**Analysis**

**Audience**: Architects, Application and smart contract developers, Business professionals

Let’s analyze commercial paper in a little more detail. PaperNet participants such as MagnetoCorp and DigiBank use commercial paper transactions to achieve their business objectives – let’s examine the structure of a commercial paper and the transactions that affect it over time. We will also consider which organizations in PaperNet need to sign off on a transaction based on the trust relationships among the organizations in the network. Later we’ll focus on how money flows between buyers and sellers; for now, let’s focus on the first paper issued by MagnetoCorp.

**Commercial paper lifecycle**

A paper 00001 is issued by MagnetoCorp on May 31. Spend a few moments looking at the first **state** of this paper, with its different properties and values:

Issuer **=** MagnetoCorp

Paper **=** 00001

Owner **=** MagnetoCorp

Issue date **=** 31 May 2020

Maturity **=** 30 November 2020

Face value **=** 5M USD

Current state **=** issued

This paper state is a result of the **issue** transaction and it brings MagnetoCorp’s first commercial paper into existence! Notice how this paper has a 5M USD face value for redemption later in the year. See how the Issuer and Owner are the same when paper 00001 is issued. Notice that this paper could be uniquely identified as MagnetoCorp00001 – a composition of the Issuer and Paper properties. Finally, see how the property Current state = issued quickly identifies the stage of MagnetoCorp paper 00001 in its lifecycle.

Shortly after issuance, the paper is bought by DigiBank. Spend a few moments looking at how the same commercial paper has changed as a result of this **buy** transaction:

Issuer **=** MagnetoCorp

Paper **=** 00001

Owner **=** DigiBank

Issue date **=** 31 May 2020

Maturity date **=** 30 November 2020

Face value **=** 5M USD

Current state **=** trading

The most significant change is that of Owner – see how the paper initially owned by MagnetoCorp is now owned by DigiBank. We could imagine how the paper might be subsequently sold to BrokerHouse or HedgeMatic, and the corresponding change to Owner. Note how Current state allow us to easily identify that the paper is now trading.

After 6 months, if DigiBank still holds the commercial paper, it can redeem it with MagnetoCorp:

Issuer **=** MagnetoCorp

Paper **=** 00001

Owner **=** MagnetoCorp

Issue date **=** 31 May 2020

Maturity date **=** 30 November 2020

Face value **=** 5M USD

Current state **=** redeemed

This final **redeem** transaction has ended the commercial paper’s lifecycle – it can be considered closed. It is often mandatory to keep a record of redeemed commercial papers, and the redeemed state allows us to quickly identify these. The value of Owner of a paper can be used to perform access control on the **redeem** transaction, by comparing the Owner against the identity of the transaction creator. Fabric supports this through the [getCreator() chaincode API](https://github.com/hyperledger/fabric-chaincode-node/blob/master/fabric-shim/lib/stub.js" \l "L293). If golang is used as a chaincode language, the [client identity chaincode library](https://github.com/hyperledger/fabric-chaincode-go/blob/master/pkg/cid/README.md) can be used to retrieve additional attributes of the transaction creator.

**Transactions**

We’ve seen that paper 00001’s lifecycle is relatively straightforward – it moves between issued, trading and redeemed as a result of an **issue**, **buy**, or **redeem** transaction.

These three transactions are initiated by MagnetoCorp and DigiBank (twice), and drive the state changes of paper 00001. Let’s have a look at the transactions that affect this paper in a little more detail:

**Issue**

Examine the first transaction initiated by MagnetoCorp:

Txn **=** issue

Issuer **=** MagnetoCorp

Paper **=** 00001

Issue time **=** 31 May 2020 09:00:00 EST

Maturity date **=** 30 November 2020

Face value **=** 5M USD

See how the **issue** transaction has a structure with properties and values. This transaction structure is different to, but closely matches, the structure of paper 00001. That’s because they are different things – paper 00001 reflects a state of PaperNet that is a result of the **issue** transaction. It’s the logic behind the **issue** transaction (which we cannot see) that takes these properties and creates this paper. Because the transaction **creates** the paper, it means there’s a very close relationship between these structures.

