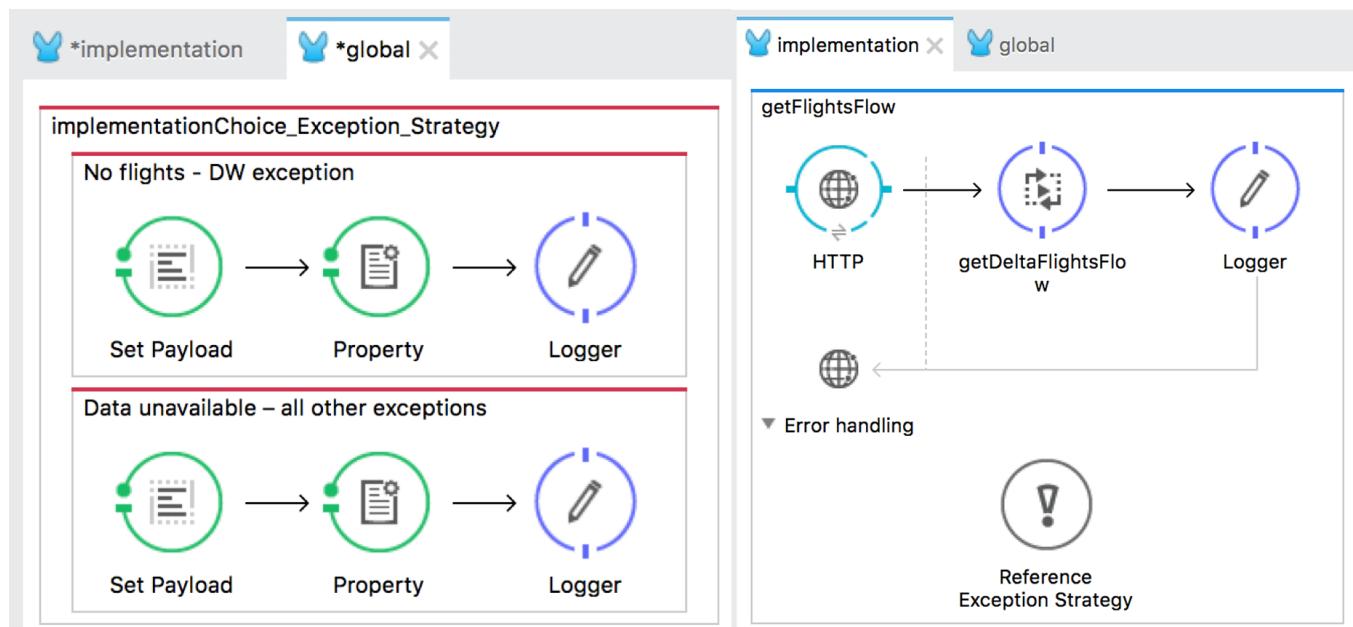


# Module 8: Handling Errors



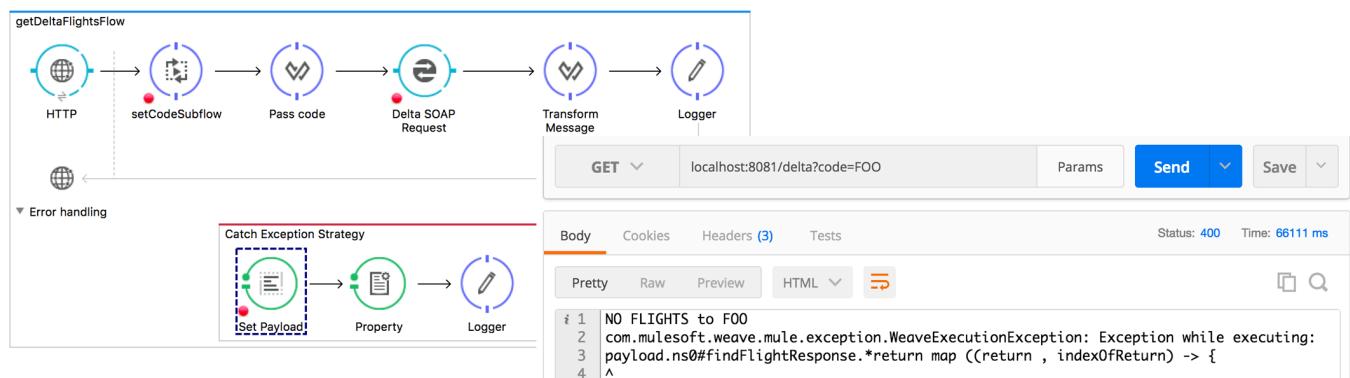
## Objectives:

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

## Walkthrough 8-1: Handle a messaging exception

In this walkthrough, you handle an exception thrown by the Delta flow when a destination with no flights is used. You will:

- 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.



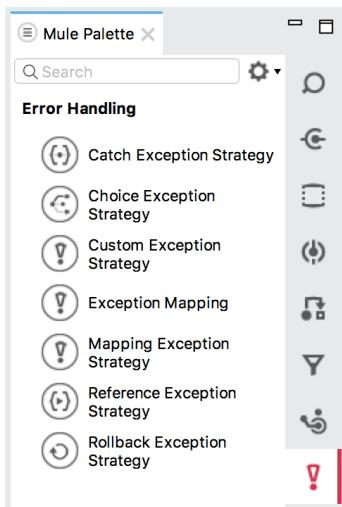
## Debug the application for a request with a non-existent destination

1. Return to getDeltaFlightsFlow.
2. Make sure there is a breakpoint on the flow reference component.
3. Debug the project.
4. In Postman, make another request to <http://localhost:8081/delta?code=FOO>.
5. In the Mule Debugger, step through the application and when you get the exception, drill-down into the exceptionThrown object.

6. Click the Resume button.

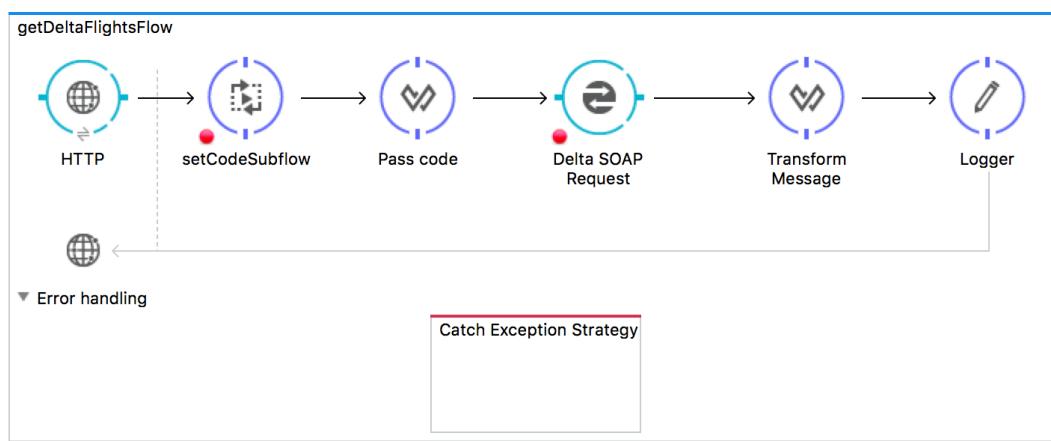
## Browse the error handling elements in the Mule Palette

7. In the Mule Palette, select the Error Handling tab.
8. View the available error handling processors.



## Add a catch exception strategy

9. In getDeltaFlightsFlow, click the arrow to expand the Error handling section.
10. Drag a Catch Exception Strategy from the Mule Palette into the error handling section of the flow.

