Transferable Digital Notes System

Practical Solution for Digitizing Fiat Currency

George Dobrescu
gdobrescu@tdnsys.com
703.568.7739
Alexandria, Virginia, USA
www.tdnsys.com

September 8, 2020

Patent Pending 39633946

Update history

03/01/2009	Document the idea of solving the double spending of digital tokens problem using a centralized system and based on token reissuance.
05/01/2010	Document the basic idea and possible applications of Transferable Digital Tokens.
11/01/2014	Defined Transferable Digital Tokens specifications.
03/01/2018	Document possible implementation of currency digitalization.
04/01/2018	Finalize documentation Transferable Digital Promissory Notes.
06/04/2018	File provisional patent application.
02/02/2020	Change the documentation Transferable Digital Promissory Notes into documentation of a digitize form of fiat currency called TDN and a system for processing caked TDNSYS.
06/05/2020	Refile provisional patent application.
08/23/2020	Add documentation for the Demo.
09/0202020	Create a tutorial using the Demo website.
09/08/2020	Last version – distribution.

Introduction

All central banks are currently evaluating the feasibility of issuing what it is called Central Bank Digital Currency or CBDC. The name seems to imply that a new currency will be put into circulation by the central bank. The currency of the Unites States is the US dollar, a fiat currency, and it is less likely that a new currency will be in circulation soon, if ever. To clarify, the Transferable Digital Notes Project is proposing a digital form of the US Dollar, not a digital currency. It is more appropriate to call this effort *digitizing fiat currency* or creating *fiat currency in digital form*.

All efforts to implement CBDC are influenced by the technologies behind what it is called 'crypto or digital currencies' and consists of patching these technologies. It is said that all the problems will be solved over time. Currently, there are mature, proven, state of the art technologies used in the Financial Systems Infrastructure and there is no reason not to use them when new systems are implemented.

There is a finite amount of digital currency based on blockchain/DLT that can be put into circulation, making it very similar to gold and silver coins minted by the government with the exception that they are not legal tender. If the government started issuing Bitcoins and decrees as legal tender, then they would be as useless in economic activities as the Gold Eagles.

A different approach is to create Fed accounts for everybody. This approach has many negative side effects and actually does not solve the digitalization problem. It should be noted that because digital devices and systems are used for processing money transactions, it does not make the money digital in the same way that processing money using mechanical means before computers did not make them 'mechanical money.'

The digital form of US Dollar proposed is called Transferable Digital Note or TDN, and it will be the third form of money in the monetary base beside paper notes/coins and Fed reserve accounts. The system processing TDNs is called Transferable Digital Notes System or TDNSYS.

Creating currency in digital form is contingent on finding a solution to *the double spending problem.* This is a problem that arises when using digital tokens. For example, to reinforce single use licensing, a software company issues a digital key (digital tokens) to the buyer. The problem is that the key can be used on more than one installation, in other words it can be 'double spent'. To prevent double spending a key can be generated based on hardware specific to the machine where the system is to be installed or by using a license server.

Another similar situation arises when using numbered, anonymous bank accounts (if they still exist). The owner of an account can transfer the ownership of the value by transferring the account number. At this point the new and previous owners both have access to the account. To prevent double spending the new owner changes the account number.

There are other situations of double spending and analyzing them helps to find a solution for preventing the double spending of digital tokens and enabling the use of digital tokens as digital currency.

Table of Contents

Table of Figures

Section 1 Currency Digitalization

1.1 Analysis and Basic Concept

Currency digitalization is a process that can be used by a Central Bank for creating and putting into circulation fiat currency in digital form.

Definition

Digital form of fiat currency establishes a relation between a value measured in the main currency unit and fractional units, and some digital data called a digital token. A digital token is a long, unique, and impossible to guess string of bytes. This is similar to a serial number on a note. A serial number is unique but is very easy to guess because these serial numbers are sequential. Digital tokens can be processed as any other digital data. It can be stored on digital devices, processed by programs, stored in databases, and moved over digital networks. It can be printed in some standard encoding. In the TDNSYS the digital token is called TDN Signature. TDN Signature and TDN terms are interchangeable. Here is an example of a TDN Signature printed as ASCII text and as a QR barcode

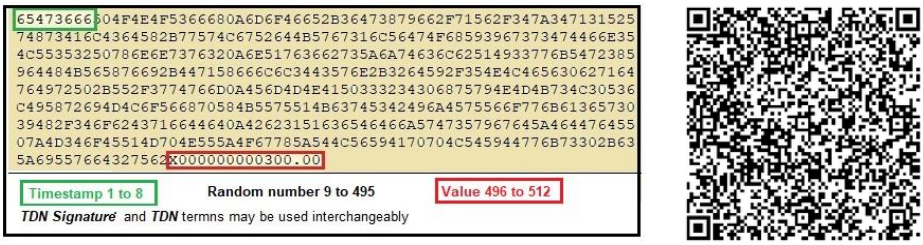


Figure 1 – TDN Signature

Note*:* The TDN Signature is not a coin, account and does not encode any information. It is just a label on an amount of currency. As you will see later, this label can be removed and a new label can be applied to this amount of currency.

In order to find the best way of implementing fiat currency in digital form we have to start with analyzing the initial requirement:

> **REQUIREMENT**
>
> Create a digital form of fiat currency as part of monetary base having properties similar to Fed accounts and cash.

Note: It is necessary to clearly state the objective and requirements of digitizing currency. Most of the time other requirements are discussed even though they are not properties of money and the Fed charter is not required to reinforce. (Ex. payments, liens, seizure, garnish, contacts, etc.)

Definition of fiat currency properties

Note: It is assumed that the reader is familiar with the economics of money and banking.

- ***Issuance***

  Fiat currency issued by the Fed has only two forms:

  a. Fed accounts – Establish a relation between a unique Fed account number and an amount of currency. The Fed guarantees that the owner of an account, which is identified by a unique number, is the owner of the amount of currency recorded in the account balance.

  Note: Bank accounts and credit are not part of the monetary base.

  b. Bills and coins – Establish a relation between a Fed issued paper note or coin and an amount of currency. The Fed guarantees that the person who physically possesses a note is the owner of the amount of currency printed on the note. The note has a unique serial number.

- ***Validation***

  The validity of fiat currency is guaranteed by the *Fe*d. For Fed accounts, the validation is performed by the Fed by balancing the accounts. The validity of a paper note is performed by eliminating the possibility of counterfeiting. This can be superficial in everyday transactions or extensive if necessary.

- ***Ownership***

  Fiat currency always has an owner. Fo*r* Fed accounts, the owner is a legal entity bound to the account number. The owner of a paper note is the entity physically in its possession. If the paper note is sitting in a Fed vault, it does not have an owner as it is not in circulation; it is not fiat money yet. It is interesting to note that when lost, the ownership of a paper note is ambiguous.

- ***Ownership transfer***

  For Fed accounts, the ownership of a sum of fiat currency is performed by the Fed. This is based on the owner's instructions by debiting the owner's account and crediting the new owner's account (changing the accounts' balances). The ownership of a note is transferred by physically transferring the note. It should be noted that the transfer may occur without the owner's consent.

- ***Removing from circulation***

  Notes are removed from circulation when banks send them back to the Fed and have their reserve accounts debited. The notes are destroyed if worn out or stored and later put back into circulation. The Fed can also remove currency from circulation by debiting reserve accounts.

Taking into consideration the properties of fiat currency, we can identify the properties of a digital form of fiat currency and the minimum implementation requirements. Let us call this form of fiat currency Transferable Digital Note or TDN. The digital token associated with it is a TDN Signature and the system processing TDNs is the Transferable Digital Note System or TDNSYS.

- ***TDN Issuance***

  The Fed issues TDNs to member banks. The Fed has a computerized system for generating and keeping track of TDNs. Using mathematical algorithms, the system generates TDN Signatures. For each TDN an entry is created in a repository (a simple database table) containing the TDN Signatures, the associated value, status as Active, the creation date, the '*initial issue'* flag set to TRUE and the bank ID

- ***TDN Validation***

  A TDN is valid and can be used in transactions if the status is Active. If the status is Canceled or Blocked (under investigation) the TDN cannot be used in any transaction. Anybody in the possession of a TDN Signature should be able to query the repository for its status. This can be done using the Fed website or using applications developed using TDNSYS API.

- ***TDN Ownership***

  The owner of a TDN is the entity physically in possession of the TDN Signature. The owner can store the TDN Signature as any other digital data on a digital device or print it on a piece of paper as ASCII text or barcode. Similar to paper notes, the ownership of TDNs can be ambiguous. This happens when two entities are in the possession of the same TDN Signature.

- ***TDN Ownership Transfer***

  TDN ownership transfer is similar to paper notes transfer with the difference being that the transfer can be performed using digital devices and over digital networks. The owner of the TDN can also transfer it by printing it on a piece of paper as ASCII text or barcode. After the transfer, the ownership is ambiguous. To solve the ambiguity and prevent double spending the new owner has to connect to the Fed TDNSYS and request TDN ownership. Applications design using the TDNSYS API perform this ownership transfer instantly. The Fed system will update the status of the transferred TDN to Canceled and create an Active TDN for the same value and return its TDN Signature to the new owner. It should be noted that all currency transfers are based on trust. If the new owner trusts the previous owner, he/she may not need to request the ownership as described above especially for small amounts.

- ***Removing from circulation***

  A bank can redeem a TDN from the Fed. The bank's reserve account is debited for the value associated with the TDN, the status is set to Canceled and the flag '*redeemed*' is set to TRUE.

This equation represents the total value of TDNs in circulation

$$\sum \text{TDN issued} - \sum \text{TDN redeemed} = \sum \text{TDN active} + \sum \text{TDN blocked}$$

Note: An interesting observation is that TDNs have all the properties of paper notes and some of the Fed accounts. *Figure 2- Money transfer comparison* shows a sample of recording ownership transfer for Fed accounts and the same transaction using TDNs. In this example, the sum of $100.00 is transferred from **A** to **B** using reserve accounts and using TDNs. The first two lines in *Figure 5 – TDNSYS repository dump on Pg. 4*, show how this transaction is recorded in the TDNSYS repository. The two recordings are marked with the same transaction id *100000*. This transfer requires two TDNSYS repository operations performed in a few microseconds.

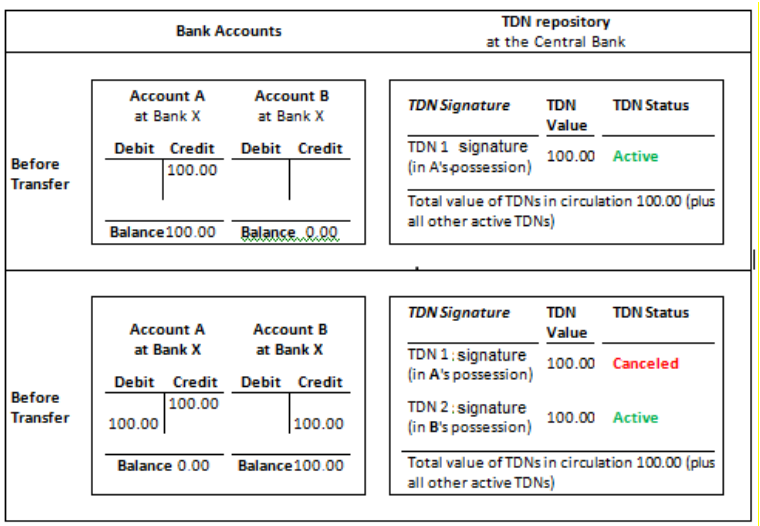


Figure 2 - Money transfer comparison

There are many ways to conduct transactions using TDNs. The simplest way is to connect to the Fed Website and perform transactions as described in *Appendix B.d - Request TDN Ownership pg. 28.*

References: *Specifications pg. 5, TDNSYS API pg.22, Transaction description Request TDN Ownership pg 11.*

Note: It is very important not to confuse creating a TDN with issuing a TDN. A TDN is issued and put into circulation by the Fed through member banks in the same way as paper money. When a TDN is issued, the total value of TDNs in circulation is increased by that TDN's value. The total value of TDNs in circulation is the total value of active and blocked TDNs. Creating a TDN is part of one side of a transaction where TDNs are canceled and created. After that transaction, the total value of TDNs created is equal to the total value of TDNs canceled. The total value of TDNs in circulation does not change

1.2 Extend the system functionality with two more transactions

1.2.1 Split TDN

Splitting a TDN is similar to breaking a paper note into smaller denominations. The difference is that a TDN value can be any dollar or cent amount. The owner of a TDN can request a split of his/her TDN in two TDNs.

