Transit Agency
Application Programming Interfaces

User Guide

Table of Contents

1 Referenced Documentation

This document, **TA APIs User Guide**, referrers to Appendices presented in Document “UniTiAg User Guide Appendices”. Whenever an Appendix is referenced in this guide, you can find the corresponding details in that document.

Appendix 1 contains a glossary, and a list of acronyms commonly used throughout this User Guide.

2 Introduction to Transit Agency APIs

Please refer to https://unitiag.com/unitiag-architecture/ for UniTiAg Business Architecture diagram and https://unitiag.com/apis/ for the general APIs diagram.

We will explore two edge TA strategies for utilizing this API architecture. TAs can adapt these strategies based on their ticketing system needs and regional conditions.

The first strategy, illustrated in the following diagram, applies to TAs where the TANB API latency is insufficient for Validator throughput. In this case, the TA ticketing system periodically syncs OTRB data to local memory and propagates it to wireless validators. This approach is also suitable for robust ABT systems requiring additional data to calculate concession fares, period passes, and discounts.

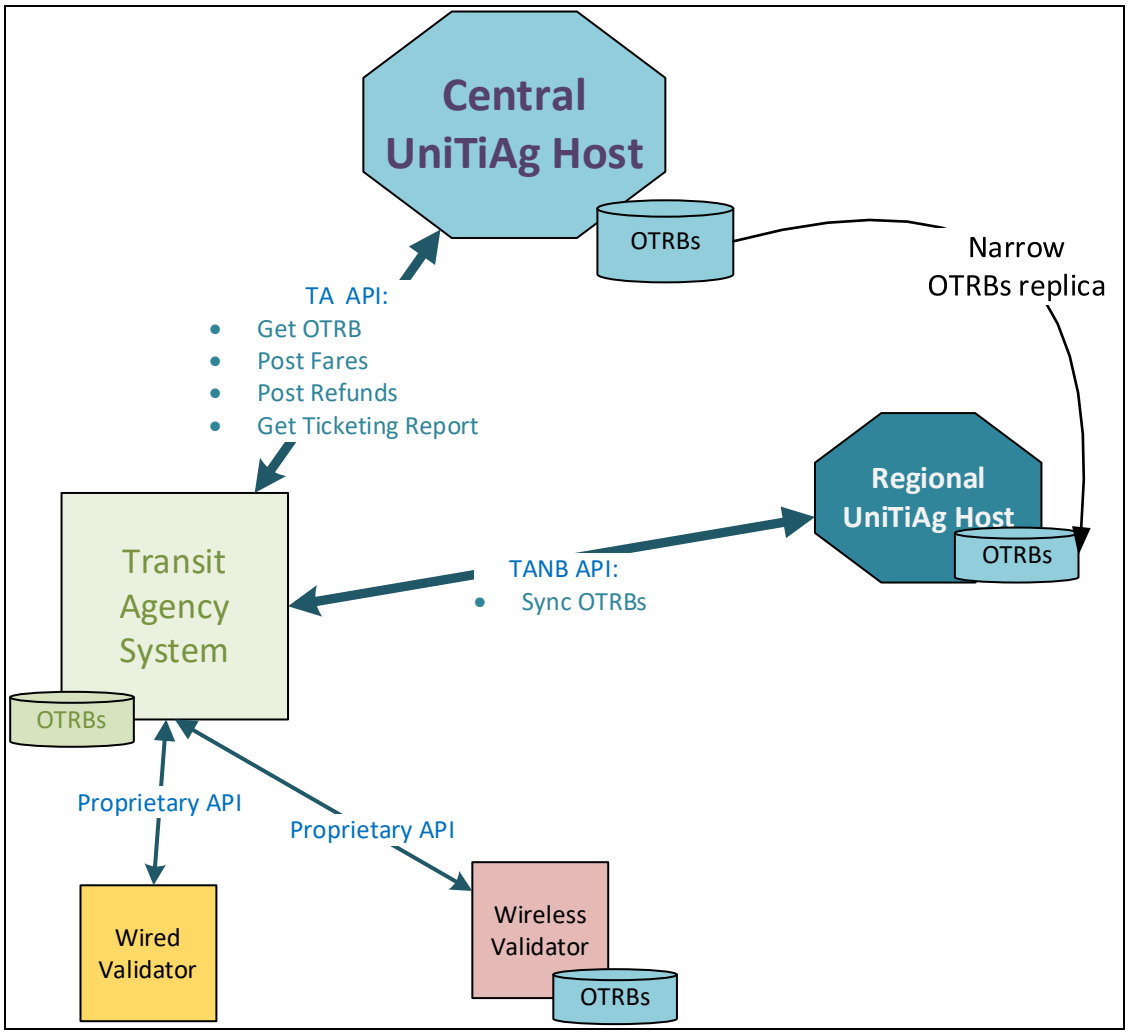

Figure 1. Business architecture of an ABT system with robust functionality, relying minimally on UniTiAg APIs.

The second edge strategy, depicted in the next diagram, involves using the TANB API for TAs that do not handle concessions, discounts, or refunds. All fares are calculated at the Validator level, and the

Regional UniTiAg Host provides low-latency API calls. In this case, the TA relies fully on UniTiAg's targeted OTRB data replication.

