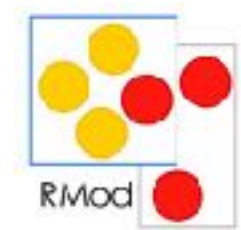


A living programming environment for a living blockchain



by Santiago Bragagnolo - PharoDays - 2017
santiago.bragagnolo@gmail.com
santiago.bragagnolo@inria.fr
[@skype:santiago.bragagnolo](https://www.skype.com/people/santiago.bragagnolo)
[@sbragagnolo](https://www.github.com/sbragagnolo)





Disclaimer!

This is not a blockchain mechanisms talk!
(Sorry disappoint you :))

General technology explanation



Starting by the fruit: Smart contracts

- Digital reification of contracts
 - Emulate the logic of contractual clauses
 - Self-executing
 - Self-enforcing
- Reduce transactional costs
- Minimise exceptions



Following by the branch: Ethereum

- Blockchain based technology
- Open source & public network
- Smart contracts
 - State stored in a blockchain
 - Byte-code executed in the turing complete EVM
 - Many development languages (solidity, serpent, etc)



Arriving to the trunk: Blockchain

- Open and distributed ledger
- Records a constantly-growing list of transactions in between two parties. (blocks)
- Resistant to modification by design
- Cryptocurrency: Paying to reinforce the social engagement with the security

