The Clearing House®

DDA Token Service Implementation Guide

Version 1.0

September 16, 2022

DISCLAIMER

The Clearing House believes that the content and information furnished herein is accurate and reliable as much care has been taken in its preparation. However, all content and information herein is provided on an "as-is" basis and The Clearing House expressly disclaims all liability and all warranties of any kind with respect thereto. Users of the content and material provided herein should exercise due care, including consultation with legal counsel where appropriate, to assure that use of the information will be in full compliance with the laws, rules, and regulations of the jurisdictions where it is used.

Use of Tokens is subject to the Token Service Rules, as well as the RTP Participation and Operating Rules (collectively "RTP Rules") and EPN Membership and Operating Rules (collectively "EPN Rules"), as applicable.

CONTENTS

1 About This Implementation Guide

1.1 Intended Audience

The primary audience for this guide is business and technical personnel at:

(i) Financial institutions that participate in The Clearing House's DDA Token Service (Token Participants),
(ii) An agent/technical service provider to a Token Participant or Sponsored Token Requestor (Technical Agent)

1.2 Related Training and Documentation

See the "STE DDA API Specifications for Banks" and "STE DDA API specifications for Token Requestors" for coding and field level information. Available on the TCH website at www.theclearinghouse.org

1.3 Customer Support

Contact TCH Operations Support at 800-875-2242
Email: *OperationsClientServicing@theclearinghouse.org*

2 About DDA Tokenization

2.1 DDA Tokenization Overview

The Clearing House offers an account tokenization service –the DDA Token Service (or "Token Service") "Secure Token Exchange (STE)" – for Participants in the RTP Network and Electronic Payments Network (EPN).[1] Tokens replace a real DDA account number with a randomized account number and are used with a dedicated Routing Transit Number specified by a Token Participant (Token RTN). The use of a tokenized account number instead of the real DDA number provides an additional layer of security to Participants by both limiting the distribution of real account data, and also through the use of controls on a token that enable the token to be suspended, deleted or replaced if compromised. For more information on the benefits of tokenization, please refer to the TCH website at www.theclearinghouse.org.

Key terms described in this DDA Token Implementation Guide include:

- **Tokens** - An identifier issued by TCH that is associated with a Token Participant's eligible account numbers and may be used in lieu of an account number in RTP Messages, ACH Entries, or both.[2]
- **Token Participants** (also referred to as "Participants" in this document) – RTP and/or EPN Participant banks that enroll in the DDA Token Service to allow tokens to be issued for their

[1] This Implementation Guide does not cover TCH issued Tokens that may be used for payments over the card networks.

[2] UPICs issued under Appendix 1-B of the EPN Rules are not Tokens under the Token Service Rules.

customers' accounts. Token Participants may also request tokens and have certain lifecycle management capabilities.

- **Token Requestor** – Either (i) a customer of a Token Participant that is sponsored by the Token Participant to enable such customer to request tokens from the Token Service (referred to as a "Sponsored Token Requestor"), or (ii) a Token Participant (bank) when it requests a token from the Token Service (referred to as a "Participant Token Requestor").
- **Token User** – A Person that uses a token in Messages, Entries, or both
- **Technical Agent** – An agent of a Token Participant or Sponsored Token Requestor that is appointed using a process TCH specifies, and which provides the technical means by which such Token Participant or Sponsored Token Requester may obtain and manage tokens on behalf of the Token Participant or Sponsored Token Requestor under these Token Service Rules. A Technical Agent may be a financial institution, technology service provider, or other organization approved by TCH.

The definitions provided in this Implementation Guide are intended to be "plain English," are provided for reference only, and do not alter the meaning of such terms under the Token Service Rules or a Token Participant's obligations under such rules. See the Token Service Rules for the full legal definitions of these terms. .

2.2 Token Attributes

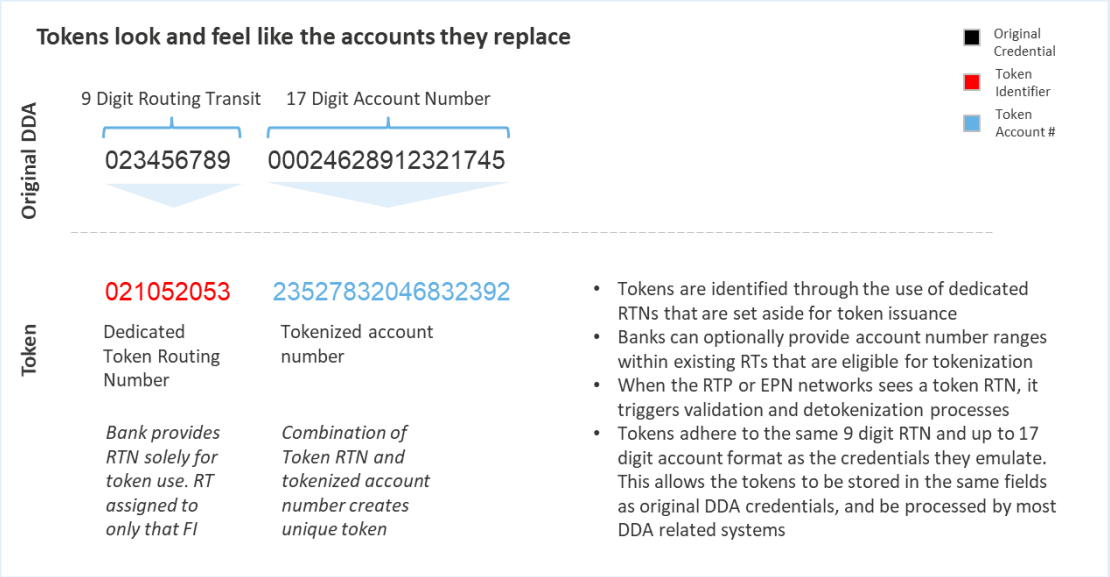
Some general attributes of tokens include:

Token Attribute	Description
Form Factor Preserving	Tokens use the same RTN/account number structure as "real" DDA accounts. Tokens are issued under real RTNs licensed to a bank and use conventional length numeric account numbers so that they can process normally over TCH's RTP & EPN networks as well as FedACH.
Token Participant Identifier (also referred to as "Token RTN" within this document)	Each Token Participant must provide a new or (empty) existing RTN to be designated as a Token RTN. All tokens will be created using this RTN and the Token RTN is used to identify tokenized account information. A list of tokenized RTNs is available on the TCH website.
Token Numbers are Static	Once issued, token numbers stay the same. This allows the tokens to be stored and utilized in the same ways as the original account number that they replace
Token Status	Tokens use various "lifecycle states" as part of their core controls. These statuses include, active, suspended, deleted, and resumed (if suspended). The ability to suspend or delete tokens without impacting the underlying real DDA is a primary advantage to tokenization
Counterparty Restrictions *(RTP Only)*	Tokens can (optionally) be restricted so that only RTP transactions between two specific accounts will be processed. For a counterparty restriction to be added to a token, the account-holding bank (Token Participant) must

	specify the counterparty account/RTN that may use the token. Counterparty restrictions can be used to restrict a token to an ongoing payment relationship, e.g., bill pay with a specific biller, B2B, etc. Counterparty restrictions are only validated on the RTP network. If a token with a counterparty restriction is presented in ACH, the counterparty restriction will not be validated as part of processing. When requesting a counterparty restricted token, the Token Requestor must also take certain actions. TCH recommends that financial institutions consult with the Token Service team before implementing counterparty restrictions. For more information see the "STE-DDA API Specifications for Banks" available on the TCH website.
Multiple Tokens per DDA Account	Multiple tokens may be issued for the same real account. This allows a different token to be issued to different entities which in turn allows tokens to be suspended or deleted without impacting all other tokens that have been issued for that real account. A different token is generated for each separate Token Requestor ID (TRID). This includes: • *Token Requestor/User Specific* – Each token assigned to a Token Requestor/User (e.g., biller, disburser, aggregator, etc.) is issued its own TRID.In general the token service will assign a TRID to the most specific known entity identified in the provisioning information (Token Requestor, third party, fourth party, etc.). If that is a Token Requestor, the TRID is assigned there. If a Token Requestor passes the token to a token user who is identified by an ID, the TRID is assigned to the token user. If the user is unknown, the TRID stays assigned to the token requestor • *RTP Network TRID* – Used for in-network tokens (only one token per account is issued for all in-network requests) • *Bank TRID* – Used for General-Use tokens which are requested by the account-holding Token Participant (only one token per account is issued for all general-use requests) • *Token Label* – Can be used to generate separate tokens even if the TRID is the same. This may be useful for joint accounts that need a separate token for each account holder
Network Interoperability and Token Restrictions (RTP Settings, ACH Settings)	DDA tokens issued by TCH may be used in messages or entries on the RTP and EPN networks respectively if the Token Participant is a participant in, and has implemented its Token RTN on, both systems. This approach to interoperability mirrors the function of the real account number. However, Token Participants can request that