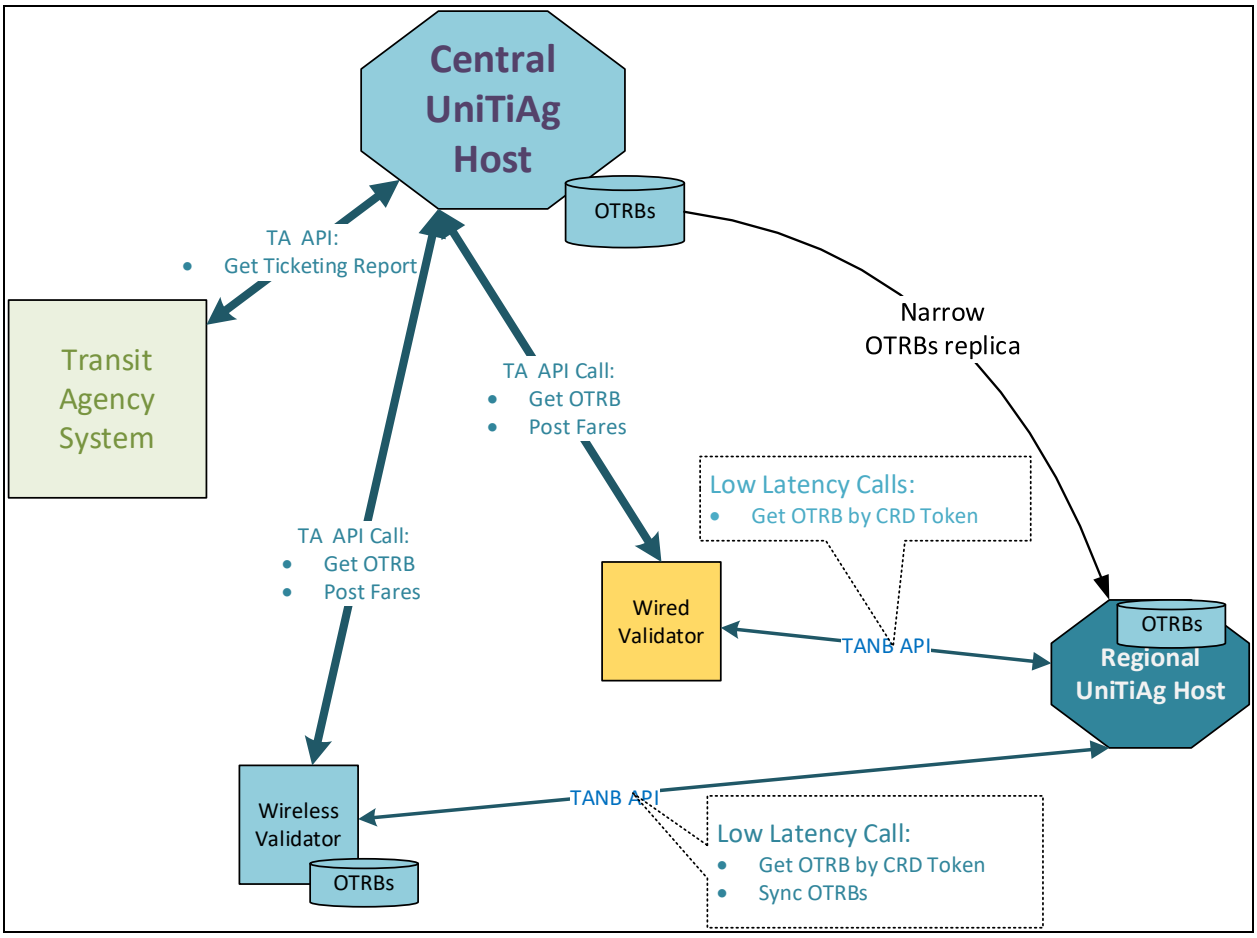

Figure 2. Business architecture of an ABT system leveraging UniTiAg APIs, with core functionality primarily dependent on UniTiAg

In practice, a TA's ABT system strategy will likely combine elements of both approaches.

Appendix 5 provides a flowchart illustrating one possible fare validation strategy using TA APIs.

3 General TA and TANB API Description

3.1 API Requests

The TA and TANB API calls utilize the REST HTTP POST method, with URLs configured separately for each environment. The current implementation is built on AWS's RESTful API Gateway infrastructure.

3.2 Request Headers

In general, the POST headers must be as follows:

Header	Necessity	Meaning and Purpose
Content-Type	m	application/json
X-API-Key	m	Used for API request authentication. Assigned for each environment separately.
taid	m	Transit Agency ID, as assigned by UniTiAg
takey	c	This key must be present unless the TA IP address is whitelisted. The latter might not be possible if the API call is made by a Validator.
validatorid	o	Required for TA and TANB API calls which are made by Validators. This header must not be present in the following API TA API calls, that can be initiated solely by TA, not Validators. • Post Refunds • Deactivate Me • Get Ticketing Report
valkey	o	Required when 'validatorid' header is present. Both headers are used for validator authentication. The value of this header is calculated by TA or Validator as described in Appendix 4.3.

All headers must be UTF-8-encoded strings within the ASCII range (not more than 1 byte per character).

3.3 Request Body

API call request 'body' is a string comprising call-specific JSON content. The body's Json attributes may be presented as Strings and Numbers:

- "String" means a set of UTF-8 symbols encoded in the range of ASCII. UTF16 String means a set of UTF-8 symbols encoded in the range of UTF-16.
- "Number" means an integer or a floating-point number.

3.4 URI Path

The calls are to be HTTPS calls, sent to URI <API URI Base><path>. The <path> component is call-specific, as described in the Sections below.

