Blockchainization
A first period

http://rmod.lille.inria.fr

Stéphane Ducasse
I think we should build a blockchain.

Uh-oh.

Does he understand what he said or is it something he saw in a trade magazine ad?

What color do you want that blockchain?

I think mauve has the most RAM.

I’ve hired a consultant to help us evolve our products to use blockchain technology.

Blockchain! Blockchain! Blockchain! Blockchain! Blockchain!

It’s as if you’re a technologist and a philosopher all in one!

Blockhain.

Sidechains.
A word of presentation
Expert in language design

Traits in Php7.0, Perl60
Influenced Traits in Scala
Pharo
Expert in Software Evolution

Program analysis, metamodels, visualization, testing, refactorings, metrics, smart contracts....
Roadmap

Blockchain in 3 slides
Challenges for Blockchainization
The real face of legacy
How we fight the dark force
Blockchain Software Engineering
Blockchain...

the new graal
What is Blockchain?
From an User’s Perspective

A secured open distributed ledger that promotes privacy
What is Blockchain?  
From an User’s Perspective

An open distributed Ledger that promotes privacy
• Ledger - Records economic transactions of cryptocurrency (BitCoin, Ether).
• Open - anyone can participate on and inspect the transactions
• Distributed - the ledger is shared and synchronised among several “people”
• Privacy - the ledger transactions promote anonymity
What is Blockchain?
From a Developer’s Perspective

Roughly, an append-only globally shared transactional database
What is Blockchain? From a Developer’s Perspective

Roughly, an append-only globally shared transactional database

• Database — stores and manages digital resources
• Transaction — ACID properties
• Append-only — immutable data (cannot delete or alter), just append new information
• Globally Shared — managed by a P2P network, all peers store a complete copy of the database
Smart Contracts
[Fritz Henglein, Smart Contracts…, 2017]

“Smart Contracts are neither Smart nor Contracts”

• Smart Contracts are self-executing programs coded in complex Turing-complete language.

• Rules & Actions intermixed
  (not really a contract)

• Low-level programs hard to analyze
  (not really smart)
Smart Contracts
[Henrique Rocha, CBSoft, 2018]

Smart Contracts are objects deployed in the blockchain.

• Code and data (like a class) residing at a Blockchain address.
• Transactions created by function execution.
• Contracts are sandboxed in the Blockchain.
Challenges for blockchainisation

Data
Flow
Architecture
Architecture of trust
At what cost?
Challenges ... data

How data created/manipulated/merged?
What part of data stored?
Is the data manipulated/stored till needed?
Where data is stored?
Challenges … flow

How data flows between subcomponents/subsystems?

How to capture existing process to hook into?
Challenges … trust

What are the trustees?
What is the impact of trust on the architecture (off-chain/on-chain)?
The Real Faces of Legacy
Lack of Knowledge

- **Obsolete** or **no** documentation
- **Departure** of the original developers or users
- Disappearance of **inside** knowledge about the system
- **Limited** understanding of entire system
- **Missing** tests
- **No** versioning
Code symptoms

- Duplicated code (up to 60% on real systems)
- Dead code (up to 2/3 on real systems)
- Code smells
- Big build times
Process symptoms

- *Too long* to turn things to production
- Need for *constant* bug fixes
- Maintenance dependencies
- Difficulties separating products
- Simple changes take too long
Some true examples

• Company C has a super large VB application. No idea of the use of underlying windows API. Can they migrate incrementally?

• Company B has 60% code duplication of Cobol. The single developer left to competitor. No tests. 65 M CHF to replace the system (SAP).
Some true examples

• Company K has 700,000 fine. Legacy is a post cobol system language running half of their business: 4500 programs. No other information.
• Company X has no versioning control, no automated tests, ship to clients new version without patch Y. Pascal + form from 90s.
• Company Z has a large Java6 app with many dependencies on obsolete/unmaintained “solutions”. Hell of dependencies.
Software is Complex
The big ball of mud
An instant of fun

MY NEW DESIGN WILL MEET ALL OF OUR CUSTOMERS’ CURRENT AND FUTURE NEEDS.
Laws of software evolution

Continuing change
A program that is used in a real-world environment must change, or become progressively less useful in that environment.

Increasing complexity
As a program evolves, it becomes more complex, and extra resources are needed to preserve and simplify its structure.
Software is a living entity...

- Early decisions were certainly good at that time
- But the context changes
- Customers change
- Technology changes
- People change
We only maintain useful successful software
Maintenance is continuous Development

Between 70% and 90% of global effort is spent on “maintenance”!