	tokens issued for the Participant's accounts be limited to one or more of the following if a particular use case merits the limitation of the token: • RTP – On/Off • EPN Credit – On/Off • EPN Debit – On/Off *Note: The decision to limit token interoperability can result in the rejection of otherwise valid transactions. Because of this fact, TCH strongly recommends full interoperability unless a clear reason exists to limit a token's network acceptance. TCH recommends engaging with the Token Service team to discuss network interoperability, particularly if a Token Participant does not participate in both the RTP network and EPN.*

Figure 1: Token Form Factor



Tokens may be issued to a Token Participant (in which case the bank is a "Participant Token Requestor"), or to a Sponsored Token Requestor. Both a Token Participant and a Sponsored Token Requestor may use a Technical Agent to connect to the Token Service to request and receive tokens and perform certain other functions on behalf of the Token Participant or Sponsored Token Requestor.

The DDA Token Service unmasks (de-tokenizes) a Token Participant's own tokens during RTP and ACH transactions which are delivered to that Token Participant. This enables each Token Participant to receive their real account information during a transaction to facilitate normal processing and posting. In RTP, tokens used by counterparty banks (for their own customer accounts) will not be detokenized during transactions. However, "token mapping" (to identify the real account number) can be provided after the fact if an RTP Participant or EPN Participant needs the account number for regulatory or compliance purposes.

2.3 Prerequisites

Participants must meet the following prerequisites before they use DDA Token Service in TCH's production environments:

Prerequisites	Comments:
Enroll in Secure Token Exchange (STE)	Enrollment establishes a bank as a Token Participant and enables tokens to be generated for its accounts. Banks that enroll as Token Participants authorize TCH to issue tokens for the bank's accounts to any Token Requestor.
Designate Token RTN	Participants must select at least one dedicated RTN as a Token RTN. All tokens created by the DDA Token service for a Token Participant's accounts will be associated with that Token Participant's Token RTN. This RTN acts as an indicator to the RTP and EPN Networks that a token is present and therefore cannot be used for any non-tokenized activity. This RTN should not be set as "On-Us" for bank processing.
Designate Token RTN as EPN endpoint	For tokens to be used on EPN, the Token RTN must be set as an EPN endpoint (in both EPN and FedACH records) so that ACH entries containing tokens originated through FedACH are delivered to EPN.
Pass API certification criteria	Participants or Sponsored Token Requestors using APIs for provisioning or lifecycle management must successfully complete token vault test cases in TCH's bank test environment for related APIs.
Support/handle RTP Report Specification 2.10 for tokenization *(RTP Only)*	Review the 2.10 reporting specifications to determine changes in reports specific to RTP tokenization.

2.4 Onboarding

TCH will coordinate enrollment with a prospective Token Participant to enable the DDA Token Service.

The following comprises much of the functions and information gathered during the onboarding process:

- Contact information for Business, Operations, and Information Security
- Token Participant will provide TCH with a bank RTN dedicated for token issuance
    - The RTN must be owned by the Participant that is enrolling in the DDA Token service
    - This "Token RTN" must be used solely for tokenization purposes and should not be used for the issuance of "real" accounts
    - The Token RTN should only be used for payments on the RTP and EPN networks

  - For tokens to be used on EPN, the Token RTN must be set as an EPN endpoint
  - Participant must indicate the settlement RTN for intra-EPN tokenized ACH Entries (originated through EPN) if it is different than the Token RTN. *Note: if a tokenized Entry is an inter-operator transaction (originated through FedACH), the Entry will settle through FedACH settlement to the Federal Reserve account associated with the Token RTN.*

- Participants must identify the Eligible Account Numbers and related RTNs that TCH is authorized to issue tokens for. Participants have the option of limiting token issuance to certain real account RTNs and account ranges within an RTN. If this option is desired, the Participant will need to provide the real account ranges that should be eligible for token creation
- Participant security related configuration will be performed by TCH (define keys to securely store generated tokens per bank) and exchanged with the bank Participant
- Participant will need to certify in TCH's test environment (only required for API access to the service or in-network provisioning method)
- Participant will choose one of the following two options for token management (e.g., to manage Token Requestor settings and the ability to suspend/delete a token, etc):
  - Connect to the STE Customer Service Portal – Set up authorized users, user permissions and deliver RSA tokens
  - Code to Lifecycle Management (LCM) API – Requires testing of the API

3 Provisioning (Token Creation & Delivery)

Token Participants will initially have four provisioning methods available to generate DDA tokens:

1. **On-Behalf-Of Provisioning**[3] – All Token Participants agree to allow TCH to generate and deliver tokens for their accounts to Token Requestors (other participating banks or Sponsored Token Requestors) without the Token Participant's direct involvement. This "OBO" or on-behalf-of service enables TCH to swap real account numbers held on-file held at entities like billers, disbursers, or aggregators with tokens. TCH recommends that Token Participants utilize available reports and or the customer service portal to be aware of and manage issued tokens.
2. **Direct Provisioning** – The second method is direct provisioning between the Token Participant (or their Sponsored Token Requestor) and TCH via an API. In this option, the bank (Participant Token Requestor) requests a token to use or distribute themselves for either its own accounts or another Token Participant's accounts. This option is particularly useful for Token Participants wanting to offer a white-label token service built around the STE product.
3. **Provisioning via a Technical Agent** – The third option is for a Token Participant to use an agent to request a token on behalf of the Token Participant.
4. **In-network Provisioning** *(RTP Only)* – Allows the Sender of an RTP Payment Message or Non-Payment Message (RTP Message) to set a "flag" requesting that their account be tokenized during transactional processing, and the token generated for the sending account is then sent during leg two of the RTP Message to the Receiving Participant/Message Receiving Participant.

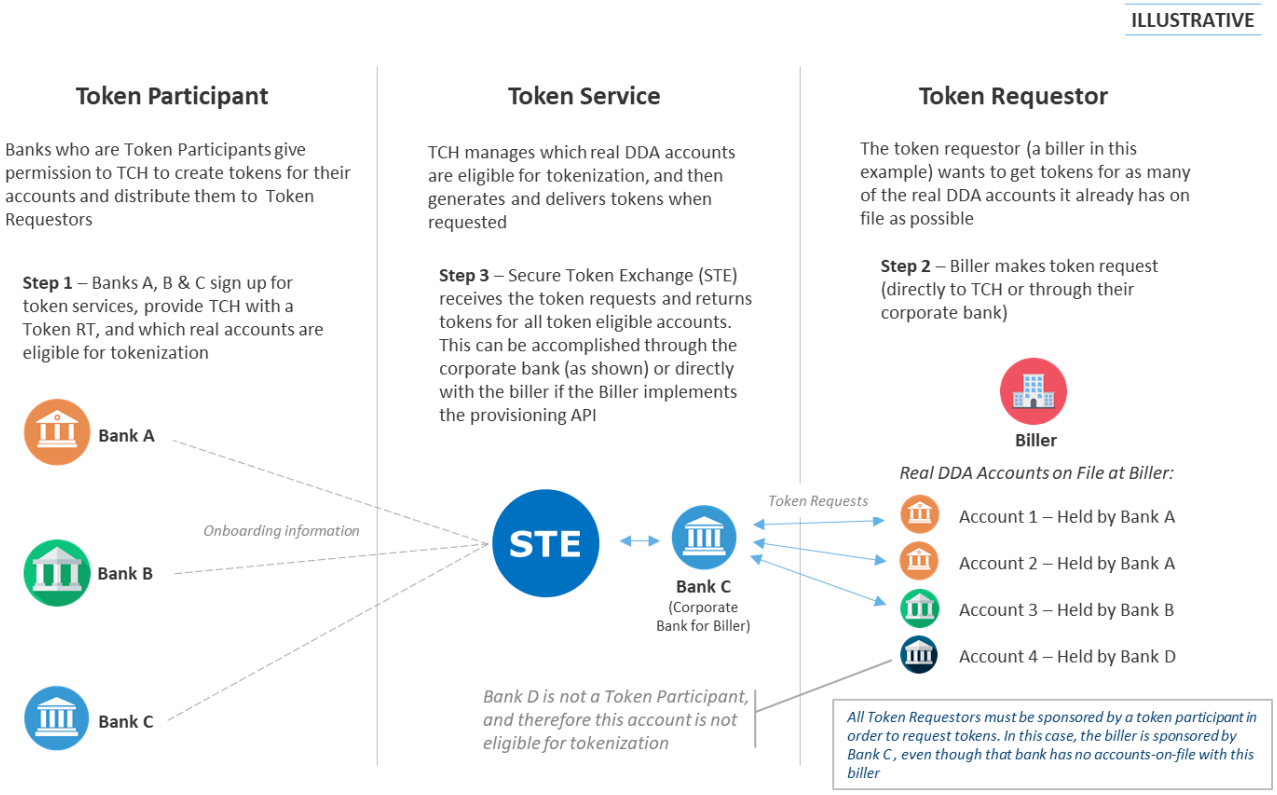
[3] On behalf of provisioning refers to the perspective of the Token Participant that holds the related accounts. Tokens are issued to a Token Requestor without the Token Participant taking any action.

