ethereum

vienna

Microsoft's Blockchain Workbench

Swarm Summit Recap

May 17th 2018

RIAT is an institute for research, development, communication and education in the fields of cryptoeconomics and the blockchain.

RIAT BLOCKCHAIN ACADEMY

Smart Contract Development with Ethereum
4 day course returns!

June 5th-8th

Agenda

1. Microsoft's Blockchain Workbench
2. Swarm Summit Recap + Incentives Update

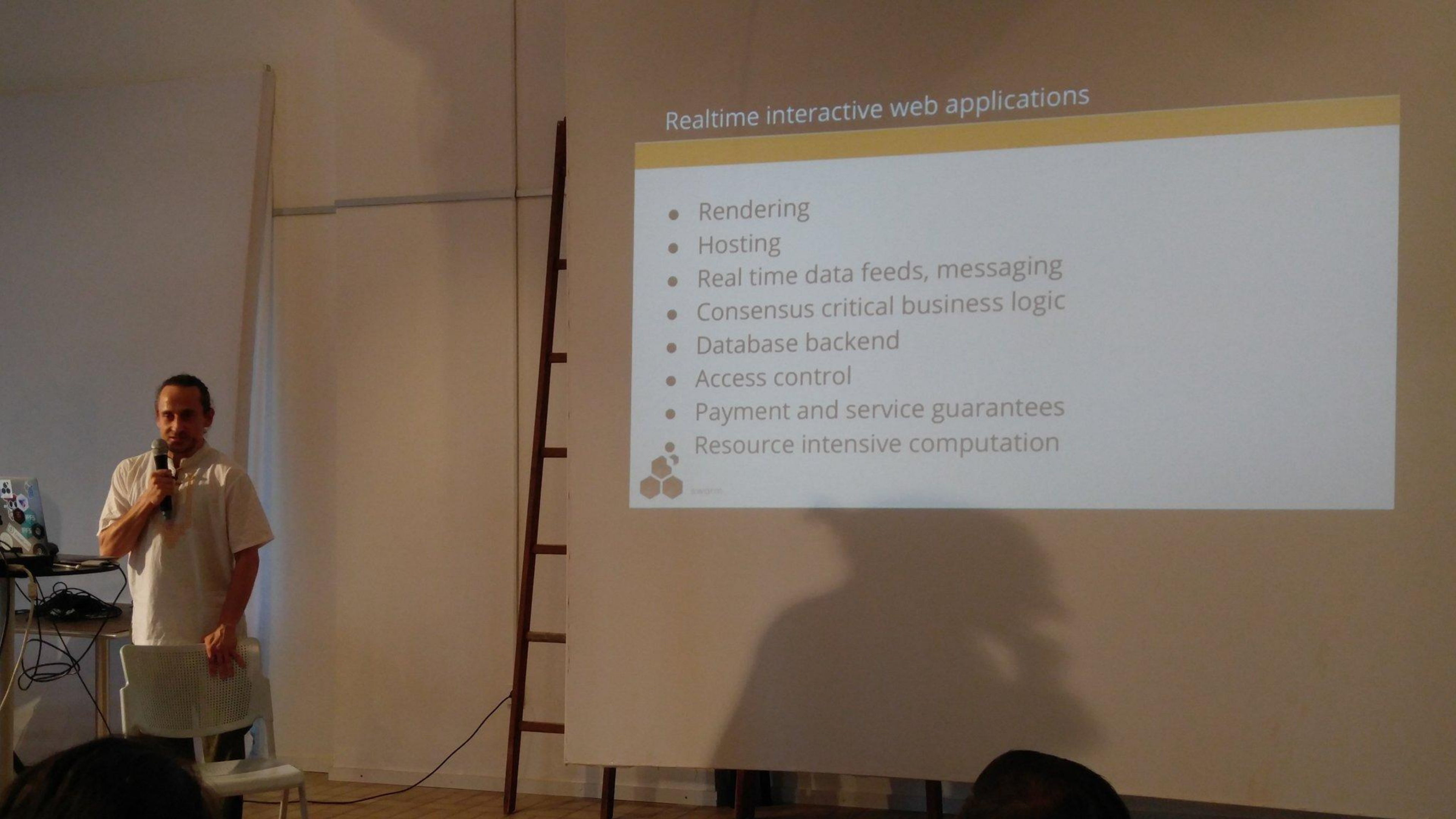
Swarm Summit Recap + Incentives

Ethereum Vienna Meetup
May 17th 2018

Agenda

1. Swarm Summit Recap
2. Swap Overview
3. Swear & Swindle Overview

Part 1:
Swarm Summit



Realtime interactive web applications

- Rendering
- Hosting
- Real time data feeds, messaging
- Consensus critical business logic
- Database backend
- Access control
- Payment and service guarantees
- Resource intensive computation