When a TDN is split, it is marked in the repository as Canceled and two Active TDNs are created for the total amount of the canceled TDNs based on the owner request.

For example, **A** owns a $100.00 TDN and wants to transfer $10.00 to **B**. First **A** needs to 'get change' and he/she splits the $100.00 TDN into two TDNs, one for $90.00 and one for $10.00. After that he/she transfers the $10.00 TDN to **B**. The lines 6, 7, and 8 in Figure 5 - TDNSYS repository dump on pg 4, show how a split transaction is recorded in the TDNSYS repository. This transaction requires five TDNSYS repository operations performed in a few microseconds.

Note: Later in Chapter 4.1 I will give more details about how TDNs are stored and manipulated. Section 4. Use Cases shows practical uses of TDNs.

References:

Appendix B.d- Demo TDN Split pg. 285, Specifications pg. 5, TDNSYS API Pg.22, Transaction description TDN Split Pg. 10, Request TDN Ownership pg. 11.

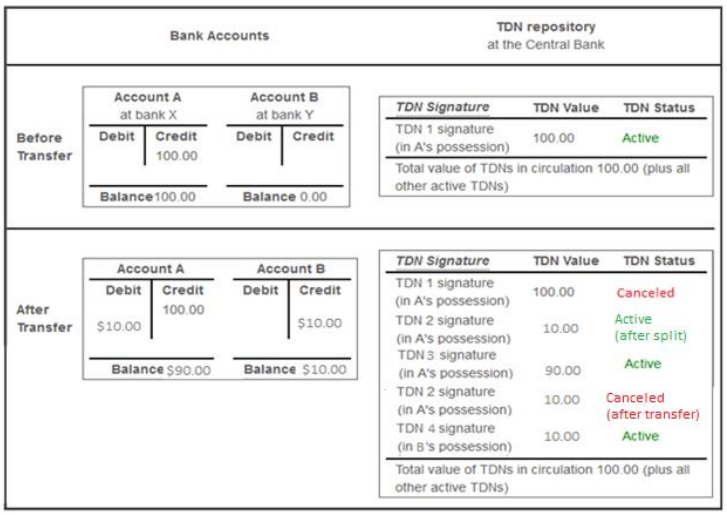


Figure 3 - Split and transfer

1.2.2 Consolidate two TDNs

The owner of two TDNs can request a TDN for a value that is equal to the values of the two TDNs combined. When two TDNs are consolidated, they first are marked as canceled in the repository and a new TDN is created for the combined value of the two TDNs canceled. For example, **A** is in the possession of two TDNs, TDN1 for $90.00 and TDN2 for $10.00 and wants to transfer $100.00 to **B**. First **A** will consolidate the two TDNs into a $100.00 TDN and then transfer it to **B**.

References:
Appendix B.c- Demo Consolidate TDNs pg. 27, Appendix B.d- Demo Request TDN Ownership Pg.27, Specifications. Pg. 5, TDNSYS API pg.23, Transaction description Consolidate TDN pg. 10, Request Ownership pg. 12

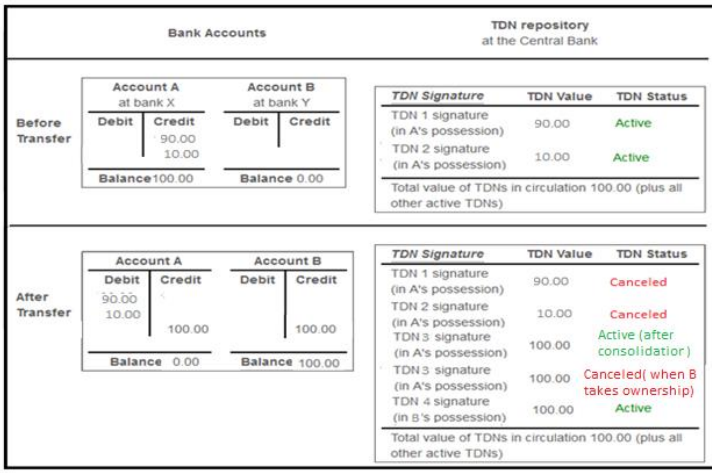


Figure 4 - Consolidate and transfer TDN

Here are how transactions are recorded in the TDNSYS repository:

Status: 0 = canceled, 1 = active, 2 = blocked=

```
tdn_id       tdn_issue_date      tdn_token        status value    Status_chg_date  Transct_ID
------------ ------------------- ---------------------- - -------- ------------------- ------
4A464C593... 2020-06-11 19:07:07 4A464C59344D4E5668... 0 00010000 2020-06-11 07:07:20 100000 Get TDN
624352323... 2020-07-20 11:06:10 64E5668414D0A24352... 1 00010000 2020-07-20 06:06:10 100000 Ownership
6A2B7A337... 2020-06-10 20:07:06 6A2B7A33744A657468... 0 00009000 2020-06-10 07:07:19 100001 Consolidate
6F316B657... 2020-07-19 10:06:19 6F316B657077735834... 1 00010000 2020-07-19 06:06:06 100001 two TDNs
232307591... 2020-06-11 19:07:07 848360A664A464C593... 0 00001000 2020-06-11 07:07:07 100001
A33744A65... 2020-07-20 11:06:11 6848360A6641654262... 1 00001000 2020-07-20 06:06:17 100002 Split TDN
B65707773... 2020-06-10 20:07:06 6A2B7A33744A657468... 1 00009000 2020-06-10 07:07:06 100002
C59344D4E... 2020-07-19 10:20:20 6F316B657077735834... 0 00010000 2020-07-19 06:06:07 100002
4A464C593... 2020-06-11 19:10:07 14D0A5A4462674974A... 0 00030000 2020-06-11 07:07:96 100003 Get TDN
344D4E6F3... 2020-07-20 06:19:10 5A4462674970546243... 1 00000340 2020-07-20 06:11:17 100003 Ownership
64C59344D... 2020-06-10 07:06:06 6A2B7A33744A657468... 0 00053476 2020-06-10 07:20:06 100004
352323075... 2020-07-19 06:07:07 6F316B657077735834... 2 00050000 2020-07-19 06:10:07 100005 Blocked TDN
```

Figure 5 - TDN repository dump

1.3 Extend the system with security

1.3.1 Set PIN

The owner of a TDN set is given a PIN to protect his/her TDN. All the operations on the TDN will require entering the correct PIN in order to complete the operation. The PIN can be changed or removed by the owner of the TDN.

1.3.2 Use Public/Private Key (PPK) Authentication

Setting a PPK is similar to protecting a TDN with a PIN. The owner of the TDN generates a public/private key pair and sets the public key of the PPK instead of a PIN. The validation is performed by exchanging encrypted strings between the owner and TDNSYS. The PPK can be changed or removed by the owner of the TDN.

Public/Private Key (PPK) Authentication does not mean that the owner of a TDN secured this way is not anonymous. The '*Authentication'* refers to the process of verifying that entity initiating a transaction on the TDN is in the possession of the private key paired with the public key.

PKI can be used to identify special TDN owners like banks and merchants. For regular owners, the TDN may be restricted to a maximum value of $500.00. The Fed can issue digital certificates the banks and merchants and the PKI is set to the public key of the certificate. The TDNs with the public key set to a digital certificate will be allowed to have of much larger values.

1.4 Repository record layout

Field	Type	Null	Key	Comment
tdn_id	char(20)	NO	PRI	Usually the first 20 char. of the TDN Signature
tdn_creation_date	datetime	NO		Record creation timestamp
tdn_cancelation_date	datetime	YES		Cancelation timestamp
tdn_signature	text	NO		A unique, impossible to guess long string of numbers associated with a money amount (tdn_dollar_value).
tdn_status	text	NO		Active, Canceled or Blocked
tdn_initial_issue	char	YES		Set to 1 if initial release otherwise to null
Tdn_redeemed	char	YES		Set to 1 if redeemed otherwise to null
tdn_dollar_value	int(8)	NO		The Dollar amount associated with the TDN
tdn_status_timestamp	datetime	YES		Timestamp when status changed
tdn_status_change_by	text	YES		Status changed by by, if available
tdn_pki	char(20)	YES		Private Key of PKI is set or Digital Cert if the owner is a bank or merchant.
tdn_pin	text	YES		Security PIN
tdn_bank	text	YES		If the TDN is owned by a bank
tdn_connection_info	text	YES		If available
tdn_note	text	YES		Optional
tdn_transaction_id	int(11)	YES		Unique number. Groups TDN involved in a transaction.

Figure 6 - Repository record layout

1.5 Specifications

These specifications are generic and they can be applied to other applications. The best application is for fiat currency in digital form.

1. A *Guarantor* is an entity who can legally hold deposits and issue *Transferable Digital Notes* to its customers.

2. A *Guarantor Partner* is an entity who has an account with the *Guarantor*.

3. A *Transferable Digital Note* is the bearer's claim on a specified value deposited at the *Guarantor*.

4. The *Transferable Digital Note* is issued by the *Guarantor* for a value deposited at the *Guarantor (see 7)*.

5. A *Transferable Digital Note* may have a *Maturity Date* or may be *Due on Demand*.

6. A *Transferable Digital Note* has a unique and impossible to guess signature consisting of a large, unique, and impossible to guess numbers and it may be secured with a *Personal Identification Number* (PIN) or a *Private/Public Key (PPK) Authentication*.

7. The *Guarantor* shall receive deposits and issue *Transferable Digital Notes* only to the *Guarantor Partners*. This operation is called *Initial TDN Issuance*.

8. The *Guarantor* shall release the value of a *Transferable Digital Note* only to *Guarantor Partners*. This operation is called *Redeem TDN*.

9. The *Guarantor* may collect and store information about the entity in the possession of the *Transferable Digital Note* if available or supplied by the owner.

10. A *Transferable Digital Note* has a *Status*. The *Status* can be *Active*, *Canceled*, or *Blocked*. Only *Active Transferable Digital Notes* can be redeemed or transferred.

11. A *Transferable Digital Note* has an *Issue Timestamp*.

12. A *Canceled Transferable Digital Note* may have a *Cancellation Timestamp*.

13. A *Blocked Transferable Digital Note* has the *Issue Timestamp* and a *Blocking Timestamp*.

14. The *Guarantor* keeps a database of all issued *Transferable Digital Notes*, their values and status, security information, and any other related data.

15. If requested by the *Owner* of a *Transferable Digital Note*, the *Guarantor* shall cancel the *Transferable Digital Note* in the owner's possession and issue to the *Owner* a new *Transferable Digital Note* for the same *Value* of the *Transferable Digital Note* canceled. This operation is called *Note Reissuance*.

16. If requested by the *Owner* of a *Transferable Digital Note*, the *Guarantor* shall cancel the *Transferable Digital Note* in the owner's possession and issue to the *Owner* two or more *Transferable Digital Notes* for a total *Value* of the *Transferable Digital Note* canceled. This operation is called *Note Split*.

17. If requested by the *Owner* of more than one *Transferable Digital Notes*, the *Guarantor* shall cancel all *Transferable Digital Notes* specified in the owner's request and issue to the *Owner* one *Transferable Digital Note* for the total value of the *Transferable Digital Notes* canceled. This operation is called *Notes Consolidation*.

18. During the transfer of a *Transferable Digital Note*, the new owner takes possession of the *Transferable Digital Note* from the previous owner. If set, the PIN has to be removed before the transfer to the new owner. If a *Private/Public Key(PPK) Authentication* is set, it must be removed before the transaction or the previous owner must give the private key to the new owner. The new owner may check the validity of the *Transferable Digital Note* by requesting its *Status* from the *Guarantor*. To prevent double spending by the previous owner, the new owner may request from the *Guarantor* a *Note Reissuance.*

19. The *Guarantor* shall release information about a *Transferable Digital Note* when presented with the *Transferable Digital Note Serial Number*. If the inquiry originates with a *Guarantor Partner* or an *Authorized Authority*, all information may be released, otherwise only the *Value* and *Status* shall be released.

