Failed Event Management Wrapper

Author: Dave Arnold

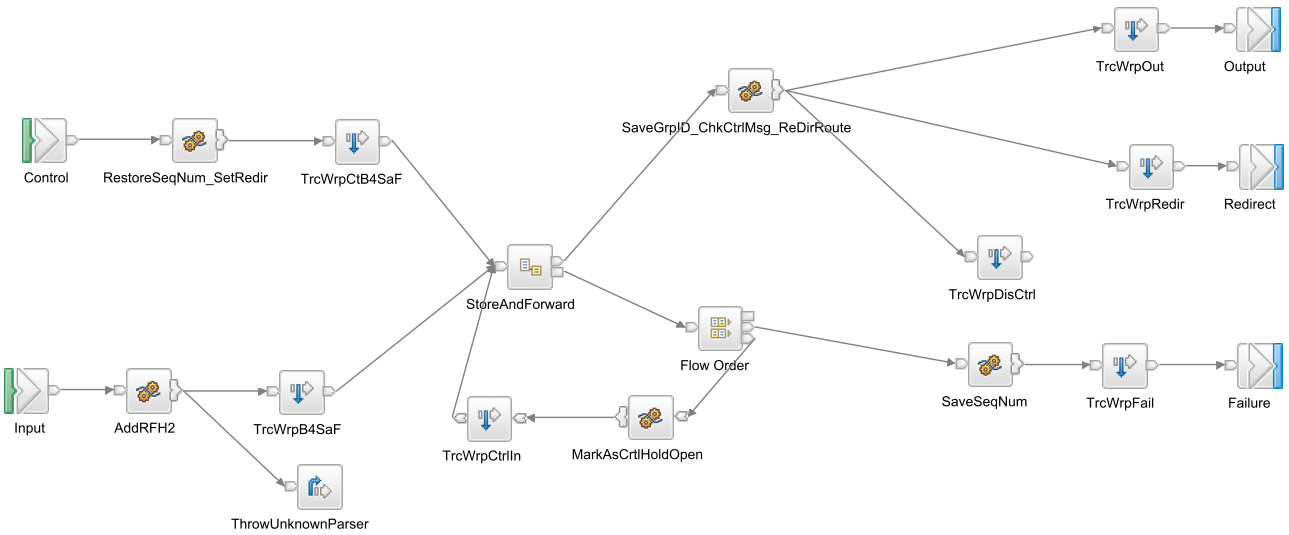
Date: 09/01/2015

Table of contents

1 Message Flow "IOBKWrapper"

/IOBKApplication/IOBKWrapper.subflow

1.1 Overview



1.1.1 Short Description

Wrapper provides the failed event management capability by adding state information to an MQRFH2 header such that downstream failed events are caught propagated to the failure terminal. Any future me

1.1.2 Long Description

The Wrapper accepts new messages via the In terminal. The Wrapper accepts replayed messages via the Control Terminal.

This Wrapper Control terminal supports the following Failed Event Management operator actions which must be set in the MQRFH2.usr.IOBK.ReplayAction field by Tooling Wrapper or message flow/subflow that drives the control terminal.

ReplayOnly: Replay the original failed event directing it to is original out path and release all blocked events of the same GroupID to the original out path
RedirectFirstOnly: Redirect the original failed event to an alternative out path and release all blocked events of the same GroupID to the original out path
RedirectAll: Redirect the original failed event and all blocked events of the same GroupID to the alternative out path.
RedirectAllPermanent: Redirect the original failed event, all blocked events of the same GroupID and any new events of the same GroupID to the alternative out path
ClearRedirectAllPerm: Manually clear the RedirectAllPermanent for the GroupID - this function does not propagate the event to an out path.

The wrapper injectss an MQRFH header before the original data payload.
The usr folder is then used to house state information.

This information will survive a "trip" through the Failed Event Management tooling such that a replayed event message is identified by a key i.e. the sequence group if belongs to.

MQRFH2.usr.IOBK.SeqGroupID - populated from the LocalEnvironment.Sequence.Group which was set by the resequence node in the StoreAndForward subflow - The Sequence GroupID is the key by which the

Wrapper and StoreAndForward (resequence node) subflows identify message grouping. This property of the resequence node is promoted to the top level containing message flow and set by the developer.

MQRFH2.usr.IOBK.SeqNum - populated from the LocalEnvironment.Sequence.Number which was set by resequence node in the StoreAndForward subflow

The Wrapper will feed back a copy of a failed event message to the StoreAndForward (resequence node) subflow as a control message to "hold the group open". This is marked as a control message such that it can be identified and filtered out when the group is released on replay.

