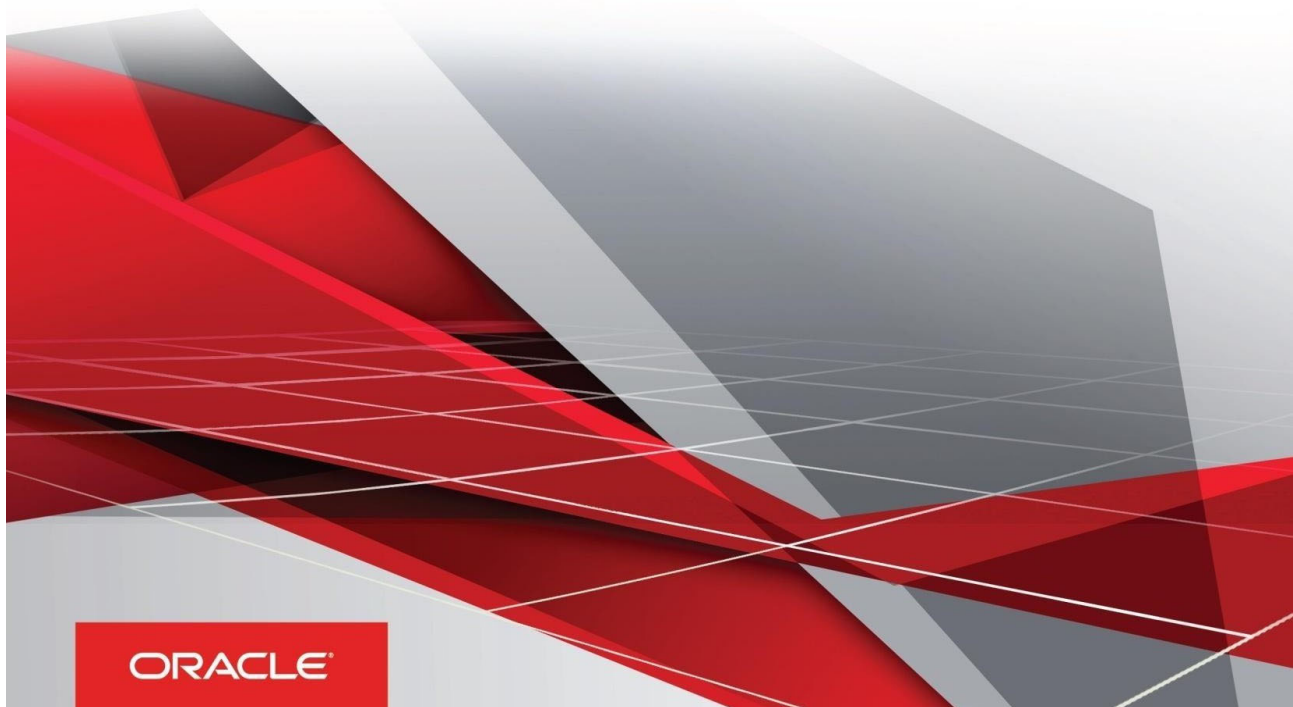# ORACLE®

Financial Services

Enterprise Case Management Application Pack

API Services Guide

July 2020

Document Versioning

Version	Date	Change Reference
01	July 2020	Created: Draft published

Table of Contents

Preface

This Preface provides supporting information for the Oracle Financial Services Enterprise Case Management Application Pack API Services and includes the following topics:

- Purpose of this Document
- Intended Audience
- Documentation Accessibility
- Related Documents

Purpose of this Document

This document contains the API services information for Oracle Financial Services Enterprise Case Management Application Pack (OFS ECM). The objective of this API Services Guide is to provide information about the REST APIs for ECM. The REST APIs described in this guide allow users to perform create events, create a case for an event, and extend the existing case with new event.

This guide assumes that the audience has technical and functional expertise in using and working with REST APIs. This document does not teach REST concepts.

Intended Audience

This document is intended for users of OFS ECM Application Pack. User should be able to utilize these API service to populate events generated from in-house application or any other third party monitoring systems.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

This section identifies additional documents related to OFS ECM You can access from the Documentation Library (OHC).

- *Oracle Financial Services Enterprise Case Management Installation Guide*
- *Oracle Financial Services Enterprise Case Management User Guide*
- *Oracle Financial Services Enterprise Case Management Admin Guide*

Overview

OFSS API is an event as well as case creation service which helps you to populate the events which are identified in your source systems into ECM layer. In transaction monitoring systems, events are generated using various scenarios through batches. Then, the generated events are processed further and cases are created. This whole process takes time. There may be some outliers for which a case can be created or extended to an existing case. This service helps users avoid the lengthier process of transaction monitoring systems.

