DLPC Working Group
of the
Innovation Council

DLPC

Distributed Ledger Payment Commitment

Technical Best Practices

Proposed Specifications for Trial Use

April 8, 2019

Table of Contents

Tables of Figures

Tables

Foreword

The Distributed Ledger Payment Commitment Working Group (DLPC Working Group) was established by the BAFT Innovation Council in 2016 to address the payment commitment within the context of trade instruments being processed on a distributed ledger network. Upon feedback from BAFT members, the working group looked into the needs for the digitalization of the full spectrum of trade instruments, from a simple Draft to a complex Letter of Credit. Based on this research, the working group has created best practice specifications for an atomic component of a trade transaction processed on a distributed ledger, the DLPC, which enables a digitized implementation of a wide range of trade instruments involving a promise to pay. BAFT believes that the DLPC component should be an industry standard and is a legally enforceable means to realize electronic negotiable instruments.

The specifications consist of this document together with the accompanying Business Best Practices. The current release of these specifications is titled *Proposed Specifications for Trial Use,* to reflect the fact that we are, as an industry, at an early stage in the implementation of trade instruments on distributed ledger. The workgroup requests feedback from early implementers of these specifications to help advance them into industry standards. We would prefer to receive comments by June 30, but will accept them through August 31. Please direct all submissions to industryfeedback@baft.org.

BAFT would like to express its gratitude to the DLPC Working Group captains and other members who generously contributed their expertise, time, and effort to develop these DLPC Technical Best Practices. These contributors and their affiliations are listed below.

Kitt Carswell	CGI
Hector Fuenzalida	CGI
Andrew Garner	Wells Fargo
Whit Knapp	GTBInsights
Ranga Krishnan	Skuchain
Dave Sutter	TradeIX

Definitions

Committee: The party to the DLPC who is proposed to become or is actually an obligee, to which effect the DLPC carries the required attestations.

Committer: The party to the DLPC who is proposed to become or is actually an obligor, whose attestation to that effect is carried in the DLPC.

DL: Distributed Ledger

DLT: Distributed Ledger Technology

DLPC: A record of a Payment Commitment on a Distributed Ledger that conforms to these Business and Technical Best Practices. A DLPC has a life-cycle comprising the following states: (1) DLPC Created (2) Contingent DLPC (3) Actual DLPC and (4) DLPC Discharged. This life-cycle is described in more detail in Section 2.2.

Network: A DL Business Network is a group of parties that use a Distributed Ledger to conduct business transactions among themselves. As a precondition to joining the Network, these parties must all agree to conduct such business on the Distributed Ledger compliant to the Network rules and governance.

1. Introduction

A DLPC is a record compliant to these specifications, of a payment commitment associated with a trade instrument recorded on a distributed ledger. The associated DLPC Business Best Practices document describes how DLPCs can be used to digitalize common data elements of trade instruments. The DLPC thus becomes a key component that enables the creation of digitalized trade instruments. This document specifies how the DLPC itself is to be implemented on a distributed ledger. The DLPC is designed so that it would be compatible with any DLT platform and can support the digitalization of any trade instruments that include a promise to pay.

