Columbus United

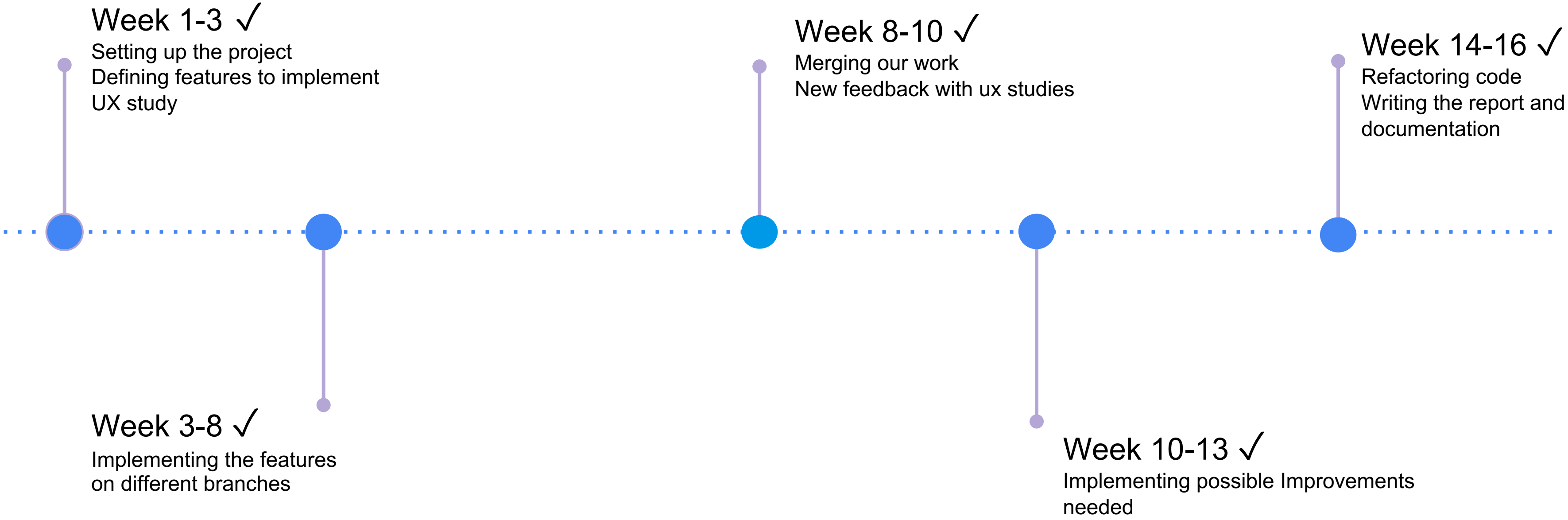
A Byzcoin vizualisation project

Supervised by Noémien Kocher
Lucas Trognon
Sophia Artioli

Objectives

- ❑ Creating a visualization tool for ByzCoin and its underlying Skipchain
- ❑ Usability and accessibility for all kinds of users
- ❑ Help users acquire more knowledge about ByzCoin

