UniTiAg User Guide Appendices

Table of Contents

Appendix 1 Glossary and Acronyms

Term	Definition
ABT (system)	Account-Based Ticketing (System). It communicates with UniTiAg using TA and TANB APIs.
API	Application Programming Interface, also known as protocol. It regulates interactions between computerized systems.
App	An application on some device platform and operating system.
Card Number	cEMV card PAN or DPAN. UniTiAg APIs accept Card Numbers in an encrypted form. UniTiAg securely stores Card Numbers in the encrypted form and never discloses them in APIs. Non-cEMV CRD cards may have card numbers too. They are treated in UniTiAg as CRD Tokens as their card numbers are readable by humans, card readers, and are not PCI DSS-protected. If a cEMV card is not a payment card (that is, its PAN is not protected by PCI DSS regulations). It is treated hereafter as a non-cEMV card.
CRD - Contactless Rider Device	A contactless device that communicates with the Validator to present its unique CRD Token, proving the rider’s right to access transit services. In UniTiAg realm, the CRD is not a payment tool but a means of attributing service entitlement to the rider. Examples include, but are not limited to, contactless EMV cards, UWB-enabled smartphone apps, Calypso cards, and QR code-based solutions. If a smart device (e.g. a phone or watch) may produce multiple Card Token, they are considered as multiple CRDs.
CRD Token	A unique identifier that CRD produces to the validator at CRD tap. When CRD is not a cEMV card, the Rider presents the same identifier to the TSMP during the OTRB creation, directly via a UI or indirectly, via the TSMP app. In case of cEMV cards, CRD Token is an irreversible hash generated from cEMV data that does not comprise “cardholder data” and “sensitive authentication data” as defined by PCI DSS. In this case, UniTiAg links the OTRB to the CRD Token after the first cEMV card tap in the UniTiAg Validator. See Appendix 4.1 for more details.
cEMV	Contactless EMV process as regulated by Contactless Specification governed by EMVCo.
Central UniTiAg Host	The UniTiAg Host supporting TSMP API and TA API.

Term	Definition
DPAN	Device PAN, a tokenized substitute for PAN introduced in Google and Apple wallets.
CSV	Comma-separated value.
ISO 4217	A standard by the International Organization for Standardization (ISO) that defines codes for currencies and their format. The format may vary by currency; for example, USD main unit is divided into 100 sub-units, while other currencies may have 0, 1,000, or 10,000 sub-units.
MVP	Minimum Value Product.
Necessity	Specifies the required appearance of an attribute in an API call. The possible values are: ‘m’ – mandatory (must always be included) ‘c’ – conditional (its presence depends on specific conditions) ‘o’ – optional (at the caller’s discretion).
Targeted Replication	A process that replicates only essential OTRB data from the Central UniTiAg Host to Regional UniTiAg Hosts. This replication is specific to OTRB data needed for fare validation within a given region. Targeted replicas within the same region are TA-specific, ensuring that only relevant data is replicated for each Transit Agency.
ODA	Offline Data Authentication – a cEMV process for ensuring that the tapped cEMV card is genuine. Broadly speaking, The ODA a specific CRD process for offline CRD authentication by Validator.
OTR	Open-To-Ride
OTRB	Represents funds and other attributes as outlined in Appendix 3.3. From the TSMP's perspective, an OTRB may consist of prepaid funds, pre-authorized amounts, portions of a credit line, loyalty points, or a combination of these elements.
OTRB ID	A unique identifier assigned by UniTiAg during OTRB creation, ensuring uniqueness across all TSMPs.
PAN	Primary Account Number, also known as card number.
Payment Scheme	A service and infrastructure that facilitates payments using credit, debit, ATM, and prepaid cards. Examples of Payment Schemes include Visa, MasterCard, American Express (AmEx), Discover, STAR, NYCE, and Interac.
PCI DSS	Payment Card Industry Data Security Standards
Regional UniTiAg Hosts	UniTiAg servers located in geographical regions where TAs operate, supporting the TANB API. They store targeted replicas of OTRBs specific to each TA in their region. The geographical proximity and TA-specific data storage help reduce latency in TANB API calls.
Rider	A user of the transit service. The Rider owns at least one OTRB associated with a CRD.
SaaS	Software as a Service