Also, these API services help Analysts to create the ad-hoc events if they come across any negative news on a customer or some other external incidents for an interested party for further investigation.

Note: These services are available from ECM 8.0.6.0.1 release.

Process Flow

Select and identify the service

↓

User Authentication

↓

Validate JSON with configuration Tables

↓

Perform respective changes in JSON or Configuration Table (if any)

↓

Call the Service
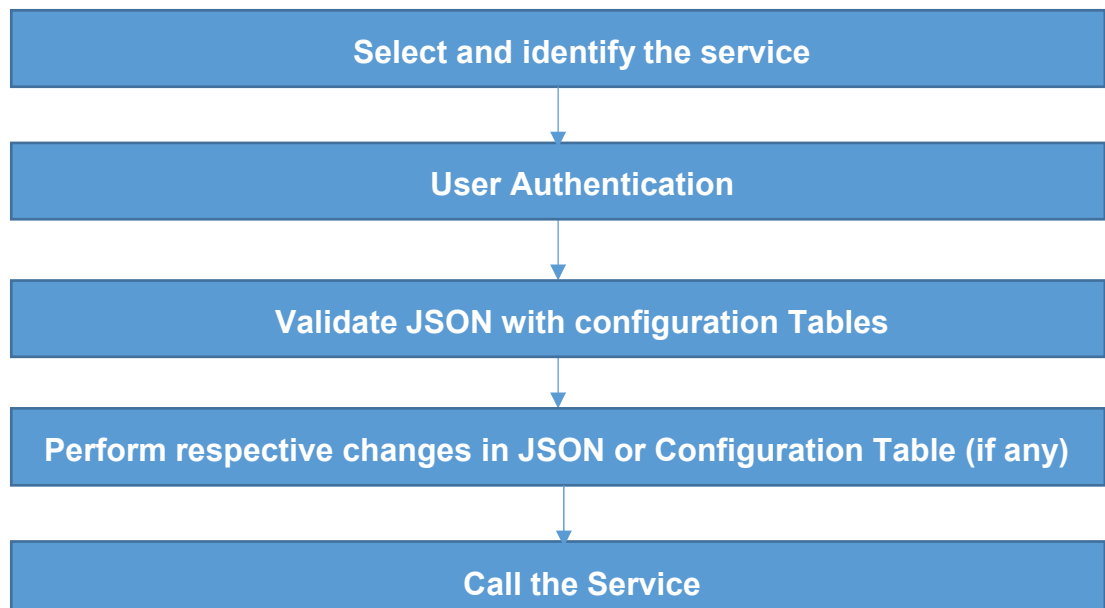
Table 1 provides additional information and links to specific documentation for each task in the flowchart.

Table 1

Task	Details and Documentation
Select and identify the service	Select and Identify the service which serves your purpose from the list of services. For more information, see *Types of Services*.
User Authentication	Identify the ECM user, to use the service. User credentials must be set in the BASIC AUTH HEADER of the service request.
Validate JSON with configuration Tables	Compare the sample JSON with your required configuration. If it matches, then same can be used else needs to be modified.
Perform respective changes in JSON or Configuration Table (if any)	Identify the necessary columns and add missing information to the configuration table as required. The columns added must be mapped back to the JSON. For more information, see *Services*.
Calling the Service	The Service is called using any REST Client.

Types of Services

The following services are supported:

- **Create Event**: This service saves the event generated in the source system to the consolidation layer. The saved event is available for correlation when the next batch is triggered.
- **Create Event and Create New Case:** This service creates a new case for the event generated from the source system. For example, if an event is created for a customer in the Customer Screening application, on triggering this service, a case is created for the event on the ECM layer.
- **Create Event and Extend to Existing Case:** This service links a generated event to an existing case mentioned in the service. For example, an event generated in AML which is related to another event for which a case is already created in ECM. On triggering this service, the new event is added to the existing case.

Prerequisites

The following are prerequisites - to use the services:

1. ECM setup should be installed.

2. User should have appropriate privilege to access the services.

3. Technical and functional knowledge to understand and execute the REST APIs and configuration knowledge.

4. Knowledge of REST concepts, JSON, browser based REST client.

5. A prior knowledge of REST programming is required to understand the examples, samples, scenarios, and reference sections.

6. Knowledge of ECM Data model

Configuring Tables

You should configure specific tables to populate and save events on the Consolidation and Case layers. By configuring these tables, you will get the flexibility to select which data (such as evented entity and entity details) should be saved. The OFSS clients can configure these tables before utilizing the API service. The following tables should be configured:

- FCC_RT_EVENT_ENTITY_TABLES
- FCC_RT_EVENT_ENTITY_TABLES_COL
- FCC_RT_EVENTTYPE_PTC
- FCC_RT_EVENTTYPE_PTC_QUERIES
- FCC_RT_EVNTTYP_CORRELATION_MAP
- FCC_CORRELATION_CASE_TYPE_MAP

The following sections describe these tables.