Timeline



Week 1-3 ✓
Setting up the project
Defining features to implement
UX study

Week 3-8 ✓
Implementing the features
on different branches

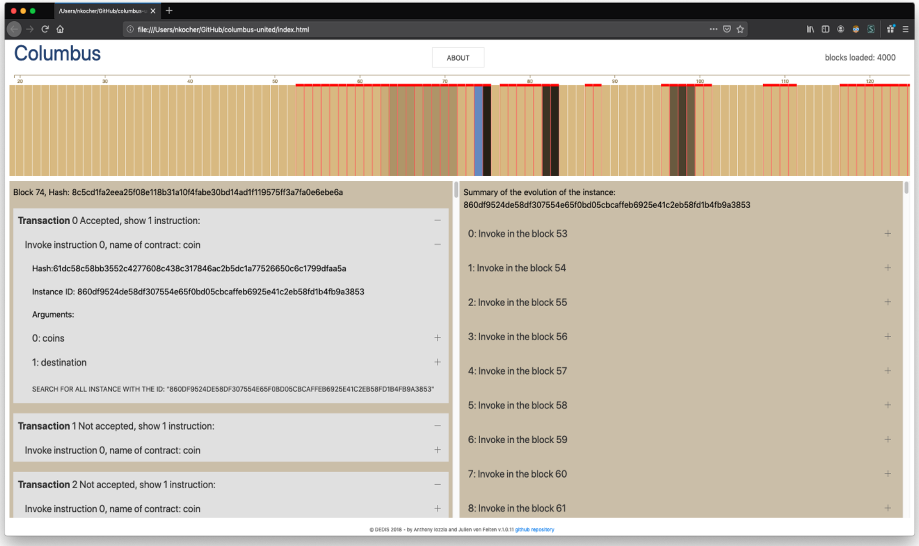
Week 8-10 ✓
Merging our work
New feedback with ux studies

Week 10-13 ✓
Implementing possible Improvements
needed

Week 14-16 ✓
Refactoring code
Writing the report and
documentation

Initial feedback on the previous explorer

- ❑ No visibility over the structure of the Skipchain
- ❑ Navigation is rigid
- ❑ Lacking key information
- ❑ Lacking readability and usability
- ❑ Instance tracking not working


/Users/nkocher/GitHub/columbus-u
file:///Users/nkocher/GitHub/columbus-united/index.html
Columbus
ABOUT
blocks loaded: 4000
20
30
40
50
60
70
80
90
100
110
120

Block 74, Hash: 8c5cd1fa2eea25f08e118b31a10f4fabe30bd14ad1f119575ff3a7fa0e6ebe6a

Transaction 0 Accepted, show 1 instruction: —

Invoke instruction 0, name of contract: coin —

Hash:61dc58c58bb3552c4277608c438c317846ac2b5dc1a77526650c6c1799dfaa5a

Instance ID: 860df9524de58df307554e65f0bd05cbcaffeb6925e41c2eb58fd1b4fb9a3853

Arguments:

0: coins +

1: destination +

SEARCH FOR ALL INSTANCE WITH THE ID: "860DF9524DE58DF307554E65F0BD05CBCAFFEB6925E41C2EB58FD1B4FB9A3853"

Transaction 1 Not accepted, show 1 instruction: —

Invoke instruction 0, name of contract: coin +

Transaction 2 Not accepted, show 1 instruction: —

Invoke instruction 0, name of contract: coin +

Summary of the evolution of the instance:
860df9524de58df307554e65f0bd05cbcaffeb6925e41c2eb58fd1b4fb9a3853

0: Invoke in the block 53 +

1: Invoke in the block 54 +

2: Invoke in the block 55 +

3: Invoke in the block 56 +

4: Invoke in the block 57 +

5: Invoke in the block 58 +

6: Invoke in the block 59 +

7: Invoke in the block 60 +

8: Invoke in the block 61 +

© DEDIS 2018 - by Anthony Iozzia and Julien von Felten v.1.0.11 github repository

Features implementation

TS Typescript as the front-end language

RXJS as the react library

Uikit as the CSS framework

D3 as the visualization library

Planned features: Chain visualization

Chain visualization	
Search bar: search by block height, transaction hash/ID, contract name	1
Dedicated space for the last validated block	1
On-hand information on the last validated block	1
Build dedicated space for chain and display block heights	1
Highlight forward links and backlinks between blocks by adding arrows	1
adding “colored stickers” to blocks on chain representing transactions	2
Make arrows between blocks clickable to display at the right scale all blocks linked	2
“go to home” reloads the latest block and displays the chain from block 0	3
Statistics: - average number of transactions per block - average validation time - most popular contracts - number of accepted transaction vs non accepted	3

Implementation & Challenges

Chain

- Displaying blocks by their heights
- Adding arrows between blocks
- Starting the vizualisation from the end of the chain