3.5 API Call Response

Name	Type	Meaning / Condition
HTTP Status Code	Number	• 200: UniTiAg fulfilled the request. • 201: UniTiAg already fulfilled the same request (idempotent case). The response body is the same as it would be with Status Code 200. • 202 – 299: request--specific, as described below. • 300 – 599: the call is rejected; the response body may contain a String with explanations of errors in the request.
HTTP Response Body	String	Request-specific.

4 TA API Details

4.1 API Call List

The following calls are supported by the TA API:

TA API Call Name	Description
Get OTRB	When the OTRB is not yet in the TAs targeted OTRB replica the TA or its Validator makes this call to request a single OTRB details and to further add this OTRB to the targeted replica.
Post Fares	The TA or its Validator reports one ore several fares. The fares are always caused by CRD taps.
Post Refunds	The TA reports one or several refunds to OTRBs, to apply discounts, day caps, etc. The refunds are never caused by CRD taps. They are initiated by the TA, not Validators
Get Ticketing Report	Request a report that TA can further use to match with the TSMPs reconciliation reports. It is initiated by the TA, not Validators

The following sections describe each TA API call.

4.2 Get OTRB

4.2.1 Request path

/get-otrb-bypan

NOTE: the name of the request path mentions “PAN”. However, the PAN is required only for cEMV cards.

4.2.2 Purpose

1. After the CRD is tapped at (or registered in any other contactless way by) the Validator, the TA / Validator calculates the CRD Token from the tap data. If the TA / Validator cannot find the OTRB identified by this CRD Token in its local memory or in its targeted replica at the Regional UniTiAg Hosts, the TA makes this call to retrieve the OTRB from the central UniTiAg Host providing the CRD Token and (only for cEMV cards) the encrypted Card PAN.
2. This call will make this OTRB active at this TA, so the next time the CRD is tapped the OTRB will become available in the targeted OTRB data replica.
3. The call response comprises the OTRB data required for the TA’s validation process.