Swarm Orange Summit 2018

Took place May 7th-11th 2018 in Ljubljana, Slovenia

Mini developer conference focused exclusively on Swarm and apps built on top

Videos not yet available (except Mainframe)

Will be available on Swarm soon*

* also on youtube

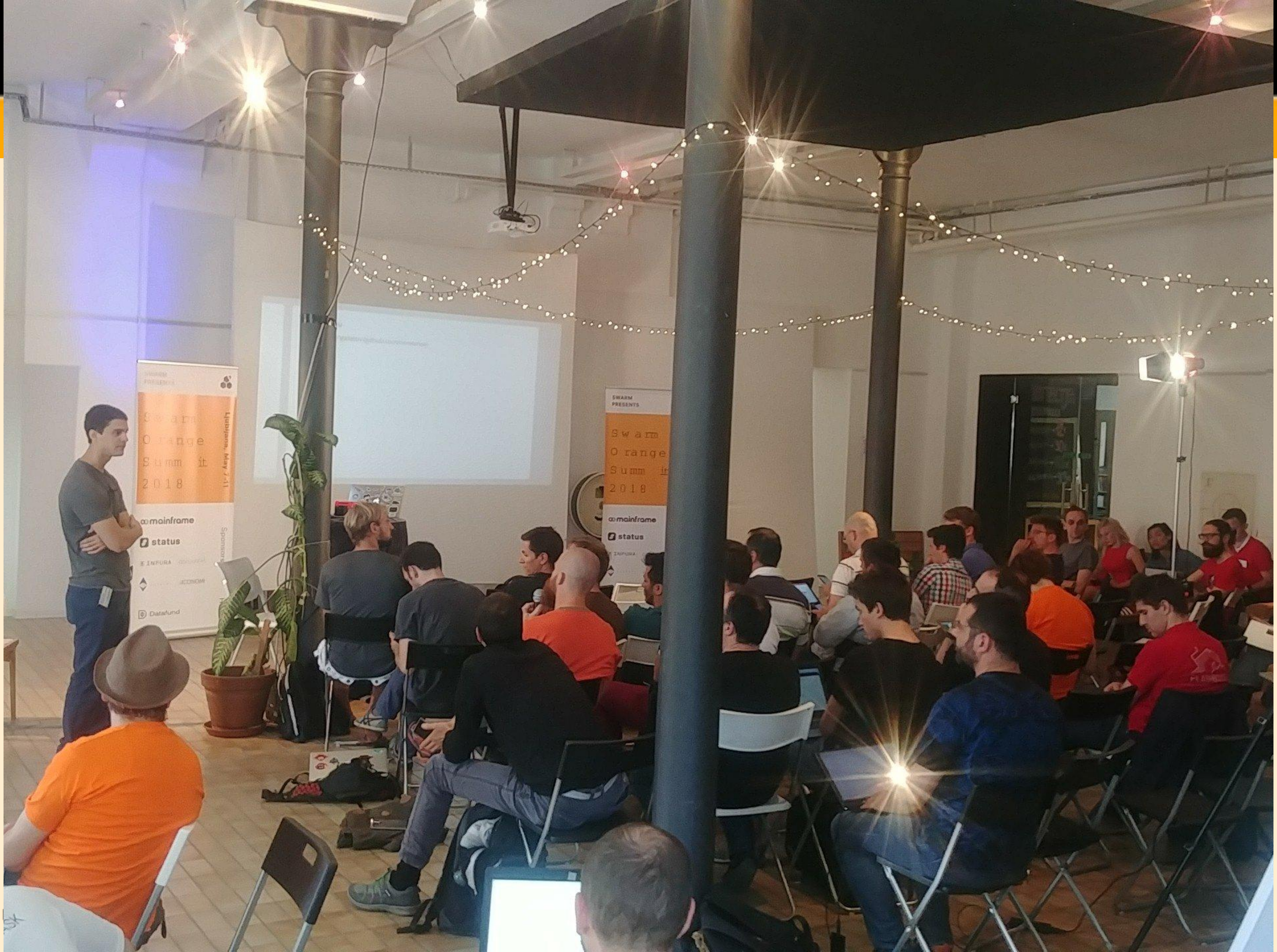
swarm

