|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-86](https://mife-axiata.atlassian.net/browse/MIFE-86) | Record SP login history | 2014/10/31 |  |

NOTE:  
This deployment effects only Mobile Connect version

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. OpenIDTokenBuilder (GIT/<dialog>/OpenIDTokenBuilder)  
2. Gsma-authenticators (GIT/<dialog>/gsma-authenticators)

Database Script

gsma\_connect.sql (GIT/<mifehub>/dbscripts)

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build OpenIDTokenBuilder and deploy it to

<IS-HOME>\repository\components\lib

1. Build Gsma-authenticators and deploy it to

<IS-HOME>\repository\components\ dropins

1. Execute the gsma\_connect.sql on CONNECT\_DB

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-206](https://mife-axiata.atlassian.net/browse/MIFE-206) | View SP login history | 2014/11/03 |  |

NOTE:  
This deployment effects only Mobile Connect version

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. MediationTest.war (GIT/<dialog>/IS-artifacts/MediationTest)  
2. Jiggery apps

1. Dashboard -( GIT/<mifehub>/jaggery-apps/dashboard)
2. Portal -( GIT/<mifehub>/jaggery-apps/portal)

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build ‘MediationTest’ and deploy it to

<IS-HOME>\repository\ deployment\server\webapps

1. Deploy ‘dashboard’ and ‘portal’ jiggery apps to

<IS-HOME>\repository\ deployment\server\jaggeryapps

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-179](https://mife-axiata.atlassian.net/browse/MIFE-179) | Display Default Application for each new user on Approval History | 2014/11/07 |  |

NOTE:  
This deployment effects only Mife Hub version

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. org.dialog.custom.hostobjects-1.0.0.jar

(GIT/<mifehub>/ org.dialog.custom.hostobject)  
  
---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build ‘org.dialog.custom.hostobject’ and deploy it to

<AM-HOME>\repository\components\ dropins

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-68](https://mife-axiata.atlassian.net/browse/MIFE-68) | Handle unmapped API requests | 2014/11/07 |  |

NOTE:  
This deployment effects only Mife Hub version

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. AxiataMediator-1.0.0.jar

(GIT/<mifehub>/AxiataMediator)  
  
---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build ‘AxiataMediator’ and deploy it to

<AM-HOME>\repository\components\ dropins

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |
| --- | --- | --- |
| **JIRA-ID** | **Description** | **Note** |
| [MIFE-181](https://mife-axiata.atlassian.net/browse/MIFE-181) | Performance error rates report and modify the southbound data publisher to save the response exception code and description on to BAM |  |

NOTE:  
This deployment effects to the sb\_api\_response\_summary table in the stat\_db.

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. org.dialog.custom.hostobject (GIT/<mifehub>/org.dialog.custom.hostobject)  
2. AxiataMediator (GIT/<mifehub>/AxiataMediator)

3. SouthboundDataPublisher (GIT/<mifehub>/SouthboundDataPublisher)

4. MIFE\_Southbound\_Analytics.tbox (GIT/<mifehub>/BAM-toolboxes)

5. manage (GIT/<mifehub>/jaggery-apps)  
  
---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build org.dialog.custom.hostobject and deploy it to

<AM-HOME>\repository\components\dropins

1. Build AxiataMediator and deploy it to

<AM-HOME>\repository\components\dropins

1. Build SouthboundDataPublisher and deploy it to

<AM-HOME>\repository\components\dropins

1. For the fresh BAM instance.

Deploy MIFE\_Southbound\_Analytics.tbox to

<BAM-HOME>\repository/deployment/server/bam-toolbox

**For the preconfigured BAM instance do the following I to XIII steps.**

1. Log in to BAM management console and select List in BAM Toolbox menu.
2. Uninstall the existing MIFE\_Southbound\_Analytics.tbox .
3. Select Add in Analytics menu.
4. Go to the Script Editor in the window that opens.
5. Execute the following script.

drop table SBAPIRequestData;

drop table SBAPIRequestSummaryData;

drop table SBAPIResponseData;

drop table SBAPIResponseSummaryData;

1. Select Browse in Registry menu.
2. Go to \_system → governance .
3. Select the StreamDefinitions.
4. Delete following components using Actions button.

mife.statistics.southbound.request

mife.statistics.southbound.response

1. Stop the BAM console.
2. Delete the Cassandra database (**This action will delete your previous data**).
3. Drop following tables in stat(mysql) database;

sb\_api\_request\_summary

sb\_api\_response\_summary

1. Restart the BAM console.
2. Deploy MIFE\_Southbound\_Analytics.tbox to

<BAM-HOME>/repository/deployment/server/bam-toolbox

1. Deploy manage module to

<AM-HOME>\repository\deployment\server\jaggeryapps

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-157](https://mife-axiata.atlassian.net/browse/MIFE-157)  [MIFE-200](https://mife-axiata.atlassian.net/browse/MIFE-200) | Operator wise Rate Card  Decouple Rate Card & Tier XML | 2014/11/10 |  |