3.1 On-Behalf-Of (OBO) Provisioning

OBO provisioning is the required base token service, i.e., all Token Participants agree to OBO (i.e., to allow TCH to issue tokens for your accounts to any Token Requestor (whether a Participant Token Requestor or Sponsored Token Requestor). It does not require active involvement from Token Participants beyond basic onboarding (i.e., providing a Token RT and which real accounts are eligible for tokenization).[4] The low barrier to entry for OBO provisioning makes participation in the Token Service possible for banks of all sizes and profiles, and it acts as a stepping-stone for other active provisioning methods described later. Importantly, the OBO process provides a means to grow the token ecosystem by increasing the number of Token Participants which in turn increases the value of participation for Token Requestors.

If a Token Requestor requests a token for any token eligible account, then the Token Service will generate and deliver the token to the Token Requestor without interaction from the account-holding Token Participant. An illustrative diagram below provides a high-level overview of the process:

Figure 2: On-Behalf-of Provisioning



3.2 Direct Provisioning

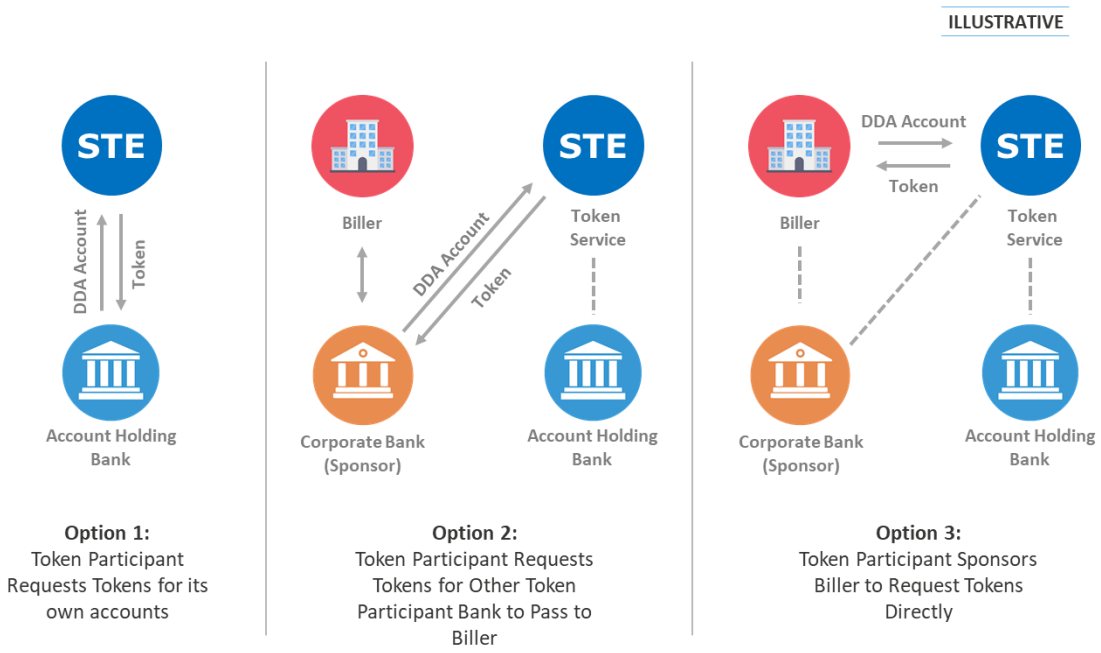
The use of the STE-DDA Provisioning API allows tokens to be requested directly from STE, hence, the "direct provisioning" moniker. Several benefits are derived from API based provisioning. First, tokens can be individually configured and assigned. This allows the Token Participants to assign tokens to a certain Token Requestor (such as the bank's customer), or to have restrictions placed on a single token. Second,

[4] Token Participants may wish to utilize lifecycle management functions for Tokens issued for their accounts.

the API can be used to request tokens for either a Token Participant's own accounts, or that of any other Token Participant. This allows banks the ability to offer token services to their own clients, and white label those services if desired.

In addition to direct provisioning to a Token Participant, a version of the provisioning API can be extended to a Sponsored Token Requestor, who can then request tokens directly from STE. These different options are illustrated below:

Figure 3: Direct Provisioning Options



3.2.1 Option 1: Token Participant requests tokens for its own accounts

The simplest form of direct provisioning involves a Token Participant using the "STE-DDA API Specification for Token Requestors" to request tokens for their own accounts. When Token Participants request tokens for their own accounts, they have a couple options of how to configure the token:

- Token Requestor Specific – Many tokens are assigned to whatever entity will store and use the tokens for payments. This may include customers of a Participant Token Requestor or other entities (such as billers, disbursers, aggregators, etc.). Each of these entities must be added in STE prior to an API based provision. STE only needs the name of a Token Requestor to add them into the database. Once added, STE will provide Token Participants with a "Token Requestor ID" (TRID) that can be used to identify that Token Requestor in the API. While optional, the benefit of linking a Participant's own token to a specific Token Requestor and assigning the token to a TRID is that if there is a breach or hack of that Token Requestor all tokens that have been issued to that Token Requestor can be easily identified and suspended or deleted if needed.
- General-Use Tokens – If the Token Participant does not have a specific entity to assign the token to, then the token is considered a "general-use" token and assigned to the bank itself. General-Use tokens *cannot* have counterparty restrictions but *can* define interoperability settings (described later in this document). Examples of potential use cases for general use tokens include a Token Participant providing a token to their consumer customer to use instead of their real DDA. The consumer might share the token with multiple parties undermining the ability to assign a single TRID. Another example

would be a UPIC-like usage where a bank requests a general use token with a debit block for a corporate client who then shares that token with multiple business partners for accounts receivables purposes.

3.2.2 Option 2: Token Participant requests tokens for another Token Participant's Accounts

In addition to their own accounts, Token Participants can request tokens for the token eligible accounts of any other Token Participant. This is primarily used where the Token Participant requesting tokens wants to offer token services to a corporate client who holds real DDA accounts on file (e.g., biller, disburser, aggregator, etc.) and wants to swap those real DDAs for tokens. The corporate client provides their Token Participant with a list of real DDAs the client already holds on-file and then the bank would use the API to request tokens for each of those accounts. If one of the real DDA account numbers belongs to another Token Participant, and the RT and account range have been made token eligible by the other bank, then STE will generate and return a token, which can then be passed to the corporate client to store instead of the real DDA.

When requesting tokens for another Token Participant's accounts, the requesting bank must identify that customer to TCH prior to the first request. STE will provide a TRID for that customer and the Token Participant must include that TRID in each token request. No general use tokens are allowed for other banks.