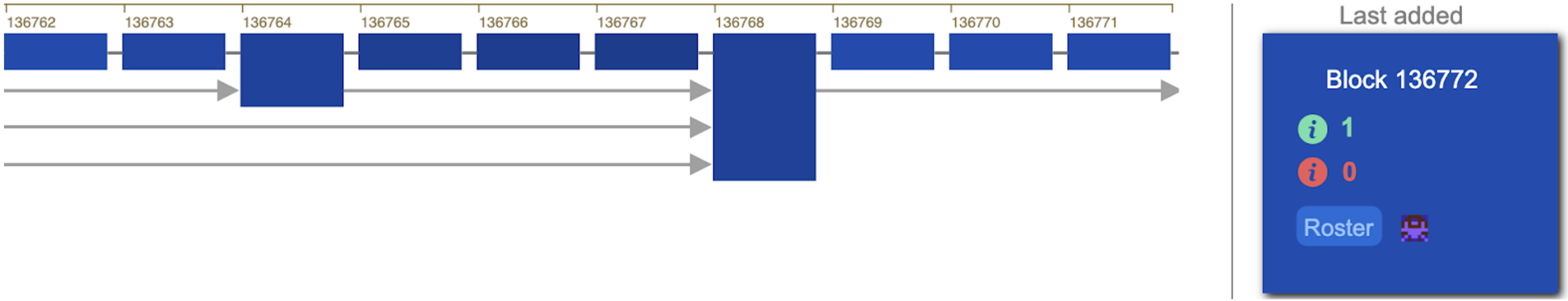
Last added block

- Building a dedicated space
- Display of information on-hand

Search-bar and translation

- Building the search bar
- Making requests
- Chain translation
- “Chunk” loading of the chain

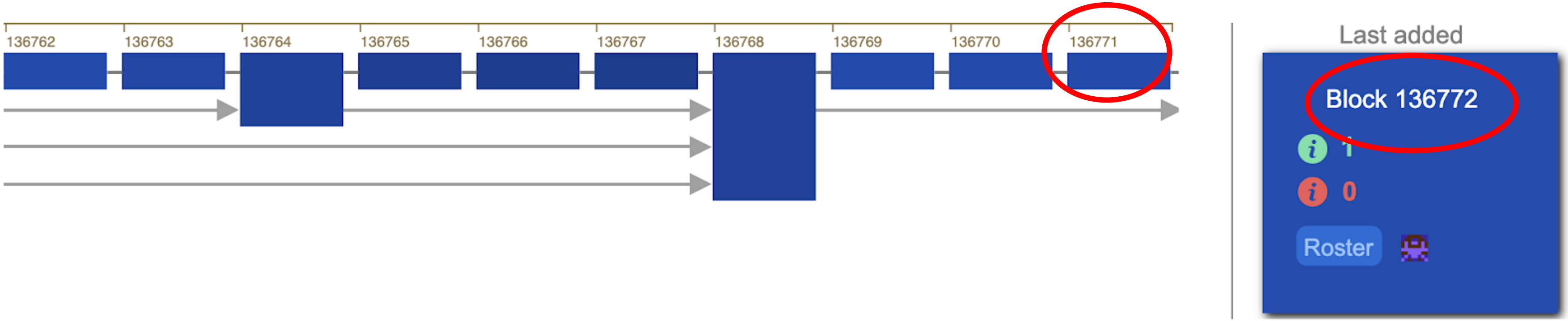
Skipchain visualization: Block heights and arrows



Challenges

- Fit all block heights in the container
- Linking the blocks
- Keeping the proportions & making it aesthetic
- No overlaps

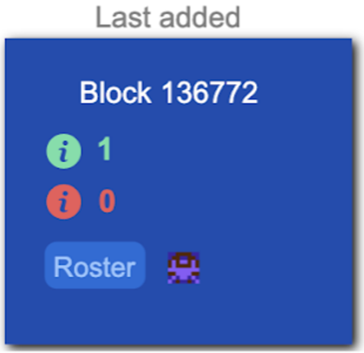
Skipchain visualization: End of chain display