- 18% Corrective (fixing reported errors)
- 18% Adaptive (new platforms or OS)
- 60% Perfective (new functionality)
- 4% Other

“Maintenance”
50% of development time is lost trying to understand code!

Between 50% and 80% of the overall cost is spent in the evolution.

We lose a lot of time with inappropriate and ineffective practices.
How can we fight the dark force?
We design tools and analyses to tame software
Expertise

code analysis, metamodeling, software metrics, reverse engineering, program visualization, evolution analysis, refactoring, quality, program understanding, changes analysis, commit, dependencies, rule and bug assessment, semi-automatic migration, example-based transformations, test selection, rearchitecturing blockchains, \textit{ui-migration}

Collaborations

Pleiad (Chile), UFMG (Brazil), SCG (Switzerland), Soft-VUB (Belgium)
You would not see this dentist!

Why doing it for your software?
Building dedicated tools

Data → Model

Develop Analysis

Analysis

Take decision

an analysis should lead to a decision
Example: Who is behind package X?
Step 1 - Model Creation/Import

Definition of a model to represent entities
Data Extraction (CVS...)

(1) Extraction
(2) Modèle
(3) Analyses
(4) Visualisation
Step 2 - Analyses

(1) Extraction

(2) Modèle

(3) Analyses

(4) Visualisation

Who wrote how many lines of code?
Step : 3 - Creating the Map
All JBoss at a glance

Interactive tool

Data in perspective
A *meta* platform to create tools

McCabe = 21
LOC = 753,000
classes select: #isGod
Evolution in the large: Automated Rules to support migration
How to help migrate from version to version?
Mining API Change Rules

1. Extracting deltas

Rev 1 → Rev 2 → Rev 3 → ... → Rev n

revisions

deltas
Mining API Change Rules

1. Extracting deltas

- Rev 1
- Rev 2
- Rev 3
- Rev n

- deltas

2. Discovering rules

- Select deltas
- selected deltas
- Create rules (data mining)
- rules
Mining API Change Rules

Format of the changes

- `deleted-invoc(context-id, signature)`
- `added-invoc(context-id, signature)`

Diff of method `foo()` between version 1 and 2

- `− self.add(MooseModel.root().add(model));`
- `+ self.add(model.install());`

Formatted changes

- `deleted-call("foo()-rev2", "MooseModel.root()")`
- `deleted-call("foo()-rev2", "MooseModel.add(MooseModel)")`
- `added-call("foo()-rev2", "MooseModel.install()")`
Mining API Change Rules

Request: foo()
Mining API Change Rules

Request:  foo()
1st step: selecting deltas
Mining API Change Rules

Rule:  
\[
\text{foo() } \rightarrow \text{ bar() } \quad \text{Confidence = 75%}
\]

1\text{st} step: selecting deltas

2\text{nd} step: discovering rules
Mining API Change Rules

isNil().ifTrue(*) → ifNil(*)

keys().do(*) → keysDo(*)

intersect(*) → intersectIfNone(*,*)

Scanner.new().scanTokens(*) → parseAsLiteralToken()

RegisterAsApplication(*) →
   WAAdmin.registerAsApplicationAt(*,*)

Character.cr() → ROPlatform.current().newLine()
Tool Support
Validation

Are the generated rules valid to experts?
Yes between 46% to 86%

<table>
<thead>
<tr>
<th></th>
<th>Rules</th>
<th>Valid</th>
<th>Invalid</th>
<th>Don’t know</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pharo2</td>
<td>62</td>
<td>17</td>
<td>20</td>
<td>25</td>
<td>46%</td>
</tr>
<tr>
<td>Pharo3</td>
<td>95</td>
<td>68</td>
<td>24</td>
<td>3</td>
<td>74%</td>
</tr>
<tr>
<td>Moose</td>
<td>50</td>
<td>43</td>
<td>7</td>
<td>0</td>
<td>86%</td>
</tr>
<tr>
<td>Glamour</td>
<td>23</td>
<td>15</td>
<td>8</td>
<td>0</td>
<td>65%</td>
</tr>
<tr>
<td>Roassal</td>
<td>36</td>
<td>28</td>
<td>8</td>
<td>0</td>
<td>78%</td>
</tr>
<tr>
<td>Seaside</td>
<td>76</td>
<td>61</td>
<td>15</td>
<td>-</td>
<td>80%</td>
</tr>
</tbody>
</table>
Evolution in the large: GWT to angular

PhD CIFRE of Benoit Verhaeghe for Berger-Levrault
GWT: An industrial Case Application