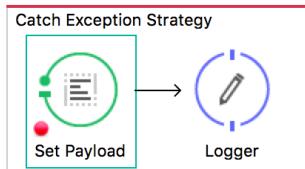


11. Add a Set Payload transformer to the catch exception strategy.
12. In the Set Payload properties view, set the value the following MEL expression:

```
NO FLIGHTS to #[flowVars.code + '\n' + exception]
```

13. Make sure there is a breakpoint on the transformer inside the catch exception.

14. Add a Logger after the transformer.

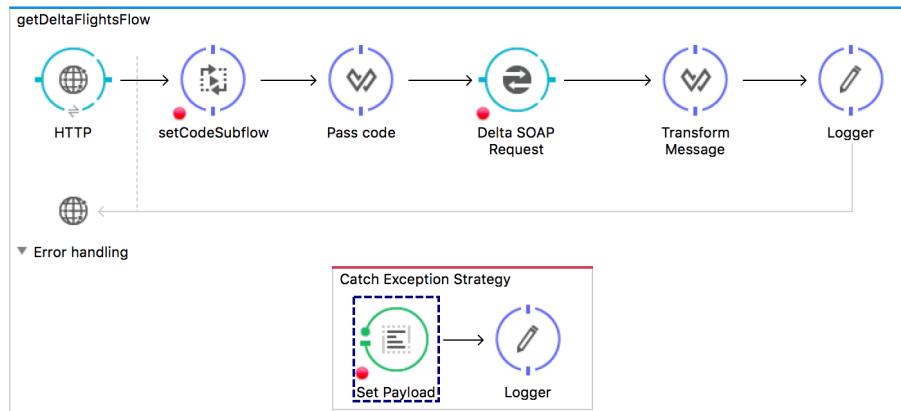


## Test the application

15. Redeploy the application in debug mode.

16. In Postman, make another request to <http://localhost:8081/delta?code=FOO>.

17. Step through the application; you should see the exception thrown in getDeltaFlightsFlow and handled by the exception handler.



18. Step to the end of the application.

19. In Postman, you should get a 200 response with your custom message.

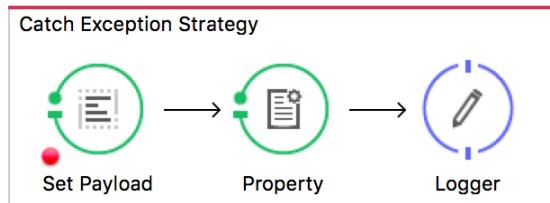
A screenshot of the Postman interface. At the top, there is a search bar with "localhost:8081/delta?code=FOO" and a "Send" button. Below the search bar, the "Body" tab is selected, showing a text area with the following content:

```
i 1 NO FLIGHTS to FOO
2 com.mulesoft.weave.mule.exception.WeaveExecutionException: Exception while executing:
3 payload.ns0#findFlightResponse.*return map ((return , indexOfReturn) -> {
4 ^
5 Type mismatch for 'MultiValue Selector (*.*)' operator
6     | found :string, :name
7     required :object, :name or
8     required :array, :name|
```

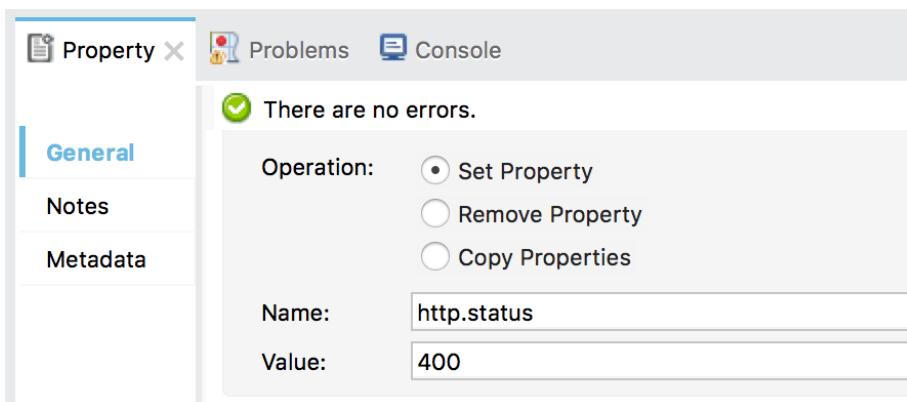
At the top right of the body panel, it says "Status: 200 OK" and "Time: 35082 ms".

## Set the http status code

20. Return to getDeltaFlightsFlow.
21. In the catch exception strategy, add a Property transformer after the Set Payload transformer.

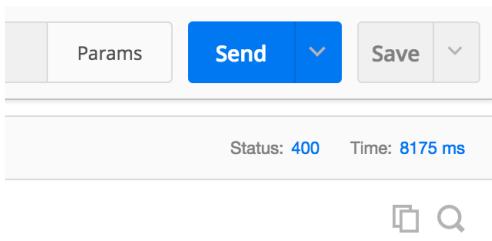


22. In the Property properties view, select Set Property.
23. Set the name to http.status and the value to 400.



## Test the application

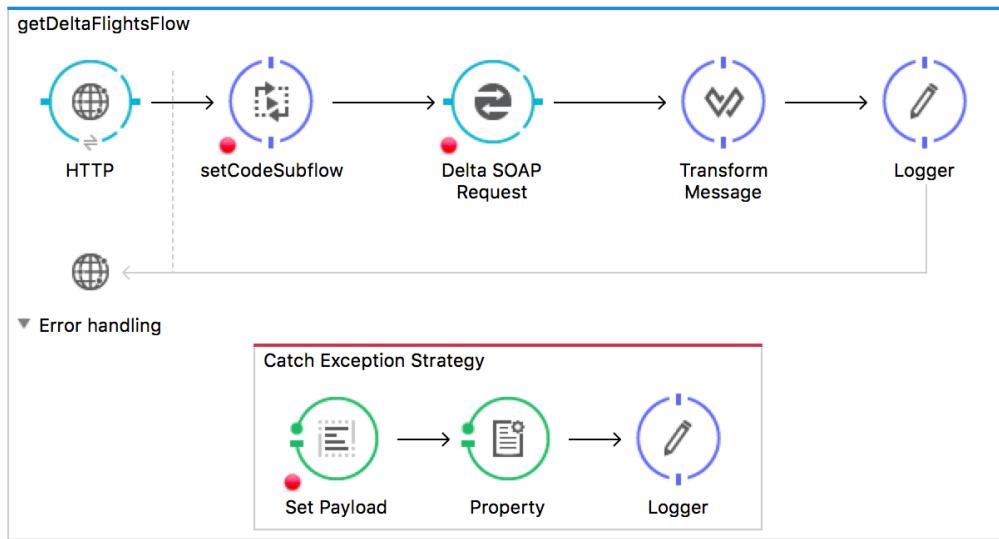
24. Redeploy the application.
25. In Postman, make another request to <http://localhost:8081/delta?code=FOO>.
26. In the Mule Debugger, step through the application.
27. Return to Postman; you should now get a 400 status code.