FCC_RT_EVENT_ENTITY_TABLES

This table stores entities that the service accepts. For example, if this table has entry for only entities such as Account and Customer, then API service will accept only events generated which have a focus of Account and Customer.

Column Name	Description
N_ENTITY_TYPE_ID	Entity Type unique ID.
V_ENTITY_TYPE	Entity type name. It should be the same as which is sent in the JSON. For example, Customer, Account, and so on.
V_EVENTED_TABLE_NAME	Name of the entity evented table where evented data is stored.
V_ENTITY_TABLE_NAME	Name of the corresponding CA business table name. For example, FCC_CUST.
V_ENTITY_LOOK_UP_TABLE_NAME	Name of the entity lookup table.
IS_LOOK_UP_PER_REQD	Determines if the entity details are required to copy to the entity lookup table. You can set this to Y or N. If flag is Y, then it will update the corresponding entity lookup tables. If you are adding a new entity and this flag is Y, then it will check the newly added entity in the lookup table and will not update it in the table if it is duplicate.
IS_ENTI_PER_REQD	Determines if the entity details are required to copy to the entity table. You can set this to Y or N. If this flag is N, then entity information at the time of event generation will not be updated to the corresponding entity tables.

IS_EVENT_PER_REQD	Determines if the entity details are required to copy to the event table. That is, you can choose in which table data is to be populated. You can set this to Y or N. If this flag is N, then entity information at time of event generation will not be updated to the corresponding event tables
V_LOOKUP_SEQUENCE_NAME	Sequence name to skey generation of the lookup. This is required if lookup persistence is required. You can find the sequence from the Sequences (CM_CUST_SEQ) option on the left panel of the database.

FCC_RT_EVENT_ENTITY_TABLES_COL

This table maps the lookup, evented, and business entity information required to be captured for the respective entities, that is, the columns of the lookup, evented, and business entities which are required on the JSON. The mapping should match the data accepted in the relevant entity tables.

Column Name	Description
N_ENTITY_TYPE_ID	Sequence ID of the entity. It should be same as defined in the FCC_RT_EVENT_ENTITY_TABLES table.
V_TABLE_NAME	Table name for the entity which is given in the FCC_RT_EVENT_ENTITY_TABLES table.
V_COL_BUSINESS_NAME	Business name of the entity attribute.
V_COL_NAME	Column name of the entity attribute.
V_DATA_TYPE	Data type of the entity attribute.
V_COL_TYPE	Column type of the entity attribute.
V_IS_SURROG_KEY	Defines the surrogate key.

FCC_RT_EVENTTYPE_PTC

This table maps the case type and corresponding T2T query IDs which should be triggered on case creation or case extension. These queries are triggered only if the service triggered needed to create or extend a case.

Column Name	Description
V_CASE_TYPE	The type of the case. For example, AML_DD

N_SEQUENCE	Order in which the queries should run for the corresponding case type.
V_T2T_CODE	Code name of the corresponding T2T query.

FCC_RT_EVENTTYPE_PTC_QUERIES

This table stores T2T queries against corresponding T2T codes. Queries are triggered on case service call, but only the queries mapped to the case type for which the event is triggered will be called.

Column Name	Description
V_T2T_CODE	Code name of the corresponding T2T query.
C_T2T_QUERY	T2T query which populates data to the ECM tables.

FCC_RT_EVNTTYP_CORRELATION_MAP

This table maps event type to the correlation rule, so the case generated through the service will be of case type (AML_SURV, CS_SAN, and so on) to which correlation rule is mapped. For example, Correlation rule 1 is mapped to case type AML_SURV. If the ingesting AML event has to generate a case type of AML_SURV, entry has to be passed in this table mapping event type and correlation rule.

Column Name	Description
N_EVENT Type	The type of the event used for correlation.
N_CORRELATION_RULE_SKEY	This is the correlation rule unique Identification number. The value of N_CORRELATION_RULE_SKEY column (rule number) should be same as defined in FCC_CORRELATION_RULE table.

FCC_CORRELATION_CASE_TYPE_MAP

⧫ This table maps correlation rule with case type, so cases created through this correlation rule will be all of the mentioned case type (For more information on how to design the case type, see the *Oracle Financial Services Enterprise Case Management Admin Guide*).

Column Name	Description
V_CASE_TYPE	This is the type of case.

`N_CORRELATION_RULE_SKEY`	This is the correlation rule unique Identification number. The value of N_CORRELATION_RULE_SKEY column (rule number) should be same as defined in FCC_CORRELATION_RULE table.