4.2.3 Request Parameters

The request body parameters are as follows:

Name	Type	Necessity	Restrictions	Meaning
cardNumber	Depends on encryptionType	c	Required only for cEMV cards	An object encapsulating the encrypted PAN or DPAN captured by the Validator. (see Appendix 4.2).
encryptionType	Decimal	c, o	Used only with cEMV cards. If	Identifies algorithm and cardNumber format used in this call.

Name	Type	Necessity	Restrictions	Meaning
			omitted – '0' is assumed	
cardToken	Base64-encoded string	m		CRD Token captured by the Validator. (see Appendix 4.1)

4.2.4 API Response

If the call is not rejected, and the OTRB is retrieved successfully, UniTiAg responds with http statusCode 200, and the following object in response body:

Name	Type	Meaning / Condition
ID	String	OTRB ID
otrbName	String	User-friendly OTRB Name as created by the Rider/TSMP
otrbAmount	Number	OTR balance expressed in major units of TA Currency, without rounding.
otrbStatus	Number	OTRB Status: 'active', 'cancelled', or 'onhold'.
trustRate	Number	Trust Rate set by the TSMP for this OTRB, between 0 and 100.
tsmpId	Number	TSMP ID of TSM that owns this OTRB

An example of such a response is shown below. Note, that body value will be stringified in the response.

```
{
  "statusCode": 200,
  "body": {
    "ID": " a971c204-2dbc-42ff-8720-df93250eb23e"
    "otrbName": "My UK OTRB",
    "otrbAmount": 44.234678,
    "otrbStatus": "active",
    "trustRate": 50
  },
  "headers": {
    "Content-Type": "application/json"
  }
}
```

If the call is not rejected, and there was an issue with retrieval of this OTRB, the response body is empty, and the http statusCode is as follows.

statusCode	Reason
211	This CRD is not associated with any OTRB of any of TSMPs.
212	Found OTRB does not belong to any TSMP which has this TA as a merchant.
213	Only for cEMV cards. A different CRD Token is already associated with the OTRB linked to this Card Number. Most likely, the rider received the card replacement with the same PAN, a DPAN has been changed, and the rider has not reported this change to their TSMP before tapping the card/device.

214	Only for cEMV cards. The cardToken is not unique, that is, it belongs to another cardNumber / OTRB. This may happen if some TA or TSMP do not follow the algorithm of cardToken creation (see Appendix 4).

If the call content is malformed or there was an issue with the call authentication, UniTiAg rejects the call with http statusCode between 300 and 599.

4.2.5 UniTiAg Actions

1. UniTiAg places this TA to the OTRB's list of active TAs. As a result, it becomes available in the targeted replica, at relevant Regional UniTiAg Host.
2. Only for cEMV cards. If this was the first time when UniTiAg obtained the CRD Token of this OTRB, UniTiAg also starts replicating this OTRB data to the Regional UniTiAg Hosts to the targeted replicas of all TAs active for this OTRB.

4.2.6 TA Actions

1. The TA and Validator can immediately, and only once, use otrbAmount, otrbStatus, trustRate for the current trip validation.
2. The TA must not validate rides if otrbStatus is not 'active'. Such ride fares will never be reconciled.
3. The TA should further use faster TANB API call *CRD Token Get OTRB by CRD Token* or retrieve OTRBs from the local OTRB lists obtained by TANB API Call *Sync Otrbs* for validation purposes.
4. For cEMV cards, the TA and Validator should discard Card Number in the Validator and in the TA ABT system databases, to decrease PCI DSS scope. The cardNumber is not required in any further TA API or TANB API calls.
5. The TA ABT system must store the returned parameter ID as a reference to this OTRB for further communications with the TSMP for reconciliation purposes and communications with UniTiAg for posting refunds.
6. The TA ABT System should store the otrbName as a reference for further communications with the Rider.
7. The TA may farther use the CRD Token and/or OTRB ID as OTRB identifiers for the purpose of supporting concession logic. Please note, that only CRD Token, not (OTRB ID), is available to the TA from the Tap Data.
8. The TA may use local stop-lists of cancelled OTRBs, using the CRD Token as a reference to OTRBs, to increase the validation throughput.

