



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.

IRD to Composer Migration

Target Manipulation Function Migration

12/18/2025

Target Manipulation Function Migration

The table below describes how migration is handled for IRD Functions in the Target Manipulation category as defined in the *Universal Routing 8.1.x Reference Manual*.

- The `_genesys` Data Subcategory is described in the [Orchestration Server Wiki](#).

Function Name	Functional Module Mapping	Automatic Migration	Update Required	Comments/ Manual Steps Required
BlockDN	<code>_genesys.queue.reserveTarget()</code>	Auto		
CCTExtractTargets	<code>_genesys.queue.cCTExtractTargets()</code>	Auto		
DeliverCall	-	Manual		Not Supported. This function was initially developed to shuttle interactions between IVR and the original routing point. During Migration, the user should consider using either one of the routing blocks within composer to emulate the required and desired behavior. It may occur that the Force block is the most appropriate. When using this function a new block must be created and linked appropriately by the end user for the desired behavior.
DeliverToIVR	Not supported	-		
GetRemoteAccessCode	Not supported	-		
IncrementPriority	<code>_genesys.queue.incrementPriority()</code>	Auto		
IncrementPriorityEx	Not supported	-		
KeepQueue	Not supported	-		Use the URS Function block in Composer to call the URS HTTP interface methods (for example,

Function Name	Functional Module Mapping	Automatic Migration	Update Required	Comments/ Manual Steps Required
				KeepQueue). For more information on parameters and syntax, refer to the Universal Routing 8.1 Reference Manual .
NMExtractTargets	<code>_genesys.queue.nmExtractTargets()</code>	Auto		
Priority	Assign to Priority attribute of Target block	Manual		Assign to appropriate Target block
PriorityLimits	-	Manual		Use ORS function: <code>_genesys.queue.priorityLimits()</code>
PriorityTuning	<code>_genesys.queue.priorityTuning()</code>	Auto		
RouteCall	SCXML State Block	Manual		Use a Routing block: Target or Route Interaction block.
Routed	<code>_genesys.queue.routed()</code>	Auto		
SelectDN	SCXML State Block	Manual		Use a Routing block: Target or Route Interaction block.
SetTargetThreshold	Assign Block	Manual		Threshold is now exposed within the Target block and Route Interaction block
SetVQPriority	Assign Block	Manual		The SetVQPriority function is typically used prior to target selection, therefore this can be replaced with a variable to indicate the desired priority and then passed into the target selection as a part of the Target block.
SuspendForDN	Not supported as a function.	Manual		SuspendForDN[timeout] cannot be migrated automatically. A Target block with the correct timeout may be used.

Function Name	Functional Module Mapping	Automatic Migration	Update Required	Comments/ Manual Steps Required
SuspendForTreatmentEnd	Not supported as a function	Manu		This will map onto a transition event for the end of treatment. It is therefore expected that any logic required to wait for treatment end be linked off of such transitions. Such events are for example <code>dialog.stop.done</code> . This may have to be combined with a delay event to simulate timeouts if the desired behavior cannot be correctly modeled with the exposed Composer treatment blocks.
TargetSelectionTuning	<code>_genesys.queue.targetSelectionTuning()</code>	Auto		
Translate	<code>_genesys.queue.translate()</code>	Auto		

Note on the SelectDN Function

The SelectDN URS function's parameters differ between SCXML and IRD usage. From Composer, the SelectDN function can be invoked via either the URS Function block or the SCXML State block. Also, URS handles the request differently when the request is received from an external source, such as from ORS. Due to this difference in behaviour, ORS and Composer strategy responses might differ from that of iRD/URS.

In such cases where the SelectDN function is called from externally, the function must have an implicit integer parameter in its first place, used to refer to a specific entrance call into a queue.

In such cases where the function is used externally, users must specify an integer in the first place. For example,
`[1,"VirtualQueue1","10","RStatAgentsReadyvoice","StatSelectMax","agent3601@.A"]`.

When called externally as shown above, the SelectDN will place the call into a queue but does not return a ready target. It exits immediately with the result, `timeout`. Though the call is placed in a queue, the function does not wait for the result of the target selection process. As the target selection process could be a lengthy operation and might include sub-processes such as, agent reservation, and attaching data, it might not be possible to accommodate all this within the processing time used by the SelectDN function.

Important

ORS does not have a native equivalent to the SelectDN URS function. iRD/URS behaviour is different from that of Composer/ORS behaviour. Strategy design might have to be different to achieve similar outcomes. The Composer strategy can use two Target blocks - the first block to queue to the VQ and request a dummy target, and the second block to request for the actual target.