

META AND SPECIFIC Transactions Explained With an Example

Those who are new to Service Virtualization, its always very intriguing to understand the META and SPECIFIC Transactions. Before, we get into to the content, please read this introduction about META tx.

<https://docops.ca.com/devtest-solutions/10-4/en/using/using-ca-service-virtualization/ca-service-virtualization-concepts/vse-transactions/logical-transactions>

As you may have read from the docops link, every time when a request hits a VS, the logic transaction set is loaded. Then the following check is performed:

1. The first META transaction is consulted if it matches the request and can respond. If yes, then all the specific transactions that are part of that META tx were checked. If any specific TX affirms, then the specific response is returned. If no specific TX affirms, then the META response would be returned.

This is better explained with an example:

While working on a TCP Virtualization, I had this following META response. Few important point to NOTE:

1. META transactions doesn't support MATCH tolerance as EXACT. The supported ones are SIGNATURE and OPERATION
2. In order to easily identify that the response comes from META, a separate attribute has been added (see pic)

The screenshot shows the Service Virtualization interface with several annotations:

- Transaction Basics:** Match style is set to **Signature**. A red box highlights this, with a callout: "Only allowed match styles are Signature and Operation for META".
- Request Data:** A table lists request attributes. A red box highlights the table, with a callout: "Added an identification to META response to easily identify".
- Response 1 of 1:** Shows the response body with XML content. A red box highlights the `META="YES"` attribute in the XML, with the same callout as above.
- META Transaction:** A red box highlights the **META** transaction type in the left pane.

Name	Name in Session	Comparison Operator	Value	Magic Stb
val1			<nrf:NRFTransaction version="...	<input type="checkbox"/>
nrf_NRFTransaction_nrf_Request_nrf_raveGetIsDutiable_destCountry			US	<input type="checkbox"/>
nrf_NRFTransaction_nrf_Request_nrf_raveGetIsDutiable_destPostalCode			90210	<input type="checkbox"/>
nrf_NRFTransaction_nrf_Request_nrf_raveGetIsDutiable_origCountry			US	<input type="checkbox"/>
nrf_NRFTransaction_nrf_Request_nrf_raveGetIsDutiable_origPostalCode			07652	<input type="checkbox"/>

There is one SPECIFIC transaction with MATCH tolerance as EXACT

The screenshot shows the Service Image Transactions interface. On the left, there are three request snippets for 'raveGetIsDutiable', 'raveGetCurrencyInfo', and another 'raveGetIsDutiable'. The main pane shows 'Transaction Basics' with 'Match style: Exact' and 'Operation: raveGetIsDutiable'. A table under 'Request Data' lists arguments like 'destCountry', 'destPostalCode', and 'origCountry' with comparison operators set to 'Anything'. The 'Response 1 of 1' pane shows an XML response with a 'SPECIFIC="YES"' attribute. A 'META' pane at the bottom shows a request snippet with 'origPostalCode="076521"'. Annotations include: 'EXACT match style used which compares values of arguments from the request', 'Added an identification to SPECIFIC response for easy identification', and a red box around the 'META' pane.

Now, lets look at the Request and matching response as returned by the VS

Request	Response
<pre><nrf:NRFTtransaction version="07.01.0001" RequestMethodName="raveGetIsDutiable" RequestMethodResponseName="raveGetIsDutiableResponse" transactionID="15523212610091" xmlns:nrf="http://ups.com/nrfServerInterface"><nrf:Request><nrf:raveGetIsDutiable origCountry="US" origPostalCode="07652" origCity="" destCountry="US" destPostalCode="90210" destCity=""/></nrf:Request></nrf:NRFTtransaction></pre>	<pre><nrf:NRFTtransaction xmlns:nrf="http://ups.com/nrfServerInterface" RequestMethodName="raveGetIsDutiable" RequestMethodResponseName="raveGetIsDutiableResponse" transactionID="15523212610091" version="07.01.0001"><nrf:Response><nrf:raveGetIsDutiableResponse isDutiable="N" SPECIFIC="YES"></nrf:Response></nrf:NRFTtransaction></pre>
<pre><nrf:NRFTtransaction version="07.01.0001" RequestMethodName="raveGetIsDutiable" RequestMethodResponseName="raveGetIsDutiableResponse" transactionID="15523212610091" xmlns:nrf="http://ups.com/nrfServerInterface"><nrf:Request><nrf:raveGetIsDutiable origCountry="US" origPostalCode="076521" origCity="" destCountry="US" destPostalCode="90210" destCity=""/></nrf:Request></nrf:NRFTtransaction></pre>	<pre><nrf:NRFTtransaction xmlns:nrf="http://ups.com/nrfServerInterface" RequestMethodName="raveGetIsDutiable" RequestMethodResponseName="raveGetIsDutiableResponse" transactionID="15523212610091" version="07.01.0001"><nrf:Response><nrf:raveGetIsDutiableResponse isDutiable="N" META="YES"></nrf:Response></nrf:NRFTtransaction></pre>

```
<nrf:NRFTransaction version="07.01.0001"  
RequestMethodName="raveGetIsDutiable"  
RequestMethodResponseName="raveGetIsDutiableResponse"  
transactionID="15523212610091" xmlns:nrf="http://ups.com/nrfServerIn  
terface"><nrf:Request><nrf:raveGetIsDutiable origCountry="US" origP  
ostalCode1="07652" origCity="" destCountry="US"  
destPostalCode="90210"  
destCity=""/></nrf:Request></nrf:NRFTransaction>
```

NO MATCH FOUND