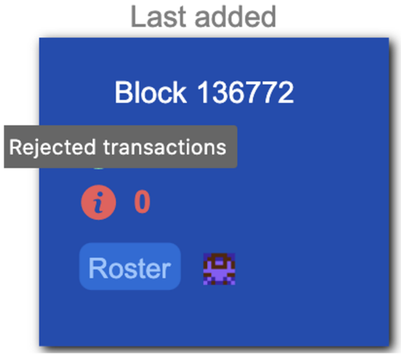
Challenges

- Broken links
- Pagination requests when close to the end of the chain

Skipchain visualization: Last added block

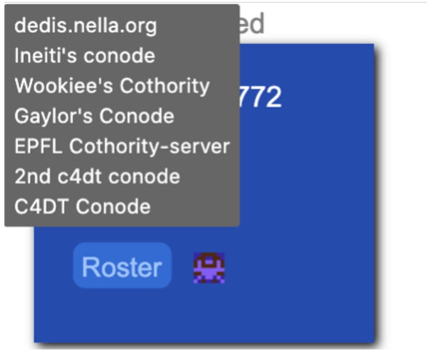


No pointer interaction



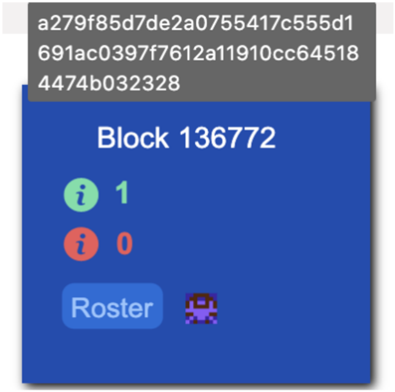
Rejected transactions

Pointer on the information icons



dedis.nella.org
Ineiti's conode
Wookiee's Cothority
Gaylor's Conode
EPFL Cothority-server
2nd c4dt conode
C4DT Conode

Pointer on the “Roster”



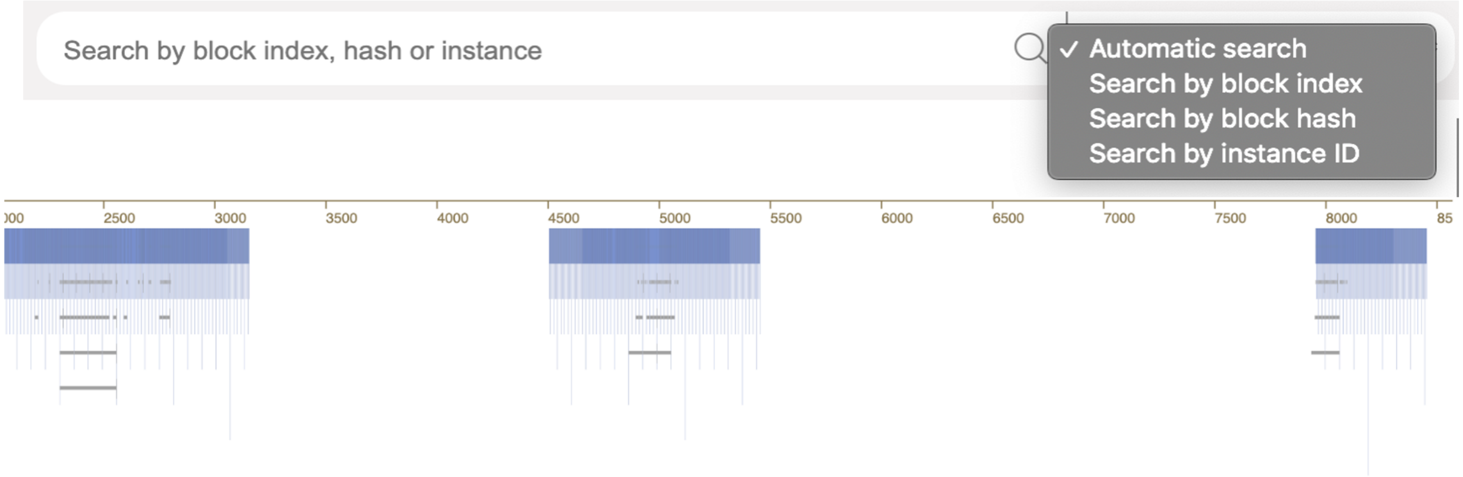
a279f85d7de2a0755417c555d1
691ac0397f7612a11910cc64518
4474b032328