Use Cases for Configuring Tables

You can use below scenarios to understand the services. Before modifying the JSON, you need to configure the tables with appropriate values.

Adding an Entity and **Adding Persistence to an Entity** use cases explain how to add a new entity in FCC_RT_EVENT_ENTITY_TABLES.

Adding an Entity, Adding Persistence to an Entity, and **Adding PTC for Entity** use cases explain how to use Create Event service.

Adding an Entity, Adding Persistence to an Entity, Adding PTC Event, and **Adding Event Type** use cases explain how to Correlation Map use case explains Create Event and Attach to Existing Case.

Adding Event Type to Correlation Map use case explains how to map event type to correlation rule.

This section includes the following topics:

- Adding an Entity
- Adding Persistence to an Entity
- Adding PTC for Entity
- Adding Event Type to Correlation Map

Adding an Entity

To add an entity, follow these steps:

1. Add a new entry in the FCC_RT_EVENT_ENTITY_TABLES table, as shown in the following example:

N_ENTITY_TYPE_ID	V_ENTITY_TYPE	V_EVENTED_TABLE_NAME	V_ENTITY_TABLE_NAME	V_ENTITY_LOOK_UP_TABLE_NAME	IS_LOOK_UP_PER_REQD	IS_ENTI_PER_REQD	IS_EVENT_PER_REQD	V_LOOKUP_SEQUENCE_NAME
102	ECM_CUSTOMER	FCC_CUST_EVNT	FCC_CUST	FCC_CUSTOMER_LOOKUP	Y	Y	Y	

> **NOTE:** Oracle recommends setting the flag to Y for `IS_LOOK_UP_PER_REQD` and `IS_EVENT_PER_REQD` columns. `V_ENTITY_TABLE_NAME table` names should be the same as mentioned in the backend ECM data model.

2. Add the evented details in the FCC_RT_EVENT_ENTITY_TABLES_COL table for the newly added entity, as shown in the following example:

N_ENTITY_TYPE_ID	V_TABLE_NAME	V_COL_BUSINESS_NAME	V_COL_NAME	V_DATA_TYPE	V_COL_TYPE	V_IS_SURROG_KEY
102	FCC_CUST_EVNT	custId	CUST_INTRL_ID	CHAR		

Here, the entry in the V_TABLE_NAME column should be same as either the V_EVENTED_TABLE_NAME, V_ENTITY_TABLE_NAME, or V_ENTITY_LOOK_UP_TABLE_NAME column of the FCC_RT_EVENT_ENTITY_TABLES table.

Adding Persistence to an Entity

For the FCC_RT_EVENT_ENTITY_TABLES table, the following scenarios are possible:

- If columns `IS_LOOK_UP_PER_REQD`, `IS_ENTI_PER_REQD`, and `IS_EVENT_PER_REQD` are set to Y, then the corresponding Entity and Event tables will be populated after lookup.

N_ENTITY_TYPE_ID	V_ENTITY_TYPE	V_EVENTED_TABLE_NAME	V_ENTITY_TABLE_NAME	V_ENTITY_LOOK_UP_TABLE_NAME	IS_LOOK_UP_PER_REQD	IS_ENTI_PER_REQD	IS_EVENT_PER_REQD	V_LOOKUP_SEQUENCE_NAME
102	ECM_CUSTOMER	FCC_CUST_EVNT	FCC_CUST	FCC_CUSTOMER_LOOKUP	Y	Y	Y	

- If columns `IS_LOOK_UP_PER_REQD`, and `IS_EVENT_PER_REQD` are set to Y, then the corresponding Event tables will be populated after lookup. If `IS_ENTI_PER_REQD column` is set to N, then the UI may not display correct data.

N_ENTITY_TYPE_ID	V_ENTITY_TYPE	V_EVENTED_TABLE_NAME	V_ENTITY_TABLE_NAME	V_ENTITY_LOOK_UP_TABLE_NAME	IS_LOOK_UP_PER_REQD	IS_ENTI_PER_REQD	IS_EVENT_PER_REQD	V_LOOKUP_SEQUENCE_NAME
102	ECM_CUSTOMER	FCC_CUST_EVNT	FCC_CUST	FCC_CUSTOMER_LOOKUP	Y	N	Y	

Adding PTC for Entity

To add the PTC for an entity, follow these steps:

Add a new entry in the FCC_RT_EVENTTYPE_PTC table. Before adding the entries in FCC_RT_EVENTTYPE_PTC table, populate the FCC_RT_EVENT_ENTITY_TABLES and FCC_RT_EVENT_ENTITY_TABLES_COL tables.