## Create a different type of error

28. Return to getDeltaFlightsFlow.

29. Delete the Pass code processor.



## Test the application

30. Redeploy the application.
31. In Postman, delete the query parameter and send the request.
32. In the Mule Debugger, step through the application until you get the exception and move into the exception handler; you should see the exception is handled by the same exception handler.
33. Drill-down into the Exception object in the debugger.
34. Click Resume.
35. In Postman, you should get the same message, but this time saying there are no flights to SFO even though that was not the problem.

GET localhost:8081/delta Params Send Save

Status: 400 Time: 8658 ms

Body Cookies Headers (3) Tests

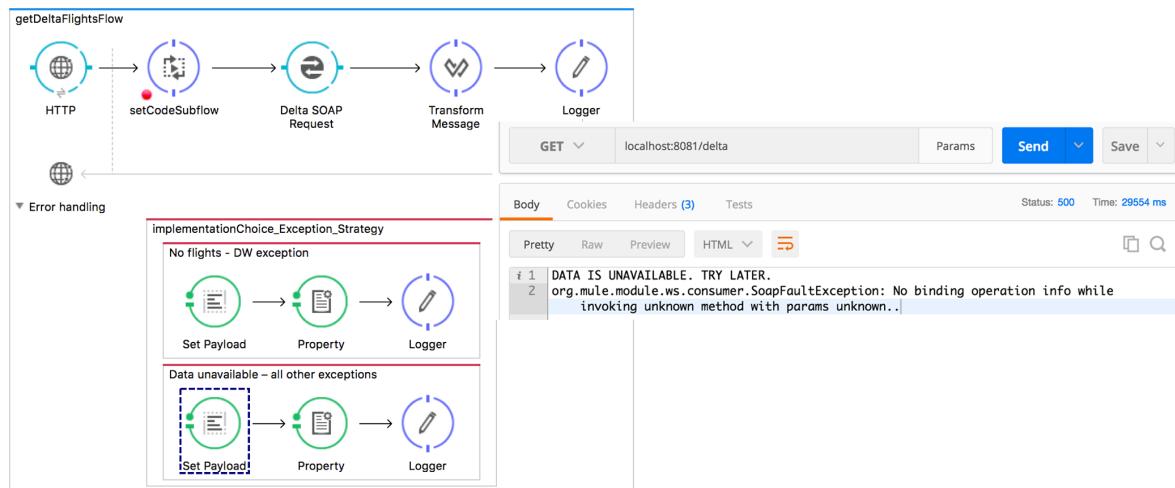
Pretty Raw Preview HTML HTML Copy Search

i 1 NO FLIGHTS to SFO  
2 org.mule.module.ws.consumer.SapFaultException: No binding operation info while  
invoking unknown method with params unknown..|

## Walkthrough 8-2: Handle different types of messaging exceptions

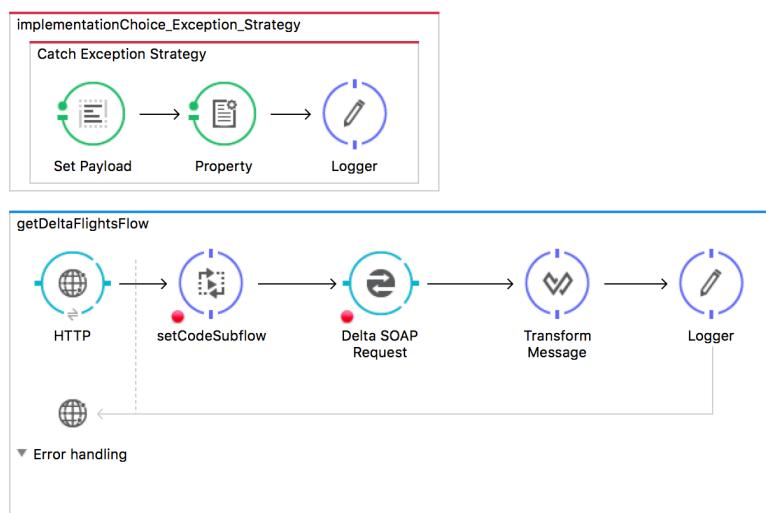
In this walkthrough, you handle multiple types of exceptions thrown by the Delta flow. You will:

- 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.



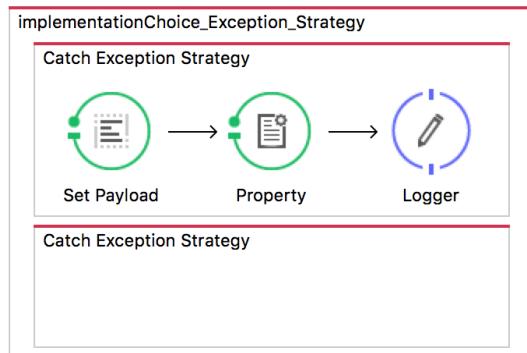
### Add a choice exception strategy

1. Return to implementation.xml.
2. Drag a Choice Exception Strategy from the Mule Palette and drop it above getDeltaFlightsFlow.
3. Drag the catch exception strategy from getDeltaFlightsFlow and drop it in the choice exception strategy.



4. Drag a second Catch Exception Strategy from the Mule Palette and drop it in the choice exception strategy.

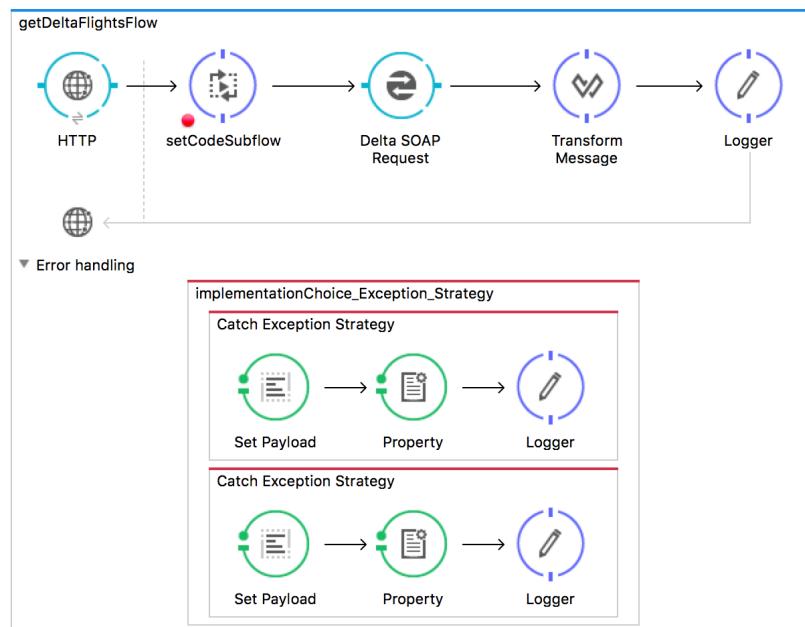
*Note: If you are having difficulty adding it, try dropping it on the catch exception strategy already there.*



5. Drag the choice exception strategy and drop it in the error handling section of getDeltaFlightsFlow.

*Note: If you are having difficulty adding it because the canvas is scrolling, try changing the order of the flows on the canvas and then dragging and dropping.*