20. An *Authorized Investigative Authority* may block a *Transferable Digital Note* during an investigation by setting the status to *Blocked*. If the TDN is used in a transaction, the transaction initiator will receive a message showing that the TDN is *Blocked* and instruction on how to contact the *Investigative Authority*.

Section 2 TDNs Transactions

It is very important to understand that we are not talking about payments, money transfers or other transactions performed with money. TDN transactions are simply TDNSYS repository transactions. For example, changing the TDN Signature associated with a money amount. They do not involve accounts or physical entities.

2.1 Examples of operations with TDNs

To better understand TDN transactions, let us take a look at some operations performed with TDNs and how they are recorded in the TDNSYS Repository. It should be noted that these operations can be performed by using the Fed website directly or they can be automated by applications designed using the published TDNSYS API. A person can use some application to complete a purchase or money transfer without being aware that the underlying transactions are based on TDNs. (See Section 4 - Use Cases)

- **Bank X *requests a TDN from the Fed***

  Transaction description:
  Bank X requests a TDN from the Fed. The reserve account of Bank X is debited for an amount equal to the value of the TDN requested. The TDNSYS generates a TDN for this value and records it in the TDNSYS Repository as Active and with the flag '*initial issue'* set tot TRUE. Bank X updates its accounts accordingly and stores the TDN Signature in its systems.
  This is how TDNs are put into circulation by the Fed.

Specification reference	See 1.5 – 7 Initial TDN Issue pg. 5
TDN transaction reference	See 2.4 TDN Initial Issue pg. 8
TDNSYS API reference	See Appendix A – TDN API pg. 22

- **Bank Y *redeems a TDN from the Fed***

  Transaction description:
  Bank Y presents a TDN to the Fed and requests a credit on its reserve account for the value of the TDN. The reserve account of Bank Y is credited for an amount equal to the value of the TDN. The TDNSYS marks it as Canceled and sets the flag '*redeemed'* to TRUE in the TDNSYS Repository. Bank Y updates its accounts accordingly and removes the TDN Signature from its systems or updates the status.
  This is how TDNs are removed from circulation by the Fed.
  The Fed should issue digital certificates to the banks and merchants in order to be able to identify them.

Specification reference	See 1.5 – 8 Redeem TDN pg. 5
TDN transaction reference	See 2.5 - Redeem TDN pg. 9
TDNSYS API reference	*See Appendix A – TDN API pg. 22*

- **Entity A withdraws $100.00 from his/her Bank X account in TDN form**

  Transaction description:
  Bank X splits a TDN with a $100.00 value from a TDN in its inventory, debits A's account for $100.00 and presents A with the TDN Signature. A may or may not request the TDN ownership because Bank X should be trusted not to double spend it and in case it mistakenly does, an investigation will immediately find the problem. A can protect the TDN with a PIN or PPK.

Specification	See 1.5 – 16 Split TDN pg. 5
	See 1.5 – 18 Get TDN Ownership pg. 5
Transaction reference	See 2.6 - Split TDN pg. 10
	See 2.8 - Get TDN Ownership pg. 11
API reference	See Appendix A – TDN API pg. 22
Demo	See Appendix B – Fed TDN Website pg. 25

- **Entity A owns a $100.00 TDN and transfers $40.00 to Entity B**

  Transaction description:
  A splits $100.00 TDN into a $60.00 TDN and a $40.00 TDN, and transfers the $40.00 TDN Signature to B. The transfer can be performed in many different ways, with specially designed applications or simply by using e-mail or SMS. If the $40.00 TDN is protected by a PIN or PPKPPK, A has to either remove the PIN or PPK or send it to the new owner. Upon receipt B may or may not request ownership of the $40.00 TDN depending on the trust between the two entities. B can protect the TDN with a PIN or PPK.

Specification	See 1.5 – 16 Split TDN pg. 5
	See 1.5 – 18 Get TDN Ownership pg. 5
Transaction reference	See 2.6 - Split TDN pg. 10
	See 2.8 - Get TDN Ownership pg. 11
API reference	See Appendix A – TDN API pg. 22
Demo	See Appendix B – Fed TDN Website pg. 25

- **Entity B owns a $200.00 TDN and the transferred $40.00 TDN from Entity A and deposits them to Bank Y**

  Transaction description:
  B consolidates the $200.00 TDN and the $40.00 TDN into a $240.00 TDN. B presents the $240.00 TDN Signature to Bank Y to be deposited into his/her account. Bank Y credits B's account for $240.00 and consolidates the $240.00 TDN into a TDN in its inventory.

Specification	See 1.5 – 16 Split TDN pg. 5
	See 1.5 – 18 Get TDN Ownership pg. 5
Transaction reference	See 2.6 - Split TDN pg. 10
	See 2.8 - Get TDN Ownership pg. 11
API reference	See Appendix A – TDN API pg. 22
Demo	See Appendix B – Fed Website pg. 25

2.2 TDNSYS Transactions Overview

The following transactions can be performed with TDNs using the API published by the TDNSYS:

2.2.1 Public API Calls:

- Validate TDN
- Split TDN
- Consolidate TDNs
- Request TDN Ownership
- Set TDN PIN or PPK

2.2.2 API Calls available only to the member banks:

- Initial TDN Issue
- Redeem TDN
- Request TDN Data

2.2.3 API Calls available only to the investigative authorities:

- Change TDN Status
- Request TDN Data

2.2.4 TDN Signature fields:

Issue timestamp	8 bytes
Random unique number	488 bytes
Value	16 bytes
Routing number	Optional if the database is mirrored in multiple locations.

2.2.5 TDN status values:

- ACTIVE
- CANCELED
- BLOCKED

2.2.6 TDN status transitions:

Transition	Database Operation
Set to Active	Set TDN status to Active when the TDN is created.
Active to Canceled	Update TDN status to Canceled and set Cancelation Date.
Active to Blocked	Update TDN status to Blocked and set Blocked Date. Make an entry in the investigation log.
Cancel to Active	NOT ALLOWED.
Cancel to Blocked	Update TDN status to Blocked and set Blocked Date. Make an entry in the investigation log.
Blocked to Active	Update TDN status to Active. Make an entry in the investigation log.
Blocked to Canceled	Update TDN status to Canceled and set Cancelation Date. Make an entry in the investigation log.

2.3 Validate TDN

A TDN is valid and can be used for payment or transfer if its status is Active. If the status is Canceled, the TDN has been used in a Split, Consolidate, Request Ownership, or Redeem transaction. If the status is Blocked, the TDN is part of an investigation by an authorized authority. When the status is Canceled or Blocked, the TDN cannot be used for a payment or transfer.

The easiest way to check the status of a TDN is by using the Fed's website. Anybody in the possession of a TDN can check if it is valid on this website. The user will cut and paste the TDN Signature into the appropriate field of the 'Validate' webpage or scan the TDN barcode if the device accessing the site has a webcam. The return value consists of the TDN status and its value. If the status is Blocked, a message with the contact information of the investigative authority that blocked the TDN will appear.

TDN Applications or eCommerce websites use the Validate TDN API call to implement validation. The return value of the call is used in the application as appropriate. When initiating a TDN transaction with an invalid TDN, the API call will return an error code and the transaction will not be performed.

TDN validation is just a simple search of the TDN Signature in the repository. The execution is very fast.

A TDN is considered compromised if the TDN Signature is in the possession of more than one entity. A valid TDN does not guarantee that the TDN is not compromised. It is impossible to know if a TDN is compromised until a transaction is performed on that TDN. If the legal holder of the TDN performs the transaction (except validation), the compromised TDN is canceled and the fact that it was compromised is obviously irrelevant. If, however, a fraudulent holder of the TDN uses it, then the legal holder is in the possession of a canceled TDN. He/she may request an investigation to identify the fraudulent user

of the TDN.
There may be more than one owner of a TDN when the parties involved in a transaction trust each other and the new TDN holder does not request ownership of the TDN. Theoretically, the TDN is considered compromised but in this situation, it is less likely a fraudulent transaction will be performed on that TDN. The new owner of the TDN assumes that the previous owner is not unknowingly in the possession of an already compromised TDN.

2.4 TDN Initial Issue

TDNs are issued when a request for a TDN withdrawal is received by the Fed from a member bank. When a TDN is issued a record is inserted in the TDNSYS repository with status Active

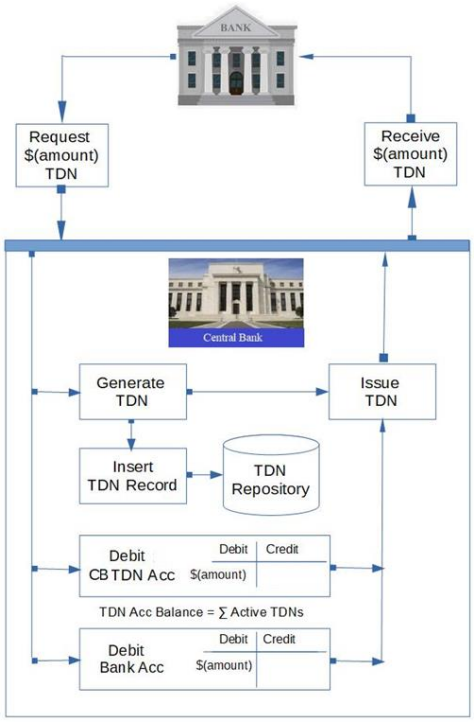


Figure 7 - TDN initial issue

and the field `tdn_issued` is set to 1 (TRUE).

The TDN Signature is sent to the bank, and the reserve account of the bank at the Fed is debited for the TDN's value. This is similar to when a bank requests paper money from the Fed. Each Fed member bank has a digital certificate issued by the Fed. The public key of this certificate is set for all the TDNs issued to the bank. When requesting a TDN, the bank's system has to satisfy an SSL/TSL challenge initiated by the TDNSYS with the bank's certificate.

Only Fed member banks may be issued and redeem TDNs. Most of the time a bank will request a large value TDN and then split it into smaller value TDNs as needed, and when redeeming, it will consolidate multiple TDNs into a larger TDN.

When a TDN issued the value of TDNs in circulation is increased with the value of that TDN.

2.5 Redeem TDN

This transaction is similar to a member bank depositing paper money into its Fed reserve account. Of course,when working with TDNs, the difference is that there is no need for expensive and time-consuming paper money transportation, the transaction is performed instantly. The Fed cancels the TDN deposited and credits the bank's reserve account with the overall TDN value. The field `tdn_redeemed` is set to 1 (TRUE).

Each Fed member bank has a digital certificate issued by the Fed and all TDN in its possession must have the PPK set to the public key of this certificate. When redeeming a TDN, the bank system has to satisfy an SSL/TSL challenge initiated by the TDNSYS with the bank's certificate.

Only Fed member banks may be issued and may redeem TDNs. Most of the time a bank will request a large value TDN and split smaller value TDNs as needed and when redeeming it will consolidate multiple TDNs into a larger TDN.

When a TDN is redeemed the value of TDNs in circulation is reduced with the value of that TDN.