2. DLPC Lifecycle

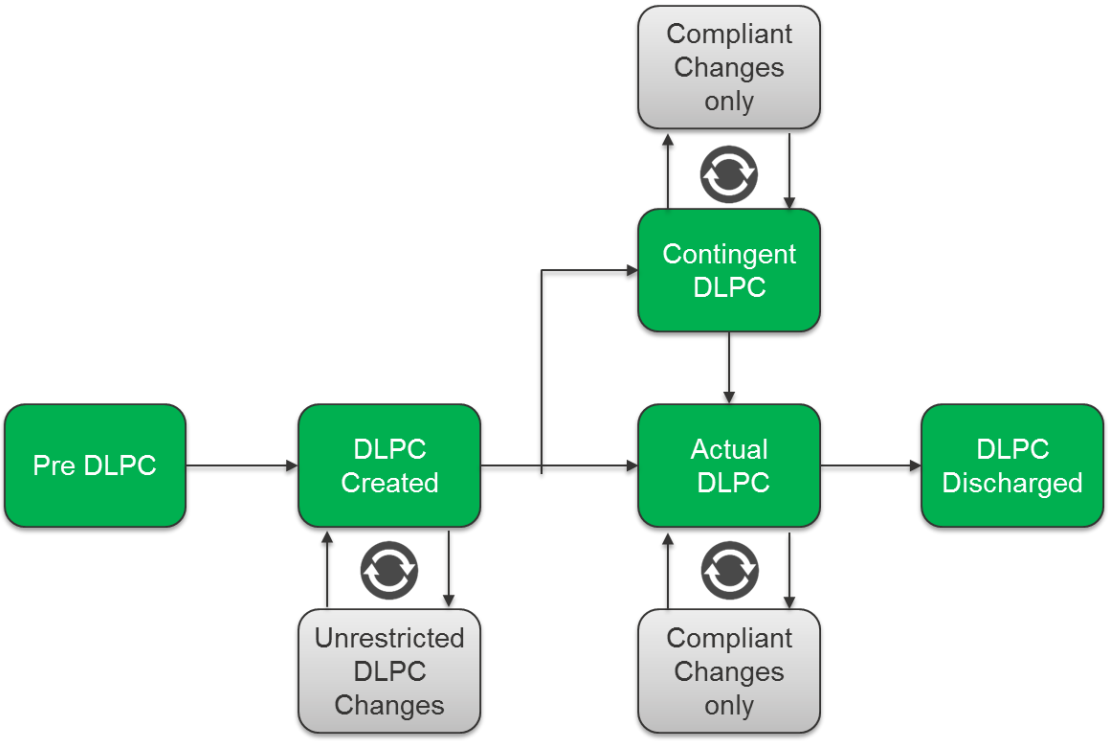

Figure 1: Lifecycle of a DLPC

A DLPC goes through the lifecycle shown in Figure 1. A DLPC comes into existence only when some of the specified data fields for it have been recorded on a distributed ledger. To reach the DLPC Created state, at a minimum, the record must contain the DLPC ID field. The DLPC ID is a globally unique ID which is designed to be unique across networks and ledgers. The DLPC is linked to a specific instrument via the Parent Instrument ID. Prior to reaching the DLPC Created state, any preparatory stages that maintain DLPC data fields in the internal IT systems of any party, are pre DLPC stages as shown in Figure 1.

Once created, if a DLPC does not meet the minimum requirements to be legally binding specified in Section 3 of this document, it can continue to be updated without restriction, until the minimum requirements are met. When the minimum requirements to be legally binding are met, the DLPC would become either a Contingent DLPC or an Actual DLPC. Contingent DLPCs and Actual DLPCs allow only Compliant Changes as specified in Section 4 of this document. One such Compliant Change is the negotiation of an Actual DLPC upon the requisite attestations being made, which could result in a new person being designated as the Committee. The conversion of a Contingent DLPC to an Actual DLPC is another example of a Compliant Change.

An Actual DLPC transitions to a Discharged DLPC when the Committee indicates that they have received the promised payment or are otherwise relieving the Committer of their promise to pay.

3. DLPC Data

Each data field in a DLPC carries attestations from the involved parties. The data fields in the DLPC are designated as mandatory [M] or optional [O] and carry an indication of the attestations required. In order for a DLPC to represent a legally binding commitment:

(a) All mandatory fields must be present

and

(b) All fields must have the required attestations

When a DLPC is a legally binding commitment, it is either in the Contingent DLPC or Actual DLPC states. Each field of a DLPC, together with whether it is mandatory [M] or optional [O], and whether that field needs to be attested to by the Committer [Cr] and/or the Committee [Ce], is described in section 3.1 below.

3.1. Data Fields

1. **DLPC ID** [M, Cr, Ce]: Unique identifier for the DLPC record. This unique identifier or handle allows all state changes of a specific DLPC to be tied together.

2. **Parent Instrument ID** [O, Cr, Ce]: ID of a parent trade instrument, if it exists. The parent instrument governs all its component DLPCs.

3. **Committer** [M, Cr, Ce]: Identifier of the party making the commitment.

4. **Committee** [M, Cr, Ce]: Identifier of the party benefiting from the commitment.

5. **Currency** [M, Cr, Ce]: The currency unit in which the commitment is denominated.

6. **Amount** [M, Cr, Ce]: The amount committed to.

7. **Start Date** [M, Cr, Ce]: The date on which the DLPC becomes effective.

8. **End Date** [M, Cr, Ce]: The date after which the DLPC is no longer effective.

9. **Commitment Type** [M, Cr]: Contingent or Actual indicating whether the commitment is conditional or unconditional.

10. **Commitment Discharged** [O, Ce]: Yes or No indicating if Committer has been discharged. If this field is absent, the committer has not been discharged.

11. **Applicable Rules** [M, Cr, Ce]: A URL that points to the rules governing the DLPC.

3.2. Logical DLPC state

Different distributed ledgers will represent the DLPC data fields and the transactions that change them through different mechanisms. They may also choose different digital signature algorithms and cryptography to record attestations by parties to the DLPC. For this reason only the attestation time stamp is significant for this specification and not the content of the attestation itself. While implementations compliant to this specification have such freedom, they shall on demand provide the logical state of the DLPC shown in Table 1. In this DLPC logical state, the DLPC data fields shall be in the format specified in section 3.1 above. The attestation time stamp shall be in the same format as the time format specified for the Start Date and End Date fields. The attestation timestamp shall be empty/blank if the attestation is not present.

