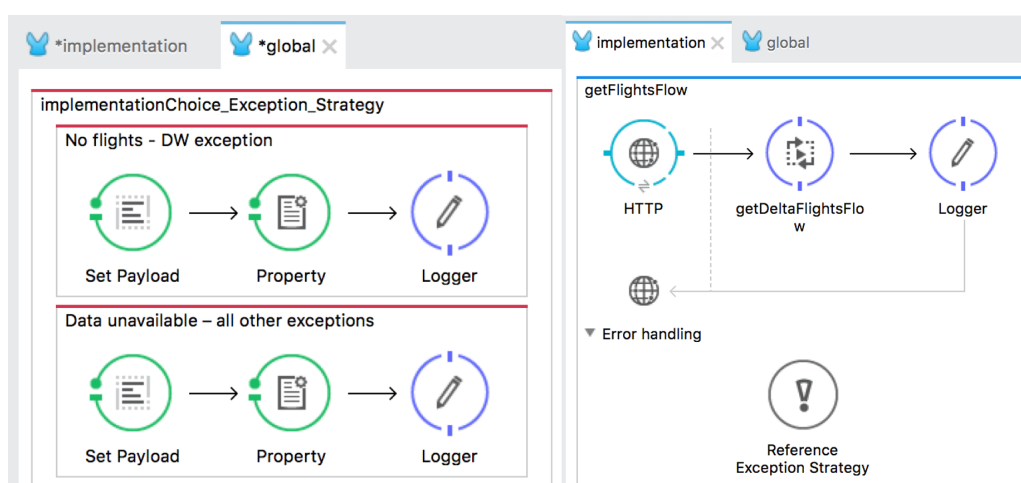




Module 8: Handling Errors

Goal



Objectives



- Describe the different types of exception strategies
- Handle messaging exceptions in flows
- Create and use global exception handlers
- Specify a global default exception strategy

All contents © MuleSoft Inc.

3

Handling system exceptions



Types of exceptions



- System exceptions
 - Thrown at the system-level when *no* message is involved
 - Exceptions that occur
 - During application start-up
 - When a connection to an external system fails
- Message exceptions
 - Thrown within a flow whenever a message is involved

Handling system exceptions



- When a system exception occurs, a system exception strategy is invoked
 - Non configurable
 - Logs the exception
 - If the exception was caused by a connection failure, executes the reconnection strategy

Reconnection strategies



- Set for each connector
- Some connectors have reconnection options in the Global Element properties GUI
- For most others, you set connector properties in XML
 - Set attempt count and frequency in ms
 - Set it to blocking or non-blocking
 - Notify registered listeners

```
<jms:activemq-connector name="AMQConnector">
  <reconnect count="5" frequency="1000"/>
</jms:activemq-connector>
```

Setting reconnection properties



Global Element Properties

Salesforce: Basic authentication
Global Basic authentication configuration information. Salesforce connector configuration that uses basic authentication for establishing connection with Salesforce system.

General Pooling Profile **Reconnection** Notes

Strategies

Define how mule should attempt to handle a connection failure

☒ Do not use a Reconnection strategy

☐ Standard Reconnection

Frequency (ms):

Reconnection Attempts:

☐ Reconnect Forever

☐ Custom Reconnection

Class: ...

Properties

Name	Value	Reference

Test Connection... Cancel OK

Handling messaging exceptions

The default exception strategy

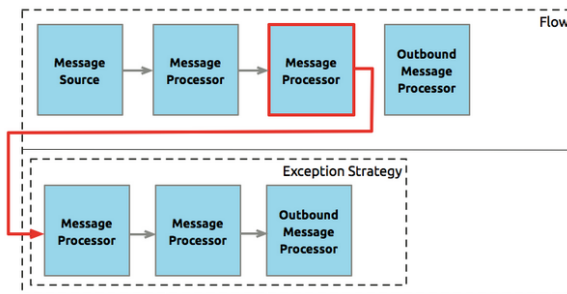


- If there is no exception strategy explicitly defined, Mule's default exception strategy is used
- The default exception strategy
 - Implicitly and globally handles all messaging exceptions thrown in Mule applications
 - Stops execution of the flow and logs the exception
 - Cannot be configured
 - Can be replaced with your own global default exception strategy
 - We will do this later this module