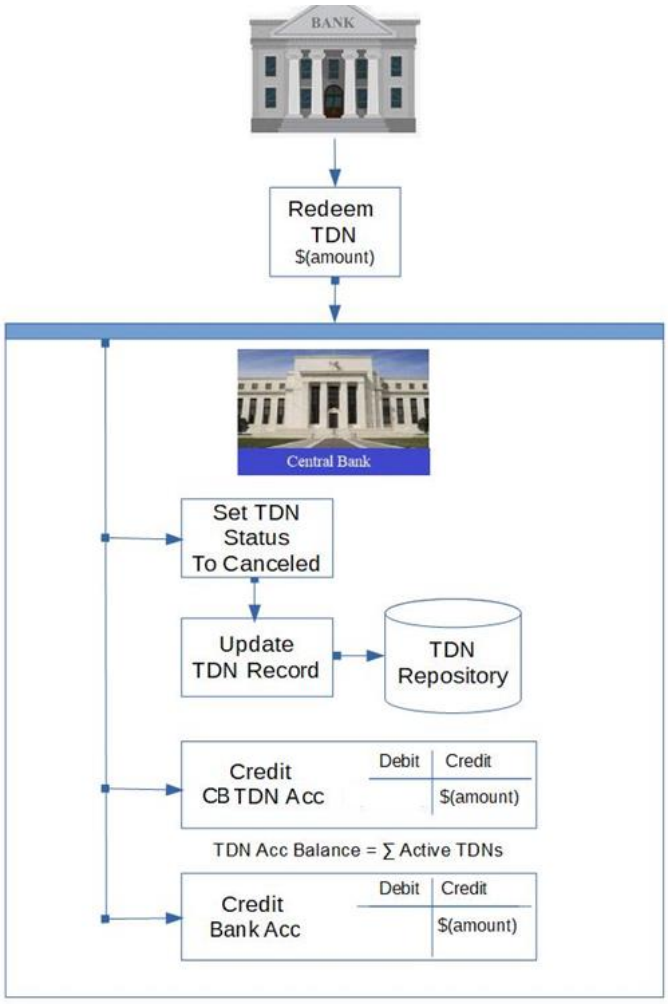


Figure 8 - Redeem TDN

2.6 Split TDN

Splitting a TDN is like getting change for paper money. The difference is that TDNs do not come in predefined denominations. A TDN is split into two TDNs with the total being equal to the original TDN. The Fed can set the fraction of the TDN to hundreds (cents) or any other value. Sometimes it is necessary to split a TDN in two TDNs in order to make a payment or transfer. This can be done using the Fed website. TDN Applications perform splits for the amount necessary to make a payment. The original TDN is canceled, one split is used for the payment and the second remains in the possession of the TDN holder using the application.
Banks and merchants prefer to keep only one TDN of a larger value and split it into smaller value TDNs when necessary. When receiving a TDN, it is consolidated with a largerr, existing TDN.
The Split transaction is very fast, requiring only a few database operations. When the TDN is protected by a PIN or PPK some extra processing is necessary. If any steps of the transaction failed, the API Call will return an error message.

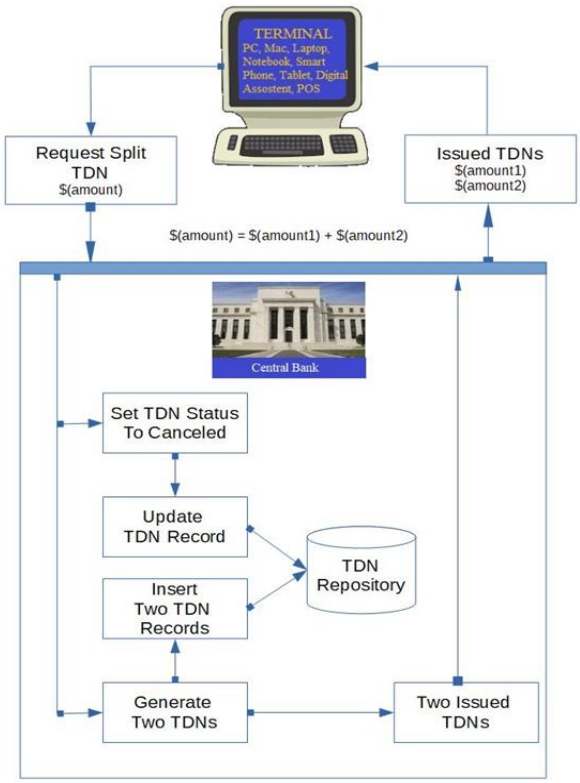


Figure 9 – Split TDN

2.7 Consolidate TDNs

Two TDNs can be consolidated into one TDN. The two TDNs are canceled and a new TDN is created for the total value of the canceled TDNs. It is more practical to hold onto only one TDN and split a TDN of a specific value when needed.

Banks and merchants prefer to keep only one TDN of a larger value and split it into smaller value TDNs when necessary. When receiving a TDN, it is consolidated with a larger, existing TDN.

The Consolidate transaction is very fast, requiring only a few database operations When the TDN is protected by a PIN or PPK, small extra processing is necessary. If any of the steps of the transaction failed, the API Call will return an error message.

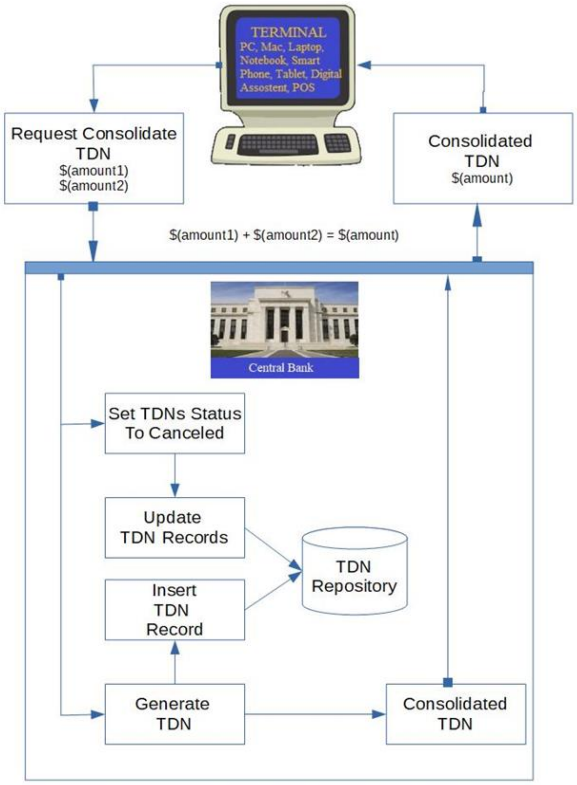


Figure 10 – Consolidate TDNs

2.8 TDN Ownership Request

Ownership request is the TDNSYS process's way of preventing double spending. Anybody in the possession of a TDN Signature may request ownership of the value associated with that TDN. TDNSYS will cancel the TDN and create a new Active TDN for the same value. It should be noted that splitting and consolidating a TDN with another TDN is also a way to obtain TDN ownership.

Sometimes the owner of a TDN may request ownership in order to be sure that the TDN has not been compromised.

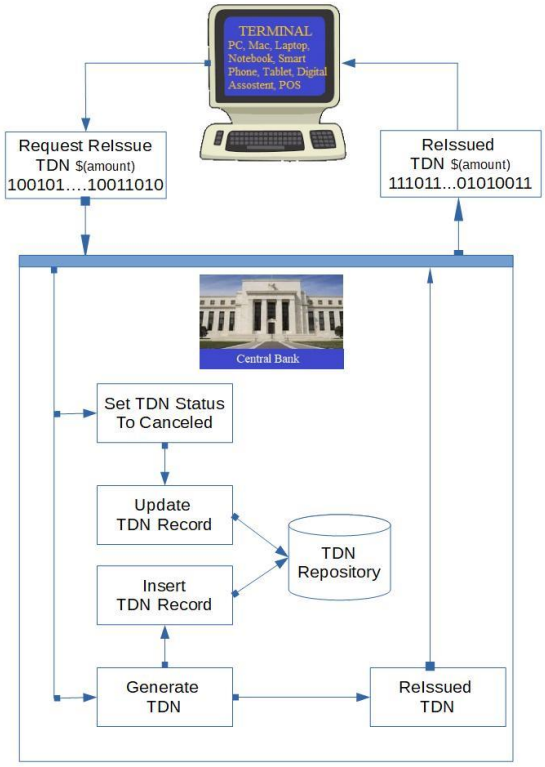


Figure 11 - Request TDN ownership

2.9 Set/Change TDN PIN/PPK

A TDN can be secured with a PIN or PPK. It is not possible to set both a PIN and a PPK for a TDN. In general, it is optional to setup a PIN or a PPK for a TDN. If a TDN is in the possession of a bank or a merchant, the TDN must have the PPK set to the public key of the digital certificate issued by the Fed.

If a PIN or PPK is set for a TDN, it must be supplied when a transaction is initiated. When validating a TDN, it is not necessary to have the PIN or PPK.

The easiest way to protect a TDN with a PIN is using the Fed Website (Appendix B-e pg. 28). A PIN is a number of four to eight digits. The PIN is set for one TDN at a time. The same PIN can be used for as many TDNs as desired to make it easier to remember. When using a TDN application, the PINcan be set by the application. The application may store that PIN and use it when a transaction is initiated.

Setting up a PPK is more difficult because a private/public key pair has to be generated. This requires advanced computer skills. The public key is set for the TDN and the private key has to be kept secret. Every time a transaction is initiated for a TDN protected with a PPK the initiator of the transaction must satisfy the PPK challenge issued by the TDNSYS using the private key.

When a TDN is issued by a Fed to a member bank, the TDN PPK is set to the public key of the bank digital certificate issued by the Fed. When a member bank redeems a TDN, it must satisfy the PPK challenge issue by the TDNSYS.

The API documentation describes the PIN and PPK validation in greater detail. This is the same for all transactions. The TDN Validation does not require this validation.

Section 3 Implementation

3.1 TDNSYS Infrastructure

The TDNSYS system is based on the client server model. It is a centralized system. The infrastructure supporting the TDNSYS relies on existing, reliable, and proven state of the art technologies used in the Financial Market Infrastructures (FMIs). It is very similar to the infrastructure required for Credit or Debit Cards. The communications between the Server and the Clients takes place over private or public networks.

The ***Server*** side is implemented and maintained by the Fed and exposes a public API which supports all the transactions related to TDNs.

The ***Clients*** consist of a Fed website and hardware/software systems developed by IT companies based on the API. Holders of TDNs can request ownership, split, consolidate, and assign a PIN or PPK to these TDNs using the Fed website or the proprietary hardware/software clients.

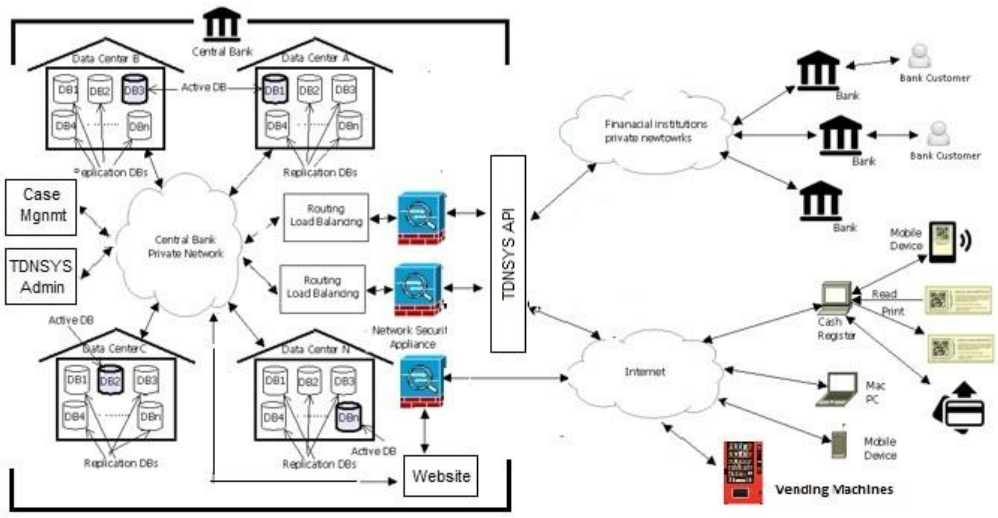


Figure 12 - TDNSYS Infrastructure

In Figure 12, redundancy and disaster recovery is implemented by storing data on multiple servers in different data centers. This makes data available almost 100% of the time. Promoters of DLT consider client server systems as having 'one point of failure.' This may be true for simple systems, but in the financial industry this is not the case. I should also mention that applying blockchain's feature 'Irreversibility of Records' to data stored on Fed systems or store this data on some computers somewhere on the internet is just plain ridiculous.

3.1.1 Server

Implementing the server software is not a technical challenge. It consists of an ISAM database that is ACID compliant, containing the main TDNs table and administrative tables. The TDNs table supports only insert and update operations. Records are never deleted. For improved performance, the canceled TDNs may be moved to a different table from time to time. The administrative tables include information available when a transaction is performed. They also support a case management system used for storing investigation information regarding complaints and fraud.

Each TDN has one record in the repository. At a minimum, the record contains the TDN Signature, the associated value, and the date when it was created. The cancel date, initial issue and redeemed flags PIN, and PPK may be null but all the other fields must have a value.

The TDN Signature is a string of numbers containing a timestamp, a long unique number, and the value zero padded. (See Fig.1 on pg. 1) The status can have the following values: ACTIVE, CANCELED, or BLOCKED. The status of a TDN can be accessed by anybody with access to the TDN Signature. The PIN and PPK support TDN security and can be set, updated or removed using

the Fed website or any other available application. If the PIN is set, it has to be included in the API request and it is validated against the store value before a transaction is performed. If the PPK is set, the validation is performed by TDNSYS by issuing a PPK challenge.