4.2.7 Other effects

The TA should expect that this OTRB will be replicated to the TA's Regional UniTiAg Host within several seconds.

4.3 Post Fares

4.3.1 Request path

/post-fares

4.3.2 Purpose

The TA or the Wireless validator use this call to report one or several fares.

4.3.3 Request Parameters

If 'valid' is present in the header, UniTiAg interprets this call as made by a Validator. In such a case, 'valkey' must also be present in the headers, and UniTiAg validates it.

Otherwise, UniTiAg interprets the call as made by the TA ABT System. In such a case fares in this call may be originated by various validators.

The call body comprises 'Items' array of fare items, with the following parameters, for each fare item.

Name	Type	Necessity	Restrictions	Meaning
frId	String	m	<=40	Fare or refund ID, unique per TA, as presented by the TA or Wireless Validator, to support call idempotency. UniTiAg maps it to requestId in table Moves.
cTk	String	m	Base64-encoded	CRD Token as calculated from the Tap Data (see Appendix 4.1)
taAmnt	Number	m	taAmnt absolute value must not exceed maxFare configured for this TA.	In main units of TA currency. Negative if the OTRB is charged. Positive if the OTRB is refunded as a result of a CRD tap, e.g. at tap-off.
opTaAmnt	Number	m		Opening OTRB amount as known to the validator/TA before the charge. Expressed in main units of TA currency.
tapAtu	Number	m	Unix time of CRD tap.	Items with late tapAtu are rejected (See below)
rtId	UTF-16 String	o	<= 20 UTF-16 characters	Route ID
stId	UTF-16 String	o	<= 20 UTF-16 characters	Stop or station ID
vhId	UTF-16 String	o	<= 20 UTF-16 characters	Vehicle ID
why	UTF-16 String	o	<= 40 UTF-16 characters	Fare description
tLat	Number	o	It is desirable that the validator measures and provides this data for further research and latency optimization	Tap latency in msec. This includes time of CRD reading and obtaining Tap Data.
vLat	Number	o	It is desirable that the validator measures and provides this data for further research and latency optimization	Validator latency in msec. This includes time of the CRD reading, time needed to retrieve the OTRB balance, and time the Validator spent on making a validation decision (before posting the Fare).
valId	String	c	<= 40 UTF-8 characters	Validator ID. If the call header validatorid is not specified, valId must be present in each Item in the body. valId is ignored when the call header validatorid is specified.

4.3.4 Validating request

1. The API call has a limited number of items configured for this TA in settings Table *tas* as postItemsLimit. In such a case UniTiAg rejects the entire call before processing any item. The TA or Validator must break the API call into several calls if this limit is exceeded.
2. If OTRB status is not 'active', and taAmnt is negative (fare charge), the UniTiAg rejects the item.
3. UniTiAg rejects items with otrbId pointing to the OTRB which belongs to a TSMP having merchant relations with this TA.

4.3.5 Request body example

The following is an example of body value. Please note that this JSON structure must be stringified.

```
{"Items": [
      {
        "frId": "1286988223",
        "cTk": "OMhH13iS4z/ESrK7coJa",
        "taAmnt": -3.5,
        "opTaAmnt": 30.89,
        "tapAtu": 1722346335011,
        "rtId": "1",
        "stId": "1",
        "vhId": "Bus on Route",
        "why": "Flat Tap Fare",
        "tLat": 2167
      },
. . .
    ]
}
```

4.3.6 API Response

HTTP statusCode:

statusCode	Reason
200	OK. There still can be rejected items.
300 - 499	The call content is severely malformed or there was an authorization issue

For each rejected item, the reason for rejections is listed in the response. Below is an example of response with stratusCode=200. Note, that body value will be stringified in the response.

```
{
  "statusCode": 200,
  "body": {
    "rejects": [
      {
        "rejected-item": 15,
        "errors": "Error: Validation failed: property blaaaaa is not
allowed.; Invalid tapAtu: late request."
      },
      {
        "rejected-item": 23,
        "errors": "Error: Validation failed: Invalid tapAtu: late request."
      }
    ],
    "total-accepted-items": 120
  },
  "headers": {
    "Content-Type": "application/json"
  }
}
```

