[XOCL](#h.t2xnu6ed3wl5)

[APIs](#h.ub4xeqljmlne)

[XOCLDOLocation](#h.weoe57gbk3bj)

[XOCLDOOperatorId](#h.7af997rg1ild)

[XOCLDOPayment](#h.b4q4munfoikl)

[XOCLSms](#h.hy8ts0khh110)

[Sequences](#h.skak1sk424up)

[Discovery](#h.hp7dj6nyvh7t)

[Local-Entries](#h.ijwybt6ymewk)

[SACL](#h.qdcalsmw36wo)

[APIs](#h.6mz10zdyrquh)

[SACLLocation](#h.7mjilq7mpnsw)

[SACLOperatorid](#h.9jtewsfb0z00)

[SACLPaymentSACLLocation  
SACLOperatorid  
SACLPayment  
SACLSmsmessaging](#h.ai5um9n11rmc)

[SACLSmsmessaging](#h.71me8wsxyy0v)

[Sequences](#h.p6q8zgfckuoa)

[RequestValidation](#h.m73pdzju3rmq)

[TransactionLogging](#h.h89858odyrm8)

[TransactionLoggingPayment](#h.w7s8htq263dl)

[Local-Entries](#h.qwc78d5lrfrc)

## 

## XOCL

This is an adaptor which has following functionalities.

1. API Conversion: Custom to Standard OneAPI

2. Endpoint Discovery : Done in the “Discovery Sequence”. Since Axiata wants to integrate only this part as they are already OneAPI compliant, this has done as self serving.

### APIs

#### XOCLLocation

#### XOCLOperatorId

#### XOCLPayment

#### XOCLSms

### Sequences

#### Discovery

This sequence will send a request to discovery endpoint and get discovery details and pass back to calling api/proxy-service. If the request fails it will send a fault to the calling sequence.

The following values should be in headers in the request to this sequence,

1. Authorization: The application’s authorization value

2. mccmnc: The mccmnc value of the subscriber operator

The parameters taken from local-entries

1. exchange\_do\_host

### Local-Entries

1. exchange\_do\_host: This the host name for the Exchange endpoint

- Reason for having endpoint and local-entry for the same endpoint is because apigee exchange directly appends to the service path.

2. so\_endpoint: This is configured for testing purposes.

## SACL

This is an adaptor which has following functionalities.

1. API Conversion: Standard OneAPI to Custom

2. Request Validation : Done in the “RequestValidation Sequence”. Since Axiata wants to integrate only this part as they are already OneAPI compliant, this has done as self serving.

3. Transaction Logging: Done in the “TransactionLogging/TransactionLoggingPayment Sequence”. Since Axiata wants to integrate only this part as they are already OneAPI compliant, this has done as self serving.

### APIs

#### SACLLocation

#### SACLOperatorid

#### SACLPayment

#### SACLSmsmessaging

### Sequences

#### RequestValidation

This sequence validates the request from the exchange, and pass the response to the calling artifact.

The following values should be in headers in the request to this sequence:

1. clientid: The application credentials which would have obtained in discovery call by Discovery sequence

2. clientsecret: The application credentials which would have obtained in discovery call by Discovery sequence

3. mccmnc

The parameters taken from local-entries

1. exchange\_so\_host

2. organization

The parameters taken from secure-vault

1. exchanage\_hub\_credentials: The access token given to hub for accessing exchange “RequestValidation” API

2. hub\_token: The access token to access hub

#### TransactionLogging

This sequence will be used by the Location, SMS and OperatorID APIs for logging. This is not handled from GSMA side but it will useful in future for cross-operator scenarios.

The following values should be in headers in the request to this sequence:

Please find a way to provide these details for above APIs.

1. transactionBatchId

2. tx\_id

3. tx\_status

The parameters taken from local-entries

1. exchange\_so\_host

2. organization

The parameters taken from secure-vault

1. exchanage\_hub\_credentials: The access token given to hub for accessing exchange “RequestValidation” API

#### TransactionLoggingPayment

The following values should be in headers in the request to this sequence:

1. transactionBatchId: amountTransaction.clientCorrelator from Payment response

2. tx\_id: amountTransaction.referenceCode from Payment response

3. tx\_status: amountTransaction.transactionOperationStatus from Payment response

The parameters taken from local-entries

1. exchange\_so\_host

2. organization

The parameters taken from secure-vault

1. exchanage\_hub\_credentials: The access token given to hub for accessing exchange “RequestValidation” API

### Local-Entries

1. exchange\_so\_host: The hostname of the SO provisioned in the Exchange

2. hub\_endpoint: The hostname of the hub

3. organization: The company name provisioned in the Exchange for the hub