When a case type is added, the corresponding KDD_CASE tables will be populated.

The following example shows how to add new entries:

V_CASE_TYPE	N_SEQUENCE	V_T2T_CODE
AML_DD	1	t2t_KDD_CASE_ACCOUNTS
AML_DD	2	t2t_KDD_CASE_CUSTOMERS
AML_DD	3	t2t_KDD_CASE_INSTN_MASTER
AML_DD	4	t2t_KDD_CASE_CLIENT_BANK
AML_DD	5	t2t_KDD_CASE_EXTERNAL_ENTITY

Adding Event Type to Correlation Map

To add an event type to correlation, follow this step:

Add a new entry in the FCC_RT_EVNTTYP_CORRELATION_MAP table, as shown in the following example:

N_CORRELATION_RULE_SKEY	V_EVENT_ TYPE
12	AML_SURV

Services

The following services are available:

- Create Event
- Create Event and Promote to Case
- Create Event and Extend to Existing Case

Create Event

This service saves the event generated in the source system to the consolidation layer. The saved event will be available for correlation when the next batch is triggered.

The following topics explain how to configure and use the **Create Events** in JSON.

- HTTP Link
- Service Type
- Request Parameters
- Request JSON Sample
- Response Parameters
- Response JSON Sample

HTTP Link

http:// <Application URL>/rest-api/CMRestService/RealTimeCaseCreationService/saveEvents

Service Type

The service type is POST.

Request Parameters

Below table describes the details of parameters:

First level	Second level	Third Level	Details
events			It accepts basic information of the Event.
events	eventCode		This accepts event id information.
events	eventScore		This accepts information about event score.
events	dataOrigin		This accepts information about Data Origin of the event. It stores the source system from where data is extracted.
events	jurisdictionCode		This accepts information about Jurisdiction code of the event. It represents geographic or legal entity monitoring or data restrictions.
events	businessDomain		This accepts information about Business Domain of the event. Business Domain represents the yours

events	type		operational data restrictions (that is, potential Chinese walls between business units).
events	type		This accepts information about 'Type' of the event. For example, AML, KYC, etc
events	priority		This accepts information about 'Priority' of the event. It is the Priority of the Event and the risk it poses to a customer or an institution. LOV: low, medium, high
events	scenarioClass		This accepts information about 'Scenario Class' of the event. For example, ML,FR, etc
events	comments		This accepts the additional comments provided which are related to the generated Event and details of the event or assignment which led to the creation of an external event that is formatted for display.
events	entities		This is second level tag under JSON. It accepts basic information related event and entity mapping.
events	entities	entityCode	This accepts entity code of the entity. It is the Entity identifier on whom the event is generated from the external system. It can be alphanumeric. For Example: Customer identifier, Account identifier on whom the event is generated.
events	entities	entityType	This accepts entity type of the entity. For example: Account, Customer, Employee, External_ Entity, Household
events	entities	entityTypeCode	This accepts entity type code of the entity.
events	entities	entityName	This accepts name of the entity.
events	entities	focusFlag	This accepts the value whether the entity is Primary Focus of the event or not. List of Values of Y or N. Only one entity of an event will have the value of Y rest all will have N
events	entities	actId	This accepts account internal ID of the Account.
events	details		This is second level tag under JSON. It accepts additional details related to event.
events	details	reason	This accepts event highlights.
events	details	scenarioName	This accepts event scenario Name.
events	bindings	bindingName	This accepts the event binding name.
events	bindings	bindingActualValue	This accepts the binding actual value
events	bindings	bindingBaseValue	This accepts the binding base value.
events	bindings	bindingType	This accepts the binding type.

Request JSON Sample

This section contains a request JSON sample for creating events.

> **NOTE:** The entries in below sample are only for reference purpose.

```
{
```

```
"events": [{

        "eventCode": "09875456",

        "eventScore": "",

        "dataOrigin": "DLY",

        "jurisdictionCode": "AMEA",

        "businessDomain": "c",

        "type": "AML_SURV",

        "priority": "",

        "scenarioClass": "AML",

        "comments": "",

        "entities": [{

                "entityCode": "XXXACFTNEXTENAC-123007",

                "entityType": "ACCOUNT",

                "entityTypeCode": "ECM_ACCOUNT",

                "entityName": "BARRY",

                "focusFlag": "Y",

                "actId": "XXXACFTNEXTENAC-123007"

        }],

        "details": [{


            "reason": "INR2000 Suspicious Transaction to
            Terror Outfits",

                "scenarioName": "User Defined Event"

        }],

        "bindings": [{

                "bindingName": "Effectv_Risk_Lvl",

                "bindingActualValue": "50",

                "bindingBaseValue": "5",

                "bindingType": ""

        }]

}],

"entities": [{

        "entityCode": "XXXACFTNEXTENAC-123007",

        "entityType": "ACCOUNT",
```

```
                    "jurisdiction": "AMEA",

                "entityTypeCode": "ECM_ACCOUNT",

                "entityName": "BARRY",

                "actId": "XXXACFTNEXTENAC-123007",

                "dataOrigin": "DLY",

                "accName": "BARRY",

                "types": "SAV",

                "registration": "IND",

                "ownerType": "",

                "status": "A",

                "branch": "BRANCH-001",

                "taxCode": "N",

                "householdGroupId": "",

                "pimeBroker": "N",

                "taxId": "T",

                "firmId": "MANTAS",

                "legalEntity": "BGMC_ACCT_001",

                "statusEffective": "31-12-2014 00:00:00",

                "lastActivity": "30-11-2015 00:00:00",

                "openData": "31-12-2014 00:00:00",

                "closeDate": ""

        }]

}
```