6. Add a Set Payload transformer, a Property transformer, and a Logger to the new catch exception strategy.



7. In the Set Payload properties view, set the value the following MEL expression:

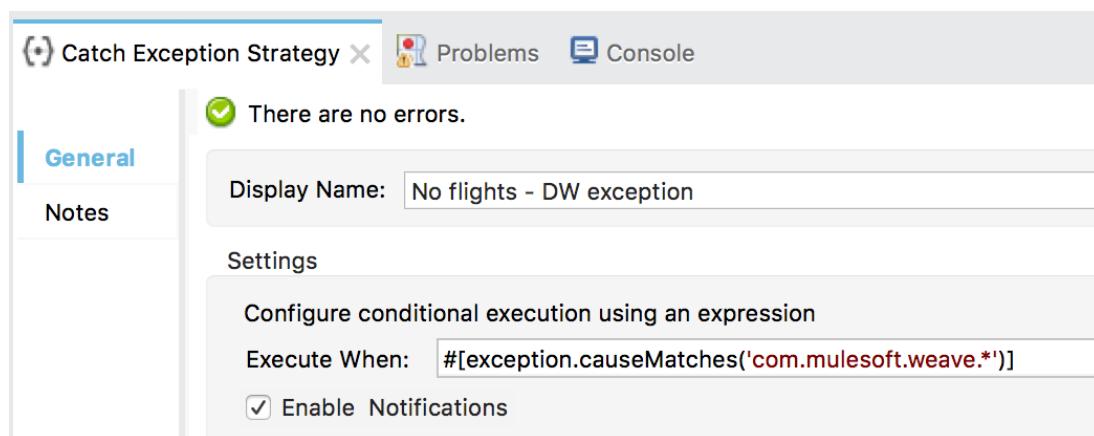
```
DATA IS UNAVAILABLE. TRY LATER. #[ '\n' + exception ]
```

8. In the Property properties view, select Set Property.
9. Set the name to http.status and the value to 500.

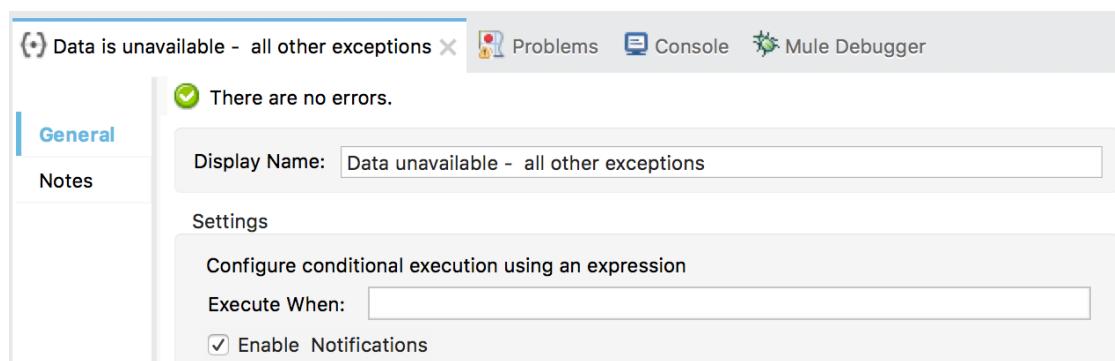
## Configure the choice exception strategy

10. In the Properties view for the first catch exception strategy, set the display name to No flights – DW exception.
11. Set the execute when value to an expression for when a com.mulesoft.weave.\* exception is thrown; use the causeMatches() method.

```
##[exception.causeMatches('com.mulesoft.weave.*')]
```



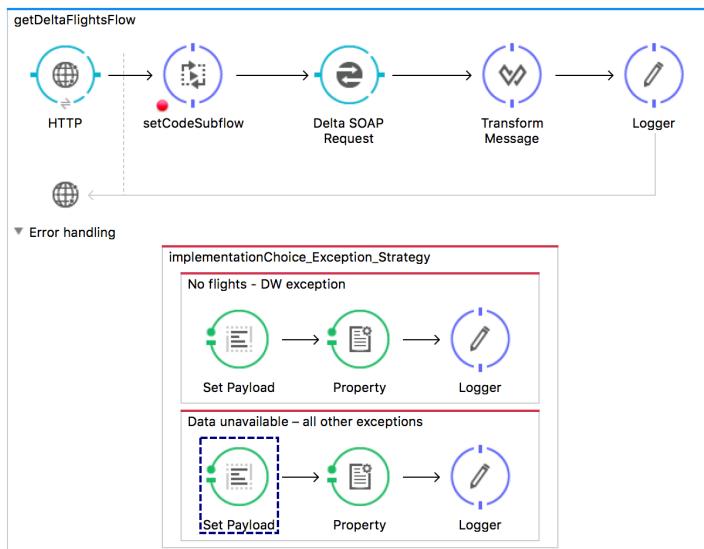
12. In the Properties view for the second catch exception strategy, set the display name to Data unavailable – all other exceptions.
13. Leave the execute when value blank.



## Test the application

14. Debug the application.
15. In Postman, make another request to <http://localhost:8081/delta>.

16. In the Mule Debugger, step through the application; the exception should be handled by the second catch block.



17. Step to the end of the application.

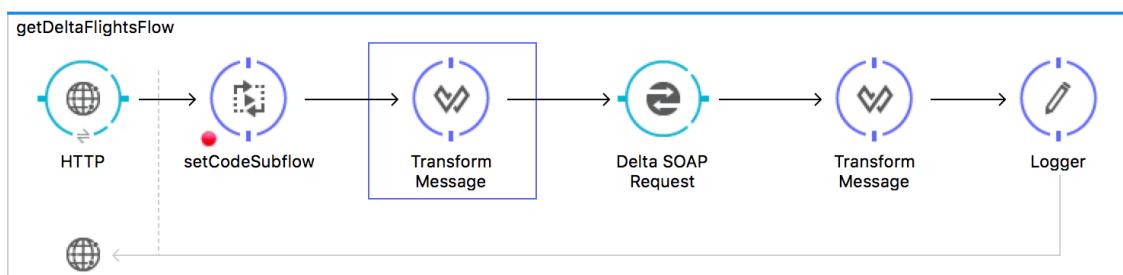
18. Return to Postman; you should see the second error message.

A screenshot of the Postman application interface. At the top, there are buttons for 'GET', 'localhost:8081/delta', 'Params', 'Send', 'Save', and a dropdown. Below this is a table with tabs for 'Body', 'Cookies', 'Headers (3)', and 'Tests'. The 'Body' tab is selected. The status bar at the top right shows 'Status: 500' and 'Time: 29554 ms'. The 'Body' section has tabs for 'Pretty', 'Raw', 'Preview', and 'HTML'. The 'Pretty' tab is selected, showing the error message: 'i 1 DATA IS UNAVAILABLE. TRY LATER.' and '2 org.mule.module.ws.consumer.SoapFaultException: No binding operation info while invoking unknown method with params unknown..|'. There are also icons for copy and search.

## Fix the request error

19. Return to getDeltaFlightsFlow.

20. Add a Transform Message component before the Delta SOAP Request processor.



21. In the Transform Message properties view, drag the code flow variable in the input section to the destination element in the output section.