Handling messaging exceptions



- When a message being processed through a Mule flow throws an exception
 - Normal flow execution stops
 - The message is passed to the first processor in the exception strategy



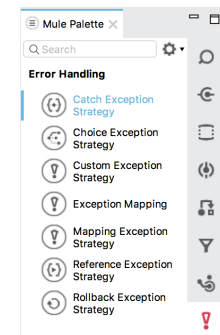
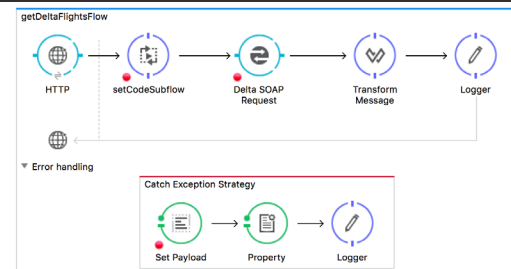
All contents © MuleSoft Inc.

11

Defining messaging exception strategies



- Exception strategies are added to the error handling section of a flow
- Each flow can contain only one exception strategy
 - Choice exception strategies can contain one or more catch and/or rollback exception strategies
- Each exception strategy can contain any number of message processors



All contents © MuleSoft Inc.

12

Referencing the exception inside the strategy



- Inside an exception strategy, you can reference the exception object
 - `<logger level="ERROR" message="#[exception]"/>`
- Use methods to get different amounts of detail about it
 - `exception.getSummaryMessage()`
 - `exception.getVerboseMessage()`
 - `exception.getMessageCode()`
 - `exception.getDetailedMessage()`

Return status codes

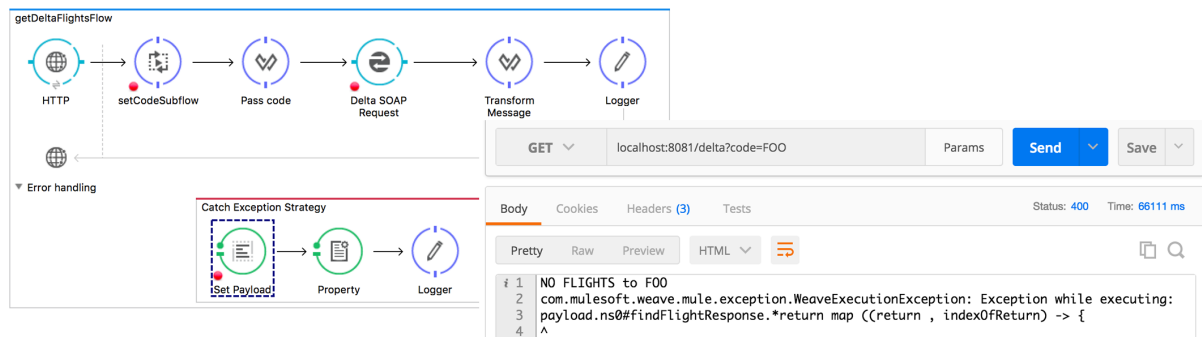


- By default, when a message is handled, an HTTP status code of 200 is set and returned
- You can change this by setting outbound property
 - `message.outboundProperties.'http.status'`
- You can also use the mapping exception strategy
 - We will take a look at the one generated earlier with APIkit

Walkthrough 8-1: Handle a messaging exception



- Add a catch exception strategy to a flow
- Catch the exception and set the payload to send an error message back
- Reference the exception object inside an exception handler
- Set an HTTP status code inside the exception handler



Handling different types of exceptions

The choice exception strategy



- The choice exception strategy must contain two or more catch and/or rollback strategies
- The individual catch and/or rollback strategies specify a condition for when they should be evaluated
- The choice strategy routes the message to the *first* exception strategy whose condition evaluates to true
- If none of its exception strategies can handle the error, the message is routed to Mule's default exception strategy

Error Handling

- Catch Exception Strategy
- Choice Exception Strategy
- Custom Exception Strategy
- Exception Mapping
- Mapping Exception Strategy
- Reference Exception Strategy
- Rollback Exception Strategy

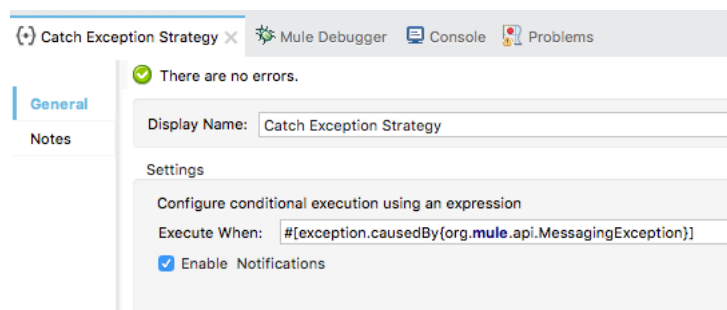
All contents © MuleSoft Inc.