Term	Definition
TA	A business entity and its ABT System that provides transit services, interacting with UniTiAg through the TA and TANB APIs for fare collection. From the TSMP's perspective, the TA acts as an online marketplace seller (merchant). However, when collecting fares exclusively through UniTiAg SaaS, the TA is not a merchant from the payment schemes' perspective.
TA API	An API provided by the Central UniTiAg Host, allowing Transit Agencies to communicate data that impacts OTRBs and needs to be shared among various TAs. This API facilitates data exchange to ensure accurate and synchronized fare processing across multiple agencies.
TANB API	The Transit Agency Near-By API, supported by Regional UniTiAg Hosts, allows for reduced API call latency by leveraging regionally stored OTRB data in targeted replicas, ensuring low call latencies.
Tap Data	Data generated by CRD and Validator during a contactless tap, e.g. in accordance with the cEMV process and regulations, Calypso specifications, etc.
TSMP	A business entity serving two types of customers: a) Wallet holders, shoppers, consumers, taxi riders, or payers. b) Online merchants offering goods and services to customers in category a). Examples of two-sided marketplaces include Amazon, Uber, and Walmart Online.
TSMP API	An API implemented on the Central UniTiAg Host to facilitate communication between TSMPs and UniTiAg. It manages the OTRB lifecycle and provides reconciliation data.
Trust Rate	A metric set by the TSMP, expressed as a percentage, representing the likelihood of recovering an overdraft of an OTRB. This helps TAs in fare validation decisions when OTRB overdrafts occur. In addition to the Trust Rate, the TSMP may establish fare overdraft limits and other conditions outlined in its merchant agreement with the TA.
UI	User Interface.
Validator	A device that interacts with Cards during taps (or using some other technology) to grant or deny Riders access to transit services. Each Validator is associated with one TA. It communicates with UniTiAg either: • Directly via the TANB API and TA API, or • Indirectly through the TA's ABT system using a proprietary protocol, which is outside the scope of UniTiAg.
UniTiAg – Universal Ticket Agent	A SaaS, providing Riders with frictionless access to public transit services and providing TAs and their ticketing system vendors with open-loop ticketing capabilities.

Appendix 2 Business Architecture

Please refer to online resource https://unitiag.com/unitiag-architecture/ .

Appendix 3 Business Data Models

3.1 TSMP Business Data Model

The following diagram presents business data that the TSMP should adhere to, to acquire UniTiAg capabilities.

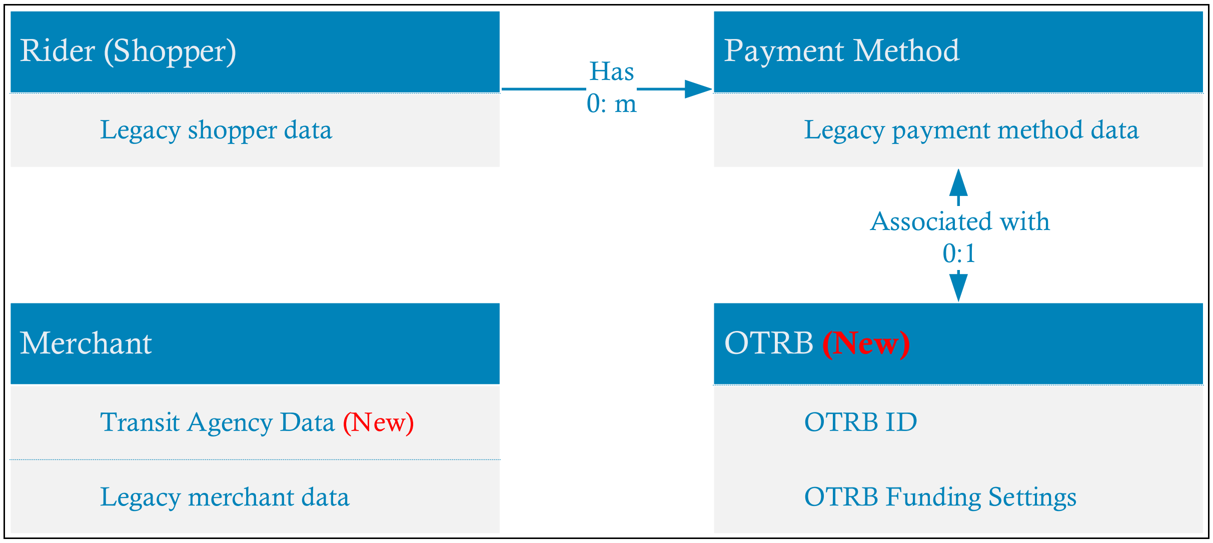

Figure 1. TSMP Business Data related to UniTiAg Implementation

It is assumed that the TSMP already holds legacy shopper and payment method data, which describe users such as wallet holders, payers, or riders (e.g., in the context of Amazon or Uber). The TSMP may fund OTRBs using various methods like "pay later," "pay before," credit lines, or loyalty points. Funding policies are managed by the TSMP and fall outside UniTiAg's scope.