- 26 000 classes
- 450 web pages
- 974 000 invocations
- 1 063 163 LOC (UI)
GWT little bac a sable

bigKitchen ~ 8979 éléments
Approach
Approach

GUI Model

GUI extraction

Code extraction
Metamodel
Results

Export

- Ouvrir un onglet
- Ouvrir une boîte de dialogue modale
- Ouvrir une boîte de dialogue non modale
- Ouvrir une boîte de dialogue unique non modale
## Results

### Export

<table>
<thead>
<tr>
<th>Etiquettes formatées (pour les listes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Montant : 10 000 000,00 €</td>
</tr>
<tr>
<td>Pourcentage : 1,50%</td>
</tr>
<tr>
<td>Booléen (1) : Oui</td>
</tr>
<tr>
<td>Booléen (0) : Non</td>
</tr>
<tr>
<td>Date : 26/06/2018</td>
</tr>
<tr>
<td>Durée : 3 ans, 2 mois et 1 jour</td>
</tr>
<tr>
<td>Enumération : jour(s)</td>
</tr>
<tr>
<td>Entier : 999999999</td>
</tr>
<tr>
<td>Entier long : 99999999999999999</td>
</tr>
<tr>
<td>Entier short : 999</td>
</tr>
</tbody>
</table>

---

```ruby
Etiquettes formatées (pour les listes)
Montant :Pourcentage :Booléen (1) :Booléen (0) :Date :Durée :Enumération :Entier :Entier long :Entier short :
```
Tool to explore
Tools for Blockchain Software Engineering
Warning!

Research effort started two years ago
Financed by UTOCAT
SmartAnvil: First platform to build blockchain software analysis tools
Still ongoing work on many points
Solidity Static Analysis
Ethereum Inspector
Smart Debug
Blockchain Query Language
Smart Contract Metrics
Contract Visualization
SmartAnvil: Open-source platform for build tools for SE-Blockchain
Can we build a smart contract dashboard?
Blockchain Visualisation
How to understand a bug in deployed smart contracts?

What is the value of x in function y for contract 0xb45787ac....?
Deployed Smart Contracts are opaque
Debugging contracts very tedious and time-consuming.
No printf :)

Contract inspection can help developer & companies
# Ethereum Inspector

**Inspector View**

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>'owner'</td>
<td>&quot;0xb4ebf466889c4a0239379125a7d0f9c4e8bf2a14&quot;</td>
</tr>
<tr>
<td>'pollTable'</td>
<td>an Array [10 items] (a Dictionary(&quot;choice&quot;-&gt;&quot;NEUTRAL&quot; &quot;hasVoted&quot;-&gt;false &quot;user&quot;-&gt;'0xb4ebf466889c4a023937'...)</td>
</tr>
</tbody>
</table>

**Items View**

<table>
<thead>
<tr>
<th>Index</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEUTRAL' 'hasVoted'-&gt;false 'user'-&gt;'0xb4ebf466889c4a0239379125a7d0f9c4e8bf2a14&quot;)</td>
</tr>
<tr>
<td>2</td>
<td>a Dictionary [3 items] ('choice'-&gt;'POSITIVE' 'hasVoted'-&gt;true 'user'-&gt;'0x3f8b33142f2012e523738e814ff0a1167b3a3f45')</td>
</tr>
<tr>
<td>3</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0x582de4eefa99797cbbd98793a67b3a3f45')</td>
</tr>
<tr>
<td>4</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0x29151a5a1a175087f3b510f022e85b45')</td>
</tr>
<tr>
<td>5</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0xef253dfa1a637e030aca5ca0ba5de4a1f')</td>
</tr>
<tr>
<td>6</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0xed1ec6dd8d69306ad4f4033e00b74b5f')</td>
</tr>
<tr>
<td>7</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0xb0ecac0a16fe9e6778560028e8c1a9b')</td>
</tr>
<tr>
<td>8</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0x3580e54232f65744183f257c5ed0598b3')</td>
</tr>
<tr>
<td>9</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0x6ef18baf53e79a538b88d67770296b1')</td>
</tr>
<tr>
<td>10</td>
<td>a Dictionary [3 items] ('choice'-&gt;'NEGATIVE' 'hasVoted'-&gt;true 'user'-&gt;'0xf7e9a6a300b0a948e7b6f475b831a25')</td>
</tr>
</tbody>
</table>
Ethereum Inspector

Pluggable
Interactive
Consistent
Reusable
Full type support
Ethereum Inspector Architecture
How to debug a smart contract?
Smart Debug