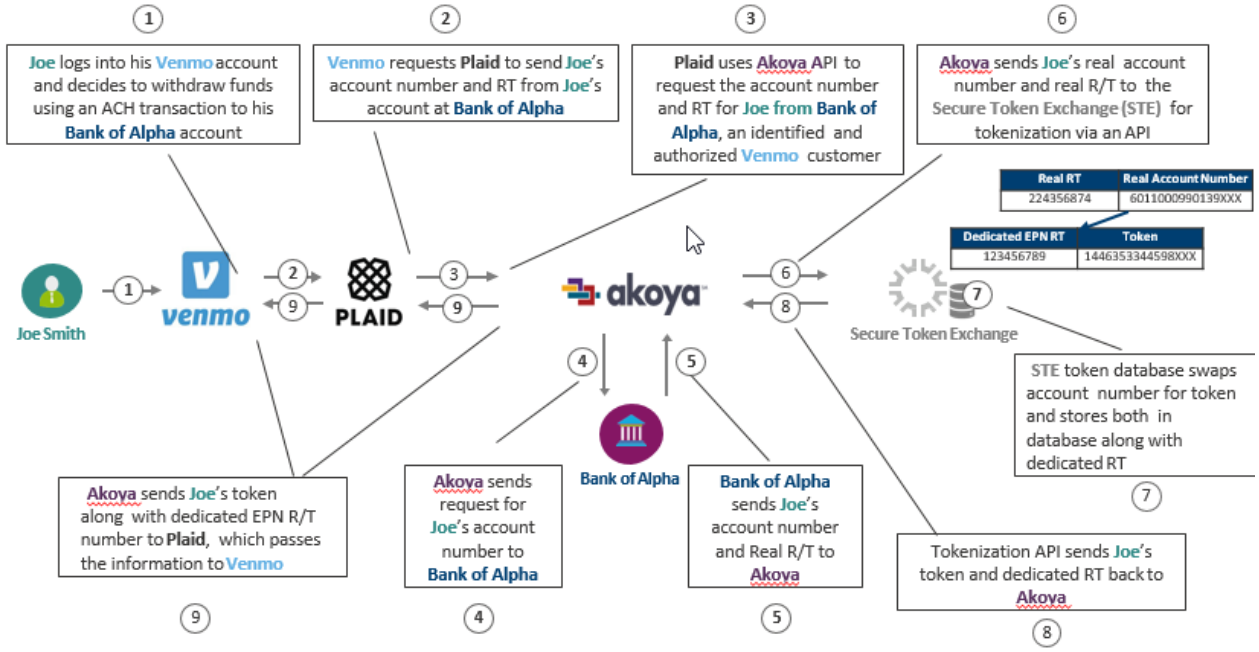
3.2.3 Option 3: Token Participant Sponsors Token Requestor to Request Tokens Directly

The final option for Direct Provisioning allows a Token Requestor, e.g., a biller, to have direct connectivity to STE. This allows the biller (or other Sponsored Token Requestor) to make token requests themselves, but it does require the Token Requestor to implement the provisioning API: the "STE-DDA API for Token Requestors". This option is preferrable when the volume of requests made by the Token Requestor is high enough to merit the direct connection, or if better performance is required (by eliminating the Token Participating bank as an extra leg in the token request). Token Requestors making a direct connection to STE must be a customer of a Token Participant and sponsored by that Token Participant, a step that ensures that there is some bank oversight. Token Participants interested in sponsorship can request the "STE Token Requestor Sponsorship Form".

3.3 Provisioning via a Third-Party Token Provider

Token Participants can designate an agent referred to as a Technical Agent to request tokens from TCH on their behalf. Depending on the agreed upon arrangement, the Technical Agent may also act as a pass-through to hand off the Token to the end Token User. In this model, both connectivity to TCH, and implementation of the "TCH Token Requestor API" become the responsibility of the Technical Agent instead of the Token Participant as the agent performs this function for the Token Participant. In its role as an agent, the Technical Agent will be responsible for obtaining the Token Participant's real DDA account number prior to sending a request to STE for a Token for that account.

Figure 4: Provisioning via Akoya



3.4 In-network Provisioning *(RTP Only)*

In-Network Provisioning creates the token as part of the RTP transaction flow. When the RTP system receives a RTP message with the In-Network Provisioning flag, a call is made to STE to provision a token or look up a previously provisioned token. The RTP message (either a Credit Transfer (pacs.008) or Request for Payment (RFP) (pain.013) or Remittance Advice (remt.001)) will then be tokenized by replacing the Sending Participant's RTN and account number with the Sending Participant's token RTN and a token, thus masking the Sender's account information in the outbound RTP message to the Receiving Participant.

In-Network Sender provisioning prevents the Receiving Participant from obtaining the real RTN and DDA account information for the Sending Participant's Customer. By using the In-Network provisioning flag in a RTP message it sends to the RTP system, the Sending Participant is instructing the RTP system to replace the Sender's account information with a token in the outbound RTP message that the system sends to the Receiving Participant. The In-Network provisioning process does not require a separate provisioning connection (API connectivity) with STE.

To use the In-Network provisioning functionality, the Sending Participant sends a Credit Transfer (pacs.008) or a Request for Payment (pain.013) or Remittance Advice (remt.001) message to the RTP system with the Sender/ Message Sender routing number and customer account number. The Sending Participant includes the code "PRTK" (the In-Network Provisioning flag) in the "Instruction for Creditor Agent" field. This code "flags" the request for the RTP system to tokenize the Sender/ Message Sender account and routing and transit number before sending the outbound message to the Receiving Participant.

3.4.1 Instruction for Creditor Agent Codes

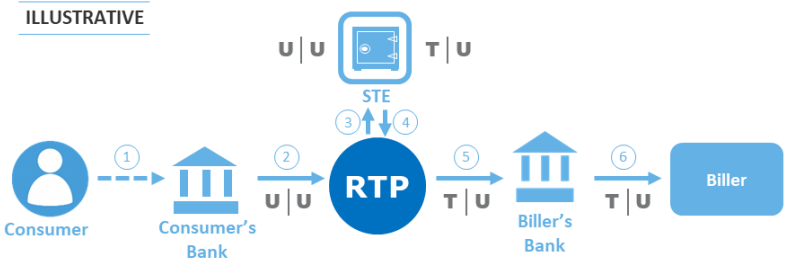
The codes listed below are used with the Instruction for Creditor Agent field in the RTP payment message and Debtor Agent field in non-payment messages (request for payment and remittance advice messages). These fields are used with both the In-Network Sender Tokenization and RTP Account Detokenization services:

- **PRTK** – *Sender Token Requested* – This code is sent by the Sending Participant (or Message Sending Participant) to indicate an In-Network Sender Tokenization request. This is a request from the

Sending Participant to tokenize the Sender/ Message Sender account and routing and transit number before sending the outbound message to the Receiving Participant (or Message Receiving Participant).

- **TOKN** – When a token that was provisioned through Direct Provisioning is being detokenized in an outbound ("leg 2" in the message flow below) Credit Transfer (Creditor Account ID) or an RFP or remittance advice (Debtor Account ID) message, the TOKN code will be present in the message and the Instruction Information field will be populated by the RTP system with the Token that was used in the inbound RTP message ("leg 1" in the message flow below). *Note: in a message that is being detokenized, in the outbound/ leg 2 message the Creditor/Debtor Account ID will carry the actual Account Number.*

Figure 5: In-Network Provisioning



1. *Consumer uses bank application to initiate a payment to a Biller. The consumer account is initially un-tokenized (Illustrative, as the consumer is likely simply selecting their account in the bank application)*
2. *The Consumer's Bank sends a transaction to the Network, and sets a flag to request an in-network token be used to replace their consumer's account information*
3. *RTP® network recognizes the flag requesting a token for the Consumer account and calls out to STE to generate a token*
4. *STE returns a token for the Consumer's account*
5. *RTP® network replaces the real account information with the token for the Consumer account and forwards to the Biller's bank*
6. *Biller receives credit and Consumer's tokenized account number*

Legend:

Consumer's account on left	T | U	Biller's account on right
T	Tokenized RTN and Account number	
U	Un-tokenized (Real RTN and Account number)	

Token Usage	
Consumer Account	**Biller Account**
Tokenized	Non-Tokenized

4 Transactional Processing with Tokens

Due to the differences between the RTP and EPN (ACH) networks, transactional processing with tokens is similar but not identical on the separate networks.