4.3.7 UniTiAg Actions

1. UniTiAg accepts but does not act on idempotent fare Items, i.e. the ones having the same 'frId' and 'taid' already accepted before, within the idempWindowHours configured for this TA.
   1.1. If the call is requested by a Validator, the Validator should ensure that certain unique ID of the validator is embedded in frId, to preserve frId uniqueness within this TA, across all validators, to preserve idempotency.
2. If, as a result of negative amnt, the otrbAmount goes below the lowAmount, UniTiAg makes TSMP API call low-amount-warning to the TSMP.
3. UniTiAg makes the TA active for the OTRB affected by the accepted fare item and replicates changes in the linked OTRBs to the TA's regional OTRB replica.
4. UniTiAg creates an item in Table *Moves* for each non-rejected call item, and assigns reconStatus in the item as follows:

reconStatus

Value	Meaning	Promise
0	OK	To be reconciled
1	Overdraft. At least some portion of the fare may be not reconciled. This is caused either by the deliberate overdraft based on the Validator/TA decision or overdraft at the time of fare post, or both.	TSMP will try to recover the overdraft from the Rider and reconcile this portion if this is inline with the TSMP-Rider agreement. The overdraft may not necessarily persist, e.g. the TA may return a portion of the fare later (typical case of tap-off).

4.3.8 TA Actions

1. The TA must post fares with tapAtu within the acceptWindowHours configured for this TA or its Validators in table *tas*.
2. The TA or Validator should not post fares with the CRD Token (cTk) which is not yet in the TAs regional OTRB replica, because they do not yet know the actual OTRB's amount and status.
   2.1. It is recommended that the TA or Validator, first, makes a TA API call *GET OTRB*, to acquire the OTRB object associated with this CRD Token.
   2.2. Not following this recommendation may result in either fare item rejection or not recovering the overdraft.

4.4 Post Refunds

4.4.1 Request path

/post-refunds

4.4.2 Purpose

TAs use this call to modify OTRB balance by applying period caps, concession, loyalty, or transfer discounts, fix a technical error, or resolve a dispute with the Rider. This API call is not related to any particular CRD tap. It is originated by the TA, not the Validator.

The TA may present several refund items in one call.

4.4.3 Request Parameters

The call headers must not comprise 'validatorid' and 'valkey'.

The call body comprises 'Items' array of refund items, with the following parameters, for each refund.

Please note, that Post Refunds call identifies OTRB items to refund by OTRB ID, contrary to Post Fares call which identify OTRB items by its CRD Token.

Name	Type	Necessity	Restrictions	Meaning
frId	String	m	<=50 UTF-16 characters	Fare or refund ID, unique per TA, as presented by the TA or Wireless Validator. This is an idempotency key. UniTiAg stores it as requestId in table Moves.
otrbId	String	m	<= 50 characters	The OTRB ID of the OTRB the refund is applied to.
taAmnt	Number	m	> 0	Amount to refund, presented in main units of TA currency.
why	UTF-16 String	m	<= 40 UTF-16 characters	Refund description as presented by TA.

4.4.4 Request body example

The following is an example of body value. Please note that this JSON structure must be stringified.

```
{
    "Items": [
      {
        "frId": "sfghg1",
        "otrbId": " a971c204-2dbc-42ff-8720-df93250eb23e”,
        "taAmnt": 1.25,
        "why": "Day cap refund"
      },
      {
        "frId": "fdsfdg12",
        "otrbId": " b3200052-f453-4a18-995f-575facb7fec5”,
        "taAmnt": 5.3,
        "why": "Week cap refund"
      }
    ]
}
```

4.4.5 Validating the request

1. One API call has a limited number of items configured for this TA in table *tas* as postItemsLimit. In such a case UniTiAg rejects the entire call before processing any item. The TA must break the API call into several calls if this limit is exceeded.
2. UniTiAg ensures that otrbId points to the OTRB which belongs to a TSMP having merchant relations with this TA.

4.4.6 API Response

HTTP statusCode:

statusCode	Reason
200	The call is accepted. Some refunds may be rejected; the error description and item’s number are retuned for such items.