Pointer on the block index

Challenges

- Interaction with the client not working because of forward links
- Choice of on-hand information
- Making it pretty

Skipchain visualization: Search-bar and translation



Challenges

- Differentiating block hashes, indexes and instance ID
- Optimal loading of the chain into “Chunks”
- Translation of the chain

Finished features: Chain visualization

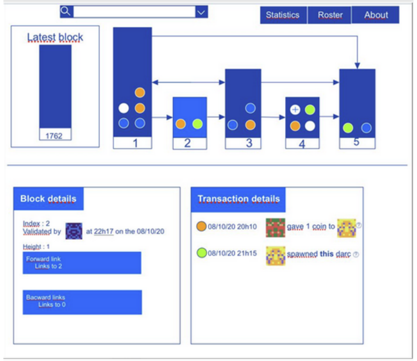
Chain visualization	
Search bar: search by block height, transaction hash/ID, contract name	1
have a dedicated space for the last validated block	1
add main information to the last validated block	1
Build dedicated space for chain and display block heights	1
Highlight forward links and backlinks between blocks by adding arrows	1
Adding “colored stickers” to blocks on chain representing transactions	2
Make arrows between blocks clickable to display at the right scale all blocks linked	2

“go to home” reloads the latest block and displays the chain from block 0	3
Statistics: - average number of transactions per block - average validation time - most popular contracts - number of accepted transaction vs non accepted	4

Planned features

Block visualization	
Overhaul the general design of the page	1
Add interactive fields to replace hashes (blockies,links, …)	2
Add more fields to the block details column (i.e. validation time)	1
Fix and redesign the instance tracker	1
Designing icons to help the user conceptualize each field	2
Reformat current fields to make it clearer & improve the arrangement of the page	1

Other	
Updating the URL to reflect chain state	3
Tutorial to introduce Columbus to new users	2



Block details

Index : 2
Validated by at 22h17 on the 08/10/20

Height : 1

Forward link
Links to 2

Bacward links
Links to 0

Transaction details

08/10/20 20h10 gave 1 coin to

08/10/20 21h15 spawned this darc

Implementation & Challenges

Block and transaction details

- Designing a usable interface
- Apprehending a large and scarcely documented project

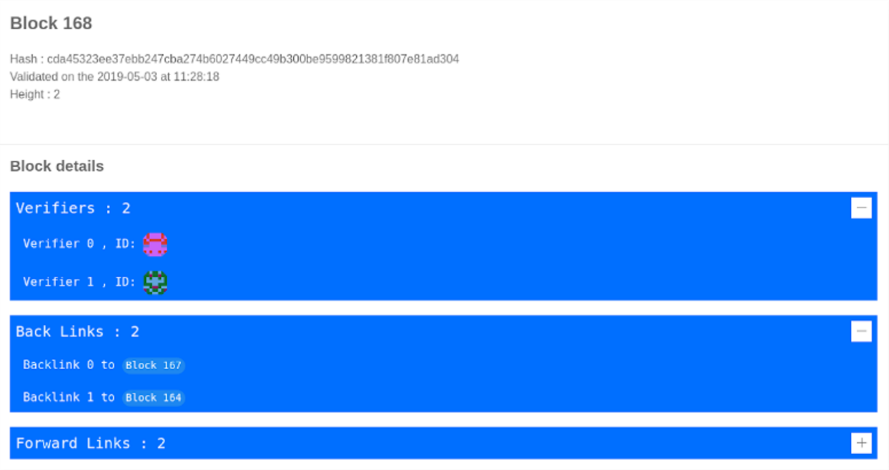
Instance tracker modernizing

- Overhauling convoluted code
- Moving the tracker to the main page
- Offering granularity in the search while keeping it simple