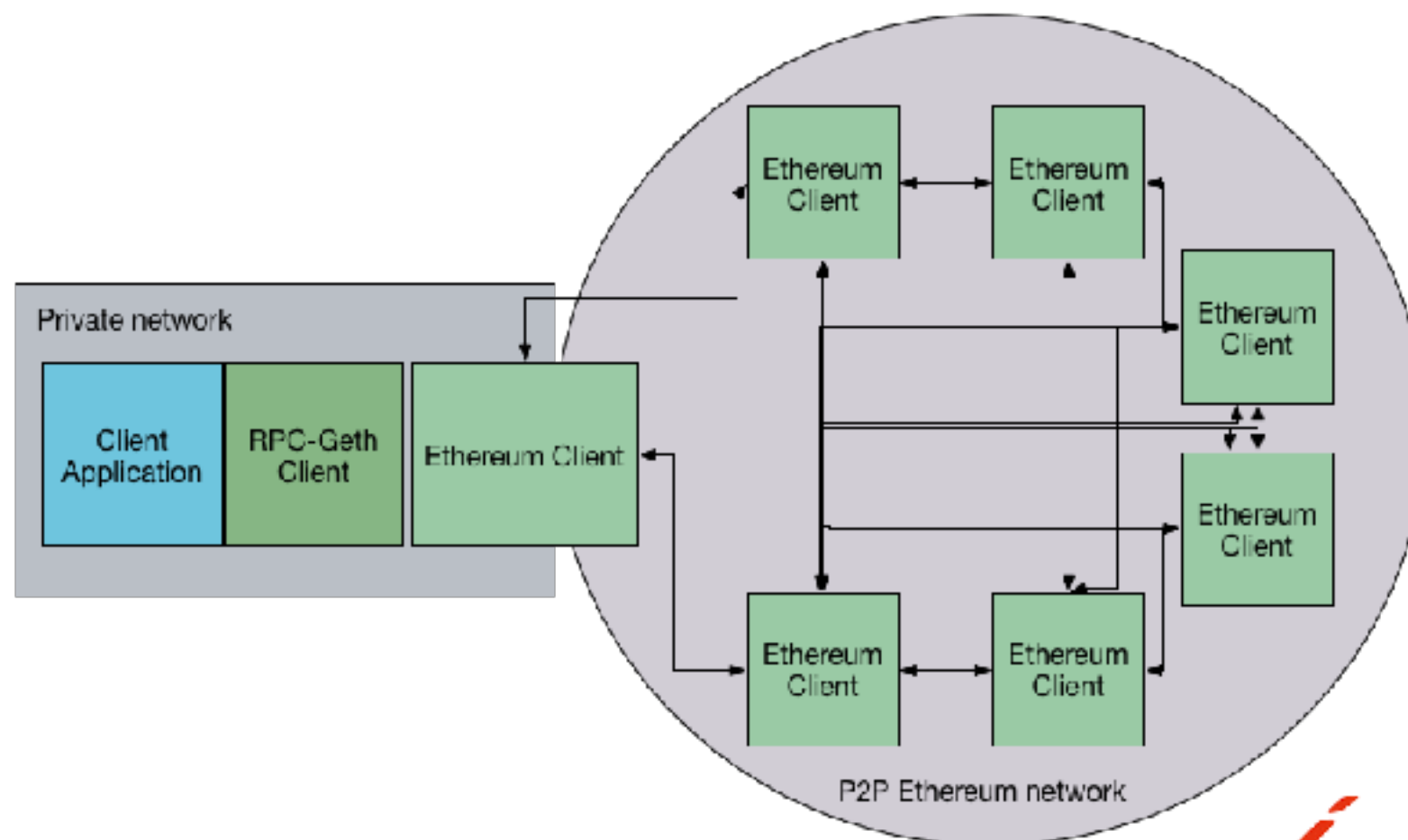


First-citizens in Blockchain

- Block: stamped batch of transactions
- Transaction: Representation of mutations of state
 - Movements of money
 - Method activation
- Account: Source and target of transactions (account in the accountancy meaning)
- Contracts (Specific in ethereum)



So what? Architecture of a proposed application



Pharo



Pharo: Why?

- Blockchain is a multiple actors always growing environment.
- Blockchain is a living environment
 - Transactions move money (ether - bitcoin) from one place to other
 - Transactions execute smart contracts
- Ethereum is a distributed runtime. Nothing better than a live environment for a living distributed runtime.
- A lot of code analysis and inspection state-of-the-art tools



Fog

- Pharo client for the Ethereum client (Geth)
- github.com/sbragagnolo/Fog



Fog - features

- Connection, communication, marshalling, etc.
- Block fetching
- Query and create transactions
- Query and create contracts
- Remote method invocation



Fog - features

- Development support
 - First-class citizen navigation (GT-Tools)
 - Accounts
 - Blocks
 - Transactions
 - Contracts
 - Automatic contract mirror generation
 - Automatic contract proxy building



Fog - features

- Cache
 - General
 - Connection
 - Session

Some fancy slides :)



Block inspection

- Navigating blocks
- Inspecting blocks individually
- Overview of a collection of blocks through statistics
- Overview of the transactions of a collection of blocks



Navigating in blocks

The screenshot displays a web application interface with two main panels. The left panel, titled 'Inspector on an OrderedCollection [10001 items]', shows a list of blocks. The right panel, titled 'an ETHBlock', provides detailed information about a selected block.

Left Panel: Inspector on an OrderedCollection [10001 items]

Variable	Value
() self	an Array [10240 items] ('Block- 0 Created 17192:14:56:53.491614 ago 7 tran...
1	'Block- 0 Created 17192:14:56:53.523614 ago 7 transaction(s)'
2	'Block- 0 Created 17192:14:56:53.527671 ago 3 transaction(s)'
3	'Block- 0 Created 17192:14:56:53.539601 ago 3 transaction(s)'
4	'Block- 0 Created 17192:14:56:53.543794 ago 0 transaction(s)'
5	'Block- 0 Created 17192:14:56:53.567485 ago 3 transaction(s)'
6	'Block- 0 Created 17192:14:56:53.56137 ago 1 transaction(s)'
7	'Block- 0 Created 17192:14:56:53.571782 ago 1 transaction(s)'
8	'Block- 0 Created 17192:14:56:53.579086 ago 1 transaction(s)'
9	'Block- 0 Created 17192:14:56:53.587617 ago 0 transaction(s)'
10	'Block- 0 Created 17192:14:56:53.59537 ago 1 transaction(s)'
11	'Block- 0 Created 17192:14:56:53.609677 ago 7 transaction(s)'
12	'Block- 0 Created 17192:14:56:53.613171 ago 0 transaction(s)'
13	'Block- 0 Created 17192:14:56:53.620486 ago 0 transaction(s)'
14	'Block- 0 Created 17192:14:56:53.628448 ago 5 transaction(s)'