Examples of TSMP funding policies:.

3.1.1 Example: OTRB as a wallet (pay before):

The TSMP creates an OTRB via the TSMP API call *Create OTRB*, charges the Rider for a specific amount, and updates the OTRB amount using TSMP API call *Update OTRB Amount.* When UniTiAg issues a ***Low OTRB Amount Warning***, the TSMP charges the rider, refills the OTRB based on the Rider's preset refill amount, and updates the OTRB amount again.

3.1.2 Example: Line of Credit (pay later)

Periodically (e.g., monthly or daily), the TSMP retrieves OTRB reports using TSMP API call *Get OTRB Report*, charges the payment method for the period's total, and refills the OTRB via TSMP API call *Update OTRB Amount*.

3.2 UniTiAg Business Data Model

UniTiAg Business Data Model is presented in the following diagram. Only main data attributes are listed there. API-specific details are provided in the next sections.

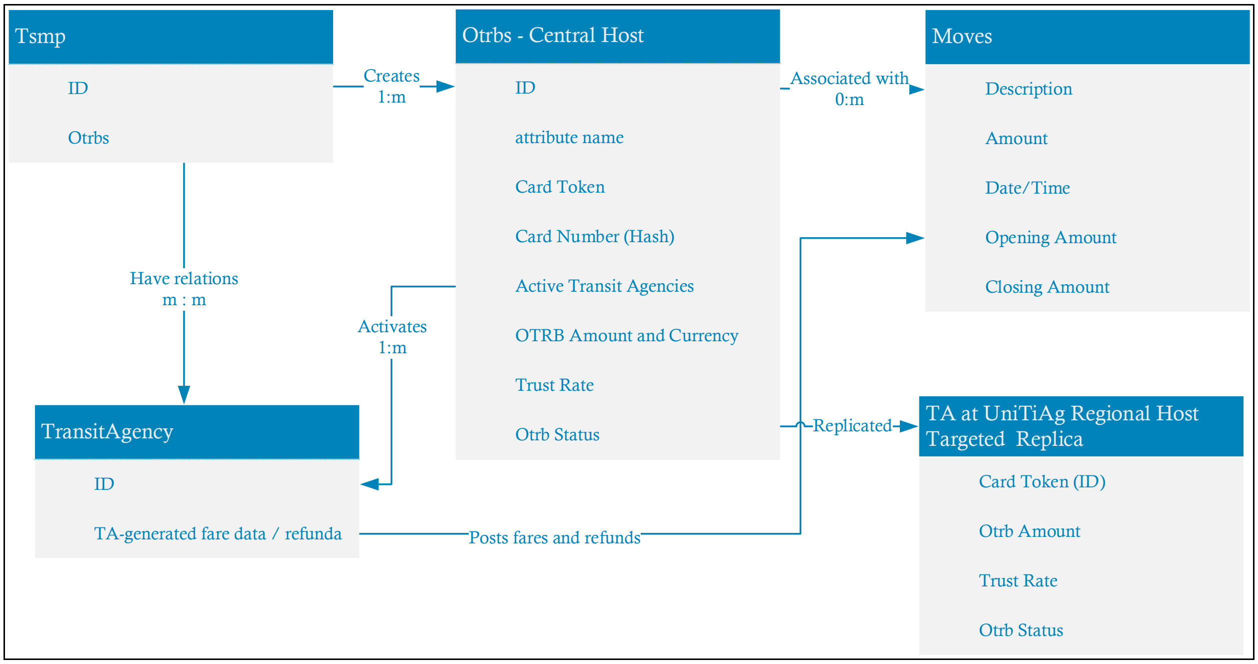

Figure 2. UniTiAg Business Data. Main Components

3.2.1 UniTiAg API-Related Data

The API-specific data objects (Tables) are described below. These attributes are references in the API User Guides.

In these Tables:

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

For better understanding the data specifics, please refer to Appendix 1.

*3.2.1.1 Data Table **Otrbs***

Attributes

Name	Type	Meaning and Limitations	Comment
ID	String	OTRB ID	
tsmpId	Number	Unique TSMP ID as assigned by UniTiAg to the TSMP	
cardNumber	String	See Card Number in the Glossary	
cardToken	String	CRD Token. Base64-encoded.	This is a unique OTRB identifier.