- Full solidity bytecode interpreter
- Storage simulation
- Step and memory access
- Call semantics
- Gas consumption
pragma solidity ^0.4.2;

contract Public3StatesPoll {
    /* Type Definition */
    enum Choice { POSITIVE, NEGATIVE, NEUTRAL }

    struct Columns { address user; Choice choice; bool hasVoted; }

    /* Properties */
    Columns[] pollTable;
    address owner;

    /* Constructor */
    function Public3StatesPoll() public {
        owner = msg.sender;
    }

    /* Functions */
    function isRegistered(address voterAccount) returns(bool registered) {
        registered = (voterIndex(voterAccount) >= 0);
    }

    function voterIndex(address voterAccount) returns(int256 index) {
        for(uint x = 0; x < pollTable.length; x++) {
            if(pollTable[x].user == voterAccount) {
                return int(x);
            }
        }
    }

    function PUSH1(0x80)
    function PUSH1(0x40)
    function MSTORE
    function CALLVALUE
    function DUP1
    function ISZERO
    function PUSH2(0x10)
    function JUMPI
    function PUSH1(0x0)
    function DUP1
    function REVERT
    function JUMPDEST
    function POP
    function CALLER
    function PUSH1(0x1)
    function PUSH1(0x0)
    function PUSH2(0x100)
    function EXP
    function DUP2
    function SLOAD
    function DUP2
    function PUSH2(0x0) 0xFFFFFFFF
    function MUL
    function NOT
    function AND
    function SWAP1
    function DUP4
    function PUSH2(0x0) 0xFFFFFFFF
How to identify one transaction?
How to identify a contract?
In Ethereum:

• **Hash Access memory**
  If you lose the hash of your transaction… you are in troubles.…

• **Massive data**
  if you have to look for it one by one it would be impossible