## Test the application

22. Run the project.
23. In Postman, make the same request to <http://localhost:8081/delta>; you should get the SFO results again.
24. Make a request to <http://localhost:8081/delta?code=FOO>; the exception is handled by the first exception strategy and you should get the NO FLIGHTS TO FOO message.

The screenshot shows a Postman interface. At the top, there's a header bar with 'GET' selected, a URL field containing 'localhost:8081/delta?code=FOO', a 'Params' button, a 'Send' button, and a 'Save' button. Below this is a main panel with tabs for 'Body', 'Cookies', 'Headers (3)', and 'Tests'. The 'Body' tab is active and displays a JSON-like structure with three items:

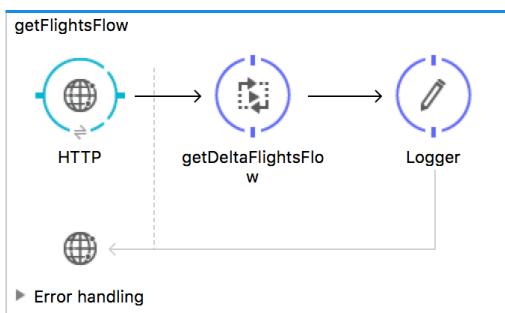
```
i 1 NO FLIGHTS to FOO
2 com.mulesoft.weave.mule.exception.WeaveExecutionException: Exception while executing:
3 payload.ns0#findFlightResponse.*return map ((return , indexOfReturn) -> {
```

At the bottom right of the main panel, there are icons for 'Pretty', 'Raw', 'Preview', 'HTML' (which is currently selected), and a copy icon. To the right of the main panel, it says 'Status: 400' and 'Time: 185 ms'.

## Call the Delta flow from another flow

25. Return to Anypoint Studio.
26. From the Mule Palette, drag an HTTP connector and drop it at the top of the canvas above all the other flows.
27. In the HTTP properties view, set the connector configuration to the existing `HTTP_Listener_Configuration`.
28. Set the path to `/flights` and the allowed methods to `GET`.
29. Change the flow name to `getFlightsFlow`.
30. Add a Flow Reference component to the flow.
31. In the Flow Reference properties view, set the flow to `getDeltaFlightsFlow`.

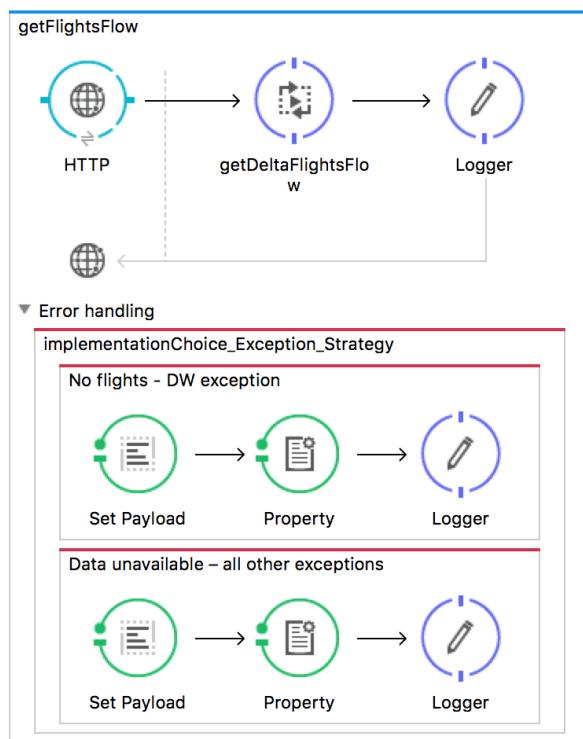
32. Add a Logger to the end of the flow.



## Move a catch exception strategy to the calling flow

33. Expand the error handling section of getFlightsFlow.

34. Move the choice exception strategy from getDeltaFlightsFlow to getFlightsFlow.



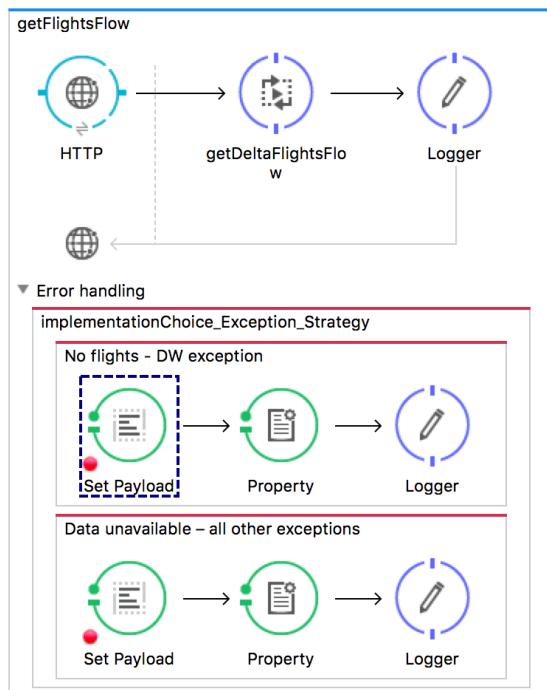
## Test the application

35. Debug the project.

36. Make sure there is a breakpoint on each of the Set Payload transformers in the catch exception strategies.

37. In Postman, make a request to <http://localhost:8081/flights?code=FOO>.

38. In the Mule Debugger, step through the application; you should see the exception thrown by the Delta flow is caught by the calling flow.

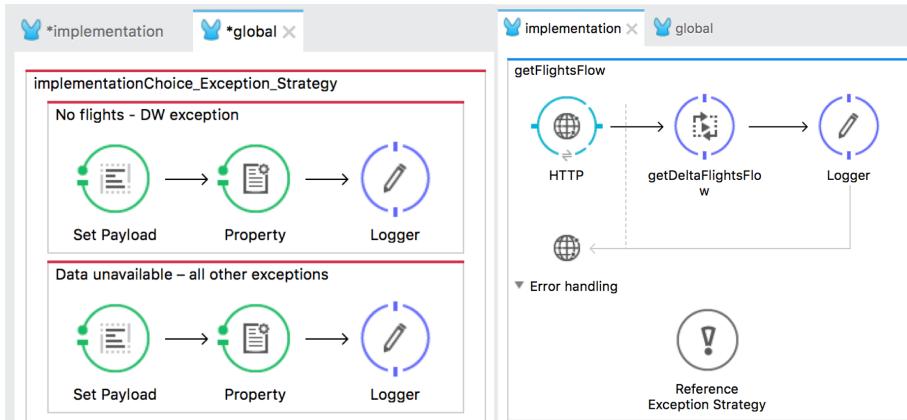


39. Step through the rest of the application and then stop the project.

## Walkthrough 8-3: Create and use global exception strategies

In this walkthrough, you create a global exception handler for the project. You will:

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



### Create a global exception handler

1. Return to implementation.xml.
2. Switch to the Configuration XML view.
3. Locate the choice exception strategy and select it and cut it.

```
<choice-exception-strategy doc:name="implementationChoice_Exception_Strategy">
  <catch-exception-strategy when="#[exception.causeMatches('com.mulesoft.weav
    <set-payload value="NO FLIGHTS to #[flowVars.code + '\n' + exception]">
    <set-property propertyName="http.status" value="400" doc:name="Property">
    <logger level="INFO" doc:name="Logger"/>
  </catch-exception-strategy>
  <catch-exception-strategy doc:name="Data unavailable &#8211; all other exce
    <set-payload value="DATA IS UNAVAILABLE. TRY LATER. #['\n' + exception]">
    <set-property propertyName="http.status" value="500" doc:name="Property">
    <logger level="INFO" doc:name="Logger"/>
  </catch-exception-strategy>
</choice-exception-strategy>
```

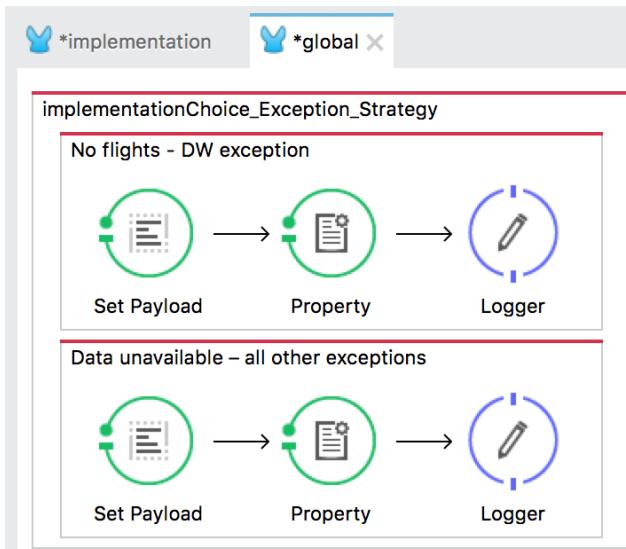
4. Go to the Configuration XML view in global.xml.
5. Place the cursor on a new line inside the start and end mule tags after the other tags.
6. Paste the choice exception strategy.

7. In the choice-exception-strategy tag, change doc:name to name; the problem should go away



```
17      </http:request-config>
18      <ws:consumer-config name="Delta_Web_Service_Consumer" service="Ticketing">
19
20      <choice-exception-strategy name="implementationChoice_Exception_Strategy">
21          <catch-exception-strategy when="#[exception.causeMatches('NO FLIGHTS')]">
22              <set-payload value="NO FLIGHTS to #[flowVars.code + '\n'" />
23              <set-property propertyName="http.status" value="400" doc:name="Set HTTP Status"/>
24              <logger level="INFO" doc:name="Logger"/>
25          </catch-exception-strategy>
26          <catch-exception-strategy doc:name="Data unavailable &#8211; all other exceptions">
27              <set-payload value="DATA IS UNAVAILABLE. TRY LATER. #[exception.cause]" />
28              <set-property propertyName="http.status" value="500" doc:name="Set HTTP Status"/>
29              <logger level="INFO" doc:name="Logger"/>
30          </catch-exception-strategy>
31      </choice-exception-strategy>
32
33  </mule>
```

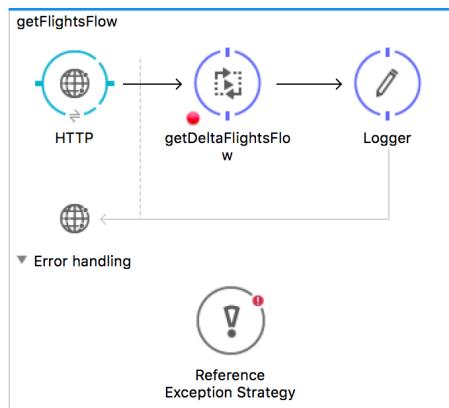
8. Switch to the Message Flow view; you should see the choice exception strategy.



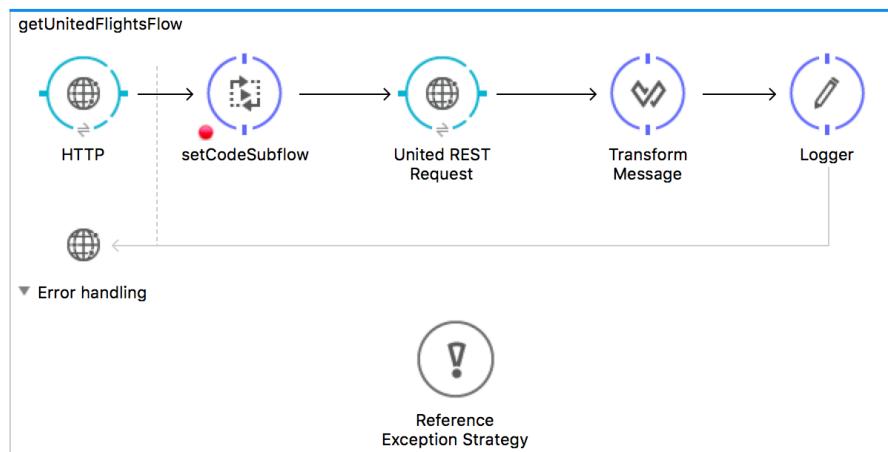
## Use the global exception handler

9. Return to implementation.xml and switch to the Message Flow view.  
10. Locate getFlightsFlow and expand its error handling section.

11. Drag a Reference Exception Strategy from the Mule Palette and drop it in the error handling section.



12. In the Reference Exception Strategy properties view, set the global exception strategy to the existing implementationChoice\_Exception\_Strategy.  
 13. Expand the error handling section of getUnitedFlightsFlow.  
 14. Drag a Reference Exception Strategy from the Mule Palette and drop it in the error handling section.  
 15. In the Properties view, set the global exception strategy to implementationChoice\_Exception\_Strategy.



*Note: You could add a reference exception strategy to the rest of the flows, but instead, you will create a global default exception strategy in the next walkthrough.*

## Test the application

16. Run the project.

17. In Postman, make the same request to <http://localhost:8081/flights?code=FOO>; you should get the same NO FLIGHTS to FOO message.

GET localhost:8081/delta?code=FOO Params Send Save

Body Cookies Headers (3) Tests Status: 400 Time: 185 ms

Pretty Raw Preview HTML `≡` □ 🔍

```
i 1 NO FLIGHTS to FOO
2 com.mulesoft.weave.mule.exception.WeaveExecutionException: Exception while executing:
3 payload.ns0#findFlightResponse.*return map ((return , indexOfReturn) -> {
```

18. Make a request to <http://localhost:8081/united?code=FOO>; you should see the same message.  
19. Make a request to <http://localhost:8081/delta?code=FOO>; you should NOT see the same message because the exception is not being handled.

GET localhost:8081/delta?code=FOO Params Send Save

Body Cookies Headers (2) Tests Status: 200 Exception while executing: &#10;payload.ns0#findFlightResponse.\*return map Time: 118 ms

Pretty Raw Preview HTML `≡` □ 🔍

```
i 1 Exception while executing:
2 payload.ns0#findFlightResponse.*return map ((return , indexOfReturn) -> {
3 ^
4 Type mismatch for 'MultiValue Selector (*.*)' operator
5     found :string, :name
6     required :object, :name or
7     required :array, :name|
```

## Walkthrough 8-4: Specify a global default exception strategy

In this walkthrough, you change the default exception handling for the application. You will:

- 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.