Response Parameters

First level	Details
Message	Shows the message of service execution
Status	Shows the status as Success or Fail

Response JSON Sample

This section contains a response JSON sample for creating events.

> **NOTE:** The entries in below sample are only for reference purpose.

```
{

    "MESSAGE": "Events are saved successfully.",

    "STATUS": "SUCCESS"

}
```

Create Event and Promote to Case

This service creates a new case for the event generated from the source system. For example, if an event is created for a customer in the Customer Screening application, on triggering this service, a case is created for the event on the ECM layer.

> **NOTE:** The defaulted priority for the cases created through the **Real Time Case Creation** service will be **Low**. System will not determine priority of the cases created through the service based on the **Case Priority** configuration.

The following topics explain how to configure and use the **Create Event and Promote to Case service** in JSON:

- HTTP Link
- Service Type
- Request Parameters
- Request JSON Sample
- Response Parameters
- Response JSON Sample

HTTP Link

http:// <Application URL>/rest-api/CMRestService/RealTimeCaseCreationService/saveEventsAndPromoteToCase

Service Type

The service type is POST.

Request Parameters

Below table describes the details of parameters:

First level	Second level	Third Level	Details
events			It accepts basic information of the Event.
events	eventCode		This accepts event id information.
events	eventScore		This accepts information about event score.

events	dataOrigin		This accepts information about Data Origin of the event. It stores the source system from where data is extracted.
events	jurisdictionCode		This accepts information about Jurisdiction code of the event. It represents geographic or legal entity monitoring or data restrictions.
events	businessDomain		This accepts information about Business Domain of the event. Business Domain represents yours operational data restrictions (that is, potential Chinese walls between business units).
events	type		This accepts information about 'Type' of the event. For example, AML, KYC, etc
events	priority		This accepts information about 'Priority' of the event. It is the Priority of the Event and the risk it poses to a customer or an institution. LOV: low, medium, high
events	scenarioClass		This accepts information about 'Scenario Class' of the event. For example, ML,FR, etc
events	comments		This accepts the additional comments provided which are related to the generated Event and details of the event or assignment which led to the creation of an external event that is formatted for display.
events	entities		This is second level tag under JSON. It accepts basic information related event and entity mapping.
events	entities	entityCode	This accepts entity code of the entity. It is the Entity identifier on whom the event is generated from the external system. It can be alphanumeric. For Example: Customer identifier, Account identifier on whom the event is generated.
events	entities	entityType	This accepts entity type of the entity. For example: Account, Customer, Employee, External_ Entity, Household
events	entities	entityTypeCode	This accepts entity type code of the entity.
events	entities	entityName	This accepts name of the entity.
events	entities	focusFlag	This accepts the value whether the entity is Primary Focus of the event or not. List of Values of Y or N. Only one entity of an event will have the value of Y rest all will have N
events	entities	actId	This accepts account internal ID of the Account.
events	details		This is second level tag under JSON. It accepts additional details related to event.
events	details	reason	This accepts event highlights.
events	details	scenarioName	This accepts event scenario Name.
events	bindings	bindingName	This accepts the event binding name.
events	bindings	bindingActualValue	This accepts the binding actual value
events	bindings	bindingBaseValue	This accepts the binding base value.
events	bindings	bindingType	This accepts the binding type.

Request JSON Sample

This section contains a Request JSON sample for creating an event and promote to case.