If privacy is not a concern of the TDN holder, the PPK field can contain a digital certificate issued by the Fed or some other authority. This may be required if the holder of the TDN accepts payments. The banks must always set the PPK to the private key of the certificate issued by the Fed.

The TDNSYS must support great volumes of transactions and store very large number of records. It has to be very fast, redundant, with high availability and no down time. TDN database transactions take only a few microseconds. There is no need to get into the details as such systems are routinely deployed in the financial industry.

3.1.2 Clients

A TDN client consists of hardware, software, and means of communication with the TDNSYS server and with other clients.

This means of communication already exists. The Internet is widely available, and the banks have their private networks ready to be used by the TDN applications.

TDN client hardware can be any of the shelf digital devices or specially designed hardware. They may be PCs, MACs, smart phones, tablets, smart watches, etc. Special hardware is needed for cash registers, bank teller terminals, ATMs, and vending machines. This hardware may already be in use and only software updates may be necessary in order to support TDN transactions.

Hardware platforms use different technologies for performing input/output operations and TDNs storage.

- Webcams - the user will be able to read the TDN barcode representation in the appropriate input fields of a webpage or application.
- Barcode readers, wand barcode scanner - standard equipment on cash registers, vending machines, ATMs, and other specially purposed hardware.
- Smart card readers - digital devices may have built-in card readers. Card readers can also be purchased separately and attached to most digital devices. Cash registers, vending machines, and ATMs already have these input devices built in.
- Direct communication between the buyer's and seller’s devices.
- Bluetooth technology is widely supported by digital devices. Wi-Fi and other local networking can also be used for this purpose.
- Near-field communication (NFC) cards and devices.
- Output digital devices can be digital displays, printers and any of the devices already mentioned above.
- Manual input/output - in the situation of a digital device with only a keyboard and mouse, the user has to manually cut and paste the TDN's ASCII text representation into the appropriate fields of an application or webpage. This may also happen when an application or website does not support other input/output devices besides keyboard and mouse.

The hardware necessary for storing TDNs can be in any device able to store digital data. This can be a hard drive, a smart card, a smart phone, a memory stick, card, or wristwatch.

TDNSYS exposes an API and applications can design and implemented using it. These applications can be deployed on a large variety of hardware platforms. Brick and mortar stores can deploy cash registers and apps for mobile devices compatible with these registers. The buyer will run this application on his/her digital device. The buyer does not need an account with the seller. A generic application may be designed for mobile devices as long as all the sellers agree to a cash register standard. The basic interaction between such a generic app and the cash registers will consist of the following operations:

- Buyer accepts the amount displayed on the seller's cash register. (This step can be skipped)
- Seller's cash register reads a barcode displayed on the buyer’s mobile device screen showing the value of the TDN stored on the device.
- If this value is the same as the amount due, the seller's cash register will request the TDN ownership.
- If the TDN value on the device is larger than the amount due, the cash register will request a split for the amount due. It will take ownership of it and display the barcode for the second split (change due).
- The buyer’s app reads the barcode from the cash register, requests its ownership, stores on the device it or consolidate it in an existing TDN..
- If any of the steps fail, the transactions are rolled back, and the payment is not successful.

Here are some examples of clients:

- Websites
- TDN Apps
- Smart cards readers
- Cash Registers
- Vending Machines
- Bank tellers

These clients are documented in Section 4 – Use Cases

3.2 Designing and implementing a prototype

It is very simple to design and implement a TDNSYS and clients prototype. I developed the demo described in Appendix B in just a couple of weeks without any help. For the repository there is no need of a relational database. A DBMS like c-tree from

FairCom allowing low level access to the records can be used. The repository can be designed and deployed in just a few days.

The API design and documentation will need more work. Implementing the

Section 4 TDNSYS Security and Privacy

4.1.1 Infrastructure Security

Server Side

The TDNSYS infrastructure is deployed using existing, reliable, and proven state of the art technologies in the Financial Markets Infrastructures (FMI). The operating systems, database and development tools, and methodologies used in the system implementation have been in use for a long time in the IT world and have been upgraded and hardened continuously to address new technical challenges and new security threats.

The security standards ruling the development, deployment, and operation of TDNSYS are established by the government agencies, who on daily basis assess new threats and threat mitigation. All government legislation regarding security and privacy have to be respected as the system is owned by the Fed. The TDNSYS Server site is as secure as any other system operated by the Fed.

It is impossible to counterfeit TDNs. It is impossible to access the TDNSYS database directly and the TDNSYS is balanced against the reserve accounts of member banks. Every time a TDN is issued or redeemed, the reserve accounts are debited or credited as appropriate.

Resources:

- NIST: National Institute of Standards and Technology
- GSA Cyber security Programs & Policy
- Federal Chief Information Officers (CIOs)
- Federal Reserve Policy on Payment System Risk

Reading any of the documents listed above is very boring. It is not as interesting and exciting as a YouTube video about Bitcoin and blockchain. Unfortunately, serious financial business is very boring most of the time.

The TDNSYS does not record information about private holders of TDNs. The bank and merchants may be required to set the PKK of a TDN to the public key of a digital certificates issued by the Fed. In this situation the security is more important than privacy. Private holders of TDNs may setup PPK instead of PIN based security. This may be the public key of a digital certificate and in this situation the anonymity of the holder is waved.

The TDNSYS may record information available when a transaction is executed. Most of the time this information is about the Internet traffic related to the transaction.

Asking what happens in case of infrastructure failure is a legitimate question. This may happen in a war situation or a natural disaster. A contingency plan should be in place to distribute paper money to the population. The Fed should have enough cash to cover such a distribution. Local banks which will perform the distribution should also have cash ready in their volts and a system in place to manually record all the transactions related to this distribution.

Client Side

The TDNSYS Server interacts with its clients over the Internet. Financial institutions may connect to the Fed through existing private networks. All the security issues associated with the Internet apply to TDNSYS clients. Clients deployed by banks comply with industry security standards and are less exposed to penetration and fraud.

The TDNSYS exposes a public API. Third parties can develop applications for TDN transfers or payments using this API. This may create the possibility of fraudulent applications. The Fed may decide to certify applications and allow transactions only for applications signed with a Fed issued digital certificate.

TDN holders must always be confident that they are dealing with a trusted party. This is similar to sending money using Western Union or any other money transfer company. The money goes to party specified by the sender.

4.1.2 Payments Security

When using TDNs for making payments, most of the security issues that apply to cash payments have to be addressed. If you pay somebody with cash, you may never see that seller again. This is why it is very important to make payments and transfer money only when you are confident that you know exactly who the other party is.

The parties involved in a TDN transaction have to ensure that the TDN is valid and that double spending is prevented. This means that when transferring a TDN, the party receiving it has to make sure it is valid and should request ownership from the TDNSYS immediately. The transactions should be considered completed only after the TDN has been validated and the ownership transferred. TDN Apps perform ownership request instantly.

Because the parties involved in TDN transactions are responsible for validation and double spending prevention, there is flexibility in the way TDNs are used. When the parties involved in a transaction trust each other or when the amount transferred is small, validation and double spending prevention can be performed later or not at all. The parties might exchange the printed TDN barcode or a TDN stored on a memory device

During the handling of a TDN as a printed ASCII text or barcode, it is possible for somebody to take a picture of it. This means that the TDN was compromised and there is another holder of the TDN Signature. The fraudulent holder may acquire its ownership locking out the rightful owner.

Because TDNSYS clients are handling TDNs using computing devices and the Internet, all the related security issues have to be addressed and all precautions have to be taken. This can be prevented printing the TDN with a cover that can be scratched later, similar to lottery tickets.

4.2 TDNSYS Fraud Prevention and Mitigation

4.2.1 Server Side

The server-side implementation of TDNSYS is very simple. It is similar to the Federal Reserve's electronic system Fedwire, and its private sector competitor, CHIPS, with the difference being that it does not have accounts for handling money transfers and there is no processing required for a netting engine. The security and possibility of penetration of TDNSYS should be identical to Fedwire.

I could not find any information on the Web about Fedwire, SWIFT, or CHIPS being compromised due to technology problems. It is possible, like with any financial system, to use them in a fraudulent way. This may happen when there is a breach in security at one of the customers at any these systems, most of the time by insiders.

An interesting example is one of the largest cyber-heists in history when hackers stole $81millions from a Bangladesh Bank account at the New York Fed. Capitalizing on weaknesses in the security of the Bangladesh bank, including the possible involvement of some of its employees, the perpetrators managed to compromise Bangladesh bank's computer network, observe how transfers are done, and gain access to the bank's credentials for payment transfers.

The TDNSYS is balanced against the reserve accounts of member banks. Every time a TDN is issued or redeemed, the reserve accounts are debited or credited as appropriate. This makes it easy to detect any inconsistencies in the system.

The TDNs may have a value below a preset threshold (ex. $500.00). In order to prevent fraud, the Fed may issue a digital certificate to banks, merchants or any other clients it may consider necessary. TDNs with the PPK set to the private key of a Fed issued certificate will be allowed to have a value over this threshold. When a TDN is issued to a bank it will have the PPK set to the private key of the digital certificate issued to that .

The TDNSYS also records all Internet traffic information related to a transaction and any other available information. TDNs involved in the transaction will be marked with a transaction ID. For example, when a TDN is split, the initial TDN and the two splits are marked with the same transaction ID. This makes it possible to 'chain' transactions.

When a fraud is reported a case is opened in the TDNSYS Case Management system and the TDN(s) affected will be marked as Blocked if still Active. If a transaction is initiated for any of these TDNs, the transaction originator is informed that the TDN is under investigation and instructed to contact the investigative authority. He/she may contact the investigative authority and participate in the investigation.

It is possible to reconstruct a tree structure of all the TDNs cascading from the fraudulent TDN. Each leaf in the tree can be traced back to the fraudulent TDN through a chain of transactions. Using this tree and all the information recorded from the nodes in the tree, the investigative authority may be able to recover the TDN and possibly identify the perpetrator. The usual criminal investigative techniques are also employed in solving and preventing TDN fraud.

An insurance system may be put in place to reimburse victims of TDN fraud partially or in full.

4.2.2 Client Side

Because TDNs are very similar to paper money, the prevention of lost or theft is the responsibility of the owner. All the rules for digital data security and privacy apply to TDNs. Transactions must be performed only with trusted parties, especially when using the Internet. When using credit cards, the credit card company may reimburse the card holder for unauthorized use. This is not the case with TDNs.

A fraud may occur when a TDN is compromised. A TDN is considered compromised if the TDN Signature is in the possession of more than one entity. It is impossible to know if a TDN is compromised until a transaction is perform on that TDN. If the legal holder of the TDN performs a transaction (except validation) the fact that it was compromised is obviously irrelevant. If, however, a fraudulent holder of the TDN gets ownership, then the legal holder will no longer be able to use it. He/she may request an investigation to identify the fraudulent use of the TDN.

There may be more than one owner of a TDN when the parties involved in a transaction trust each other and the new TDN holder does not request ownership of the TDN. Theoretically, the TDN is considered compromised but in this situation, it is less likely a fraudulent transaction will be performed on that TDN. The new owner of the TDN also assumes that the previous owner was not unknowingly in the possession of an already compromised TDN.

The party receiving a TDN has better means of protection against fraud then the sender. The TDN can be validated and the ownership transferred before the transaction is accepted. This assumes that the application or website used to perform these operations is legitimate. Merchants have very safe software/hardware systems. The unauthorized payments so frequent with credit cards do not happen with TDNs because there is no credit card number to be stolen.

Merchants must take steps to prevent employee theft. As it happens with cash, employees may have access to TDNs. The theft is noticed only when the stolen TDN is involved in a transaction. Most of the systems will allow access to the TDNs to only a limited number of employees. The best protection against this fraud is depositing the TDNs into the merchant’s bank account immediately after they are received.

Banks may be able to recover TDNs lost by their customers. If, for example, a customer withdrew a TDN from his/her bank account and loses it, a bank employee can identify the TDN Signature in question and if the status is still ‘Active,’ they can cancel it and create a new one for the customer.

TDNs based credit and debit cards have the same vulnerabilities as other cards. If lost or stolen, they can be fraudulently used if the PIN is also compromised.