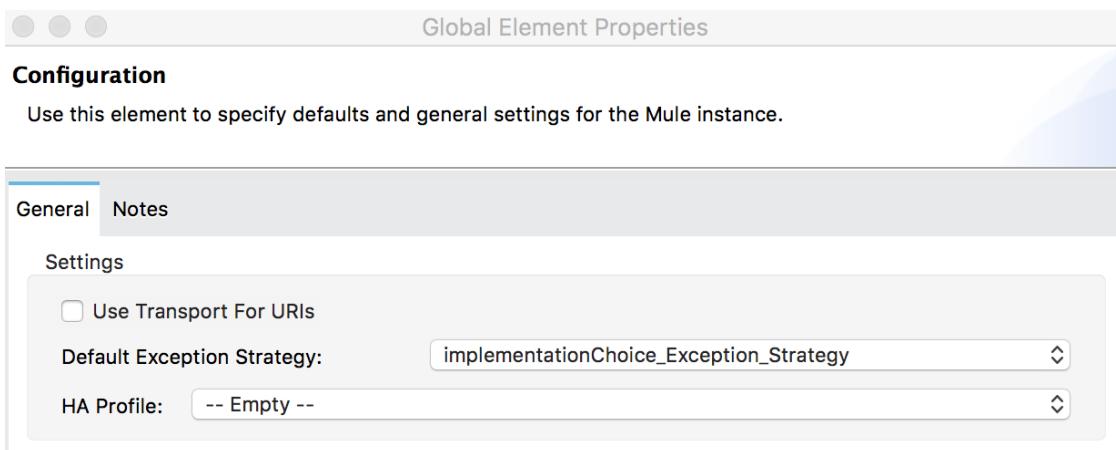
The screenshot shows the 'Global Element Properties' dialog with the 'Configuration' tab selected. The left pane lists 'Global Mule Configuration Elements' with a table of type and name. The right pane shows settings for the 'Implementation' configuration, including a checkbox for 'Use Transport For URIs', a dropdown for 'Default Exception Strategy' set to 'implementationChoice\_Exception\_Strategy', and a dropdown for 'HA Profile' set to '-- Empty --'.

### Specify a global default exception strategy

1. Return to global.xml.
2. Switch to the Global Elements view.
3. Click the Create button.
4. In the Choose Global Type dialog box, select Global Configurations > Configuration and click OK.

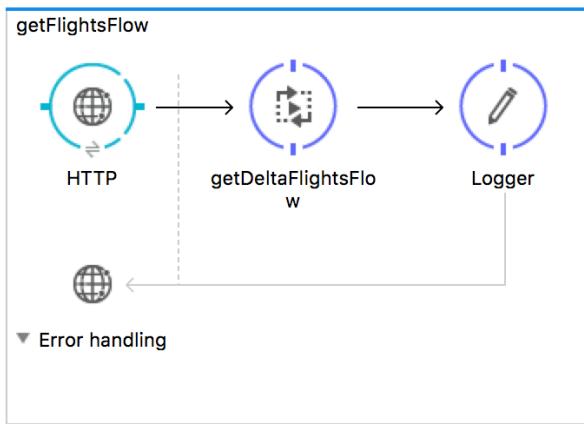
The screenshot shows the 'Choose Global Type' dialog with the title 'Choose Global Type'. It has a search bar labeled 'Filter: Search' and a tree view under 'Choose Global Type' with items like Beans, Global Configurations (selected), Configuration (highlighted), Connector Endpoints, and Processing Strategies.

- In the Global Element Properties dialog box, set the default exception strategy to implementationChoice\_Exception\_Strategy and click OK.

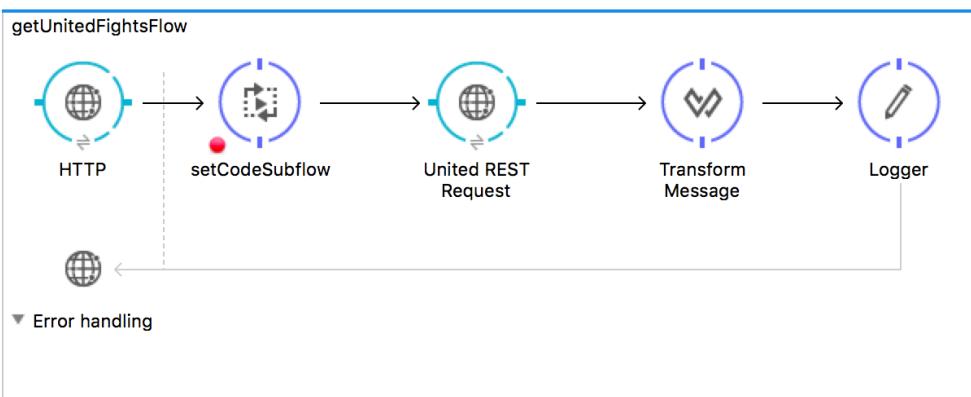


## Remove the existing exception strategy references

- Return to implementation.xml.
- Delete the reference exception strategy in getFlightsFlow.



- Delete the reference exception strategy in getUnitedFlightsFlow.



## Test the application

9. Save all the files and run the project.
10. In Postman, make a request to <http://localhost:8081/flights?code=FOO>; you should still see the no flights error message displayed.
11. Make a request to <http://localhost:8081/delta?code=FOO>; you should now see the no flights error message displayed.

GET localhost:8081/delta?code=FOO Params Send Save

Body Cookies Headers (3) Tests Status: 400 OK Time: 132 ms

Pretty Raw Preview HTML ⚡

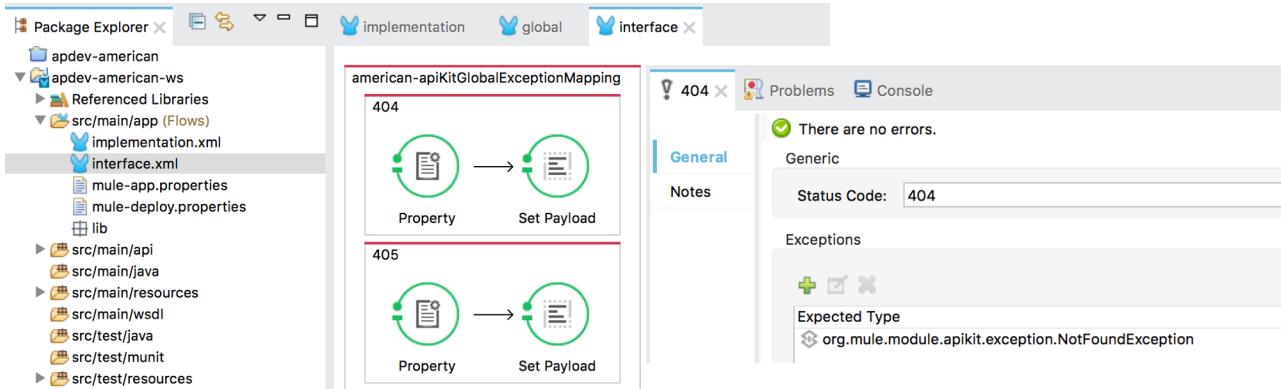
i 1 NO FLIGHTS to FOO  
2 com.mulesoft.weave.mule.exception.WeaveExecutionException: Exception while executing:

12. Return to Anypoint Studio and stop the project.

## Walkthrough 8-5: Review a mapping exception strategy

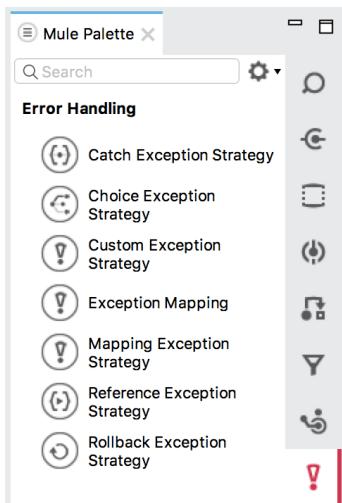
In this walkthrough, you review the mapping exception strategy that was created automatically by APIkit for the interface for the American flights web service. You will:

- Locate the mapping exception strategy.
- Review the exception mappings.