> **NOTE:** The entries in below sample are only for reference purpose.

```
{
                "events": [{
                                "eventCode": "655867843",
                                "eventScore": "",
                                "dataOrigin": "DLY",
                                "jurisdictionCode": "AMEA",
                                "businessDomain": "a",
                                "type": "AML_SURV",
                                "priority": "",
                                "scenarioClass": "AML",
                                "comments": "",
                                "entities": [{
                                                "entityCode": "1412",
                                                "entityType": "CLIENT_BANK",
                                                "entityTypeCode": "ECM_CLIENT_BANK",
                                                "entityName": "JADEJ",
                                                "focusFlag": "Y",
                                                "instId": "HSINDI-001",
                                                "instTypeCd": "BIC"
                                }],
                                "details": [{
                                                "reason": "INR7890 TEST - CORRESPONDENT
BANK",
                                                "scenarioName": "User Defined Event"
                                }]
                }],
                "entities": [{
                                "entityCode": "1412",
                                "entityType": "CLIENT_BANK",
                                "entityTypeCode": "ECM_CLIENT_BANK",
                                "entityName": "JADEJ",
```

```
                "instId": "HSINDI-001",

                "instTypeCd": "BIC",

                "dataOrigin": "DLY",

                "city": "",

                "country": "",

                "postalCode": "",

                "state": "",

                "address": "",

                "instituteCountry": "",

                "instituteName": "JADEJ",

                "largeInstitution": "N",

                "selfInstitution": "N"

        }, {

                "skey": "1412",

                "entityType": "CLIENT_BANK",

                "entityTypeCode": "ECM_CLIENT_BANK_SEC",

                "dataOriginCb": "DLY",

                "cbEfctvRiskNb": "0",

                "cbListSrcCd": "",

                "cbMatchTx": "",

                "busUnitCd": "",

                "acctMgrNm": "",

                "lastActDate": "31-12-2014 00:00:00",

                "cbListRiskNb": "0",

                "cbMatchTypecd": "",

                "instSeqId": "102",

                "jurisdiction": "AMEA",

                "busList": "a"

        }]

}
```

First level	Details

CaseId	Displays the case ID
Message	Displays the message of service execution
Status	Displays the status as Success or Fail
CaseType	Displays the type of case

Response JSON Sample

This section contains a Response JSON sample for creating an event and promote to case.

> **NOTE:** The entries in below sample are only for reference purpose.

```
{

    "caseId": "CA354",

    "message": "Events are saved successfully and promoted to case ",

    "status": "SUCCESS",

    "caseType": "AML_SURV"

}
```

Create Event and Extend to Existing Case

This service links a generated event to an existing case mentioned in the service. For example, an event generated in AML which is related to another event for which a case is already created in ECM. On triggering this service, the new event will be added to the existing case.

Below topics explain how to configure and use the **Create Event and attach it to an existing Case service** in JSON:

- HTTP Link
- Service Request
- Request Parameters
- Request JSON Sample
- Response Parameters
- Response JSON Sample

HTTP Link

http:// <Application URL>rest-api/CMRestService/RealTimeCaseCreationService/saveEventsAndExtendToCase

Service Type

The service type is POST.

Request Parameters

Below table describes the details of parameters:

First level	Second level	Third Level	Details
events			It accepts basic information of the Event.
events	eventCode		This accepts event id information.
events	eventScore		This accepts information about event score.
events	dataOrigin		This accepts information about Data Origin of the event. It stores the source system from where data is extracted.
events	jurisdictionCode		This accepts information about Jurisdiction code of the event. It represents geographic or legal entity monitoring or data restrictions.
events	businessDomain		This accepts information about Business Domain of the event. Business Domain represents the yours operational data restrictions (that is, potential Chinese walls between business units).
events	type		This accepts information about 'Type' of the event. For example, AML, KYC, etc
events	priority		This accepts information about 'Priority' of the event. It is the Priority of the Event and the risk it poses to a customer or an institution. LOV: low, medium, high
events	scenarioClass		This accepts information about 'Scenario Class' of the event. For example, ML,FR, etc
events	comments		This accepts the additional comments provided which are related to the generated Event and details of the event or assignment which led to the creation of an external event that is formatted for display.
events	entities		This is second level tag under JSON. It accepts basic information related event and entity mapping.
events	entities	entityCode	This accepts entity code of the entity. It is the Entity identifier on whom the event is generated from the external system. It can be alphanumeric. For Example: Customer identifier, Account identifier on whom the event is generated.
events	entities	entityType	This accepts entity type of the entity. For example: Account, Customer, Employee, External_ Entity, Household
events	entities	entityTypeCode	This accepts entity type code of the entity.
events	entities	entityName	This accepts name of the entity.
events	entities	focusFlag	This accepts the value whether the entity is Primary Focus of the event or not. List of Values of Y or N. Only one entity of an event will have the value of Y rest all will have N
events	entities	actId	This accepts account internal ID of the Account.
events	details		This is second level tag under JSON. It accepts additional details related to event.

events	details	reason	This accepts event highlights.
events	details	scenarioName	This accepts event scenario Name.
events	bindings	bindingName	This accepts the event binding name.
events	bindings	bindingActualValue	This accepts the binding actual value
events	bindings	bindingBaseValue	This accepts the binding base value.
events	bindings	bindingType	This accepts the binding type.