16

Setting exception strategy conditions



- Can reference the message or the exception
 - `exception.causedBy(org.mule.example.ExceptionType)`
 - `exception.causedExactlyBy(org.mule.example.ExceptionType)`
 - `exception.causeMatches('org.mule.example.*')`



All contents © MuleSoft Inc.

17

Bubbling exceptions



- All flows can have their own exception strategies
- If a flow does not have an exception strategy, the strategy of the calling flow is used

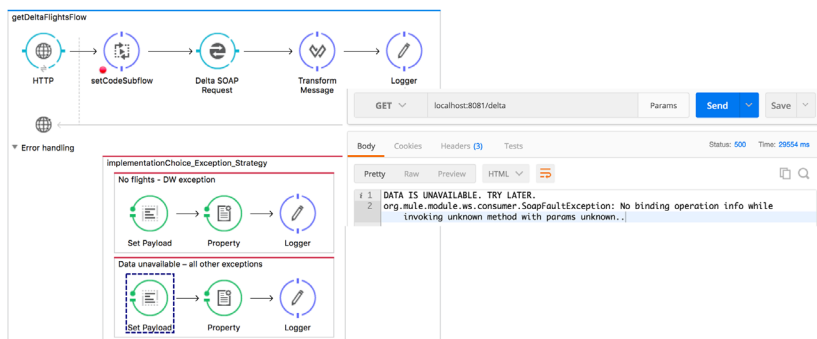
All contents © MuleSoft Inc.

19

Walkthrough 8-2: Handle different types of messaging exceptions



- Add and configure a choice exception strategy
- Get exceptions handled by both of the catch exception strategies in the choice strategy
- Create a new flow that calls a flow that has an exception so the exception can bubble up and be handled by the calling flow



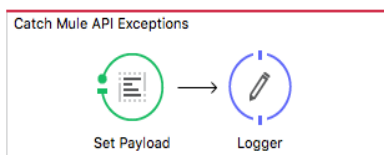
20

Defining global exception strategies

Defining global exception strategies



- You can reuse exception handling strategies by defining them outside a flow
 - You can drag them out and drop them outside any flow
 - Typically, put them in your global configuration file



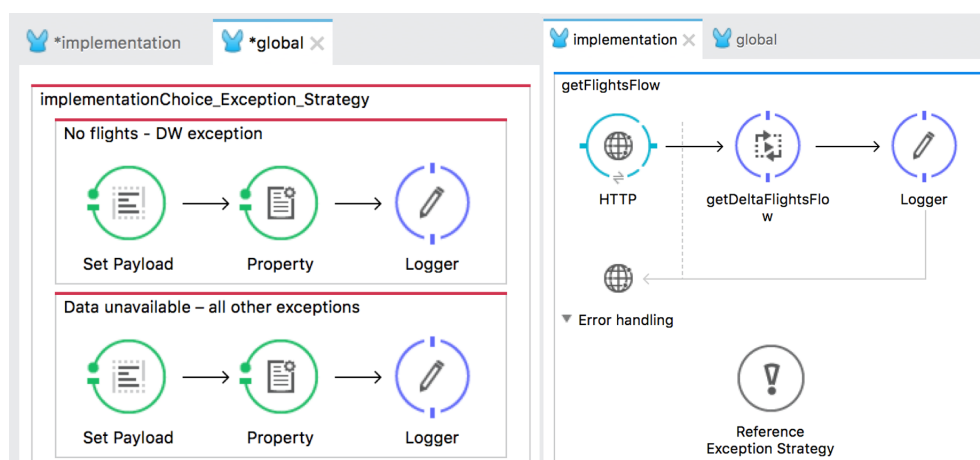
- Reference global exception handlers using the reference exception strategy



Walkthrough 8-3: Create and use global exception strategies



- Create a global exception handler
- Reference and use the global exception handler in flows