Name	Type	Meaning and Limitations	Comment
effectiveStopTime	Number	Unix time in msec when the OTRB is effectively cancelled or put on hold.	Since that time, UniTiAg allows only OTRB withdrawals. Before this time expires UniTiAg does not allow cardToken update.
lastChangedAt	Number	Unix time when the OTRB was changed. Changes that the TAs need to know are considered.	Used in TA API
otrbCurrency	String	The currency of this OTRB as selected by the Rider / TSMP	Assigned by TSMP at Create OTRB API call. Never changes.
trustRate	Number	Trust Rate	As set by TSMP
lowAmount	Number	UniTiAg must warn the TSMP when the running OTRB amount is decreased below this limit.	Expressed In otrbCurrency main units.
otrbAmount	Number	Last known OTR Balance.	Expressed in otrbCurrency main units. Not rounded.
OtrbName	UTF-16 String	Used in the Rider UIs in the TSMP and the possibly in TAs to point-out to this OTRB.	An acronym set by Rider. E. g. “Daughter’s OTRB”, “Euro OTRB”.
otrbStatus	String	OTRB Status, either: ‘active’, ‘onhold’, ‘cancelled’	
riderId	String	Unique within TSMP rider identifier	As assigned by TSMP. Never changes
tas	Number	Bitmask designating TAs active for this OTRB.	

*3.2.1.2 Regional Data Table **Otrbs** (Targeted Replica)*

This Table is TA-specific. It comprises only OTRBs active in a given TA.

Attributes

Name	Type	Meaning and Limitations	Comment
cT	String	CRD Token. Base64-encoded.	This is a unique OTRB identifier.
bal	Number	OTR Balance expressed in major units of TA Currency,	Rounded to 5th digit after the decimal point.
cAt	Number	Unix time when the OTRB was changed.	Changes that the TAs need to know are considered.
tr	Number	Trust Rate	
st	Number	OTRB Status 0: ‘active’, 1: ‘onhold’, 2: ‘cancelled’	
cAtExp	Number	Unix time of item expiration.	Is used to reduce the replica volume by excluding dormant OTRBs.

*3.2.1.3 Data Table **Moves***

UniTiAg creates an item in this Table each time the OTR balance is changed, as a result of some API call.

Common Attributes as created by ether TSMP or TA/TANB API calls

Attribute	Value Type	Meaning and Limitations	Comment
ID	String	Unique item ID	Generated by UniTiAg. The TSMP should use this ID to match reconciliation reports with the rider-centric OTRB reports.
atu	Number	Unix time (msec) at which this item was created	.
tsmpId	Number	TSMP ID.	
otrbId	String	OTRB ID	
amnt	Number	Increment of OTR Balance amount. Negative if caused by withdrawal or TAs fare. Positive, if caused by OTRB refill, TA refund, or fare return.	Expressed in OTRB currency, in main units, not rounded.
opAmnt	Number	Opening OTRB amount, before this item creation.	Expressed in OTRB currency in main units
clAmnt	Number	Closing OTRB amount. As a result of this item creation. In OTRB currency	Expressed in OTRB currency in main units
curr	String(3)	OTRB Currency	
why	UTF16 String	Description of the reason this item is created	As set by TSMP or TA
by	String	“TSMP”, “TA”, or “Tap”	Class of entities caused this item creation.
requestId	String	Unique per submitting entity (either TA or TSMP)	Supports idempotency.
moveCreateLatency	Number	Time required to UniTiAg to process this move Item, in msec	

Specific attributes as posted by TA / Validator in TA and TANB APIS.

Attribute	Value Type	Meaning and Limitations	Comment
taId	Number	TA ID	
taAmnt	Number	Fare (<0) or refund (>0) amount as presented by TA	Expressed in main units of TA currency
opTaAmnt	Number	Opening amount, i.e. OTRB amount before the tap as known to the TA at the time of tap.	Expressed in main units of TA currency
taCurr	String (3)	The TA currency.	
fxRate	Number	ForEx rate to OTRB currency	A multiplier to convert amounts from TA currency to OTRB currency.
valId	String	Validator ID	Not present if the call made by TA
rtId	UTF16 String	<= 20 characters	Route ID as presented by the TA
stId	UTF16 String	<= 20 characters	Stop or station ID as presented by the TA
vhId	UTF16 String	<= 20 characters	Vehicle ID as presented by the TA
tapAtu	Number	Unix time of CRD tap as registered by the Validator (msec)	