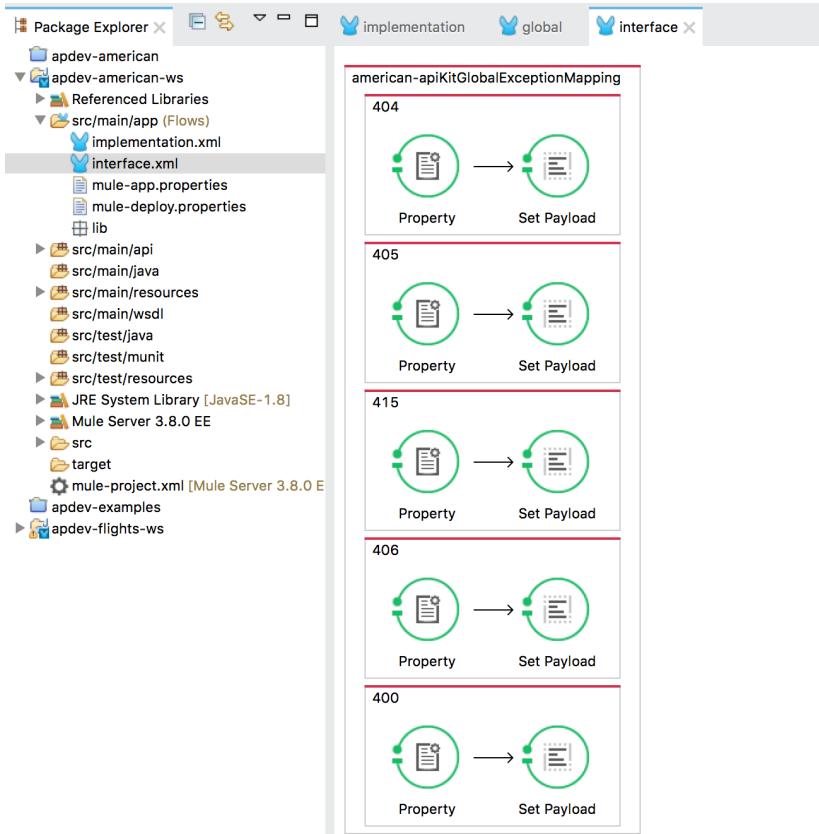
### Browse the error handling elements in the Mule Palette

1. In the Package Explorer, double-click the `apdev-american-ws` project to open it.
2. Open `interface.xml` in `src/main/app`.
3. In the Mule Palette, select the Error Handling tab.
4. Locate the Mapping Exception Strategy.

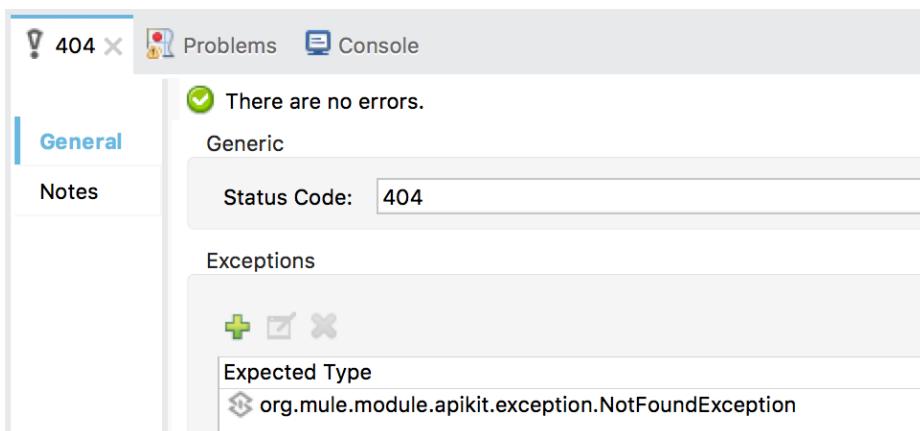


## Review a mapping exception strategy

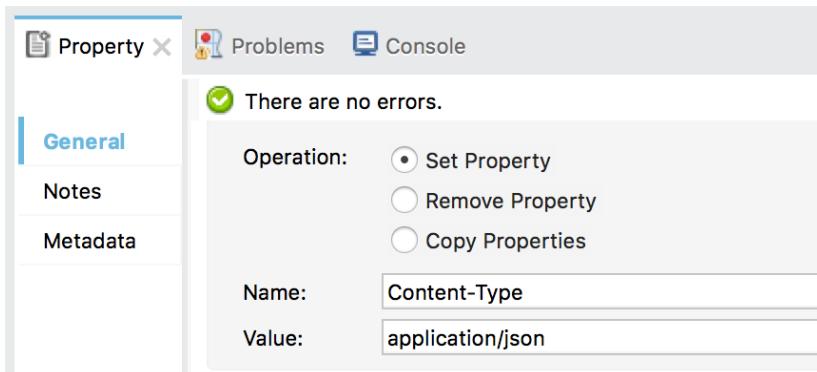
5. In interface.xml, locate the american-apiKitGlobalExceptionMapping Mapping exception strategy.



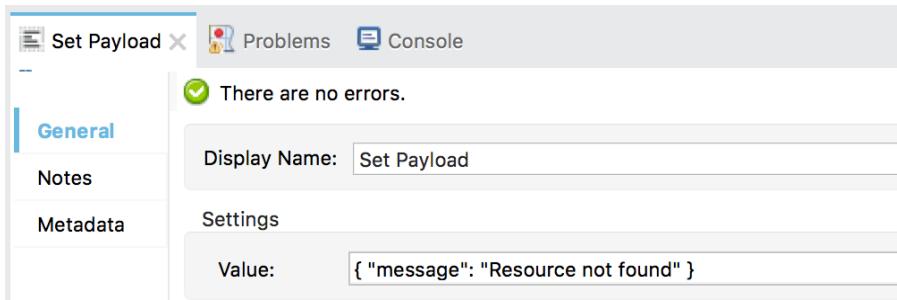
6. Double-click the 404 exception mapping and look at the settings in the 404 properties view.



7. Double-click the Property transformer in the 404 exception mapping and look at the settings in the Set Payload properties view.



8. Double-click the Set Payload transformer in the 404 exception mapping and look at the settings in the Set Payload properties view.



9. Look at the values for the Set Payload transformers in the other exception mappings.  
10. Close the project.