessage> for appropriate media
Business	<script> if done in orchestration, <session:fetch> with application server code, if done outside.
Busy	<dialog:playsound>

Function Block	SCXML and Functional Module equivalent
Call Subroutine	<code>&lt;include&gt;</code> or <code>&lt;invoke&gt;</code>
Cancel Call	<code>&lt;dialog:stop&gt;</code>
Chat Transcript	<code>&lt;msgbased:createmessage ... chattranscript="'xxx' "&gt;</code> for appropriate media
Classify	<code>&lt;classification:classify&gt;</code>
Classify (segmentation)	<code>&lt;classification:classify&gt;</code> , <code>&lt;script&gt;</code> if done in orchestration, <code>&lt;session:fetch&gt;</code> with application server code, if done outside.
Collect Digits	<code>&lt;dialog:collect&gt;</code>
Comment	<code>&lt;log&gt;</code> NOTE: As of Orchestration version 8.1.200.46 - the URS messages 22001 to 22020 are supported.
Create Interaction	<code>&lt;session:fetch srcexpr="'http://server1.com/.../customers/\${customer_id}/interactions...' method="'post'" ...&gt;</code>
Create Email Out	<code>&lt;msgbased:createmessage&gt;</code> for appropriate media
Create Notification	<code>&lt;msgbased:createmessage&gt;</code> for appropriate media
Create SMS	<code>&lt;msgbased:createmessage&gt;</code> for appropriate media
Database Wizard	<code>&lt;session:fetch&gt;</code> , specialized application server code
Date	<code>&lt;script&gt;</code> if done in orchestration (use ECMAScript standard date functions/objects or Session functional module ECMAScript date and time functions), <code>&lt;session:fetch&gt;</code> with application server code, if done outside.
Day of Week	<code>&lt;script&gt;</code> if done in orchestration (use ECMAScript standard date functions/objects or Session functional module ECMAScript date and time functions), <code>&lt;session:fetch&gt;</code> with application server code, if done outside.
Default	<code>&lt;queue:default&gt;</code>
Delete User Announcement	<code>&lt;dialog:deleteann&gt;</code>

Function Block	SCXML and Functional Module equivalent
DNIS	<code>&lt;script&gt;</code> if done in orchestration, <code>&lt;session:fetch&gt;</code> with application server code,
Do Not Call	<code>&lt;session:fetch... " method = "'post'" srcexpr="'http://server.com...req=DoNotCall'" /&gt;</code>
Entry	<code>&lt;initial&gt;</code> or <code>&lt;scxml initial attribute&gt;</code>
Error Segmentation	<code>&lt;script&gt;</code> if done in orchestration, <code>&lt;session:fetch&gt;</code> with application server code, if done outside.
Exit	<code>&lt;final&gt;</code>
External Service	<code>&lt;session:fetch&gt;</code> , specialized application server code
Fast Busy	<code>&lt;dialog:playsound&gt;</code>
Force Routing	<code>&lt;ixn:redirect&gt;</code>
Forward E-Mail	<code>&lt;msgbased:createmessage&gt;</code> for appropriate media
Function	<code>&lt;script&gt;</code> any functional module interface action or ECMAScript function.
Generic	<code>&lt;script&gt;</code> if done in orchestration, <code>&lt;session:fetch&gt;</code> with application server code, if done outside.
Identify Contact	<code>&lt;session:fetch srcexpr="'http://server1.com/cv/profiles'" method="get" ... &gt;</code>
If	<code>&lt;if&gt;</code>
IVR	<code>&lt;dialog:remote&gt;</code>
Load Balancing	<code>&lt;queue:submit&gt;</code>
Macro	<code>&lt;include&gt;</code>
MultiAssign	<code>&lt;assign using ECMAScript object properties and functions&gt;</code>

Function Block	SCXML and Functional Module equivalent
MultiAttach	<code>&lt;script&gt;</code> or <code>&lt;assign&gt;</code> using <code>_genesys.FMname.interactions[x].udata</code> object properties
MultiScreen	<code>&lt;classification:screen&gt;</code>
Music	<code>&lt;dialog:playsound&gt;</code>
Pause	<code>&lt;runttreatments&gt;</code> and <code>&lt;pause&gt;</code>
Percentage	<code>&lt;queue:submit&gt;</code>
Play Announcement	<code>&lt;dialog:play&gt;</code> and <code>&lt;prompts type="tts"&gt;</code>
Play Announcement and collect digits	<code>&lt;dialog:playandcollect&gt;</code>
Play Application	<code>&lt;dialog:start&gt;</code>
Processed	<code>&lt;session:fetch..." method = "'post'" srcexpr="'http://server.com...req=RecordProcessed'" /&gt;</code>
Queue Interaction	<code>&lt;ixn:redirect&gt;</code>
RAN	<code>&lt;dialog:playsound&gt;</code>
Record User Announcement	<code>&lt;dialog:recordann&gt;</code>
Redirect E-Mail	<code>&lt;msgbased:createmessage&gt;</code> for appropriate media
Reply E-Mail From External Resource	<code>&lt;msgbased:createmessage&gt;</code> for appropriate media
Reschedule	<code>&lt;session:fetch..." method = "'post'" srcexpr="'http://server.com...req=RecordReschedule'" /&gt;</code>
Ringback	<code>&lt;dialog:playsound&gt;</code>
Route Interaction	<code>&lt;queue:submit&gt;</code>
Screen	<code>&lt;classification:screen&gt;</code>

Function Block	SCXML and Functional Module equivalent
Screen (segmentation)	<code>&lt;classification:screen&gt;</code> , <code>&lt;script&gt;</code> if done in orchestration, <code>&lt;session:fetch&gt;</code> with application server code, if done outside.
Selection (target selection)	<code>&lt;queue:submit&gt;</code> , <code>&lt;dialog:runtreatments&gt;</code>
Send E-Mail	<code>&lt;msgbased:sendmessage&gt;</code>
Service level	<code>&lt;queue:submit&gt;</code> , <code>&lt;dialog:runtreatments&gt;</code>
Set Default Destination	<code>&lt;dialog:setdialogdefaultdest&gt;</code>
Silence	<code>&lt;dialog:playsound&gt;</code>
Statistics	<code>&lt;queue:submit&gt;</code> , <code>&lt;dialog:runtreatments&gt;</code>
Stop Interaction	<code>&lt;ixn:terminate&gt;</code>
Switch to Strategy	<code>&lt;include&gt;</code> or <code>&lt;invoke&gt;</code>
Text to Speech	<code>&lt;dialog:play&gt;&lt;prompts type="tts"&gt;</code>
Text to Speech and Collect Digits	<code>&lt;dialog:playandcollect&gt;&lt;prompts type="tts"&gt;</code>
Time	<code>&lt;script&gt;</code> if done in orchestration (use ECMAScript standard date functions/objects or Session functional module ECMAScript date and time functions), <code>&lt;session:fetch&gt;</code> with application server code, if done outside.
Update Contact	<code>&lt;session:fetch srcexpr="http://server1.com/cv/profiles" method="put"... &gt;</code>
Update Record	<code>&lt;session:fetch... method = 'post' srcexpr="http://server.com...req=UpdateCallCompletionStats' /&gt;</code>
Verify Digits	<code>&lt;dialog:playandverify&gt;</code>
Web Service	<code>&lt;session:fetch&gt;</code> with application server code
Workbin	<code>&lt;ixn:redirect&gt;</code>

<b>Function Block</b>	<b>SCXML and Functional Module equivalent</b>
Workforce	<code>&lt;queue:submit&gt;, &lt;dialog:runtreatments&gt;</code>