4.1 RTP Processing

In addition to the functions of In-Network provisioning (described above), several token processes are performed during each transaction:

- **Token Identification** – RTP looks for both a Token RTN and for the presence of an In-Network provisioning flag. If either is found, RTN and Account details from the message are sent to STE for processing (tokenization/detokenization/validation) as appropriate.

- **Detokenization** – For a credit transfer (pacs.008), request-for-payment (RFP) (pain.013), or remittance advice (remt.001), if a token is present in the field for the receiver account, STE performs detokenization and places the real account number in the Creditor Account ID and the token that was used in Leg 1 in the Instructions for Creditor Agents field using the TOKN code. The Token RTN is also replaced with the Receiving Participant's RTN associated with the account number.

  When tokenized, STE does not detokenize the sender account since it is intended for that token to be passed to the receiver.

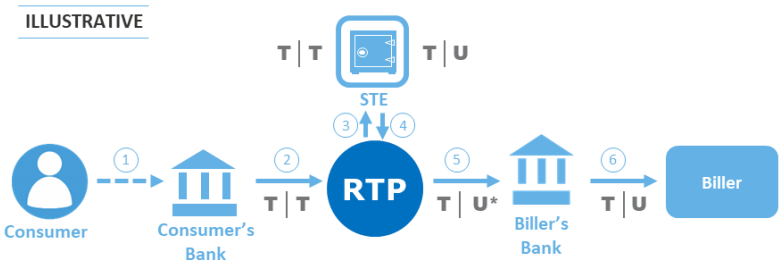
- **Validation** – STE validates the token status (active, suspended) and any (optional) counterparty restrictions placed on the token. Each Token Participant has two options for how to configure validation failures:

  - *Hard reject* – The RTP system will reject a Payment Message or Message that includes a Token when the Token does not exist in the Token Service or been suspended or deactivated, the Token is used in a manner that is not consistent with its interoperability settings or counterparty restrictions or the Token Service technology that is used to validate Tokens is unavailable. A token validation failure error code is returned to the Sending Participant or Message Sending Participant.
  - *Soft Reject* – The RTP system indicates that a token is suspended, or the counterparties do not match (if applicable) and sends a validation error code to the Receiving Participant who can determine whether to accept or reject the transaction. This option is only advantageous if the Receiving Participant can perform its own validations during its allotted network processing time. Otherwise, the hard reject option should be chosen.

Note: All sender tokens that fail validation result in hard rejects. Soft rejects are available only for tokens representing the receiving account because the Receiving Participant has an opportunity to evaluate the message prior to accepting or rejecting. For specifications on validations please refer to the "STE-DDA API Specification for Bank Participants"

Each of the functions mentioned (token identification, detokenization, validation) are performed in real-time during the RTP network's allotted processing time for each message. Any changes to token status or the creation of new tokens are updated in near real-time, allowing RTP processing to have access to the most current token information.

Figure 6: RTP Transactional Flow



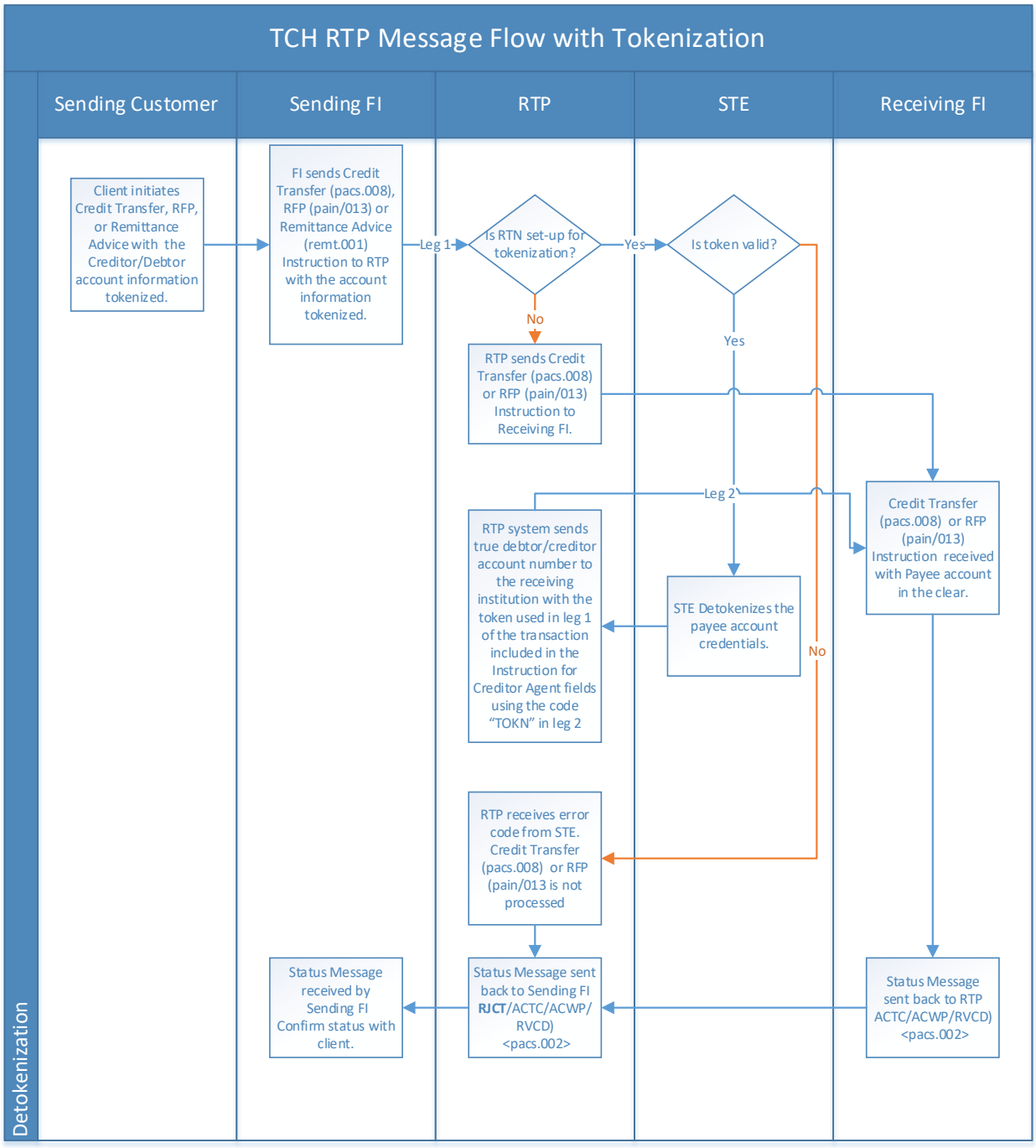
1. *Consumer uses bank application to initiate a payment to a utility company. Both the Consumer account and Biller accounts are represented by tokens (Illustrative, as the consumer is likely simply selecting their account in the bank application)*
2. *The Consumer's Bank sends a transaction to the Network containing both tokens*
3. *RTP® network recognizes both accounts as tokens and calls our to STE for detokenization of the Biller account*
4. *STE detokenizes the Biller account to facilitate processing by their bank and validates the Consumer's token, but leaves it in place*
5. *RTP® network forwards the Consumer's token. *Biller's Bank receives both the token and original DDA account information for the Biller*
6. *Biller receives funds, with token representing the Consumer account*

Legend:

Consumer's account on left	T | T	Biller's account on right
T	Tokenized RTN and Account number	
U	Un-tokenized (Real RTN and Account number)	

Token Usage	
Consumer Account	**Biller Account**
Tokenized	Tokenized

Figure 7: RTP Token Transaction Decision Matrix



FinCEN's "Travel Rule" Implications for RTP Tokens:

- FinCEN's "Travel Rule" requires sending banks to include the sender's true account number in payment messages subject to the rule (B2B RTP payments over $3,000)
- Design for RTP Tokenization allows banks to designate when to tokenize the sender's account
- Banks should not utilize tokens for their own accounts (which will represent the sender) if originating a B2B transaction over $3,000

4.1.1 RTP Token Status and Error Codes

Token Status/Errors Codes – The following new Status/Error codes have been introduced to the RTP message set in support of DDA tokenization. Please refer to the "RTP Message Specification 2.9" for additional details on the use of error codes.