Attribute	Value Type	Meaning and Limitations	Comment
tLat	Number	Tap latency in msec.	
vLat	Number	Validator latency in msec.	This includes tap latency as well as time needed to retrieve the OTRB balance, and time the Validator spent on making a validation decision (before posting the Fare).
reconStatus	Number	Reconciliation status	See explanation below
dlbRiskAmount	Number	Deliberate risk amount caused by overdraft known to TA/Validator at the time of tap	In main units of TA currency. TSMP shall attempt to recover the overdraft from the Rider if within its policies.
odRiskAmnt	Number	Overdraft recon risk amount Based on the actual OTRB amount, known to UniTiAg at the time of this item creation.	In main units of TA currency. TSMP shall attempt to recover the overdraft from the Rider if within its policies.
regAtu	Number	Unix time of this TA API call item being registered by UniTiAg (msec)	
delFareAmnt. **POST MVP**	Number	Delay charge risk amount in main units of TA currency. At risk of not to be reconciled because of late fare report or late OTRB sync	TSMP tries to recover the overdraft from the Rider, before declining reconciliation
delSyncAmnt. **POST MVP**	Number	Delay charge risk amount in main units of TA currency. At risk of not to be reconciled because of OTRB sync	TSMP tries to recover the overdraft from the Rider, before declining reconciliation

Explanation of reconStatus:

Value	Meaning	Promise
0	OK	To be reconciled
1	Overdraft. At least some portion of the fare may be not reconciled. Caused either by the deliberate overdraft based on the Validator/TA decision or overdraft at the time of fare post, or both.	The 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 persist, if the TA returned a portion of the fare later (typical case at tap-off).
2	Attempt to charge non-active OTRB	Not to be reconciled, OTRB amount is not affected as a result of the Post Fare TA API call that created this move item

*3.2.1.4 Data Table **Tsmps***

This object describes a single TSMP. Only attributes relevant to the APIs are depicted.

Attributes

Name	Type / Restrictions	Meaning	Comment
ID	Number	Created by UniTiAg unique reference to the TSMP	UniTiAg creates it during TSMP onboarding
tasMask	Number	A binary mask. Specifies all TAs this TSMP deals with.	E.g. if this TSMP has TA 2 and TA 3 as merchants, the mask is 2**2 + 2**3 = 4 + 8 = 12.
lowAmountCallback	String	A URL to warn the TSMP about low amount	
cancelledOtrbTtlDays	Number	Number of days to live for cancelled OTRBs and their move items	
lowAmountApiKey	String	The key of the secret where the API Key is stored	

3.2.1.5 Table: Tas

This object describes a single TA. Only attributes relevant to the APIs are depicted.

Attributes

Name	Type / Restrictions	Meaning	Comment
ID	Number	Unique TA Identifier.	As assigned by UniTiAg.
acceptWindowHours	Number	Time window to accept post-refund and post-fare calls from TA	In hours. Late posts are rejected.
currencyCode	String (3)	TA currency code,	
description	A set of UTF16 Strings	Description of the TA as set by it, including location, modes of operations, transit operators, etc.	As presented by TA Includes strings legend, name, URI.
idempWindowHours	Number	Window to reject duplicated API call items for post-fares and post-refunds, in hours	To support API Idempotency
maxFare	Number	Max fare or refund amount, in main units of TA currency.	
postItemsLimit	Number	Limits the number of items in one Post Fares or Post Refunds TA API call.	
region	String	AWS region code	E.g. ‘us-east-1’
regTable	String	AWS table name in Regional UniTiAg Host	
validatorAcceptWindowHours	Number	Window to accept *Post Fare* API calls from Validator	In hours. Late posts are rejected.

Name	Type / Restrictions	Meaning	Comment
tsmpsMask	Number	Present TSMPs having merchant relations with this TA. This setting must be consistent with TSMPs tas mask.	E.g. if this TA is a merchant for TSMP 2 and 3, the mask is 2**2 + 2**3 = 4 + 8 = 12.
riderDormancyDays	Number		Number of days to keep dormant CRD tokens in the regional TA table. Default 354, min 10.

3.2.1.6 Table: Settings

This object describes a single TA. Only attributes relevant to the APIs are depicted.

Attributes

Name	Type / Restrictions	Meaning	Comment
stageLowWarningSnsTopicArn	String	Must point to the SNS topic with low amount warning lambda	
maxOtrbSyncDelay	Number	Set in seconds. Should be more than TA's are allowed to delay their fare reports.	Also affects otrb.effectiveStopTime
otrbReportItemsLimit	Number	The limit of numbers in OTRB report, returned by /report-otrb TSMP API call	
maxLoadParamDelay	Number	How often settings parameters must be refreshed in Lambda function (sec).	