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. org.dialog.custom.hostobjects-1.0.0.jar

(GIT/<mifehub>/org.dialog.custom.hostobject/1.0.0/)

1. Manage jaggery app

(GIT/<mifehub>/jaggery-apps/manage)

1. rate-card.xml

(GIT/<mifehub>/config/rate-card.xml)

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build ‘org.dialog.custom.hostobjects-1.0.0’ and deploy it to

<AM-HOME>\repository\components\ dropins

1. Copy manage jaggery app to <AM-HOME>\repository\deployment\server\jaggeryapps
2. Add rate-card.xml to the AM registry at <AM\_REGISTRY>/ governance/apimgt/applicationdata/rate-card.xml

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-172](https://mife-axiata.atlassian.net/browse/MIFE-172) | Refactor discovery module for both Mediator and IS | 2014/12/02 |  |

NOTE:  
API manager changes effects only Mife Hub version

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. New Module -"Mobile Network Resolver" - mnc-resolver-1.0.0.jar

-(GIT/<mifehub>/mnc-resolver/)

-(GIT/<mifehub>/config/MobileCountryConfig.xml

2. AxiataMediator-1.0.0.jar

-(GIT/<dialog>/AxiataMediator /)

3. com.gsma.authenticators-1.0.0.jar

-(GIT/<dialog>/gsma-authenticators

Database Script

gsma\_connect.sql (GIT/<mifehub>/dbscripts)

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build ‘AxiataMediator’ and deploy it to

<AM-HOME>\repository\components\ dropins

1. Build ‘mnc-resolver’ and deploy it to

<AM-HOME>\repository\components\ dropins

<IS-HOME>\repository\components\ dropins

1. Drop the MobileCountryConfig.xml to

<AM-HOME>\repository\conf\

<IS-HOME>\repository\conf\

1. Build ‘gsma-authenticators’ and deploy it to

<IS-HOME>\repository\components\ dropins

1. Put MNC number ranges in mcc\_number\_ranges table for MCC’s defined as ‘Stage` 1 in MobileCountryConfig.xml

<Mcc>

<Code>413</Code>

<Stage>1</Stage>

<DefaultMnc>Dialog</DefaultMnc>

</Mcc>

1. IS Management Console, Identity Providers

Registered identity providers in the below format

<country prefix>:<mcc code>:<Mnc Brand>

Eg: 94:413:Dialog

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |
| --- | --- | --- |
| **JIRA-ID** | **Description** | **Note** |
| [MIFE-236](https://mife-axiata.atlassian.net/browse/MIFE-236) | Publish payment category as a separate field in the southbound data publisher module |  |

NOTE:  
This change effects only Mife Hub version

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. SouthboundDataPublisher (GIT/<mifehub>/SouthboundDataPublisher)

2. MIFE\_Southbound\_Analytics.tbox (GIT/<mifehub>/BAM-toolboxes)

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build SouthboundDataPublisher and deploy it to

<AM-HOME>\repository\components\dropins

1. For the fresh BAM instance.

Deploy MIFE\_Southbound\_Analytics.tbox to

<BAM-HOME>\repository/deployment/server/bam-toolbox

**For the preconfigured BAM instance do the following I to XIII steps.**

1. Log in to BAM management console and select List in BAM Toolbox menu.
2. Uninstall the existing MIFE\_Southbound\_Analytics.tbox .
3. Select Add in Analytics menu.
4. Go to the Script Editor in the window that opens.
5. Execute the following script.

drop table SBAPIRequestData;

drop table SBAPIRequestSummaryData;

drop table SBAPIResponseData;

drop table SBAPIResponseSummaryData;

1. Select Browse in Registry menu.
2. Go to \_system → governance .
3. Select the StreamDefinitions.
4. Delete following components using Actions button.

mife.statistics.southbound.request

mife.statistics.southbound.response

1. Stop the BAM console.
2. Delete the Cassandra database (**This action will delete your previous data**).
3. Drop following tables in stat(mysql) database;

sb\_api\_request\_summary

sb\_api\_response\_summary

1. Restart the BAM console.
2. Deploy MIFE\_Southbound\_Analytics.tbox to

<BAM-HOME>/repository/deployment/server/bam-toolbox

1. Deploy manage module to

<AM-HOME>\repository\deployment\server\jaggeryapps

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-67](https://mife-axiata.atlassian.net/browse/MIFE-67) | Category based charging reports | 2014/12/03 |  |

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. org.dialog.custom.hostobjects-1.0.0.jar

(GIT/<mifehub>/org.dialog.custom.hostobject/1.0.0/)