• **Data opaqueness**
  You may not have any help from the available tools to understand what you are looking at
Blockchain Query Language

```sql
SELECT block.parent.number, block.hash, 
    block.timestamp, block.number, 
    block.amountOfTransactions 
FROM ethereum.blocks AS block 
WHERE block.timestamp BETWEEN date('2016-01-01') 
    AND now() AND block.transactions.size > 10 
ORDER BY block.transactions.size 
LIMIT 100;
```
<table>
<thead>
<tr>
<th>Index</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-779523 - 4 transaction(s) '-&gt;'block.hash.'-&gt;'8965218 '-&gt;'block.timestamp.'-&gt;'2016-01-07T05:53:07+01:00 '-&gt;'block.number.'-&gt;'779524 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>2</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-779112 - 1 transaction(s) '-&gt;'block.hash.'-&gt;'104195555 '-&gt;'block.timestamp.'-&gt;'2016-01-07T03:51:04+01:00 '-&gt;'block.number.'-&gt;'779113 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>3</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-779505 - 0 transaction(s) '-&gt;'block.hash.'-&gt;'226022799 '-&gt;'block.timestamp.'-&gt;'2016-01-07T05:48:01+01:00 '-&gt;'block.number.'-&gt;'779506 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>4</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-782753 - 0 transaction(s) '-&gt;'block.hash.'-&gt;'96179627 '-&gt;'block.timestamp.'-&gt;'2016-01-07T11:48:34+01:00 '-&gt;'block.number.'-&gt;'782754 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>5</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-780346 - 0 transaction(s) '-&gt;'block.hash.'-&gt;'238566626 '-&gt;'block.timestamp.'-&gt;'2016-01-07T10:40+01:00 '-&gt;'block.number.'-&gt;'780349 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>6</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-779113 - 11 transaction(s) '-&gt;'block.hash.'-&gt;'101468812 '-&gt;'block.timestamp.'-&gt;'2016-01-07T03:51:20+01:00 '-&gt;'block.number.'-&gt;'779114 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>7</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-782218 - 0 transaction(s) '-&gt;'block.hash.'-&gt;'33936575 '-&gt;'block.timestamp.'-&gt;'2016-01-07T18:18+01:00 '-&gt;'block.number.'-&gt;'782219 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>8</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-786572 - 0 transaction(s) '-&gt;'block.hash.'-&gt;'129559478 '-&gt;'block.timestamp.'-&gt;'2016-01-07T01:48:18+01:00 '-&gt;'block.number.'-&gt;'786573 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>9</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-779293 - 0 transaction(s) '-&gt;'block.hash.'-&gt;'249003683 '-&gt;'block.timestamp.'-&gt;'2016-01-07T04:38:40+01:00 '-&gt;'block.number.'-&gt;'779282 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>10</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-780351 - 8 transaction(s) '-&gt;'block.hash.'-&gt;'193525535 '-&gt;'block.timestamp.'-&gt;'2016-01-07T10:21:05+01:00 '-&gt;'block.number.'-&gt;'780352 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>11</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-781319 - 8 transaction(s) '-&gt;'block.hash.'-&gt;'239274786 '-&gt;'block.timestamp.'-&gt;'2016-01-07T14:56:33+01:00 '-&gt;'block.number.'-&gt;'781320 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>12</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-779029 - 8 transaction(s) '-&gt;'block.hash.'-&gt;'188508984 '-&gt;'block.timestamp.'-&gt;'2016-01-07T07:21:49+01:00 '-&gt;'block.number.'-&gt;'779030 '-&gt;'block.amountOfTransactions (')-&gt;'11)</td>
</tr>
<tr>
<td>13</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-781583 - 8 transaction(s) '-&gt;'block.hash.'-&gt;'192796897 '-&gt;'block.timestamp.'-&gt;'2016-01-07T11:11:12+01:00 '-&gt;'block.number.'-&gt;'781584 '-&gt;'block.amountOfTransactions (')-&gt;'12)</td>
</tr>
<tr>
<td>14</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-778266 - 9 transaction(s) '-&gt;'block.hash.'-&gt;'148395039 '-&gt;'block.timestamp.'-&gt;'2016-01-07T00:55:16+01:00 '-&gt;'block.number.'-&gt;'778276 '-&gt;'block.amountOfTransactions (')-&gt;'12)</td>
</tr>
<tr>
<td>15</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-780352 - 8 transaction(s) '-&gt;'block.hash.'-&gt;'245360810 '-&gt;'block.timestamp.'-&gt;'2016-01-07T10:21:05+01:00 '-&gt;'block.number.'-&gt;'780354 '-&gt;'block.amountOfTransactions (')-&gt;'12)</td>
</tr>
<tr>
<td>16</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-781499 - 8 transaction(s) '-&gt;'block.hash.'-&gt;'157190109 '-&gt;'block.timestamp.'-&gt;'2016-01-07T15:52:34+01:00 '-&gt;'block.number.'-&gt;'781500 '-&gt;'block.amountOfTransactions (')-&gt;'12)</td>
</tr>
<tr>
<td>17</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-780917 - 3 transaction(s) '-&gt;'block.hash.'-&gt;'130847270 '-&gt;'block.timestamp.'-&gt;'2016-01-07T15:47+01:00 '-&gt;'block.number.'-&gt;'780918 '-&gt;'block.amountOfTransactions (')-&gt;'12)</td>
</tr>
<tr>
<td>18</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-781459 - 4 transaction(s) '-&gt;'block.hash.'-&gt;'255862572 '-&gt;'block.timestamp.'-&gt;'2016-01-07T15:46:40+01:00 '-&gt;'block.number.'-&gt;'781460 '-&gt;'block.amountOfTransactions (')-&gt;'12)</td>
</tr>
<tr>
<td>19</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-780697 - 5 transaction(s) '-&gt;'block.hash.'-&gt;'118747668 '-&gt;'block.timestamp.'-&gt;'2016-01-07T17:46+01:00 '-&gt;'block.number.'-&gt;'780698 '-&gt;'block.amountOfTransactions (')-&gt;'12)</td>
</tr>
<tr>
<td>20</td>
<td>an OrderedCollection [5 items] {'block.parent.'-&gt;'Block-778514 - 0 transaction(s) '-&gt;'block.hash.'-&gt;'208518231 '-&gt;'block.timestamp.'-&gt;'2016-01-07T07:21+01:00 '-&gt;'block.number.'-&gt;'778515 '-&gt;'block.amountOfTransactions (')-&gt;'13)</td>
</tr>
</tbody>
</table>
Ongoing work
What tools need business BC engineers?
First prototypes

```
pragma solidity ^0.4.16;

contract Poll {
    struct PollEntry {
        bool hasVoted;
        Choice choice;
        address user;
    }

    address owner;
    PollEntry[] pollTable;

tenum Choice {POSITIVE, NEGATIVE, NEUTRAL }
}
```
Blockchain modelisation

[PhD - S. Bragagnolo - Berger-Levrault]

How to capture and model existing process?
How to model and simulate domain of trust and possible architecture?
What is the impact of trust model on application architecture?
What are software metrics for Smart contracts?

What is the quality of a Smart contract?
How to capture potential problems?
Ongoing challenges
Existing applications?

How to support integration of existing applications and blockchain solution?

We believe that we can build tools for the identification of injection points.

If you are interested discuss with us.
Creating tools to tame software

Interested by your challenges

• migration help
• software assessment
• software map
• rearchitecturing
• service/micro service identification
• blockchainisation :)

RMoD