Appendix 4 Hash Algorithms

cardToken and cardNumber are 20-char long base64-encoded strings.

4.1 CRD Token Algorithm

4.1.1 CRD Token Derived From a cEMV Card

When a validator creates cardToken during the cEMV card tap, CRD Token must be a Base64-encoded string comprising first 15 bytes of SHA-256 hash of EMV tag: 9F46 - "ICC Public Key Certificate" value – of the first ADF in the card PPSE directory list.

Below is an example of Kotlin function implementing this algorithm.

```
fun makeCardToken(iccTagValue: ByteArray): String {
  //creates 15-byte long hash (which is 20-char-long after base64-encoding),
 //  from the value of EMV Tag 9F46 ICC Public Key Certificate

    // Get the SHA-256 message digest instance
    val digestInstance = MessageDigest.getInstance("SHA-256")

    // Hash the byte array using SHA-256
    val hash = digestInstance.digest(iccTagValue)

    // Truncate the hash to 128 bits (15 bytes)
    val truncatedHash = hash.copyOfRange(0, 15)

    // Encode the truncated hash to Base64 string
    val base64EncodedHash = Base64.getEncoder().encodeToString(truncatedHash)

    return base64EncodedHash
}
```

4.1.2 CRD Token for Non-Payment CRDs

If the OTRB is associated with a non-cEMV card, the CRD Token is created by an app or Rider UI, in a session with the TSMP.

The algorithm of CRD Token creation is negotiated with each owner of CRD Type app. It ensures the following:

- The CRD Token is unique in UniTiAg
- The TSMP creates and Validator determines the same value of the CRD Token
- The CRD Token length in binary format must be between 15 and 32 bytes, the lesser – the better.
- The CRD Token must be linked to a X.500/X.509 certificate where Common Name encodes the CRD Token

The following algorithm is recommended and is used in OTRB created in Telegram's demo mini app RideOnTonBot:

1. Build a string comprising: Prefix, User ID, and unique CRD ID, where:

   a. Prefix is an alpha-numeric symbol assigned to the app owner. It is “2” for RideOnTonBot.
   b. User ID – unique ID within the app owner
   c. CRD ID – a unique device ID within the app owner
2. Create a SHA-256 hash of this string
3. The CRD Token is the first 15 bytes of this hash.

4.2 Card Number Algorithm

UniTiAg supports PAN / DPAN encryption algorithms and formats of cardNumber presentation in APIs as negotiated with ABT system vendors and TSMPs.

If the CRD is not a cEMV cards, the card number (if exists) is usually encapsulated in the CRD Token, is a CRD Token, or is not used at all. This is up to the entity that owns or regulates this type of CRD.

4.3 Card Number Encryption Algorithm

Card Number Encryption is used for presenting the Card Number in API calls when OTRB is associated with a cEMV card, in accordance with PCI DSS regulations. The specific algorithm is negotiated:

- Between TA’s ABT system vendors and UniTiAg for TA API calls Get OTRB.
- Between TSMPs and UniTiAg for TSMP API calls Create OTRB.

Other API calls do not use Card Number attributes in calls and responses.

Appendix 5 Example of Validation Flow

The following flow chart depicts an example of validation decision-making flow on TA or Validator level.