Request JSON Sample

This section contains a request JSON sample for creating an event and attaching to case:

> **NOTE:** The entries in below sample are only for reference purpose.

```json
{

    "events": [{

        "eventCode": "999979",

        "eventScore": "",

        "dataOrigin": "DLY",

        "jurisdictionCode": "AMEA",

        "businessDomain": "c",

        "type": "AML",

        "priority": "",

        "scenarioClass": "AML",

        "comments": "",

    "entities": [{

        "entityCode": "ACDPGTOTACTAC-
        010",

        "entityType":
        "ACCOUNT",

        "entityTypeCode":
        "ECM_ACCOUNT",

        "entityName":
        "ELEVENTH",

        "focusFlag": "Y",

        "actId": "ACDPGTOTACTAC-
        010"
```

```
        }],
        "details": [{
                "reason": "INR2000 Suspicious Transaction to Terror
                Outfits",
                "scenarioName": "User DefinedEvent"
        }],
        "bindings": [{
                "bindingName":
                "Effectv_Risk_Lvl",
                "bindingActualValue":
                "50",
                "bindingBaseValue":
                "5",
                "bindingType": ""
        }]
        }],
        "entities": [{
                "entityCode": "ACDPGTOTACTAC-010",
                "entityType": "ACCOUNT",
                "entityTypeCode": "ECM_ACCOUNT",
                "entityName": "ELEVENTH",
                "actId": "ACDPGTOTACTAC-010",
                "dataOrigin": "DLY",
                "accName": "ELEVENTH",
                "types": "INV",
                "registration": "IND",
                "ownerType": "RPT",
                "jurisdiction": "AMEA",
                "status": "A",
                "branch": "BRANCH-001",
                "taxCode": "N",
                "householdGroupId": "HHSAMPLE00",
                "pimeBroker": "N",
                "taxId": "S",
```

```
       "firmId": "",

       "legalEntity": "BGMC_ACCT_001",

       "statusEffective": "31-12-2014
       00:00:00",

       "lastActivity": "",

       "openData": "31-12-2014 00:00:00",

       "closeDate": ""                                    û

       "events":[
   {

      "eventCode":"153803908863599",

      "eventScore":"",

      "dataOrigin":"DLY",

      "jurisdictionCode":"AMEA",

      "businessDomain":"c",

      "type":"AML",

      "priority":"",

      "scenarioClass":"AML",

      "comments":"",

      "entities":[
         {

            "entityCode":"CUHRTHRCPHH-002",

            "entityType":"CUSTOMER",

            "entityTypeCode":"ECM_CUSTOMER",

            "entityName":"ANDREW ELEVENTH",

             "jurisdiction" :"AMEA",

            "focusFlag":"Y",

            "custId":"CUHRTHRCPHH-002"

         }

      ],

      "details":[
         {

            "reason":"ffjfjgghgkkkkhhhk", "scenarioName":"User

            Defined Event"

         }

      ]
```

```
            }
    ],
    "entities":[
        {
            "entityCode":"CUHRTHRCPHH-002",

            "entityType":"CUSTOMER",

            "entityTypeCode":"ECM_CUSTOMER",

            "entityName":"ANDREW ELEVENTH",

            "jurisdiction" :"AMEA",

            "custId":"CUHRTHRCPHH-002",

            "dataOrigin":"DLY",

            "customerName":"ANDREW ELEVENTH",

            "customerType":"IND",

            "taxID":"TAXHRTHRCPHH-002",

            "legalStructCode":"GOV",

            "aliasName":"",

            "incomeRange":"",

            "industry":"US",

            "publicCompany":"",

            "estNetWorth":"1182552",

            "atmDailyLimit":null,

            "estAnualIncome":"168936",

            "added":"31-12-2014 00:00:00",

            "incorporationDate":"",

            "customerstatus": "A",

            "effectiverisk": "2",

            "busdomain": "a"
        }
    ],
            "caseID": "CA115"
}
```

First level	Details

Message	Displays the message of service execution
Status	Displays the status as Success or Fail
CaseType	Displays the type of case

This section contains a response JSON sample for creating an event and attaching to case:

> **NOTE:** The entries in below sample are only for reference purpose.

```
{

    "message": "Events are saved successfully and promoted to case ",

    "status": "SUCCESS",

    "caseType": "AML_SURV"

}
```

ORACLE®

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

Integrated Cloud Applications & Platform Services

Oracle is committed to developing practices and products that help protect the environment