DLPC Field Name	DLPC Data	Committer Attestation Timestamp	Committee Attestation Timestamp
DLPC ID [M]	123456		
Parent Instrument ID [O]	Draft 001		
Committer [M]	Buyer X		
Committee [O]	Seller Y		
Currency [M]	USD		
Amount [M]	$100,000.00		
Start Date [M]	7/1/2018		
End Date [M]	10/31/2018		
Commitment Type [M]	Contingent		
Commitment Discharged [O]	No		
Applicable Rules [M]	https://baft.org/...		

Table 1: DLPC logical state

4. Compliant changes to a DLPC

A field in the DLPC can be changed or added if it is not present, without restriction until the DLPC meets the requirements to be a binding legal commitment. Once the DLPC becomes legally binding only compliant changes that meet the requirements below are allowed. These requirements specify on a per data field basis what attestations are required. It is assumed for the discussion below that only one field changes at a time, and receives the required new attestations, while all other fields retain their prior values and attestations:

1. **DLPC ID** [M]: Change not allowed

2. **Parent Instrument ID** [O]: Committer and Committee

3. **Committer** [M]: New Committer and Current Committee

4. **Committee** [O]: Current Committee

5. **Currency** [M]: Committer and Committee

6. **Amount** [M]:

   (i) Increment: Committer

   (ii) Decrement: Committee

7. **Start Date** [M]: Committer and Committee

8. **End Date** [M]: Committer and Committee

9. **Commitment Type** [M]:

   (i) Change of Contingent to Actual: Committer

   (ii) Change of Actual to Contingent: Not Allowed

10. **Commitment Discharged** [O]:

   (i) No to Yes: Committee

   (ii) Yes to No: Not Allowed

11. **Applicable Rules** [M]: Change not allowed

When the end date of a DLPC is past, and the commitment type is Contingent and the amount is greater than zero, the DLPC is considered expired and the committer is no longer obligated for that residual amount. When the end date of a DLPC is past, and the commitment type is Actual and the amount is greater than zero, and the commitment has not been discharged, the DLPC is considered past due.

5. Data Format/Representation

For each of the fields in the DLPC, the data format that should be used is specified below.

5.1 DLPC ID

The structure of the DLPC ID described here, provides for a globally unique ID that would not be duplicated any party creating a new DLPC. The DLPC ID is constructed by combining 3 elements: LEI, UUID and System Time as described below.

[20 character **LEI**][16 octet (128 bit) **UUID**][**System Time** in milliseconds-Unix format]

5.1.1 LEI

A **Legal Entity Identifier** (or **LEI**) is a 20-character alphanumeric string that identifies distinct legal entities that engage in financial transactions. It is defined by **ISO 17442**. Natural persons are not required to have an LEI; they are eligible to have one issued, however, but only if they act in an independent business capacity. The LEI is a global standard, designed to be non-proprietary data that is freely accessible to all. As of January 29, 2019 over 1.3 million legal entities from more than 195 countries have now been issued with LEIs. An LEI consists of a 20-character alphanumeric string, with the first 4 characters identifying the Local Operating Unit (LOU) that issued the LEI. Characters 5 and 6 are reserved as '00'. Characters 7-18 are the unique alphanumeric string assigned to the organization by the LOU. The final 2 characters are checksum digits. Examples of LEI codes are shown in Table 2 below:

Structure of LEI codes

1 2 3 4	5 6	7 8 9 ... 18	19 20
LOU-Code	Reserved	Entity-Identification	Check-sum
G.E. Financing GmbH			
5493	00	84UKLVMY22DS	16
Jaguar Land Rover Ltd			
2138	00	WSGIIZCXF1P5	72
British Broadcasting Corporation			
5493	00	0IBP32UQZ0KL	24

Table 2: Structure of LEI codes

The Global Legal Entity Identifier Foundation (GLEIF) is not directly issuing Legal Entity Identifiers, but instead it delegates this responsibility to Local Operating Units (LOUs). These LEI issuers supply different services. Local Operating Units can have different prices for the registration services they offer. GLEIF is responsible for monitoring LEI data quality.

5.1.2 UUID

A UUID (Universally Unique Identifier) can be used for multiple purposes, from tagging objects with an extremely short lifetime, to reliably identifying very persistent objects across a network, particularly (but not necessarily) as part of an ASN.1 object identifier (OID) value, or in a Uniform Resource Name (URN). UUIDs are also known as Globally Unique Identifiers (GUIDs), but this term is not used in Rec. ITU-T X.667 | ISO/IEC 9834-8.