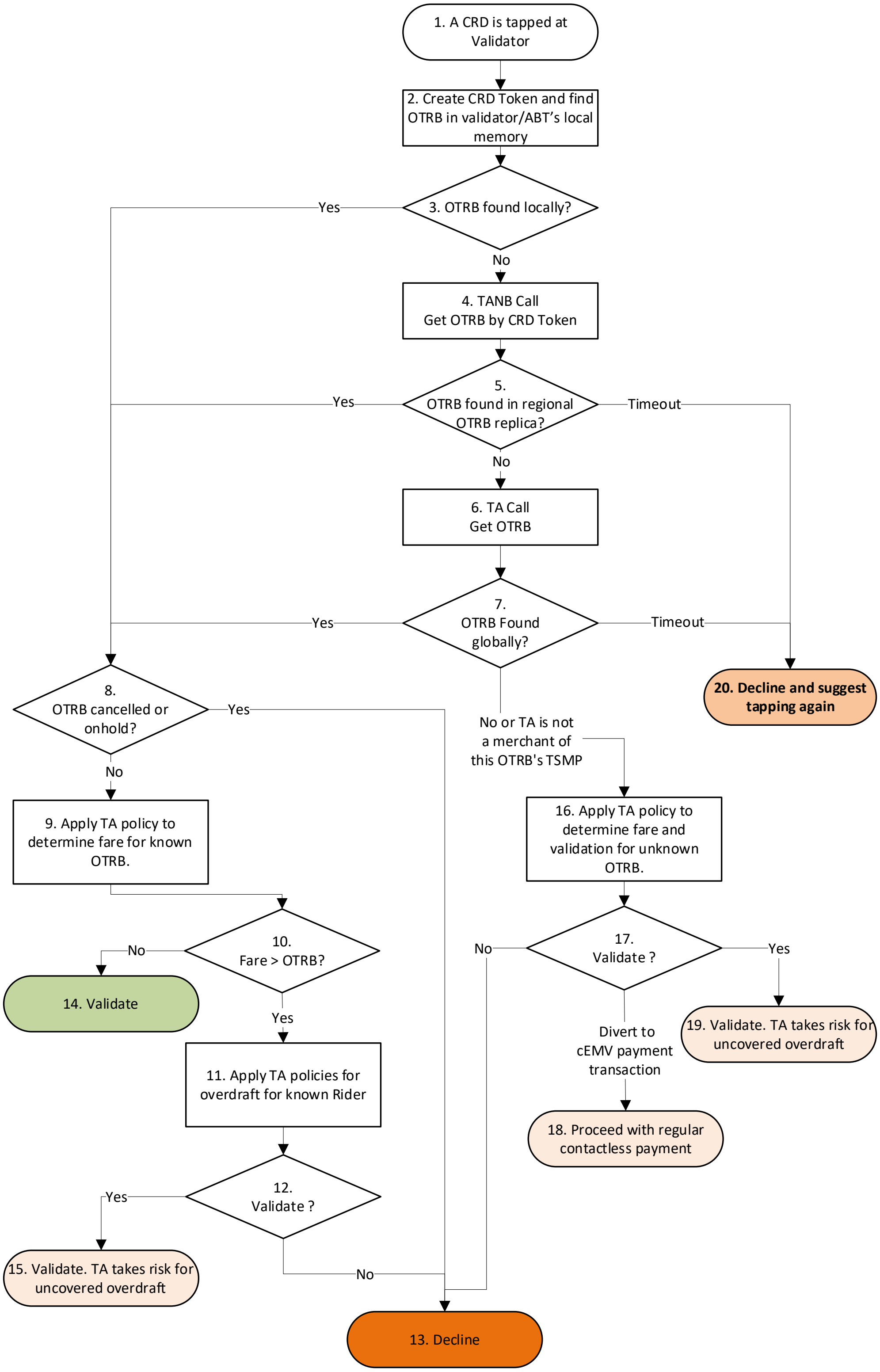
1. A CRD is tapped at Validator
2. Create CRD Token and find OTRB in validator/ABT's local memory
3. OTRB found locally?
Yes
No
4. TANB Call Get OTRB by CRD Token
5. OTRB found in regional OTRB replica?
Yes
Timeout
No
6. TA Call Get OTRB
7. OTRB Found globally?
Yes
Timeout
20. Decline and suggest tapping again
8. OTRB cancelled or onhold?
Yes
No or TA is not a merchant of this OTRB's TSMP
No
9. Apply TA policy to determine fare for known OTRB.
16. Apply TA policy to determine fare and validation for unknown OTRB.
10. Fare > OTRB?
No
14. Validate
Yes
17. Validate ?
No
Yes
Divert to cEMV payment transaction
19. Validate. TA takes risk for uncovered overdraft
11. Apply TA policies for overdraft for known Rider
18. Proceed with regular contactless payment
12. Validate ?
Yes
15. Validate. TA takes risk for uncovered overdraft
No
13. Decline

Figure 3. Validation Flow Example

Appendix 6 UniTiAg Contactless Interface Specification (HCE)

Protocol: ISO/IEC 14443-4 (Type A/B) over Android Host Card Emulation (HCE) Standard: ISO/IEC 7816-4 APDU

6.1 Overview

This specification defines the contactless communication protocol for Android applications complying with the UniTiAg standard (e.g., app “Bye-Bye Cards”). The application functions as an offline EMV-like entity using a simple state machine to authenticate the user via a stored CRD Token and a signed certificate.

6.2 Application Identifiers (AIDs)

The application supports discovery via the standard Payment System Environment (PSE) method or direct selection.