Below the table, a text area shows a snippet of the array's content:

```
"an Array('Block- 0 Created 17192:14:56:53.697659 ago 7 transaction(s) ' 'Block- 0 Created 17192:14:... 'Block- 0 Created 17192:14:56:53.915439 ago 0 transaction(s) ' 'Block- 0 Created 17192:1...etc...'
self
```

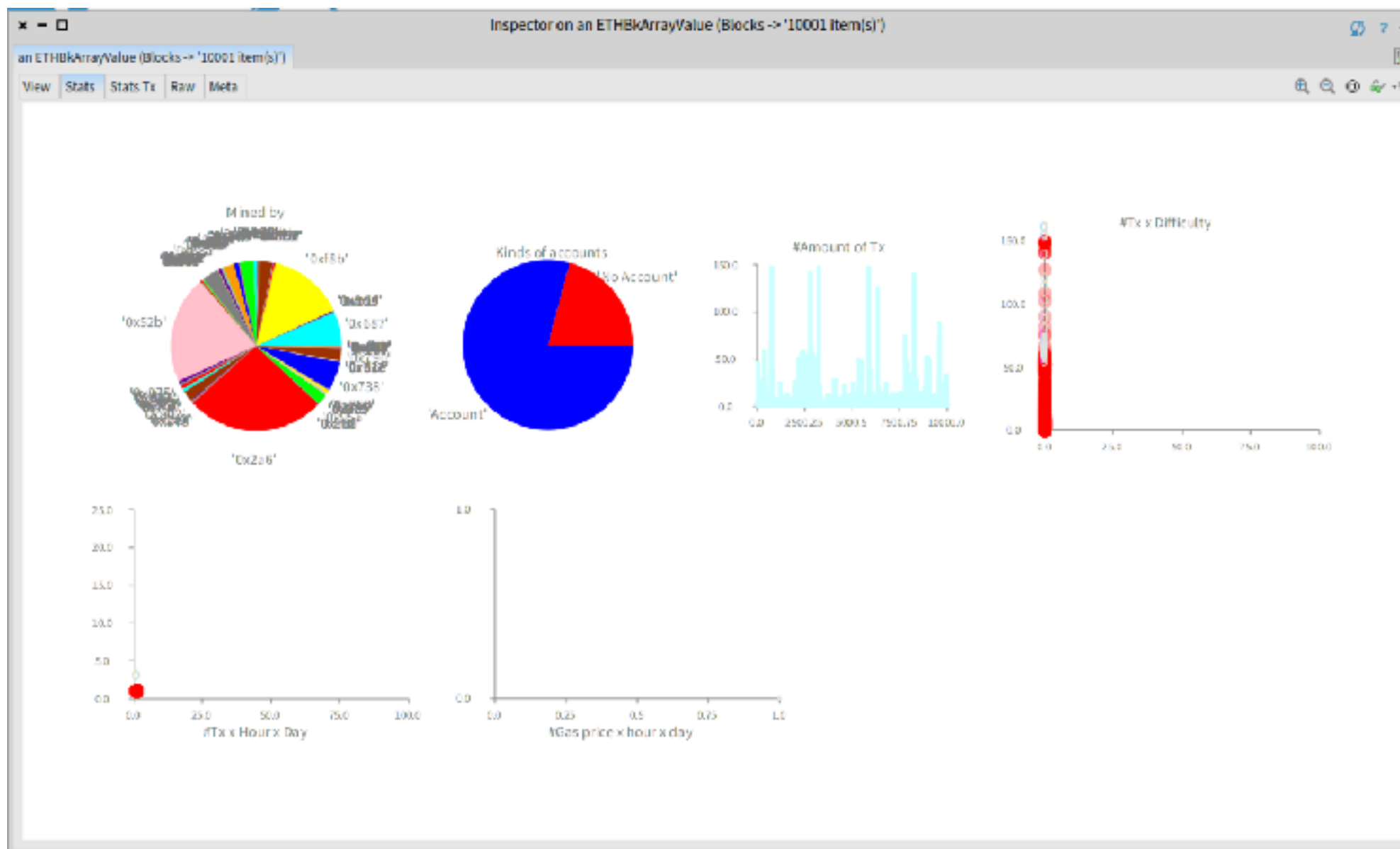
Right Panel: an ETHBlock

The right panel displays the details of a selected block:

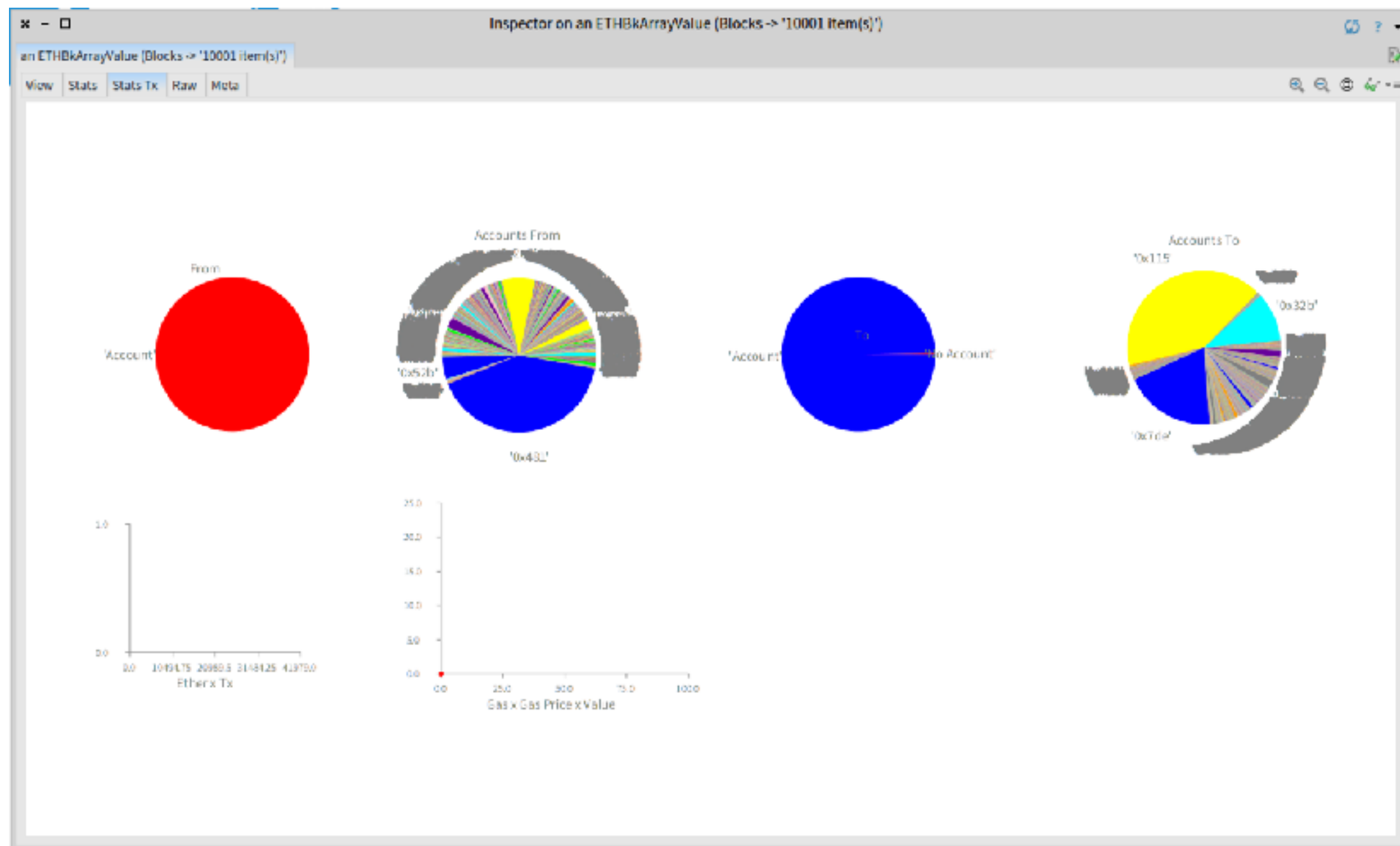
```
'Block- 0 Created 17192:14:58:11.400115 ago 7 transaction(s) '
transactions -> '7 transaction(s) AVG gas:0 AVG gas-price:0 AVG ether:0'
parent -> 'Block- 0 Created 17192:14:58:11.405058 ago 0 transaction(s) '
uncles -> '0 item(s)'
hash -> '0x025fa3d7601cc0a3a9296541c62cc09f1a689bdf508debe85da3e823796efb6d'
```