Integrating the feedback to the explorer

- Design good UX studies
- Design concise UX studies
- Make compromises

Block and transaction details

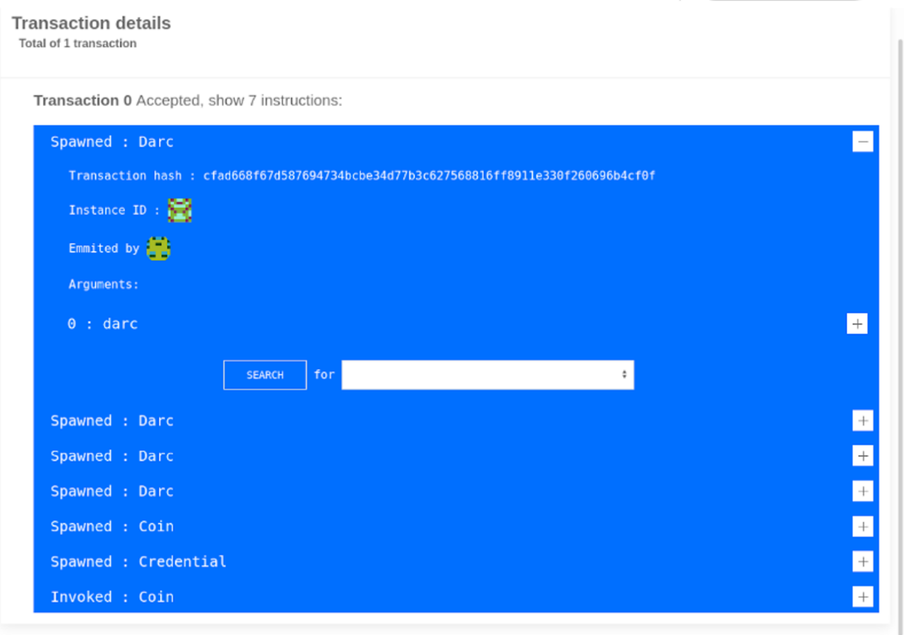


Block 168

Hash : cda45323ee37ebb247cba274b6027449cc49b300be9599821381f807e81ad304
Validated on the 2019-05-03 at 11:28:18
Height : 2

Block details

```
Verifiers : 2
  Verifier 0 , ID:
  Verifier 1 , ID:
Back Links : 2
  Backlink 0 to Block 167
  Backlink 1 to Block 164
Forward Links : 2
```