Section 5 Use Cases

5.1 How TDNs are stored and manipulated.

A TDN is a long string of numbers associated with a money amount. The long string of numbers is called the TDN Signature. Like any other digital object, a TDN can be stored as a file on the file system of any digital device, in a database, in a cloud, on a smart card, or in some proprietary format used by an application. The ASCII text or barcode representation of a TDN can be printed or displayed on the screen of a digital device. (See Fig. 1 on pg. 1**Error! Reference source not found.**)

TDNs can be deposited or withdrawn from bank accounts or exchanged for paper money. The Fed has an official website supporting all TDN transactions.

The TDNSYS exposes an API supporting the TDN transactions. This API can be used to design, implement, and deploy applications and websites facilitating payments and money transfers. Cash registers can be easily upgraded to support TDN payments.

The easiest way to use TDNs is having a TDN based debit card issued by a bank. The user does not have to be aware of the details of the TDN operations performed in the background.

It is not impossible, yet very inefficient, to type a TDN into a field of an application when making a payment or transferring money. These applications may support one or more of the following input/output devices:

- Copy and paste
- Barcode readers and printers/displays
- Smart card readers
- Near-field communication (NFC) cards and devices

TDNs can be transferred like any digital data using e-mails, SMS, file transfers, applications, etc.

It is good practice to make backups of the TDNs.

5.2 Website Transactions

A TDN enabled website uses the TDNSYS API to implement the functionality supporting TDN transactions. Websites require the user to manually handle the TDNs. The ASCII text representation of the TDN has to be cut and pasted in and from the website fields. Most digital devices have a webcam enabling the user to read a TDN barcode in a field. If the digital device is equipped with a smart card reader, the process of inputting and outputting TDNs in and from the website fields can be automated.

The simplest web client is the Fed website supporting all TDNSYS TDN transactions. This is the basic process for performing TDN validation, split, consolidate, set, or change a PIN or PPK or ownership request.

5.2.1 Websites Input Methods

The input method used when paying with a TDN on an internet store depends on how the buyer's TDN is stored, what kind of input is supported by the payment webpage, and if the buyer's digital device has a webcam or a smart card reader.

The copy and paste method is used when the TDN Signature is stored in a file on the digital device file system. The buyer will copy the TDN Signature from the file and paste it into the payment field on the eCommerce webpage. The buyer has to keep track of the spent TDN. For example, he/she can rename the TDN file from 'myTDN14.65.tdn' to 'myTDN14.65.spent' or if the file contains more than one TDN Signature, write 'spent' next to the TDN Signature.

The buyer may have an application for TDN handling installed on his/her device. This application should be able to perform validation, split, consolidate, set or change a PIN, PPK, or ownership request, deposit and withdraw TDNs from the buyer's bank account, and store or keep track of available and spent TDNs. The application will display the TDN Signature (for copying) or insert it into the clipboard. The buyer will paste the TDN Signature in the appropriate field and press a button to complete the transaction.

The application may be able to print or display the TDN's barcode. If the eCommerce site supports barcode input and the digital device accessing it has a webcam, the buyer can use the TDN barcode to input the amount due. If the barcode is printed on the device screen, it has to be a different device than the one accessing the website.

See 4.4 TDN Apps, pg.19 for more details about TDN management applications.

5.2.2 eComerce Websites

Online store developers use the TDNSYS API to implement the payment process. The buyer will supply a TDN for the payment due and the seller will instantly take ownership of the TDN. If the TDN is protected by a PIN or PPK, it has to be removed before passing it to the seller. If the TDN value sent as the payment is larger than the payment, the seller may refuse the payment or they may send back a TDN for the value of the change due.

Payments with TDNs do not require the buyer to have an account with the seller. There is one input operation necessary for the payment. This is much easier compared to the complicated process for inputting credit card information and validating it. The buyer will not be 'locked' with an eCommerce company because that company already has the payment information and there is no need to enter card information all the time.

After a payment is performed, there is no payment information left on the seller's system to be stolen by hackers. Of course, the seller has the responsibility to secure the TDNs in its possession like any other TDN holder.

The Fed does not charge any fee for TDN transactions. The only cost a merchant will encounter when receiving payments in TDNs is for the development of a payment page and an application for handling TDNs. This cost is minimal. There is no need to have a company handle the payments and pay fees to this company and the credit cards. There is no need to collect and store customer data or create account numbers. The payments are anonymous. The TDN is available to the merchant immediately after receiving the payment.

The time necessary to build a TDN payment page into a website and the supporting software for managing the merchant's TDN is very small. For building a payment page and integrating it with such an application, a professional programmer will spend around 12 to 20 hours working. Consulting fees for this kind of work rarely exceed $250/hour.

The buyer must be sure that the eCommerce site is legitimate. A payment to a fraudulent website may be recovered when using a credit card. When using TDNs the money is gone as soon as the payment is processed. Fortunately, the Fed requires merchants to register and request a digital certificate issued to them in order to receive TDN payments. This makes it easy to identify the vendor when the payment is made. This way any disputes can be handled by an authorized authority.

As long as the vendor's digital certificate is validated when the payment is made, it can be said that online TDN payments are as safe as credit card payments with the advantage that there is no credit card number to be stolen and used fraudulently. Most likely, TDN fraud prevention software will be developed, making it safer to buy online using TDNs.

5.2.3 Fed Website

This website is implemented and maintained by the Fed. It supports all TDN basic transactions. Access to the website is not restricted and all transactions performed are anonymous. The Fed may collect all available information regarding a website access and store them in TDNSYS database. For example, the IP address of the machine accessing the website will most likely be stored in relation to the transaction requested, failed transactions, traffic parameters, etc.

The website is part of the TDNSYS implemented by the Fed.

5.3 Peer to Peer Money Transfers

TDN transfers are very simple. The sending entity passes the TDN Signature to the receiving entity and the receiving entity requests the TDN ownership. If Get Ownership transaction failed, the transfer is invalid.

The TDN can be transferred as a computer file containing the TDN Signature, a barcode image file, a barcode printed on paper, or a barcode on a digital device screen. The transfer can be performed face to face or using regular mail email, SMS, social media, video telephony, FTP, or any other means of transferring digital data. For security, it is a good idea to protect the TDN with a PIN and send the PIN separately

If the two entities involved in the transfer trust each other, the receiving entity may not request the ownership of the TDN received. Theoretically, the TDN is considered compromised, but in this situation it is less likely a fraudulent transaction will be performed on that TDN. The new owner of the TDN assumes that the previous owner was not unknowingly in the possession of an already compromised TDN.

If the TDN is secured with a PIN or a PPK, it has to be removed before performing the transfer or the PIN has to be passed to the receiving entity. Sharing a PPK private key is possible, but it is not advisable.

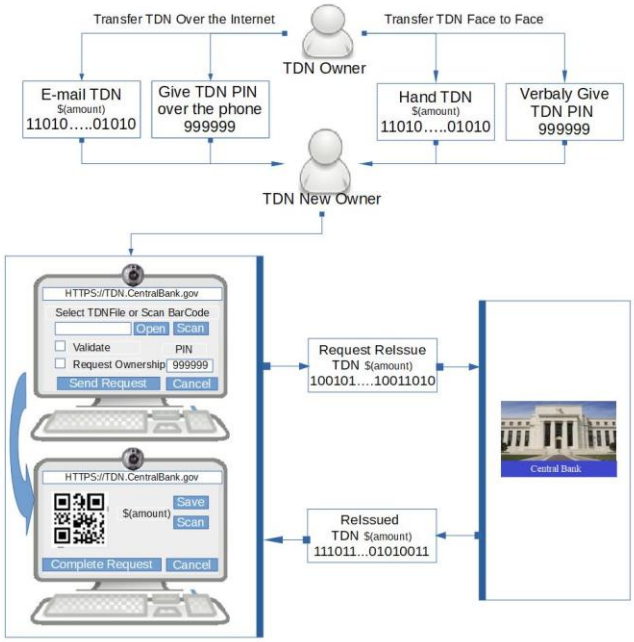


Figure 13 - Peer to peer transaction

5.4 TDN App

A TDN App is not necessary in order to use TDNs for payments or transfers. It is just an application that stores TDNs on a digital device connected to the Internet and performs TDN transactions.

Software developers have unrestricted access to the TDN API, and they can develop specific applications based on a particular user need. For example, functionality can be implemented in the TDN Application to connect to the bank where the user has an account and withdraw or deposit TDNs. The Fed may require developers to register in order to prevent abuse.

Note: TDN Apps are not digital wallets.

A TDN Application can be installed on a PC/Mac or any a mobile device. The functionality of a TDN Application may support some or all of the following transactions:

- List the values of the TDNs stored on the host device and their status.
- Removed canceled TDNs from the device (manually or automatically).
- Keep a history of transactions performed by the application.
- Select a TDN stored on the device and set or change the PIN or PPK security.
- Select a TDN stored on the device and split a TDN with a smaller, specified amount.
- Select two TDNs stored on the device and consolidate them into a single TDN.
- Connect with the bank and withdraw a TDN from the user's account.
- Connect with the bank and deposit a TDN into the user's account.
- Select a TDN stored on the device and print the TDN's barcode on the screen of the printer (if available).
- Read a TDN barcode, store it on the device, and request ownership.
- Copy and paste text or barcode input.
- Select a TDN and e-mail it as a barcode or text.
- Select a TDN and SMS it as a barcode or text.
- Perform all necessary application setup.

The application is implemented using the published TDNSYS API. Not all the above features have to be implemented on applications and other functionality may be added. TDN Apps must be acquired only from safe sources. Banks may supply them free of charge to their customers.

Using a TDN App requires some understanding of TDNSYS and computer abilities. The app users have to be aware of TDNs Security. Users who do not want to worry about this can use a TDN based Debit Card.

5.5 TDN Smart Cards

TDN Smart Cards offer an easy way of handling TDNs and performing TDN transactions. At a minimum, Smart Cards can be used to store TDN Signatures. Devices equipped with a smart card reader that run a TDN Application or access an eCommerce website can read and write TDN Signatures from and to the Smart Cards.

Banks can offer their customers TDN based debit or credit cards. This is the easiest way to make payments or transfer TDNs. The user of such a card does not have to be aware of the inner working of TDNs. When using such a card, the TDN Signatures are not stored on the card. The card has to be protected by a PIN, and during a purchase or transfer the user must supply it for validation. This is the same as any PIN assigned to a regular debit card. It is not the PIN used to secure TDNs.

If the smart card stores TDN Signatures, it may be possible to use it as a 'Gift Card'. The holder of this kind of card can load it by withdrawing TDNs from his/her bank account. Here is a diagram showing how this card works:

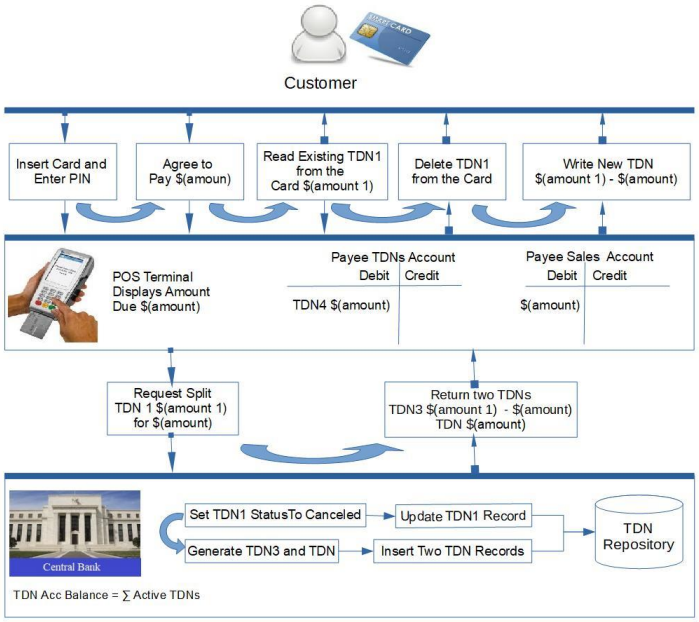


Figure 14 - Card transactions

5.6 Cash Registers Supporting TDNs

A cash register supporting TDN payments is not different than currently used cash registers. Most of the existing cash registers can be upgraded with software supporting TDN transactions in order to receive TDN payments. They already have the hardware required for TDN transactions and they can interface with accounting systems and bank accounts.