Blocks overview



Transactions overview





Contract source code

```
pragma solidity ^0.4.2;

contract StructTestContract {

    enum myenum { A, B, C }

    struct mystruct {
        bool boolean;
        myenum uservalue;
        uint32 commonvalue;
    }
    address _owner;
    bool bool1;
    int16 midint;
    mystruct simpleExample;
    bool bool2;
    mystruct[] arrayExample;

    function StructTestContract (){
        _owner = msg.sender;
        bool1 = true;
        bool2 = true;
        midint = 32;
        simpleExample.boolean = true;
        simpleExample.uservalue = myenum.B;
        simpleExample.commonvalue = 6355432;
        arrayExample.push(mystruct(true, myenum.A, 134));
        arrayExample.push(mystruct(false, myenum.B, 235));
        arrayExample.push(mystruct(true, myenum.C, 34));
    }
    function kill() {
        suicide(_owner);
    }
}
```



Inspecting contract

Inspector on an ETHContractInstanceBind

an ETHContractInstanceBind

Block View Inspector Raw Meta

Name	Value
'_owner'	'0xb4ebf466889c4a0239379125a7d0f9c4e8bf2a14'
'bool1'	true
'midint'	32
'simpleExample'	a Dictionary [3 items] ('boolean'->true 'commonvalue'->6355432 'uservalue'->'B')
'bool2'	true
'arrayExample'	'an array of 3 elements '

Quick selection field. Given your INPUT, it executes: self select: [:each | INPUT]



Inspecting structs

Inspector on an ETHContractInstanceBind

an ETHContractInstanceBind

Block View Inspector Raw Meta

Name	Value
'_owner'	'0xb4ebf466889c4a0239379125a7d0f9c4e81'
'bool1'	true
'midint'	32
'simpleExample'	a Dictionary [3 items] ('boolean'->true 'com
'bool2'	true
'arrayExample'	'an array of 3 elements'

Quick selection field. Given your INPUT, it executes: self select: [:each | INPUT]

a Dictionary [3 items] ('boolean'->true 'commonvalue'->6355432 'usei...

Items Keys Raw Meta

Key	Value
'boolean'	true
'uservalue'	'B'
'commonvalue'	6355432

Quick selection field. Given your INPUT, it executes: self select: [:each | INPUT]



Yet to implement

Inspector on an ETHContractInstanceBind

an ETHContractInstanceBind

Block View Inspector Raw Meta

Name	Value
'_owner'	'0xb4cbf466889c4a0239379125a7d0f9c4c8l
'bool1'	true
'midint'	32
'simpleExample'	a Dictionary [3 items] ('boolean' -> true 'com
'bool2'	true
'arrayExample'	'an array of 3 elements'

Quick selection field. Given your INPUT, it executes: self select: [:each | INPUT]

a ByteString ('an array of 3 elements')

String Items Raw Meta

an array of 3 elements

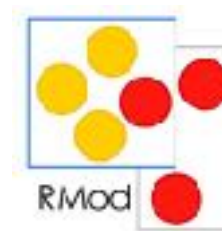


Fog - Demo



Fog - future

- Finishing session management
- Events support
- Transactional message send recognition
- New AST Definition (Henrique Rocha)



THANKS :)!

by Santiago Bragagnolo - PharoDays - 2017
santiago.bragagnolo@gmail.com
santiago.bragagnolo@inria.fr
[@sbragagnolo](https://www.skype.com/people/santiago.bragagnolo)