MQRFH2.usr.IOBK.CtrlMsg - N or Y

For original messages entering the Wrapper the ReplayAction will be initialized to 'not set'.
Replay messages will set the ReplayAction dependent on the where the operator replayed the message in the ToolingWrapper. This allows for the identification of replay messages and how they should be actioned/routed.

MQRFH2.usr.IOBK.ReplayAction - 'not set' or a supported replay action.

In conjunction with the MQRFH.usr.IOBK folder the wrapper using information populated to the LocalEnvironment by the StoreAndForward (resequence node) subflow

LocalEnvironment
Sequence
Number - (INTEGER)
Group - (CHARACTER)
LastInBatch - (BOOLEAN)
End - (BOOLEAN)

Number - sequence number
Group - sequence GroupID set by the resequence node based on the property promoted to the containing message flow for configuration by the developer.
LastInBatch - indicator identifies that a message leaving the resequence node is the last one currently stored for forwarding for a particular sequence group. This allows the Wrapper to reset a RedirectAll automatically and have subsequent new messages sent down the original (rather redirected) route.
End - Last message for a sequence group (the group is closing) the Wrapper does not use this field.

1.2 Node - "Output"

Short Description	Driving: The original processing route
Long Description	

1.3 Node - "Failure"

Short Description	Delivering: Copies of the original failed event tothe Failed EvtMgr in the ToolingWrapper
Long Description	

1.4 Node - "Control"

Short Description	Servicing: Replay messages from the ToolingWrapper
Long Description	This Wrapper Control terminal supports the following Failed Event Management operator actions which must be set in the MQRFH2.usr.IOBK.ReplayAction field by Tooling Wrapper or message flow/subflow that drives the control terminal. ReplayOnly: Replay the original failed event directing it to is original out path and release all blocked events of the same GroupID to the original out path RedirectFirstOnly: Redirect the original failed event to an alternative out path and release all

blocked events of the same GroupID to the original out path RedirectAll: Redirect the original failed event and all blocked events of the same GroupID to the alternative out path. RedirectAllPermanent: Redirect the original failed event, all blocked events of the same GroupID and any new events of the same GroupID to the alternative out path ClearRedirectAllPerm: Manually clear the RedirectAllPermanent for the GroupID - this function does not propagate the event to an out path.

1.5 Node - "Input"

Short Description	Servicing: Inbound messages from the containing message flow via the ToolingWrapper
Long Description	

1.6 AddRFH2Node - "AddRFH2"

Short Description	Insert an MQRFH2 header and initialize the usr folder values
Long Description	Insert an MQRFH2 header and initialize it Determine the original input parser such that the data payload can be reattached correctly in the message tree.
ESQL File	IOBKWrapper.esql

1.7 Node - "Redirect"

Short Description	Driving: Failed Event Message redirection
Long Description	Driven if the following ReplayActions are used: RedirectFirstOnly: Redirect the original failed event to an alternative out path and release all blocked events of the same GroupID to the original out path RedirectAll: Redirect the original failed event and all blocked events of the same GroupID to the alternative out path. RedirectAllPermanent: Redirect the original failed event, all blocked events of the same GroupID and any new events of the same GroupID to the alternative out path

1.8 TrcWrpB4SaFNode - "TrcWrpB4SaF"

Short Description	
Long Description	

1.9 TrcWrpOutNode - "TrcWrpOut"

Short Description	
Long Description	

1.10 SaveSeqNumNode - "SaveSeqNum"

Short Description	Save the Sequence Number and Sequence Group ID to the MQRFH2
Long Description	Move the Sequence Number and Sequence Group ID to the MQRFH2 such that they survive the trip throught the ToolingWrapper and tooling. The exceptionlist is passed on it will be used by the ToolingWrapper to contruct the failed event
ESQL File	IOBKWrapper.esql

1.11 RestoreSeqNum_SetRedirNode - "RestoreSeqNum_SetRedir"

Short Description	Recover state information from the MQRFH2 and set up the replay or redirection processing

Long Description	Recover the Sequence Number from the MQRFH2 for use by the StoreAndForward (resequence node) subflow. Manage (add entries to) the Redirection groupID list dependent on replayAction type. ReplayOnly: Don't add an entry to the list RedirectFirstOnly: Add a GroupID entry to the list RedirectAll: Add a GroupID entry to the list and mark it as ALL=Y in the list RedirectAllPermanent: Add a GroupID entry to the list and mark it as PERM=Y in the list ClearRedirectAllPerm: Remove an entry from the GroupID list RedirIDList SHARED ROW - IDs[n].ID or IDs[n].ID + IDs[n].ID.PERM or IDs[n].ID + IDs[n].ID.ALL RedirIDCount SHARED INTEGER - number of entries in the list
ESQL File	IOBKWrapper.esql