The greatest advantage for merchants receiving TDN payments is the instant availability of the payment received at the point of sale. The money can be deposited immediately into the bank account and is available for use. There is no need for clearance, as happens with checks, and there are no fees to pay for using TDNs. The only cost a merchant will encounter when receiving payments in TDNs is for the development or purchasing of software needed by their existing cash registers. There is no need to pay a company to handle the payments or pay fees to the credit card company.

5.7 TDN Vending Machines

Error! Objects cannot be created from editing field codes.

Vending machines can be equipped with a TDN barcode reader. Buying from these vending machines using a TDN App installed on a mobile digital device is as easy as using cash. The user splits the payment amount, displays it on the device's screen, and positions it in front of the vending machine barcode reader. The TDN used for purchases at vending machine must not be protected with a PIN or PPK.

Vending machines may even be equipped with smart card readers and could accept 'gift card' types of Smart Cards.

A paper printed barcode can also be used. The user may frequently use the same amount for purchases at vending machines and have multiple printed TDN barcodes for the same amount.

There are a lot of advantages for the owner of a TDN vending machine. It is less expensive to equip a vending machine with a barcode reader, internet connection, and the necessary software then a mechanical system and it is much easier to maintain This machine is less prone to failure. The owner of the vending machine can retrieve the money without visiting the location or it can be directly deposited into the vendor's bank account. The money cannot be stolen from the vending machine by thieves. After the TDN Signature is read by the machine and the product is released, the vendor takes ownership of the TDN and the buyer is no longer able to use it. The transaction takes just a fraction of a second.

5.8 Banking with TDNs

All the bank operations performed with paper money can be performed with TDNs. Of course, there is no need to have a bank account in order to use TDNs. Banks should convert paper money to TDNs on request for anybody.

TDN deposits and withdraws are instant and do not require a trip to the bank or an ATM when using an app supplied by the bank. The bank ATM can also offer TDN withdraws. The withdrawn TDN may be printed for the customer or it may be scanned into a mobile device with a TDN App. ATM TDN deposits may be performed in the same way.

Interbank transactions are also very simple and fast. For example, if a customer wants to transfer money between two different bank accounts, the sending bank will withdraw the amount from his/her account as a TDN and send it to the receiving bank who will deposit it in the customer's account.

Electronic bill payments are performed in the same way. There is no need for clearing the transactions and the customer does not risk overdrawing the account. The bank will not initiate the transaction if there are insufficient funds.

The bank does not need to have a large quantity of paper money on hand if TDNs are widely accepted. Requesting TDNs from the Fed against the reserve account is instant.

The Fed can collect statistical data for TDNs that was impossible to collect for paper money. Maintaining a TDNSYS is much less expensive than printing and handling paper money.

The TDNSYS specifications are generic and can be used for other purposes. For example, a system can be designed to implement banking transactions without needing clearance. In this situation the bank accepts deposits and issues TDNs for them. This is totally different than the Fed issuing TDNs. The bank may enter into agreements with other banks issuing TDNs and merchants willing to accept these TDNs as payments. The banks will accept each other's TDNs which requires a clearing process. In order for these TDNs to get accepted, it is necessary to have only one TDNSYS used by all banks and merchants.

Appendix A. TDNSYS API

The TDNSYS API is currently under development. This is not the full API.

a. TDNSYS Global Settings

- Banks
- Merchants
- Maximum Value

b. Public API Calls

These calls can be executed by any software system on the Internet. The caller must have a valid TDN and the PIN or PPK if set. The PIN and PPK are not required if the call is only for TDN public validation.
A call has the following format:

- Validate TDN
- Split TDN
- Consolidate TDNs
- Request TDN Ownership
- Set TDN PIN
- Set TDN PPK

c. API Calls available only to the member banks

Banks systems can use these calls to request the withdraw or deposit of TDNs in their reserve accounts with the Fed.

- Request TDN Data
- Initial TDN Issue
- Redeem TDN

d. API Calls available only to the investigative authorities

In case of fraud or complaints, an investigation may be authorized by the Fed. In these situations, the TDN in question has to be blocked for the duration of the investigation. These calls are implemented in the Case Management system of the TDNSYS.

- Request TDN Data
- Change TDN status

e. API Calls Syntax

The API calls are not completely defined because at this time an implementation of the TDNSYS server site is being developed. The exact call field lengths and values will be defined during the development of the TDNSYS.
The calls are documented in a table with two columns. The left column is the Client side and the right column is the Server (Fed TDNSYS) side. The request call is highlighted and has a Call ID in the first field. The right side is for the server response.
Each API call is defined as a transaction. If a step of the transaction fails, the transaction is rolled back.

Keywords	
Begin Transaction, End Transaction	
If Success, If Failed, Goto End, Roll Back	
PIN /PPK Challenge	*If PIN parameter sent: Server validates against the value set for the TDN.* *If PPK parameter sent: Server encrypts a string using the public key.* *Client sends back the decrypted string.*
CERT Challenge	All banks and Investigators must set a PPK for a TDN as the private key of a certificate issued by the Fed. PIN security is not accepted. The server encrypts a string using the public key. Client sends back the decrypted string.

Parameter Values	
009	*Call ID - Always the first field, zero padded numerical values*
(param)	*Optional parameter*
|	*Field Separator*
--	*Call step ends*
TDN	*TDN Signature*
PIN/PPK is a set of roles, policies, hardware, software, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates and manage public-key encryption. The purpose of a PPK is to facilitate the secure electronic transfer of information for a range of network activities, such as e-commerce, internet banking, and confidential email. PPK	*PIN or PPK set for TDN*
CERT	*TDNs belonging to banks must always be signed with a Digital Certificate*
TDN status	*Status: 0 = canceled, 1 = active, 2 = blocked*
Value	*TDNs' value is in the form of $dollars.cents. The Fed sets a maximum value for a publicly used TDN. All TDNs with larger values must be signed with a digital certificate issued by the Fed.*
TDN	*TDN Signature*
XML	*Field value stored in XML format*
0, 1	*Failed and Success values*

f. Public API Calls

These calls can be executed by any software system on the Internet. The caller must have a valid TDN and the PIN or PPK. The purpose of a PPK is to facilitate the secure electronic transfer of information for a range of network activities such as e-commerce, internet banking, and confidential email. PPKPIN and PPK are not required if the call is only for TDN validation.

Validate TDN

This call can be made by any client without the need of a PIN or PPK.

Client	***Server***
Begin Transaction	
001 | TDN	If Success **TDN | status | Value --** If Failed **0 --**
End Transaction	

Split TDN

Client	***Server***
Begin Transaction	
201 | TDN | PIN /PPK --	
PIN /PPK Challenge	
	If Success **TDN | Value | XML--** If Failed **0 --**
End Transaction	

Consolidate TDNs

Client	***Server***
Begin Transaction	
201 | TDN | PIN /PPK --	
PIN /PPK Challenge	
	If Success **TDN | Value | XML--** If Failed **0 --**
End Transaction	

Request TDN Ownership

Client	***Server***
Begin Transaction	
201 | TDN | PIN /PPK --	
PIN /PPK Challenge	
	If Success **TDN | Value | XML--** If Failed **0 --**
End Transaction	

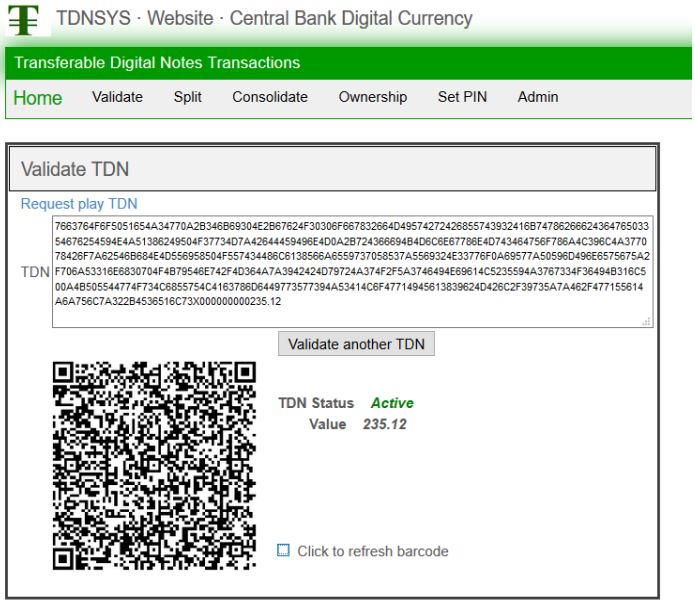
Set TDN PIN

Client	***Server***
Begin Transaction	
201 | TDN | PIN/ PPK --	
PIN/ PPK Challenge	
	If Success **TDN | Value | XML--** If Failed **0 --**
End Transaction	

Appendix B. Fed TDNSYS Website

This is a simple implementation of a Fed Website. Using this website, all TDN transactions can be performed. The user must be very familiar with TDNs and have good computer skills. Most of the time the TDN transactions will be performed using TDN apps or special terminals, but quick transfers between two parties can be very efficient using the website. Following is the documentation for the demo website on https://tdnsys.com/demo/default.php. This website is also used in the TDNSYS tutorial.

a. Validate TDN

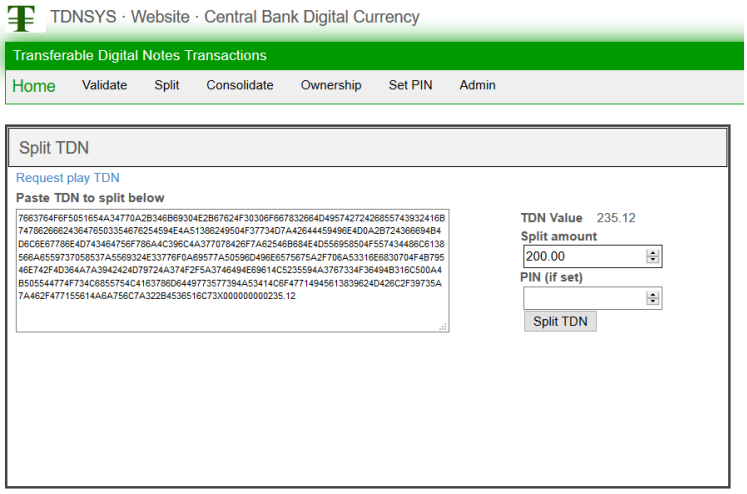


Validating a TDN is available to anybody in the possession of a TDN Signature. The TDN Signature has to be store in a text file or printed as a QR barcode. The user pastes the text or reads the QR barcode using a webcam. The system returns the status and the value of the TDN if the TDN is active. If canceled or the TDN Signature is not in the TDNSYS repository, the system returns an error.

A special case is when the status is Blocked. This means that an authorized authority is investigating possible fraud. The system will display a messge with information on how to contact the investigative authority. If, for example, your TDN is blocked, you can call and get information and maybe help in the investigation.

Figure 15 - Validate TDN screenshot

b. Split TDN



This operation is needed in order to make payments or transfers for a specific amount when the transaction initiator does not have a TDN for an exact value in his/her possession.

The user pastes the TDN Signature or reads the QR Barcode and the desired split amount. If the TDN is protected with a PIN, it must be entered before submitting the request.

The two split TDNs are displayed on the screen as text and QR code.