Figure 8: RTP Token Error Codes

Code	Used by TCH / FI or Reserved for Future	Explanation
TK01	TCH	Invalid Token
TK02	TCH	Sender Token Not Found
TK03	TCH	Receiver Token Not Found
TK04	Future	Token Expired
TK05	Both	Token Found with Counterparty Mismatch
TK06	Future	Token Found with Value Limit Rule Violation
TK07	Future	Single Use Token Already Used
TK08	Both	Token Suspended

4.2 EPN Processing

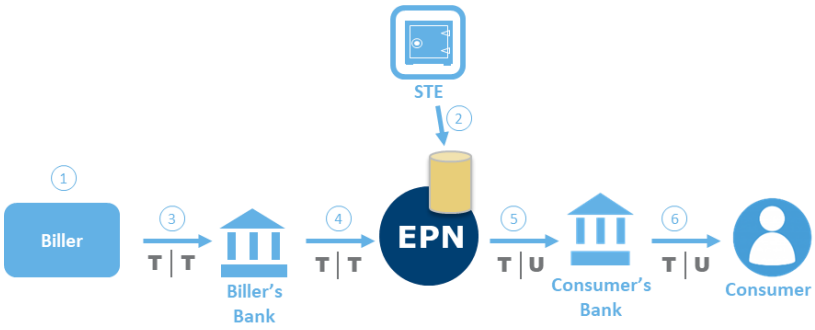
As a batch-oriented network, EPN has different token processes than the RTP system. It also has different constraints, the most obvious being a lack of available fields for token information. The unique properties of ACH processing were taken into account as token methods were designed for the network, and in some cases result in different token functionality than is available on the RTP network.

Four times throughout each EPN processing day, STE updates a token file maintained within the EPN network. The EPN token file contains token mapping (real DDA and token pairing) and token status (active, suspended, deleted). If a new token is created, or the status of an existing token changes, that information will not be reflected by EPN until the next token update file is received at the next set time that EPN processing day.

- **Token Identification** – EPN identifies tokenized Entries based on the presence of a Token RTN. EPN will add a three-character flag ("Token Flag") to the company discretionary data field of the Batch Header Record to identify the tokenized Entry. The Batch Header that contains tokenized forward payments will contain the characters "C9T" in company discretionary field, positions 38-40. For IAT transactions the C9T will be in IAT Indicator field, positions 18-20.
- **Detokenization** – If a Token RTN is found during processing, the EPN Token database detokenizes the RDFI's account (by replacing the Token with the account number and the Token RTN with the RTN associated with the account number) for delivery to the Token Participant with other ACH files distributed to that RDFI through EPN. Detokenization enables the RDFI to process the Entry normally based on the real DDA account information.
- **Validation** – EPN validates that the token is in an active status. The network will reject Entries where (i) the account number field includes a Token that is deactivated ("unlinked") or suspended, or a

number that is not an active Token, or (ii) the Entry attempts to use a Token in a manner inconsistent with the interoperability settings the Token Participant selected (e.g., to debit an account using an ACH credit only Token). EPN does not validate the counterparty restriction on a token because the ODFI account is not present in ACH transactions. Additional information about Token exceptions (i.e., Operator returns/rejects) appears below.

Figure 9: EPN Token Transactional Flow



1. *The Biller has previously provisioned a token for the consumers account and now holds that token on file*
2. *Prior to the next processing window, STE sends an update file to EPN containing the new token*
3. *The Biller initiates a debit transaction using their primary corporate bank*
4. *The Bank originates a transaction into the EPN network with tokens to represent both the consumer and biller*
5. *Using the token mapping database, EPN detokenizes the consumer's token and places the real DDA account information into the outbound file*
6. *The consumer bank receives the debit transaction and is able to process the ACH file using real DDA account information*

ILLUSTRATIVE

Legend:

Biller's account on left T | T	Consumer's account on right
T	Tokenized RTN and Account number
U	Un-tokenized (Real RTN and Account number)

Token Usage	
Consumer Account	**Biller Account**
Tokenized	Tokenized

Additional information pertaining to the use of tokens on EPN:

- ACH entries are not subject to the "Travel Rule"
- Tokenized transactions can originate on either EPN or the FedACH ACH services. Tokenized transactions originating through FedACH will route to EPN as a result of the Token RT being set as an EPN endpoint.

4.2.1 EPN Returns (by Token Participants)

EPN will use the code **"C9T"** placed in Batch Header to identify a Token return. When an RDFI (the Token Participant) returns an Entry that had been tokenized, it must include the three-character Token Flag in the company discretionary data field of the return Entry. When the three-character Token Flag is present, EPN will recognize that the original Entry had been tokenized. The history record created from the forward transaction will be used to identify the original Token and Token RT and substituted in the Return Detail. EPN will identify and use the original forward payment information when processing tokenized returns.

- If the RDFI/Token Participant does not include the three-character Token Flag, EPN will process the return using the Receiver's account number and related RTN, rather than the Token and Token RTN.

- If a token is changed from an active state to a suspended or deleted state after a transaction is processed, TCH will still process the return even though the token is suspended or deleted. This allows normal returns processing.

The forward transaction will be matched using the following fields provided in the Return:

- Original Batch Effective Entry Date
- Original ODFI RT number
- Original Trace Sequence number
- Original payment amount
- RDFI Account Number
- RDFI RT Number

An ODFI return will be matched using the following fields provided in the Return:

- Token Account Number
- Token RT
- ODFI RT Number
- Original Batch Effective Entry Date
- Trace Sequence Number
- Original Amount

RETURNS, CONTESTED DISHONORED, ACKNOWLEDGEMENTS, NOC *(See Exception Processing section)*	The actual RTN in the payment detail and trace number will be replaced with the TOKEN RTN. The account number will be replaced with the TOKEN.
DISHONORED RETURN, REFUSED NOC	The TOKEN RTN in the payment detail and trace number will be replaced with the actual RTN. The TOKEN will be replaced with the account number.
ALL RETURNS FOR TOKENIZED PAYMENTS MUST BE INITIATED THROUGH EPN **ALL RETURNS MUST BE ACCOMPANIED BY "C9T" TOKEN FLAG INDICATING A TOKEN WAS USED**	

4.2.2 EPN Exceptions (Operator Returns/Rejects)

EPN will reject Entries that contain Tokens (as well as Returns related to a forward Entry containing a Token) in certain circumstances identified below.

Forward Entry Exceptions

- A TOKEN/TOKEN RTN cannot be found, or TOKEN not active (i.e., suspended or deleted). Return Reason Code R04
- A transaction is a credit and the TOKEN is Debit Only. Return Reason Code R20
- A transaction is a debit and the TOKEN is Credit Only. Return Reason Code R20

Return Exceptions

EPN will reject the RDFI/Token Participant's Return where the original forward transaction cannot be identified from the Return. This will occur if the Token Participant fails to include the three character token flag in the return of a forward entry that had included a Token.

- Returns, NOCs, Contested Dishonored Returns that are not IAT transactions are returned with Reason Code R69
- Returns, NOCs, Contested Dishonored Returns that are IAT transactions are returned with Reason Code R26
- Dishonored Returns, Refused NOCs are returned with Return Reason Code R04

NOC transactions that use the correction code C01, C02, C03, C06, or C07 are not allowed. These are returned with Return Reason Code R26

Returns sent with Return Reason Codes R02, R03, R04 will appear on an exception report at TCH, and the bank owning the TOKEN will be contacted.