1. rate-card.xml

(GIT/<mifehub>/config/rate-card.xml)

3. AxiataMediator-1.0.0.jar

(GIT/<dialog>/AxiataMediator /)

4. Dbutils-1.0-SNAPSHOT.jar

(GIT/< mifehub>/Dbutils/)

Database Script

axiata\_db.sql (GIT/<mifehub>/dbscripts)

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build ‘org.dialog.custom.hostobjects-1.0.0’ and deploy it to

<AM-HOME>\repository\components\ dropins

1. Copy manage jaggery app to <AM-HOME>\repository\deployment\server\jaggeryapps
2. Add/modify rate-card.xml to the AM registry at <AM\_REGISTRY>/ governance/apimgt/applicationdata/rate-card.xml
3. Build ‘AxiataMediator’,’Dbutils’ and deploy it to

<AM-HOME>\repository\components\ dropins

1. Put valid payment categories to the valid\_payment\_categories table in axiata database

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-](https://mife-axiata.atlassian.net/browse/MIFE-67)213 | BAM Hive script modifications to incorporate incremental analysis | 2014/12/22 | This modification was performed as a tentative solution for BAM server getting stuck issue. |
| [MIFE-](https://mife-axiata.atlassian.net/browse/MIFE-67)238 | BAM Server is getting stuck with errors in back and too large log file. | 2014/12/22 |  |

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
1. MIFE\_Southbound\_Analytics.tbox

(GIT/<mifehub>/BAM-toolboxes/MIFE\_Southbound\_Analytics/)

2. List of patches to be applied on wso2bam-2.4.1;

WSO2-CARBON-PATCH-4.2.0-401

WSO2-CARBON-PATCH-4.2.0-0765

WSO2-CARBON-PATCH-4.2.0-0985

Follow the steps mentioned in README.txt of each patch when applying the patches.

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Compress “MIFE\_Southbound\_Analytics” folder with .tbox file extension.

2. Log in to BAM Management Console.

3. Main → Manage → BAM Toolbox

4. Uninstall the existing “MIFE\_Southbound\_Analytics”.

5. Navigate to Main → Registry → Browse → /\_system/governance/StreamDefinitions

6. Manually delete the below two stream definitions;

- mife.statistics.southbound.request

- mife.statistics.southbound.response

7. Go back to Main → Manage → BAM Toolbox.

8. Click on “Add New Toolbox”.

9. Custom Toolbox → Toolbox From File System → Choose the newly created .tbox file → Install

---------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-249](https://mife-axiata.atlassian.net/browse/MIFE-249) | Key Generation Issue | 2014/12/22 | Method to create same token for same name application and service provider. Previous method was create a SP in IS and generate keys and then create an application in the store with the that SP's name. |

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. In api-manager.xml set <ReuseApp> to false. (This tag was introduced with the patch0530. If that patch is not applied, no need to set this element)

2. Create the Application from API Store. Copy the name of the Application and the Consumer Key

3. Login to IS and create service provider with the same Application name.

4. Now type the following URL on the browser while on IS - replace the values of APPLICATION\_NAME and CONSUMER\_KEY

https://<IS\_ADDRESS>:<IS\_PORT>/carbon/application/configure-service-provider.jsp?action=update&display=oauthapp&spName=<APPLICATION\_NAME>&oauthapp=<CONSUMER\_KEY>

--------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE-83](https://mife-axiata.atlassian.net/browse/MIFE-83) | Spend limit Handler | 2015/01/29 | Ability to put a real time stop on an API usage when a usage of gross revenue target exceeded using CEP |

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
1. mife-events-extensions in [git\_mifehub]/MifeEvents/mife-events-extensions

2. mife-events-data in [git\_mifehub]/MifeEvents/mife-events-data

3. AxiataMediator in [git\_mifehub]/AxiataMediator/  
---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

**CEP configurations**

1. Copy [git\_mifehub]/MifeEvents/config/stream-definitions.xml file to <CEP\_HOME>/repository/conf/data-bridge/stream-definitions.xml
2. Copy the contents of the *eventbuilders* and *executionplans* folders in [git\_mifehub]/MifeEvents/config/ to the corresponding folders in <CEP\_HOME>/repository/deployment/server

The cron expressions for reset time and the spend limit amounts can be configured in the corresponding execution plan.