The only organization that is involved in the **issue** transaction is MagnetoCorp. Naturally, MagnetoCorp needs to sign off on the transaction. In general, the issuer of a paper is required to sign off on a transaction that issues a new paper.

**Buy**

Next, examine the **buy** transaction which transfers ownership of paper 00001 from MagnetoCorp to DigiBank:

Txn **=** buy

Issuer **=** MagnetoCorp

Paper **=** 00001

Current owner **=** MagnetoCorp

New owner **=** DigiBank

Purchase time **=** 31 May 2020 10:00:00 EST

Price **=** 4.94M USD

See how the **buy** transaction has fewer properties that end up in this paper. That’s because this transaction only **modifies** this paper. It’s only New owner = DigiBank that changes as a result of this transaction; everything else is the same. That’s OK – the most important thing about the **buy** transaction is the change of ownership, and indeed in this transaction, there’s an acknowledgement of the current owner of the paper, MagnetoCorp.

You might ask why the Purchase time and Price properties are not captured in paper 00001? This comes back to the difference between the transaction and the paper. The 4.94 M USD price tag is actually a property of the transaction, rather than a property of this paper. Spend a little time thinking about this difference; it is not as obvious as it seems. We’re going to see later that the ledger will record both pieces of information – the history of all transactions that affect this paper, as well its latest state. Being clear on this separation of information is really important.

It’s also worth remembering that paper 00001 may be bought and sold many times. Although we’re skipping ahead a little in our scenario, let’s examine what transactions we **might** see if paper 00001 changes ownership.

If we have a purchase by BigFund:

Txn **=** buy

Issuer **=** MagnetoCorp

Paper **=** 00001

Current owner **=** DigiBank

New owner **=** BigFund

Purchase time **=** 2 June 2020 12:20:00 EST

Price **=** 4.93M USD

Followed by a subsequent purchase by HedgeMatic:

Txn **=** buy

Issuer **=** MagnetoCorp

Paper **=** 00001

Current owner **=** BigFund

New owner **=** HedgeMatic

Purchase time **=** 3 June 2020 15:59:00 EST

Price **=** 4.90M USD

See how the paper owners changes, and how in our example, the price changes. Can you think of a reason why the price of MagnetoCorp commercial paper might be falling?

Intuitively, a **buy** transaction demands that both the selling as well as the buying organization need to sign off on such a transaction such that there is proof of the mutual agreement among the two parties that are part of the deal.

**Redeem**

The **redeem** transaction for paper 00001 represents the end of its lifecycle. In our relatively simple example, HedgeMatic initiates the transaction which transfers the commercial paper back to MagnetoCorp:

Txn **=** redeem

Issuer **=** MagnetoCorp

Paper **=** 00001

Current owner **=** HedgeMatic

Redeem time **=** 30 Nov 2020 12:00:00 EST

Again, notice how the **redeem** transaction has very few properties; all of the changes to paper 00001 can be calculated data by the redeem transaction logic: the Issuer will become the new owner, and the Current state will change to redeemed. The Current owner property is specified in our example, so that it can be checked against the current holder of the paper.

From a trust perspective, the same reasoning of the **buy** transaction also applies to the **redeem** instruction: both organizations involved in the transaction are required to sign off on it.

**The Ledger**

In this topic, we’ve seen how transactions and the resultant paper states are the two most important concepts in PaperNet. Indeed, we’ll see these two fundamental elements in any Hyperledger Fabric distributed [ledger](https://hyperledger-fabric.readthedocs.io/en/latest/ledger/ledger.html) – a world state, that contains the current value of all objects, and a blockchain that records the history of all transactions that resulted in the current world state.

The required sign-offs on transactions are enforced through rules, which are evaluated before appending a transaction to the ledger. Only if the required signatures are present, Fabric will accept a transaction as valid.

You’re now in a great place translate these ideas into a smart contract. Don’t worry if your programming is a little rusty, we’ll provide tips and pointers to understand the program code. Mastering the commercial paper smart contract is the first big step towards designing your own application. Or, if you’re a business analyst who’s comfortable with a little programming, don’t be afraid to keep dig a little deeper!