UUIDs are an octet string of 16 octets (128 bits). The 16 octets can be interpreted as an unsigned integer encoding, and the resulting integer value can be used as a subsequent arc of {joint-iso-itu-t uuid(25)} (or 2.25) in the OID tree. This enables users to generate OIDs without any registration procedure.

If generated according to one of the mechanisms defined in Rec. ITU-T X.667 | ISO/IEC 9834-8, a UUID is either guaranteed to be different from all other UUIDs generated before 3603 A.D., or is extremely likely to be different (depending on the mechanism chosen). The UUID generation algorithm specified in this standard supports very high allocation rates: 10 million per second per machine if necessary, so UUIDs can also be used as transaction IDs.

No centralized authority is required to administer UUIDs but automatic generation of UUIDS (using the algorithm defined in Rec. ITU-T X.667 | ISO/IEC 9834-8) is provided at Get a new UUID. An example was requested at 4:45 ET on 8/29/2018 with the result:

Online UUID Generator

Your Version 4 UUID:

300dfa9b-1b33-4798-badc-4cd9e1cff8e4

Refresh page to generate another.

UUIDs forming a component of an OID are represented in ASN.1 value notation as the decimal representation of their integer value, but for all other display purposes it is more usual to represent them with hexadecimal digits with a hyphen separating the different fields within the 16-octet UUID. This representation is defined in Rec. ITU-T X.667 | ISO/IEC 9834-8.

Example:

f81d4fae-7dec-11d0-a765-00a0c91e6bf6 is the hexadecimal notation that denotes the same UUID as 329800735698586629295641978511506172918 in decimal notation.

5.1.3 System Time

System time is measured by a *system clock*, which is typically implemented as a simple count of the number of *ticks* that have transpired since some arbitrary starting date, called the *epoch*. For example, Unix and POSIX-compliant systems encode system time ("Unix time") as the number of seconds elapsed since the start of the Unix epoch at 1 January 1970 00:00:00 UT, with exceptions for leap seconds.

System time can be converted into calendar time, which is a form more suitable for human comprehension. For example, the Unix system time *1000000000* seconds since the beginning of the epoch translates into the calendar time *9 September 2001 01:46:40 UT*. Library subroutines that handle such conversions may also deal with adjustments for time zones, daylight saving time (DST), leap seconds, and the user's locale settings. Library routines are also generally provided that convert calendar times into system times.

5.2 Parent Instrument ID

The Parent Instrument ID is an arbitrary ASCII string of up to 64 characters. It is however strongly recommended that like the DLPC ID the Parent Instrument ID begin with the LEI of the organization that created the Parent Instrument and augment that with a globally unique identifier.

5.3 Committer

It is recommended that the LEI as described in Section 5.1.1 be used to identify the Committer. If LEI is available, the content of this field should be limited to the LEI. However if LEI is not an option, the legal name of the Committer, together with their registered address may be used. Additional IDs such as Tax ID or other national/regional IDs can be included with the legal name/address, but if there is a conflict between the legal name/address and any additional ID, then the legal name/address will be used to determine identity and any conflicting IDs will be ignored.

5.4 Committee

It is recommended that the LEI as described in Section 5.1.1 be used to identify the Committee. If LEI is available, the content of this field should be limited to the LEI. However if LEI is not an option, the legal name of the Committer, together with their registered address may be used. Additional IDs such as Tax ID or other national/regional IDs can be included with the legal name/address, but if there is a conflict between the legal name/address and any additional IDs, then the legal name/address will be used and any conflicting IDs will be ignored.

5.5 Currency

The Currency shall be identified by its **ISO 4217** code

5.6 Amount

The Amount shall be a decimal number. The integer part can have up to 24 numerals. The fractional part can have up to 7 numerals.

5.7 Start Date

The Start Date shall be **ISO 8601** compliant and specifically in the format that Javascript specifies for the JSON representation of its Date object. Below is an example of this JSON output:

2012-04-23T18:25:43.511Z

5.8 End Date

The End Date shall be ISO 8601 compliant and specifically in the format that Javascript specifies for the JSON representation of its Date object. Below is an example of this JSON output:

2012-04-23T18:25:43.511Z

5.9 Commitment Type

The Commitment Type field shall contain either the ASCII string “Contingent” or “Actual”.

5.10 Commitment Discharged

The Commitment Discharged field shall contain either the ASCII string “Yes” or “No”.

5.11 Applicable Rules

The Applicable Rules field shall contain a URL that points to the rules governing the DLPC. An example of such rules is section *4.2 DLPC Contract Terms*, of the accompanying Business Best Practices Specification. To have these rules apply to a DLPC, this field shall contain a URL that points to section 4.2 of the Business Best Practices Specification hosted on the BAFT website.