Monitor Error Messages:

- R04 INVALID - TOKEN ACCOUNT NOT FOUND (RETURN)
- R04 INVALID - TOKEN ACCOUNT NOT ENABLED (RETURN)
- R69 TOKEN RETURN ACCOUNT NOT FOUND (RETURN)
- R20 TOKEN TRANSACTION CREDIT ENTRY NOT ALLOWED (RETURN)
- R20 TOKEN TRANSACTION DEBIT ENTRY NOT ALLOWED (RETURN)
- R26 INVALID REASON FOR TOKEN TRANSACTION (RETURN)
- R26 IAT TOKEN RETURN ACCOUNT NOT FOUND (RETURN)

5 Token Interoperability between RTP & ACH

Entities that hold real DDA accounts on file such as billers, disbursers, aggregators etc., want to be able to swap real DDA account for tokens. When tokenized, these entities have the desire to continue to present payments for these accounts into multiple payment networks (e.g., RTP and ACH). Whether a token will result in a completed transaction on any given network is determined in part by "token interoperability." With respect to use of Tokens in ACH, interoperability also refers to whether the Token can be used for ACH credits, debits or both.

5.1 Interoperability Considerations

TCH defaults all new tokens to full interoperability between RTP and EPN. While full interoperability is generally preferable, there are considerations that may lead Token Participants to define their own token programs to *not* enable interoperability for all tokens. The primary considerations include balancing the following two factors:

- **Desire for seamless token holder experience** – Today, billers, disbursers, and other entities that hold DDA accounts on file store a single instance of the real DDA account that can be used across networks. Merchants expecting that a token can be used on both RTP and ACH will have a negative experience if a transaction is rejected because the token is not accepted on one network or the other. *Note: To aid the merchant or other party in knowing how a token can be used, TCH requires the Token Participant who delivers the token to another party to notify that party of the initial interoperability settings (RTP, EPN or combination thereof) and any restrictions that are placed on the token.*
- **Validation of token controls & restrictions** – TCH's current token architecture allows for the generation of counterparty restrictions (and can enable other future restrictions including single use, cryptograms, etc.). These restrictions can be checked and validated during processing on RTP, but EPN tokenization does not support token restrictions beyond token status and the distinction between debits or credits. Thus, if a Token with counterparty restrictions is used in an ACH entry, EPN will not check the counterparty restrictions or reject the entry due to those restrictions.
    a. If a token does not have any restrictions (aka a non-restricted token), then this issue is avoided
    b. It is possible to have a token with restrictions that are enforced during RTP processing, but not checked/enforced during EPN processing.
- **Network Participation** – The ability for tokens to interoperate is dependent on (i) the Token Participant also being a Participant on both the RTP and EPN networks and (ii) enabling the Token RT on both networks.

5.2 Solution Overview

The STE system was designed to balance these competing needs while providing flexibility for individual Token Participant input. This was achieved by establishing interoperability "programs" determined by the provisioning method used to create the token. Core principles in that design include:

- Three interoperability options –
    - RTP acceptance – On/Off
    - EPN Debit acceptance – On/Off
    - EPN Credit acceptance – On/Off
- <u>All tokens default to full interoperability (all set to "On")</u>
- Every token will belong to one of the following interoperability programs:
    - **In-network provisioning** (RTP Only) – All tokens created via in-network provisioning are assigned to the RTP Network TRID. They are defaulted to non-restricted, fully interoperable tokens. However, the bank has the option to modify their interoperability setting for <u>*all*</u> of their own in-network generated tokens, i.e., interoperability cannot be changed at an individual token level for in-network tokens.

- **General-Use** (Tokens requested via API by the account-holding-bank) – Because these tokens are issued to the Token Participant in its role as a Participant Token Requestor (not a separate Token Requestor) they are assigned to that Token Participant's unique TRID. They are defaulted to full interoperability, but a Token Participant can modify the setting if needed. All tokens general-use tokens issued to a Token Participant for its own accounts will have the same interoperability settings.
- **Individual Token Requestor Program** (multiple) – All other tokens are created via API based direct provisioning & assigned to a single Token Requestor (by a TRID). Each separate Token Requestor with a direct connection to STE will have its own token program. These tokens are also defaulted to full interoperability.
    i. *Bulk-swap* – Tokens created under a bulk replacement for accounts already on file will be set to full interoperability, but can be modified later by token requestor or at an individual token level

5.3 Interoperability Management

As mentioned previously, the default setting for all tokens is full token interoperability. If full interoperability is not desired, then a bank can request that TCH modify the settings for any of the different interoperability programs:

- One interoperability setting for all tokens provisioned via the In-network (Sender) methodology (TRID is assigned to RTP Network)
- One interoperability setting for all general-use tokens that are provisioned directly between STE and a bank (TRID is assigned to bank)
- Separate interoperability settings for each Token Requestor with a unique TRID

Initially, Token Participants will need to email the TCH Token Product group to request changes to interoperability settings. This is meant to reduce the chance of a bank inadvertently causing rejections by turning off their token acceptance at one or more networks.

Changing interoperability settings after a token has been issued will result in the following actions:

- For RTP:
    - Turning off RTP acceptance for a particular interoperability program will disallow all tokens associated with that program to process on the RTP network. During transactional processing, STE will flag a token not eligible for RTP processing and the RTP system will then reject the transaction.
- For EPN:
    - If a new token is added to STE-DDA, it will appear in the next delta file (prior to next update time described above in Section 4.2) as new with the appropriate flags (Credit/Debit) set
    - If one option (credit or debit) was turned off but the other remained on, then the next delta file would provide an update, and the token will remain in the database with the one remaining interoperability option
    - If BOTH options (credit and debit) are turned off, then the next delta file would update EPN to remove both interoperability options. This is mean that transactions containing that token will be rejected, but the token itself will remain in the EPN database

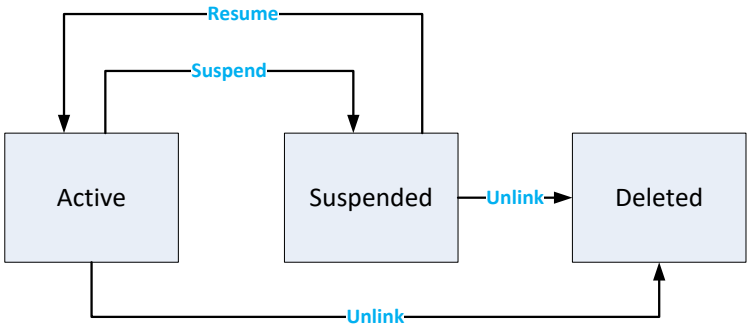
- If either option (credit or debit) was turned back on after both had been turned off, then the next delta file would once again include a record for that token and EPN would update the record to reflect the new settings

6 Lifecycle Management (LCM)

During the provisioning process, a new token is generated (in an active state) and linked to a real account number. Token Participants can change a token state of their own tokens via the customer service portal or the "STE-DDA API for Bank Participants". An active token can be suspended or unlinked from the real account number. A token in a suspended state can be re-activated (resumed) or unlinked from the real account. Unlinking a token permanently disables the token and is non-reversible.

Once an account is unlinked from a Token, the Token can no longer be used for payments and cannot be reissued to any party for 20 years. After 5 years, the unlinked Token's associated real account information is deleted. The Token is retained in the vault for an additional 15 years for audit, compliance or clearing and settlement purposes.

Figure 10: Token Lifecycle



The STE-DDA Customer Service Portal allows Participants to perform a search to retrieve the Token and account information associated with a real account number or tokenized account number. If the search is successful, a listing of tokens is displayed, the Participant can then select and change the Token's state.