Transaction details
Total of 1 transaction

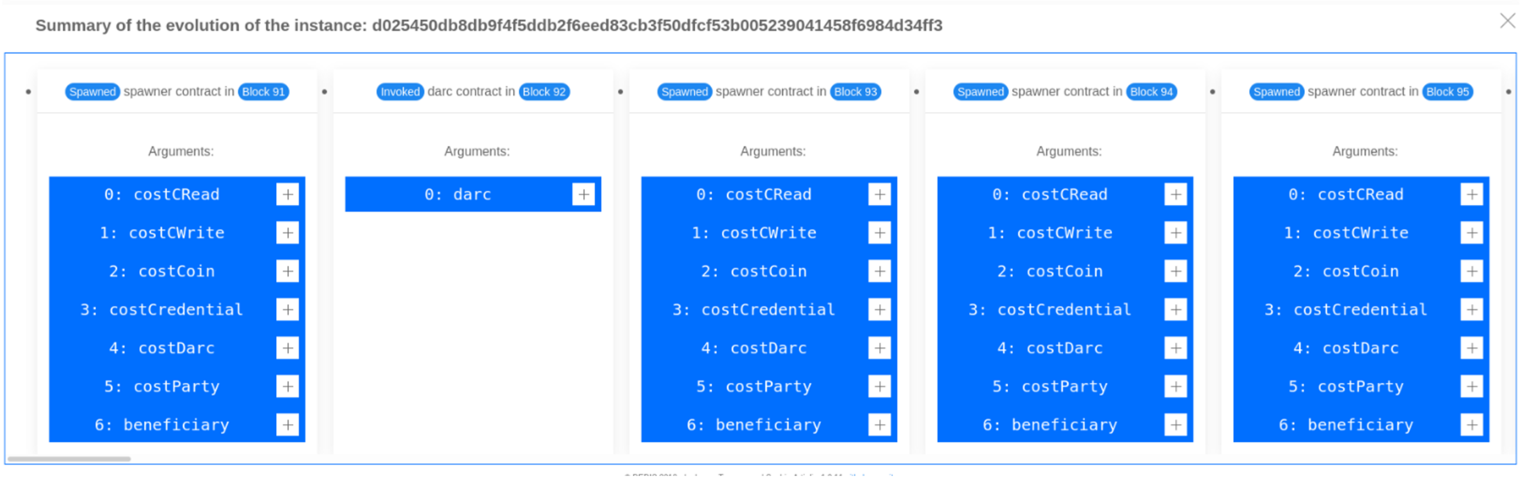
Transaction 0 Accepted, show 7 instructions:

```
Spawned : Darc
  Transaction hash : cfad668f67d587694734bcbe34d77b3c627568816ff8911e330f260696b4cf0f
  Instance ID :
  Emmited by
  Arguments:
  0 : darc
            SEARCH for
Spawned : Darc
Spawned : Darc
Spawned : Darc
Spawned : Coin
Spawned : Credential
Invoked : Coin
```

Challenges :

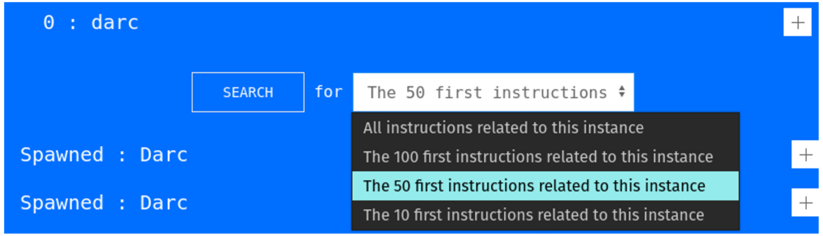
- Designing a usable interface
- Apprehending a large and scarcely documented project

Instance tracker



Summary of the evolution of the instance: d025450db8db9f4f5ddb2f6eed83cb3f50dfcf53b005239041458f6984d34ff3

Spawned spawner contract in Block 91	Invoked darc contract in Block 92	Spawned spawner contract in Block 93	Spawned spawner contract in Block 94	Spawned spawner contract in Block 95
Arguments:	Arguments:	Arguments:	Arguments:	Arguments:
0: costCRead	0: darc	0: costCRead	0: costCRead	0: costCRead
1: costCWrite		1: costCWrite	1: costCWrite	1: costCWrite
2: costCoin		2: costCoin	2: costCoin	2: costCoin
3: costCredential		3: costCredential	3: costCredential	3: costCredential
4: costDarc		4: costDarc	4: costDarc	4: costDarc
5: costParty		5: costParty	5: costParty	5: costParty
6: beneficiary		6: beneficiary	6: beneficiary	6: beneficiary



0 : darc

SEARCH for The 50 first instructions

- All instructions related to this instance
- The 100 first instructions related to this instance
- The 50 first instructions related to this instance
- The 10 first instructions related to this instance