Figure 16 - Split TDN input screenshot

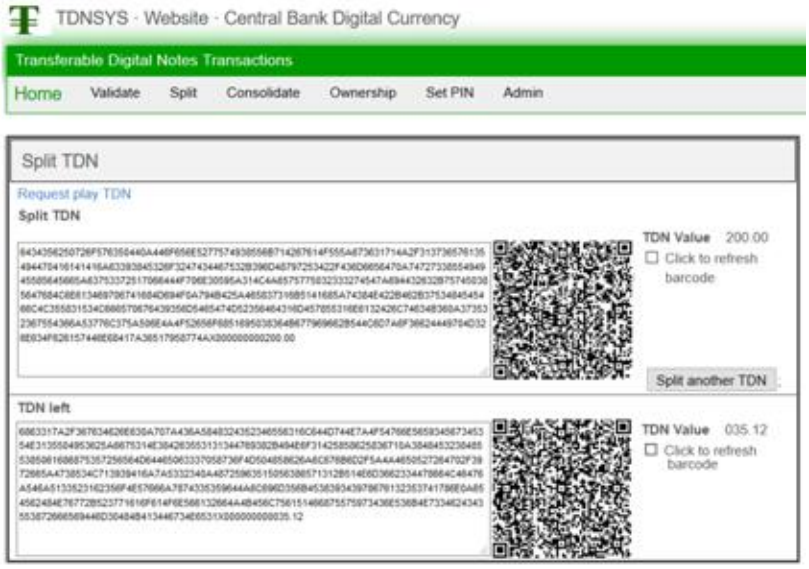
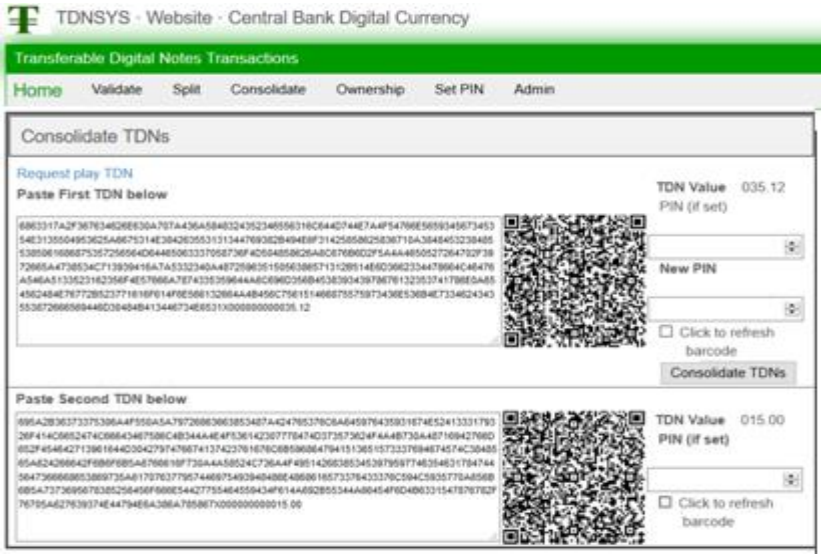


Figure 17 - Split TDN output screenshot

The user can read the QR cods in an application for managing TDNs or copy and paste it into a text file.
One TDN will have the value specified in the input and the second will have the value left from the original TDN.
If a pin was set for the TDN that was spilt, the same PIN is set for the two splits.

c. Consolidate TDNs



This transaction is the reverse of Split. The user will input two TDNs, the PINs if set, and a new PIN for the consolidated TDN.

The result is a new TDN that equals the sum of the two consolidated TDNs.

Figure 18 - Consolidate TDNs input screenshot

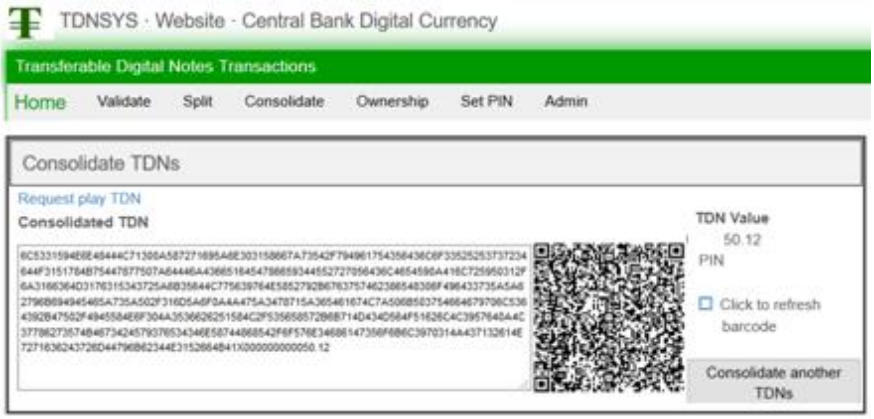


Figure 19 - Consolidate TDN output screenshot

d. Request TDN Ownership

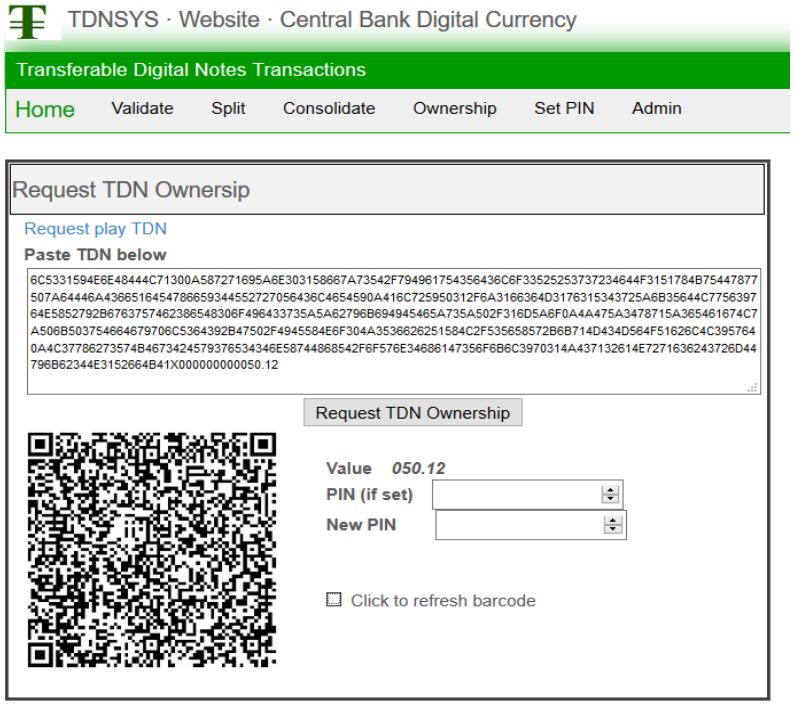


Figure 20 - Request TDN Ownership input screenshot

After getting in the possession of a TDN Signature, the owner has the responsibility to request the ownership of the TDN in order to prevent double spending. This is performed automatically when using Apps, specialized terminals, or eCommerce sites.

If the TDN is received directly in an e-mail, SMS, or by some other means, the request for ownership has to be performed unless the new owner trusts that the previous owner will not double spend or has not unwillingly transferred an already canceled TDN.

An owner of a TDN may request the ownership if he/she thinks the TDN might be compromised. If somebody else was in possession of the TDN Signature illegally after the ownership request is completed, that TDN Signature will no longer be usable.

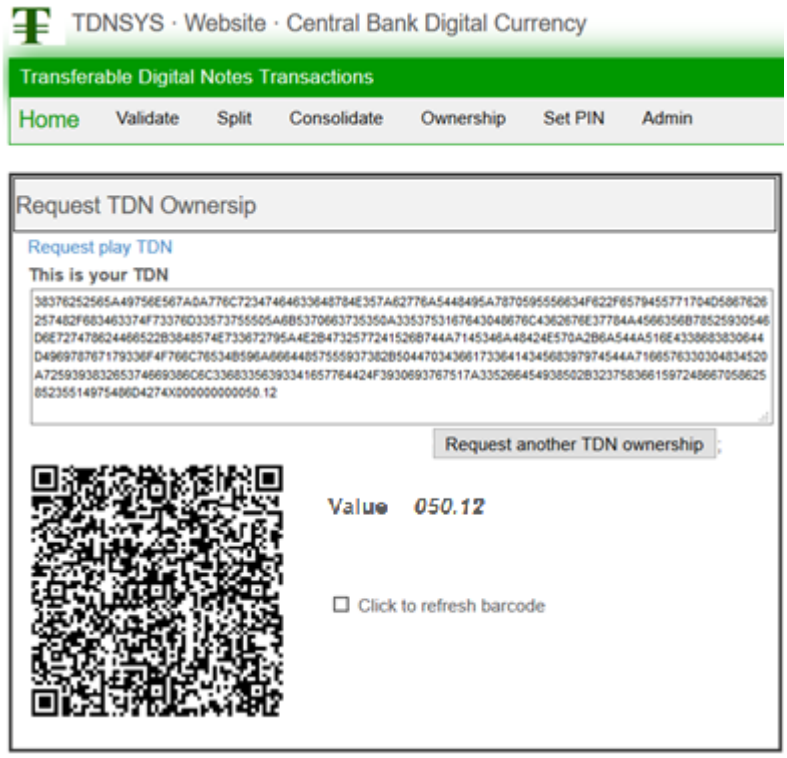
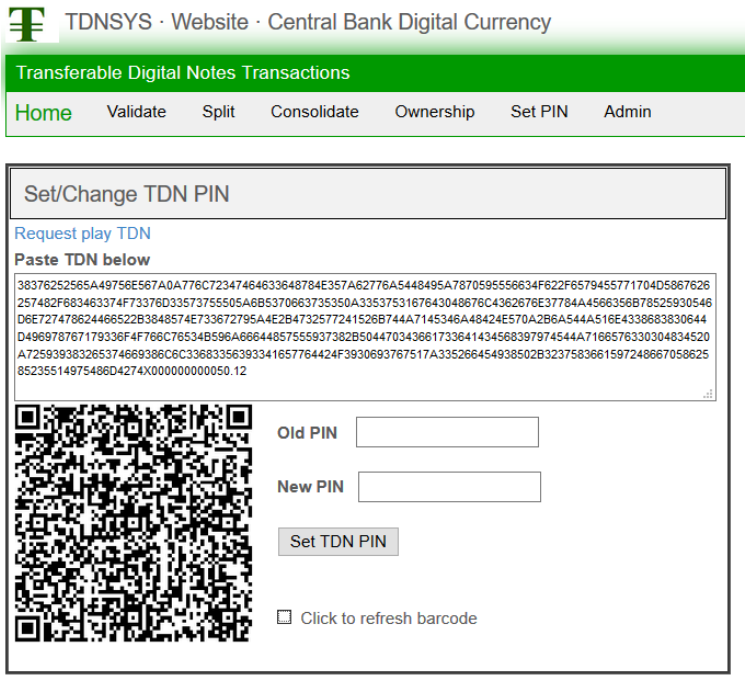


Figure 21 -- Request TDN Ownership output screenshot

e. Set TDN PIN



Changing or removing a PIN is a straight forward operation. If the TDN is not secure with a PIN only the *New PIN* has to be entered. To remove a PIN the *Old PIN* is required and the New PIN is left blank. To change a PIN both fields are required.

Set *Public/Private Key(PPK) Authentication* is not implemented in this demo. A form to process PPK setting will have a button to generate a PPK pair. The public key will be submitted to TDNSYS to update the security PPK field and the private key will be displayed as ASCII text and QR code. The user will have to store it in a safe place and us it when initiation transactions for the protected TDN. All the other forms have to be updated to support PPK Authentication.

The Fed Website should have this functionality implemented for expert users.

Figure 22 - Set TDN PIN screenshot

Most of the time *Public/Private Key (PPK) Authentication* will be implemented in Apps and it will not request user interaction.

Appendix C. Internationalization

It is less likely that there will be an international currency available soon. Each country with a stable economy and financial system has its own currency and transactions which are always performed in the country's currency. Of course, in a country with a failed financial system everybody will be happy to take your Dollars or Euros, but in any developed country you will have to be wary about converting your money into the local currency and most of the time this is not free.

If the Fed starts issuing US Dollars in digital form many other countries will do the same. This will revolutionize the currency exchanges. Such an exchange will instantly convert the TDNs of different countries directly to the owner of TDNs without the need of an intermediary. There may be many exchanges competing for business.

If other countries start issuing currency in digital form there will be multiple TDNSYS issuing TDNs valued in different currency. To differentiate between TDNs issued for different currency the TDN Signature must have a new field identifying that currency. Based on this field a client application can rout the TDN to the appropriate TDNSYS for processing.

For example, a cash register can be designed to support TDNs valued in different currencies. The cash register software will rout the TDN for processing to the appropriate TDNSYS and split, consolidate or request ownership of this TDN.

Currently, if a traveler holds US Dollars in an EU country, he/she will have to find an exchange office, pay a fee, and exchange US Dollars for Euros. This is a very expensive transaction.

If TDNs value in Dollars and Euros are in circulation, the situation is totally different. The traveler may use an app to withdraw a US TDN from his/her bank account and then use a different app to exchange it to an EU TDN. The app will find the best exchange and conversion rate.

The cash exchange offices currently in operation offer most of the time the same exchange rate during the business day. For TDNs the exchange rate will vary continuously based on offers and bids.

Most of the banks will likely offer TDN withdraw to their customers in any currency available. These are just a few possible scenarios.