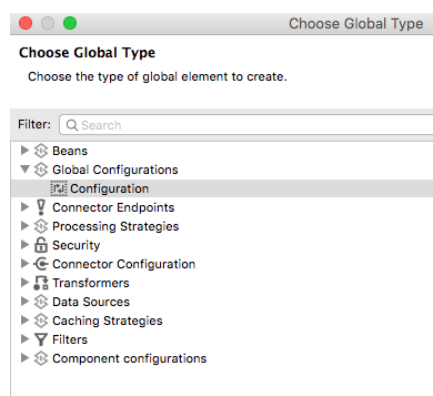
23

Defining a default exception strategy

The default exception strategy



- Recall there is a default exception strategy
 - Implicitly handles all unhandled messaging exceptions
 - Simply logs the exception
 - Cannot be configured
- Can be replaced with your own global default exception strategy
 - Create a global configuration element
 - Specify a default exception strategy in the global configuration element



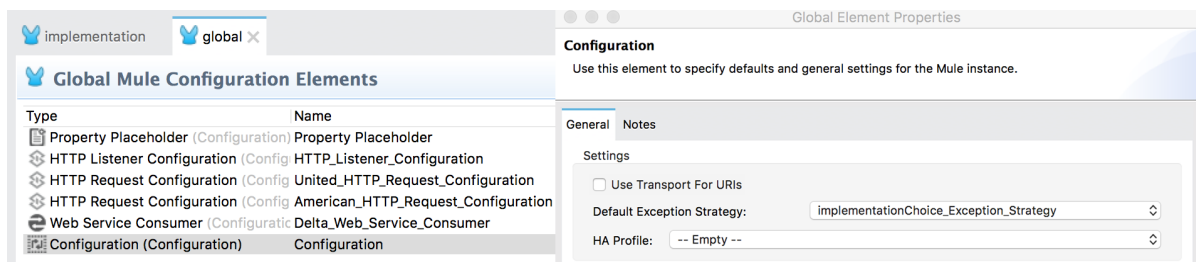
All contents © MuleSoft Inc.

25

Walkthrough 8-4: Specify a global default exception strategy



- Create a global configuration element in the global.xml file
- Specify a default exception strategy in the global configuration element
- Remove the existing exception handling strategies
- Use the default exception handling strategy



Reviewing a mapping exception strategy

Walkthrough 8-5: Review a mapping exception strategy



- Locate the mapping exception strategy that was created automatically by APIKit for the interface for the American flights web service
- Review the exception mappings

The screenshot displays the MuleSoft IDE interface. On the left, the Package Explorer shows the project structure for 'apdev-american', including 'src/main/app' and 'src/main/api'. The main editor area shows the 'american-apiKitGlobalExceptionMapping' strategy. It contains two exception mappings: 404 and 405. Each mapping is represented by a diagram with a 'Property' icon and a 'Set Payload' icon. The right-hand pane shows the 'General' tab for the 404 exception, indicating 'There are no errors.' and showing the 'Status Code' as 404. The 'Exceptions' section lists the 'Expected Type' as 'org.mule.module.apikit.exception.NotFoundException'.

Summary



Summary



- An application can have system or message exceptions
- System exceptions are thrown at the system level and involve no message
 - Occur during application start-up or when a connection to an external system fails
 - Non-configurable, but logs the exception and for connections, executes any connector reconnection strategy
- Message exceptions are thrown within a flow whenever a message is involved

Summary



- If there is no exception strategy defined, the default exception strategy is used
 - Stops execution of the flow and logs the exception
 - Cannot be configured but can be replaced with your own global default exception strategy
- If there is an exception strategy, normal flow execution stops and the message is passed to the exception strategy
 - Catch strategy catches exceptions based on conditions
 - Choice strategy selects one of multiple catch and/or rollback strategies based on conditions

All contents © MuleSoft Inc.

29

Summary



- All flows can have their own exception strategies
 - If a flow does not have an exception strategy, the strategy of the calling flow is used
 - Subflows cannot have their own exception strategies
- Create global exception strategies by defining them outside a flow (typically in global.xml)
- Reference global exception handlers using the reference exception strategy
- Set a default global exception strategy by creating a global configuration element and setting its default exception strategy to a global exception strategy

All contents © MuleSoft Inc.

30