1. Build *mife-events-extensions* component in [git\_mifehub]/MifeEvents/ folder and add the jar to <CEP\_HOME>/repository/components/dropins/
2. Add the following two lines to the <CEP\_HOME>/repository/conf/siddhi/siddhi.extension file.

com.axiata.dialog.mife.events.extensions.window.CronTimeWindowProcessor

com.axiata.dialog.mife.events.extensions.aggregator.OptimizedSumOutputAttributeAggregatorFactory

1. Add MySQL JDBC Connector jar to <CEP\_HOME>/repository/components/lib folder
2. Add Quartz library (quartz-2.2.1.jar) to <CEP\_HOME>/repository/components/lib folder
3. Add AXIATA\_MIFE\_DB datasource in the <CEP\_HOME>/repository/conf/datasources/master-datasources.xml file similar to following

<datasource>

<name>AXIATA\_MIFE\_DB</name>

<description>The custom datasource used for MIFE</description>

<jndiConfig>

<name>jdbc/AXIATA\_MIFE\_DB</name>

</jndiConfig>

<definition type="RDBMS">

<configuration>

<url>jdbc:mysql://localhost:3306/axiatadb</url>

<username>root</username>

<password>root</password>

<driverClassName>com.mysql.jdbc.Driver</driverClassName>

<maxActive>50</maxActive>

<maxWait>60000</maxWait>

<testOnBorrow>true</testOnBorrow>

<validationQuery>SELECT 1</validationQuery>

<validationInterval>30000</validationInterval>

</configuration>

</definition>

</datasource>

**Execution Plan configurations**

Three execution plans are available for the 3 different spend limit levels. MSISDNSpendLimitAnalyzer, ApplicationSpendLimitAnalyzer and OperatorSpendLimitAnalyzer.

1. Configure the cron expression used to decide the spend limit reset time. Default is *'0 0 0 \* \* ?'* which is 12 midnight.

from SpendLimitSummingStream#window.mife:cron('**0 0 0 \* \* ?**','msisdn')

1. Configure the spend limit amount, above which the APIs will be throttled.

from totalStream[reset!=true and totalAmount >= **9000**]

**API Manager Configurations**

1. Copy [git\_mifehub]/MifeEvents/config/mife-events.properties file to <AM\_HOME>/repository/conf folder and set the CEP URL, username and password in the properties file
2. Build *mife-events-data* component in [git\_mifehub]/MifeEvents/ folder and add the jar to <AM\_HOME>/repository/components/dropins/
3. Build *AxiataMediator* component in [git\_mifehub]/AxiataMediator/ folder and add the jar to <AM\_HOME>/repository/components/dropins/

--------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE3](https://mife-axiata.atlassian.net/browse/MIFE-249)07 | Dynamic LOA level changing | 2015/2/12 | Same LOA level is used for authentication when the different LOA levels are sent from the same browser |
| [MIFE3](https://mife-axiata.atlassian.net/browse/MIFE-249)05 | Authentication flow error | 2015/2/12 | Authentication flow is incorrect when timeout happens in the SMS and USSD pin authenticators |

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
  
---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Change the configuration in Identity.xml file in in <IS\_HOME>/repository/conf to the following

<SessionContextCache>   
 <Enable>false</Enable>   
 <Capacity>100000</Capacity>   
</SessionContextCache>

(set the <Enable> value to 'false')

Note: These configuration is introduced to the IS with the patch0777. So change this after applying the patch first

--------------------------------------------------------------------------------  
Test  
---------------------------------------------------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| **JIRA-ID** | **Description** | **Date** | **Note** |
| [MIFE428](https://mife-axiata.atlassian.net/browse/MIFE-428) | Refactor Mediator to support dynamic API handler loading | 2015/3/13 | Remove API specific implementation from mediator and support dynamic loading of API handlers |

---------------------------------------------------------------------------------  
 Modules  
---------------------------------------------------------------------------------  
1. AxiataMediator in [git\_mifehub]/AxiataMediator/

---------------------------------------------------------------------------------  
Steps  
---------------------------------------------------------------------------------

1. Build *AxiataMediator* component in [git\_mifehub]/AxiataMediator/ folder and add the jar to <AM\_HOME>/repository/components/dropins/
2. Edit the synapse configs of the APIs found in <AM\_HOME>/repository/deployment/server/synapse-configs/default/api/ folder. Edit the class mediator specification as follows for each API.

**SMS**

<class name="com.axiata.dialog.mife.mediator.AxiataHandlerMediator">\_x000D\_ <property name="executorClass" value="com.axiata.dialog.mife.mediator.impl.sms.SMSExecutor"/></class>

**Payment**

<class name="com.axiata.dialog.mife.mediator.AxiataHandlerMediator">\_x000D\_ <property name="executorClass" value="com.axiata.dialog.mife.mediator.impl.payment.PaymentExecutor"/></class>

**Location**

<class name="com.axiata.dialog.mife.mediator.AxiataHandlerMediator">\_x000D\_ <property name="executorClass" value="com.axiata.dialog.mife.mediator.impl.LocationExecutor"/></class>

1. For new APIs, executorClass should be written by extending *com.axiata.dialog.mife.mediator.AxiataRequestExecutor* class. And the executor class should be specified in the synapse config of the API as above.