Spawned : Darc

Spawned : Darc

Challenges :

- Overhauling convoluted code
- Moving the tracker to the main page
- Offering granularity in the search while keeping it simple

Integrating the feedback to the explorer

First round

Goals :

- Figure out what's wrong with Columbus

Design choices :

- 10 participants (novices and intermediates)
- Very broad questions
- Mostly unguided

Second round

Goals :

- Refresh our priority list
- Tailor the user workflow around the lab use cases
- Get immediate feedback while in lockdown

Design choices :

- Google form
- Streamlined
- More complex and precise questions

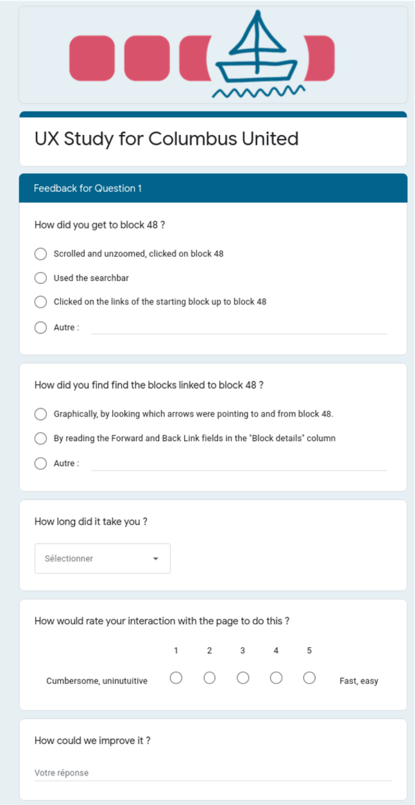
Third round

Goals :

- Validate our changes
- Find potential improvements

Design choices :

- 15 participants (both new and familiar with Columbus)
- Mostly unguided
- Precise questions on features afterward



UX Study for Columbus United

Feedback for Question 1

How did you get to block 48 ?

○ Scrolled and unzoomed, clicked on block 48

○ Used the searchbar

○ Clicked on the links of the starting block up to block 48

○ Autre : ______

How did you find find the blocks linked to block 48 ?

○ Graphically, by looking which arrows were pointing to and from block 48.

○ By reading the Forward and Back Link fields in the "Block details" column

○ Autre : ______

How long did it take you ?

Sélectionner ▾

How would rate your interaction with the page to do this ?

	1	2	3	4	5	
Cumbersome, uninutuitive	○	○	○	○	○	Fast, easy

How could we improve it ?

Votre réponse

Second round of UX study (guided)

Feedback from the lab :

- Some features were overseen or difficult to use
- Many suggestions were made
- Navigation is improved
- Overall interactions are quick and pleasant

Finished features: Block visualization

Block visualization	
Overhaul the general design of the page	1
Add interactive fields to replace hashes (blockies,links, …)	2
Add more fields to the block details column (i.e. validation time)	1
Fix and redesign the instance tracker	1
Designing icons to help the user conceptualize each field	2
Reformat current fields to make it clearer & improve the arrangement of the page (switch to a single page interface)	1

Other	
Updating the URL to reflect chain state (and make actions revertible)	3
Small tutorial to introduce Columbus to new users	2
Modal search-bar	NA
Flash messages improvements	NA
Animated translations	NA
Automated versioning	NA

Third round of UX study

Observations :

- Visibility of the features improved
- Efficient for expert users
- Learnable for beginners
- Users could infer some concepts about Byzcoin from the interface

Improvements to the Byzcoin implementation

Columbus is not only a Byzcoin visualization tool, it also helped discover bugs and strengthen the Byzcoin implementation

- Discovered some links were broken -> blocks were not accessible
- More functionalities for proxy nodes
- QoL improvements such as a beautifier
- Various bug fixes (timeouts, etc..)
- Optimizations

Conclusion - Demo

Thank you !

Q&A