1.12 TrcWrpFailNode - "TrcWrpFail"

Short Description	
Long Description	

1.13 SaveGrpID_ChkCtrlMsg_ReDirRouteNode - "SaveGrpID_ChkCtrlMsg_ReDirRoute"

Short Description	Filter out Control Messages, Determine the routing and Manage the Redirection GroupID List
Long Description	The routing/filtering/redirection list management logic is as follows: If Control Message - check if redirecting this groupID, check if LastInBatch, remove from RedirList for RedirectAll if LastInBatch but not RedirectPerm, discard CtrlMsg If not Control Message - Check if Redirecting this groupID If Redirecting this groupID - Are we RedirectFirst, RedirectAll or RedirectPerm? If RedirectFirst - remove GroupID from RedirList, strip RFH2 and propagate for redirection If RedirectAll - Are we LastInBatch? If LastInBatch - remove GroupID from RedirList, strip RFH2 and propagate for redirection If NOT LastInBatch - strip RFH2 and propagate for redirection If RedirectPerm - strip RFH2 and propagate for redirection If ClearRedirectPerm - remove GroupID from RedirList DO NOT propagate at all !!! This should have already happened in IOBKWrapper_RestoreSeqNum_SetRedir and hence never execute here end
ESQL File	IOBKWrapper.esql

1.14 ThrowUnknownParserNode - "ThrowUnknownParser"

Short Description	3001 to 3049 available for users
Long Description	3048

1.15 StoreAndForwardNode - "StoreAndForward"

Short Description	StoreAndForward contains the Resequence node configured in Retry Mechanism = 'failure' mode. This causes the resequence node to catch failed messages and "block" subsequent messages with the same sequence groupID
Long Description	StoreAndForward contains the Resequence node configured in Retry Mechanism = 'failure' mode. This causes the resequence node to catch failed messages and "block" subsequent messages with the same sequence groupID. The resequence node assigns a sequence number to messages by sequence groupID (i.e. it is tolerant of no sequence number) The path to sequence group ID on the Resequence node is a promoted property that identifies how a developer wants to group their messages - an employee ID for example. All other parameters on the Resequence node can be left as configured. The StoreAndForward subflow also uses a filter node to identify if a Failure Event is due to a downstream error being thrown back to the resequence node or of the resequence node itself has suffered an exception. If the resequence node itself suffers an exception

	then the Failed Event is not propagated on as downstream failure. Instead it is thrown back to be caught upstream as a FATAL error. It should then trigger the stopping of the containing message flow.
Subflow	StoreAndForward.subflow

1.16 TrcWrpRedirNode - "TrcWrpRedir"

Short Description
Long Description

1.17 TrcWrpCtB4SaFNode - "TrcWrpCtB4SaF"

Short Description
Long Description

1.18 Flow OrderNode - "Flow Order"

Short Description	Create two copies of the failed event message - Order of delivery is not actually important
Long Description	Create two copies of the failed event message - Order of delivery is not actually important One copy is sent to the ToolWrapper Second copy is sent back into the StoreAndForward (resequence node) subflow as a control message

1.19 MarkAsCrtlHoldOpenNode - "MarkAsCrtlHoldOpen"

Short Description	
Long Description	A control message with the correct groupID needs to be fed back into the StoreAndForward (resequence node) subflow in order that the sequence group is held open. Mark the message as a control message MQRFH2.usr.IOBK.CtrlMsg = Y such that it can be filtered out when the group id replayed. Note: On replay Control messages can be the LastInBatch (this is often but not always the case) The sequence group id is important resequence node in the StoreAndForward subflow will pick it up from the data based on the Path to Sequence Group ID
ESQL File	IOBKWrapper.esql

1.20 TrcWrpDisCtrlNode - "TrcWrpDisCtrl"

Short Description
Long Description

1.21 TrcWrpCtrlInNode - "TrcWrpCtrlIn"

Short Description
Long Description

2 Cross Reference

The tables in this section show how resources are related to one another.

3 Documentation generation settings

General settings

Title:	Failed Event Management Wrapper
Author:	Dave Arnold
Date:	09/01/2015
TOC nesting level:	7

Utilized documentation units

Source	Documentation unit	Documentation type
IOBKWrapper.subflow	SubFlow Documentation Unit	detailed