Entity	AID (Hex)	Description
DDF	325041592e5359532e4444463031	"2PAY.SYS.DDF01" (Standard Payment Directory)
UniTiAg App	F5633545180001	The specific UniTiAg Application AID.

Export to Sheets

6.3 APDU Command Flow

6.3.1 Step 1: Select PPSE 2PAY.SYS.DDF01

This step is implemented for the compatibility purposes. It is not mandatory because the card reader knows exactly which AID to select in Step 2. If this step is executed the smartphone’s operating system may direct the NFC session to an app with a higher priority, e.g. a Wallet.

APDU Request/Response Trace:

```
Select PPSE APDU Request: 00a404000e325041592e5359532e444446303100
Transceive Time=31 msec (here and below is provided for example). SW1/SW2=9000.
Response content:
6f   FCI Template, length=41
  84   Dedicated File Name, length=14
    "2PAY.SYS.DDF01"
  a5   FCI Proprietary Template, length=23
    bf0c FCI Issuer Discretionary Data, length=20
      61 Directory Entry, length=18
        4f ADF Name, length=7
          f5633545180001
        50 Application Label, length=7
          "UNITIAG"
---- End of Response Content.
```

Tag 4F comprises the AID of the CRD Token app.

6.3.2 Step 2: Select ADF

The card reader selects Application Data File for the AID F5633545180001.

APDU Request/Response Trace:

```
Select ADF APDU REQUEST: 00a4040007f563354518000100.
Transceive Time=15 msec. SW1/SW2=9000.
Response content:
6f   FCI Template, length=49
  84   Dedicated File Name, length=7
    f5633545180001
  a5   FCI Proprietary Template, length=38
    50   Application Label, length=7
      "UNITIAG"
    9f38 PDOL, length=8
      9f1c089f37049a03
    df01, length=15
      73084de18d15f9ac4734662814d4bb
---- End of Response Content.
```

In tag 9F38, the “card” requests from the card reader 9F1C (Terminal ID, 8 bytes), 9F37 (Unpredictable Number, 4 bytes), and 9A (Transaction Date, 3 bytes). In tag DF01, the “card” presents the 15 byte-long CRD Token.

At this point, the validator can initiate a parallel process to look up the CRD Token in the validator’s OTRB list (don’t forget to re-encode the binary value on DF01 to Base64-encoded string), and continue with CRD Token offline data validation (next steps).

6.3.3 Step 3. Get Processing Option

On this step, the card reader presents the data objects requested in the PDOL which the “card” signs with its private key and returns the Signed Dynamic Application Data.

APDU Request/Response Trace:

```
GPO APDU REQUEST: 80a8000011830f30303132303031325cb0b6f725122000
Transceive Time=34 msec. SW1/SW2=9000.
Response content:
77   Response Message Template Format 2, length=73
  94   Application File Locator, length=4
    08010202
  9f4b Signed Dynamic Application Data, length=64
     7f54 ... db
---- End of Response Content.
```

Tag 94 points to file 1, records 1 and 2. The card reader obtains these records on the next step and uses them for Offline Data Authentication.

6.3.4 Step 4. Read Records

The terminal reads two records specified in the AFL tag . The records comprise the public key X.509 certificate issued by the TSMP's Certification Authority. This is the last step of the NFC session.

APDU Request/Response Trace:

```
Read Record 1 in file 1 APDU REQUEST: 00b2010c00.
Transceive Time=44 msec. SW1/SW2=9000.
Response content:
70   Read Record Response Message Template, length=244
  9f46 ICC Public Key Certificate, length=240
       3082  ...  d718
---- End of Response Content.

Read Record 2 in file 1 APDU REQUEST: 00b2010c00.
Transceive Time=36 msec. SW1/SW2=9000.
Response content:
70   Read Record Response Message Template, length=190
  9f48 ICC Public Key Remainder, length=186
    0a56 ... a8c4
---- End of Response Content.
```

6.3.5 Step 5. Offline Data Authentication

Upon finishing the NFC session, using the ICC Public Key, the PDOL data, the Signed Dynamic Application Data, and the well-known Certificate Authority public X.509 certificate, the card reader completes offline data authentication.

The card reader also compares the CRD Token obtained in Select ADF stage, tag DF01 with ICC Public Key Certificate component Certificate Name "CN'. The latter also presents the CRD Token encoded as a 30 char-long string of hexadecimal digits. In this case it must be "CN=73084DE18D15F9AC4734662814D4BB"