An example of response body value of a response with statusCode = 200 is below. Note, that body value will be stringified in the response.

```
{"rejects":
 [
  {"rejected-item":5,"errors":"Error: OTRB ID not found."}
 ],
 "total-accepted-items":15
}
```


4.4.7 UniTiAg Actions

1. UniTiAg accepts but does not act on idempotent refund Items, i.e. the ones having the same ‘frId’ and ‘taid’ already accepted before, within idempWindowHours configured for this TA.

4.5 Get Ticketing Report

4.5.1 Request path

/get-ticketing-report

4.5.2 Purpose

The TA uses this call to get its ticketing report, so it can further present this report to TSMP for reconciliation or to match it with the TSMP reconciliation reports, according to its TSMP-Merchant agreement.

The report is provided for the period of 24 hours.

4.5.3 Request Parameters

Name	Type	Necessity	Restrictions	Meaning
fromAtu	Number	m	Unix time for the earliest item in the report. Must be not earlier than 50 days ago and not later than 1 day and 10 min ago	Report period start. Report period end is set automatically to 24h after the report period start.

4.5.4 API Response

The response can be interpreted as a file to download. Returned headers:

- Content-Type: 'application/octet-stream',
- Content-Disposition: 'attachment; filename="ticketing_report_<TA ID> .csv”'

Response body:

For non-rejected calls: the body string presents the report file content in CSV format, with column header line. Each row represents one item of table Moves (see Appendix 3). Column ‘taId’ is not present in the report.

The rows in the CSV file are sorted by column ‘atu’ in ascending order.

4.5.5 TA actions

Old items in table Moves are periodically removed and archived by UniTiAg. The TA should make this call within the most recent 50 days.

4.5.6 UniTiAg actions

UniTiAg ensures that taid in the POST header belongs to the TA that makes this call.

5 TANB API Details

Please note, that when there are several TAs in the same region, each TA has its own OTRB data in the Regional UniTiAg Host.

5.1 TANB API Calls

The following calls are supported by UniTiAg's TANB API:

TANB API Call Name	Description
Get OTRB by CRD Token	Wired validators or ABT systems can use this call to get OTRB properties to support validation process.
Sync OTRB	Wireless validators can use this call to download entire OTRB list or synchronize the recent updates in it, to decrease validation the latency.

The calls are described in the following sections

5.2 Get OTRB by CRD Token

5.2.1 Request path

/get-otrb-byct

5.2.2 Purpose.

Wired Validators or the TA's ticketing system can use this call to get the OTRB data to support validation process. Wireless Validators normally do not use this method, as they usually have their local OTRB Lists provided by their ABT system; the latter must use this call.

5.2.3 Request Parameters

Header 'validatorid' may be used. If it is used, header 'valkey' must be used. If both headers are not present, UniTiAg treats the call as directly initiated by the TA's ABT system and authenticates the call respectively using the TA network properties: headers taId and takey.

The following parameter is used in the body:

Name	Type	Necessity	Restrictions	Meaning
ct	Base64-encoded String	m		CRD Token as calculated from the Tap Data. (See Appendix 4.1)

5.2.4 API Response

If the CRD Token 'ct' is associated with a OTRB in the TA OTRB Regional Table, UniTiAg responds with http response code 200 and the following parameters in the response body.

Name	Type	Meaning / Condition
bal	Number	OTR Balance expressed in main units of the TA currency, rounded to 5th digit after the decimal point.
st	Number	OTRB Status: 0 - active, 1 – onhold, 2 - cancelled.
tr	Number	Trust Rate 0-100, as determined by the TSMP

Name	Type	Meaning / Condition
mp	Number	TSMP ID of the TSM which owns this OTRB

Response example is below. Note, that body value will be stringified in the response.

```
{
  "headers": {
    "Content-Type": "application/json"
  },
  "statusCode": 200,
  "body": {
    "bal": 59.79452,
    "st": 0,
    "tr": "50"
  }
}
```

If the CRD Token 'ct' is not found in the regional OTRB Table, UniTiAg responds with statusCode 220 and empty body.