A Token may have the following lifecycle states:

- Active – Token is provisioned and ready for use
- Suspended – Token is not active and any messages received in which the Token is used will be rejected
- Unlinked – Token is no longer associated with the underlying DDA account and is retained only for reporting purposes. Any message received in which an unlinked Token is used will be rejected

To manage these states, Participants can use the STE-DDA Customer Service Portal or APIs to perform the following actions:

- Inquire – A request for information about the status of a Token
- Suspend – Command to place a Token in the suspended state
- Resume – Command to move a suspended Token into the active state
- Unlink – Command to move a suspended Token into the deleted state

Token lifecycle can always be managed by the account-holding bank, but also by the Participant Token Requestor or Sponsored Token Requestor. The options are:

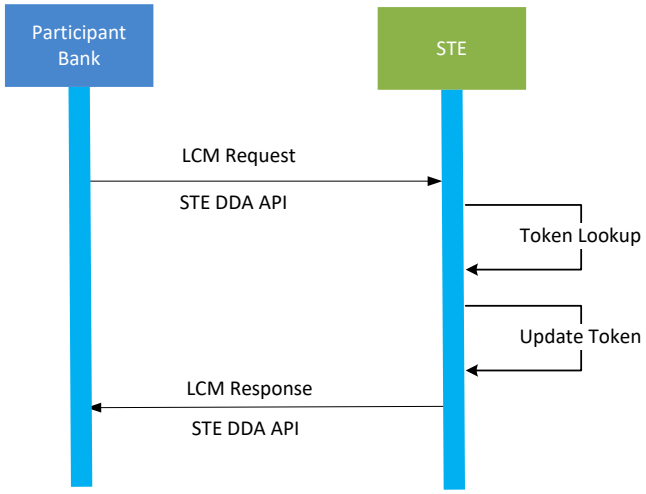
- a Token Participant can suspend, resume, and deactivate Tokens for its own accounts that have been issued to it in its role as a Participant Token Requestor;
- a Token Participant can suspend, resume, and deactivate Tokens for its own accounts that have been issued to another Token Requestor;
- a Participant Token Requestor can suspend, resume, and deactivate Tokens for another Token Participant's accounts that have been issued to it; and
- a Sponsored Token Requestor can suspend and resume Tokens for a Token Participant's accounts, except that a Sponsored Token Requestor cannot resume a Token that a Token Participant suspended under section VII.B.2.b

6.1 Lifecycle Management API

Lifecycle Management functions and account updates can also be performed using the STE DDA API. The STE DDA API messages can be used to manage Token state and inquire Token status.

The API is described in detail in the STE-DDA API specification document.

Figure 11: LCM API Flow



6.2 STE-DDA Customer Service Portal

The STE-DDA Token Customer Service Portal offers a user interface for Participants to identify and manage the current state of their own tokens. The portal also provides the complete history of the Token (audit trail), which shows all state changes, tokenization and detokenization operations applied to a Token. *See the STE-DDA Customer Service Portal Operator Manual for additional information and instructions on how to use the portal.*

6.2.1 Portal Onboarding

Onboarding onto the portal will be a part of general tokenization onboarding processes.

6.2.2 User Configuration

STE-DDA Customer Service Portal users will have access to real accounts and tokenized accounts linked to their 'own' Financial Institution. They will also have access to LCM functions for tokens that they request from other Token Participants. To make authorized user configuration as easy as possible, certain groups and users for each Participant are created automatically during the onboarding process. These groups and users are automatically restricted to view only real accounts and Token accounts for their own newly on-boarded Financial Institution.

Default Authorization Hierarchy

- *User Level 1* - Token information can be searched and retrieved. Audit information for Token state changes can be retrieved
- *User Level 2* - Level 1 role, plus ability to view log information and unlink tokens
- *Administrator* - Level 1 and Level 2 roles and they can create or delete users, modify user rights, and restrict user access

Participants can create and manage their users through the user management menu in the STE-DDA Customer Service Portal. The User Management Menu allows Participants to manage users, password policies and language presented to the user.

6.2.3 Token Search

The STE-DDA Customer Service Portal allows Token Participants to perform a search to retrieve their own tokens and associated account information. If the search is successful, a listing of tokens is displayed on the Tokens Overview page. The following information is displayed:

Property	Description
Token Reference ID	Unique Reference ID of the Account Token
Real Routing	The routing number of the real account numbers
Real Account	The real Account Number
Token Routing	The routing number of the tokenized account numbers
Token Account	Token Account number linked to the real account
Counter Party Routing	The counter party routing number that is linked to the Token
Counter Party Account	The counter party account number that is linked to the Token. If a Token is linked to a counter party account, it can only be used for payments to this account.
Expiry Date	Expiry date of the Token
State	Current lifecycle state of the Token
Last Event	Date/time of last action performed on the Token
Financial Institution	Name of the financial institution
Token Requestor ID	Identifier of the Token Requestor that is associated with the account Token

Requestor	The name of the entity associated with the TRID. This can be a Financial Institution, Sponsored Token Requestor or a Technical Agent (This entity may or may not be the party that is the Token Requestor or Token User under the Token Rules)
RequestedDate	Date/time of the Token request (=tokenization request).
Domain	Name of the domain the Token belongs to.

6.2.4 Viewing the Token History

All state changes, tokenization, and detokenization operations applied to a Token are recorded in an audit trail. To view the Token History, Participants can use the STE-DDA Customer Service Portal to search for a specific Token and select to view all the events that have been performed on the Token.

The following information about the Token event is available in the Token History:

Property	**Description**
User	User name or process name that initiated the event on the Token, a state change, tokenize or detokenize operation.
Timestamp	Date/time when the event on the Token occurred.
Event	Type of event performed on the Token.
By	User or process that initiated the event. Options are FI, TR, CSP, SYSTEM or TXN_API_BROKER.
From state	Original state before the event on the Token.
To state	New state after the event on the Token.
Reason	Reason for the event to happen. Optionally supplied by the user during the event initiation.
*Request Type**	The type of transaction in case of tokenization and detokenization: ▪ CREDIT_TRANSFER ▪ REQUEST_FOR_PAYMENT ▪ REQUEST_REMITTANCE_ADVICE
Currency	The currency of the transaction as per ISO-3166 currency codes specification.
Txn ID	Transaction Identifier as generated by the Participant that sent transaction.
Single Use	Identifier whether the Token is single use or multiple use.
*Amount**	Transaction amount.

**Not applicable for EPN(ACH)*

6.2.5 Updating the Real Account Number

The STE-DDA Customer Service Portal allows Participants to link a new real account number and real routing number to an existing active or suspended Token.

6.2.6 Updating the Counterparty Restriction

For certain Tokens used on the RTP Network, tokens can be linked to a counterparty account, which means that they can only be used for payments or non-payment messages between the two counterparty accounts provided during provisioning. Participants can use the STE-DDA Customer Service Portal to add or update a counter party account number and routing number for an active token.

6.2.7 Distinguishing between Interoperability and Lifecycle Management

It is important to note that while interoperability management and lifecycle management are complementary - and can both prevent unwanted transactions - they are separate tools with separate inputs. For example, a token could remain in an active lifecycle state, but have an interoperability setting of on for RTP, but be turned off for both EPN credit and EPN debit. Changing a token's lifecycle setting to suspend/unlink, or modifying certain interoperability settings will both result in the rejection of transactions containing tokens with those settings where the token is used inconsistently with those settings (e.g., in EPN in when the token is RTP only).

Participant CSP Administrators can use the Customer Service Portal to manage how Token Requestors are permitted to use tokens. The Token Programs settings can be changed by accessing the Configuration menu CSP.

7 Reporting

Primary reporting on token volumes and activity are generated by the STE system. However, reports generated by the RTP and EPN network will also have some token implications:

STE – Token Participants can receive the following tokenization reports:

- *Tokenization Detokenization Active Tokens Report* - This report contains an overview of tokenization, de-tokenizations and active tokens stored in the vault for a RTP Participant.
- *FI Issued Token Account Mapping Report* – Upon direct request to TCH customer service and support, this report will include the real RTN and real account number and all the tokens mapped to that RTN and account number in the event research (e.g., AML) is required.
- *Detokenization Results* – This report contains Token detokenization results for each RTP Participant including the detokenization results (success or failure) and the description of any failures.
- *ACH Active Tokens Report* – This report provides EPN eligible token counts for all tokenizations and active tokens for a selected participant.

RTP – Several reports have been updated to include reporting on token messages. An example of this is the Volume and Value Report. This report will now include additional columns that separate out the volume and value of messages that were processed with a token. Please refer to the RTP Report Specification version 2.10 for additional details.

EPN – EPN will generate a report of failed Token lookups, and failed return payment matching, and a monitor report on tokens that were rejected.

- *Token Transaction Exception report for payment activity errors:*

EPN will generate an internal exception report of failed Token lookups, and failed return payment matching. An exception report will be produced at the end of each exchange. The report will be sent to Synergy report repository.
For details refer to Appendix E: Token Transaction Exception report layout

- *The CIF listing report* has the new Token R/T indicator added. The values for the indicator will be:
  Token R/T = Y or
  Token R/T = N
  For details refer to Appendix F: CIF Listing for the sample report.
- *The monitor report* will display the errors for the tokens that are rejected (suspended/terminated/expired).
  For details refer to Appendix G: Monitor Report.

