RECORDS KEEPER

D-313, Second Floor,
Village Mohammadpur,
RK Puram, New Delhi – 110066
Cell: +91-9910433954

MultiChain

Private Blockchain Platform

Working with MultiChain Streams

For shared immutable key–value and time series databases

MultiChain streams enable a Blockchain to be used as a general purpose append-only database, with the Blockchain providing timestamping, notarization and immutability. A MultiChain Blockchain can contain any number of streams, where the data published in every stream is stored by every node. If a node chooses to subscribe to a stream, it will index that stream's contents to enable efficient retrieval in various ways.

Each stream is an ordered list of items, in which each item has the following characteristics:

- One or more publishers who have digitally signed that item.
- A key between 0 and 256 bytes in length.
- Some data, which can reach many megabytes in size.
- Information about the item's transaction and block, including its txid, blockhash, blocktime, and so on.

Like native assets, MultiChain streams can be referred to in any of three ways:

- An optional stream name, chosen at the time of stream creation. If used, the name must be unique on a blockchain, between both assets and streams. Stream names are stored as UTF-8 encoded strings up to 32 bytes in size and are case insensitive.
- A createtxid, containing the txid of the transaction in which the stream was created.
- A streamref which encodes the block number and byte offset of the stream creation transaction, along with the first two bytes of its txid.

If root-stream-name in the blockchain parameters is a non-empty string, it defines a stream which is created with the blockchain and can be written to immediately. The root stream's createtxid is the txid of the coinbase of the genesis block, and its streamref is 0-0-0.

D-313, Second Floor,
Village Mohammadpur,
RK Puram, New Delhi – 110066
Cell: +91-9910433954

For more background, please see the blog post on streams. If you are looking for documentation on quasi streams (deprecated), click here.

Reference: The above text has been taken from MultiChain website as it is. Please click here to see the actual page.

Streams actually give a very practical option to developers to build anything you want to build with MySQL, NoSQL or Dynamo DB or in-fact any kind of database system. The only difference will be that it will be a decentralized database having a key-value pair at the fundamental level in chronological order.

Step 1: Lets create a stream and try to publish & retrieve some data into or from it.

Now let's create a stream, which can be used for general data storage and retrieval. On the Node-1:

```
create stream stream1 false
```

The false means the stream can only be written to by those with explicit permissions. Let's see its permissions:

```
listpermissions stream1.*
```

So for now, only the first server has the ability to write to the stream, as well as administrate it. Let's publish something to it, with key key1:

```
publish stream1 key1 73747265616d2064617461
```

The txid of the stream item is returned. Now let's see that the stream is visible on another node. On the Node-2:

```
liststreams
```

(The root stream was in the blockchain by default.) Now we want the Node-2 to subscribe to the stream, then view its contents:

```
subscribe stream1
liststreamitems stream1
```

Now we want the Node-2 to be allowed to publish to the stream. On the Node-1:

```
grant 1bDkf8QFnAY3Sf5Fr3iNDYTWvQXNqRAK8gqNts receive,send
grant 1bDkf8QFnAY3Sf5Fr3iNDYTWvQXNqRAK8gqNts stream1.write
```

Note that the address needs both general send/receive permissions for the blockchain, as well as permission to write to this specific stream. Now let's publish a couple of items on the Node-2:

```
publish stream1 key1 736f6d65206f746865722064617461
publish stream1 key2 53747265616d732052756c6521
```

Now let's query the stream's contents in many different ways. Back on the Node-1:

```
subscribe stream1
liststreamitems stream1
```
(should show 3 items)

RECORDS KEEPER

D-313, Second Floor,
Village Mohammadpur,
RK Puram, New Delhi – 110066
Cell: +91-9910433954

liststreamkeys stream1 (2 keys)
liststreamkeyitems stream1 key1 (2 items with this key)
liststreampublishers stream1 (2 publishers)
liststreampublisheritems stream1 1bDkf8QFnAY3Sf5Fr3iNDYTWvQXNqRAK8gqNts (2 items by this publisher)

This is just a taste of the ways in which streams can be queried – for more information, please consult the API documentation.

That's it for this lecture.

If you are interested in conducting Blockchain training in your city, office or country please reach out to us using this link or drop us an email at training@recordskeeper.co.