5.2.5 UniTiAg Actions

1. None

5.2.6 TA Actions

1. The TA or Validator can use the output attributes for the validation purposes.
2. In case of response statusCode = 220, the TA should use get-otrb-bypan and expect longer response latency.
3. The TA ABT System may use local stop-lists for cancelled OTRBs to increase the validation throughput.

5.3 Sync OTRBs

5.3.1 Request path

/sync-otrbs

5.3.2 Purpose

The Wireless Validator or the TA's ticketing system use this call to synchronize the local OTRBs data with the targeted OTRB replica at the Regional UniTiAg Host. The call can be periodical or event-driven, e.g., in case of communication error recovery.

5.3.3 Request Parameters

Header 'validatorid may be used. If it is used, header 'valkey' must be used. If both headers are not present, UniTiAg treats the call as directly initiated by the TA ABT system and authenticates the call respectively using the TA headers taId and takey.

The following parameter is used in the body:

Name	Type	Necessity	Restrictions	Meaning
sinceTime	Number	o	Time in Unix format in msec. If this value is less than one year ago, it is changed by UniTiAg to one year ago which is consistent with the time to leave for items in the TA OTRB table.	If present, UniTiAg delivers only changes in the OTRB list since this time. Otherwise, it delivers the full TA OTRB list

5.3.4 API Response

The returned string comprises a comma-separated-values line for each retrieved OTRB. The values a presented in the following order:

Position	Attribute value	Meaning / Condition
1	CRD Token	Base64–encoded string
2	Balance	Current OTR Balance in main units of TA currency, rounded up to three decimal places
3	Unix time of this OTRB last modification or creation	In milliseconds
4	OTRB Status	Integer mapped to the OTRB status as follows 0 – active, 1 – onhold, 2 - cancelled:
5	OTRB Trust Rate	Between 0 and 100.
6	TSMP ID	ID of the TSMP that owns this OTRB

The response is delivered as a stream. The stream will end once all data is retrieved. Example of one streamed line:

```
HaKq16rSxAjhSbMNffMd,64.795,1719579102874,0,50,5\n
```

The stream can be an empty line if no OTRBs are present in the list.

The lines are sorted by the third attribute (Unix time of the last item update) in ascending order.

5.3.5 UniTiAg Actions

1. UniTiAg does not keep OTRBs in a TA OTRB table for more days than specified by parameter riderDormancyDays in the TA settings table or 10 days whichever is greater.

5.3.6 TA Actions

1. The TA uses this received data for validation purposes.
2. The TA may use the 6th parameter (TSMP ID) if the TA policy varies across different TSMPs.
3. If the TA requests the entire OTRB list, the received data may not be fully up-to-date as the call may take some time. The TA should identify the Unix timestamp (parameter 3) in the last received line. If

this time deviates from the current time beyond a reasonable threshold, the TA should make another sync-otrbs call, this time retrieving OTRBs updated after the determined maximum Unix time to ensure the latest data.

4. The TA should establish the aforementioned reasonable threshold based on factors such as the shortest trip duration, the minimum transfer time from another TA to this one, and the time required for synchronizing OTRBs with its validators.
5. The TA should expect the stream to be sizable, potentially around 6 MB for 120,000 active OTRBs. Transit Agencies with larger ridership may consider implementing AWS stream receivers on Wireless Validators to manage data more efficiently. Additionally, a robust post-MVP UniTiAg solution is in development to increase the capacity of sync-otrb calls, supporting several million OTRBs in future implementations.
6. TA should periodically refresh the entire OTRB list by calling this API without sinceTime, or with SinceTime = 0. If this is not done the received OTRB list may comprise OTRBs with status 1 (onold) whereas these OTRB were actually removed long time ago. The period of the entire OTRB list refreshing should be consistent with the TSMP's dormancy period for cancelled OTRBs. Another way to resolve the same issue is not to trust very old OTRB records with status 1 and and execute API TA call *Get OTRB* (this approach will cause more validation latency).

NOTE: TTL (time to leave) for the TaNB OTRB table is set by UniTiAg according to riderDormanceDays based on attribute cAtExp which is the Unix tome of the last change of this OTRB. Dormant OTRBs are automatically removed from he table and will not be retained in this call. If the dormant rider appears at this TA later, the OTRB will be returned to the table after TA API call *Get OTRB*.