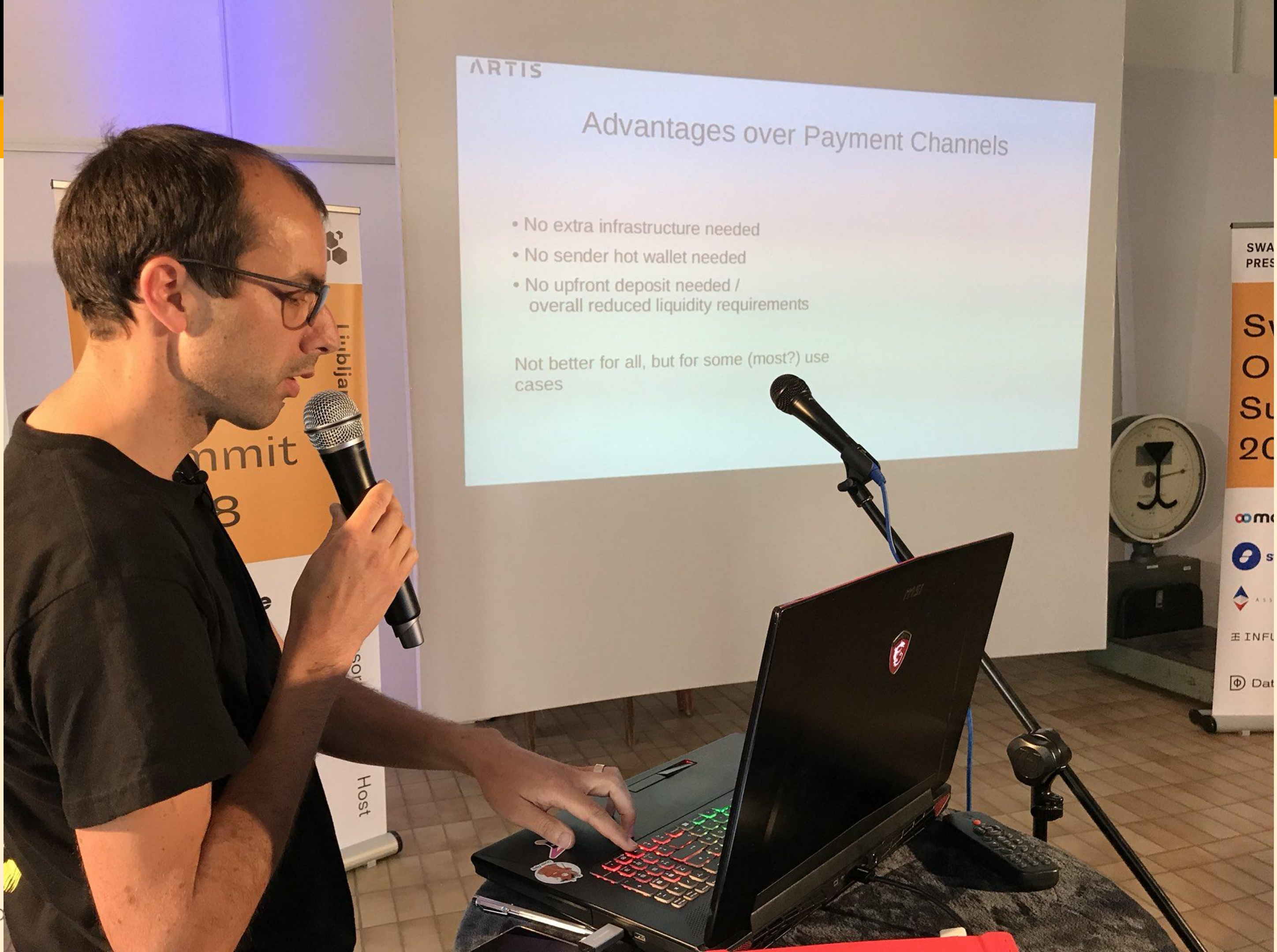
Swarm Orange Summit 2018

Some notable talks:

- Talks on Mutable Resource Updates
- Video streaming with Livepeer
- PSS updates
- Encryption on Swarm
- Swap, Swear and Swindle :-)

swarm


SWARM PRESENTS
Swarm Orange Summit 2018
Ljubljana, May 7-11
mainframe
status
INFURA
Datafund
Sponsors



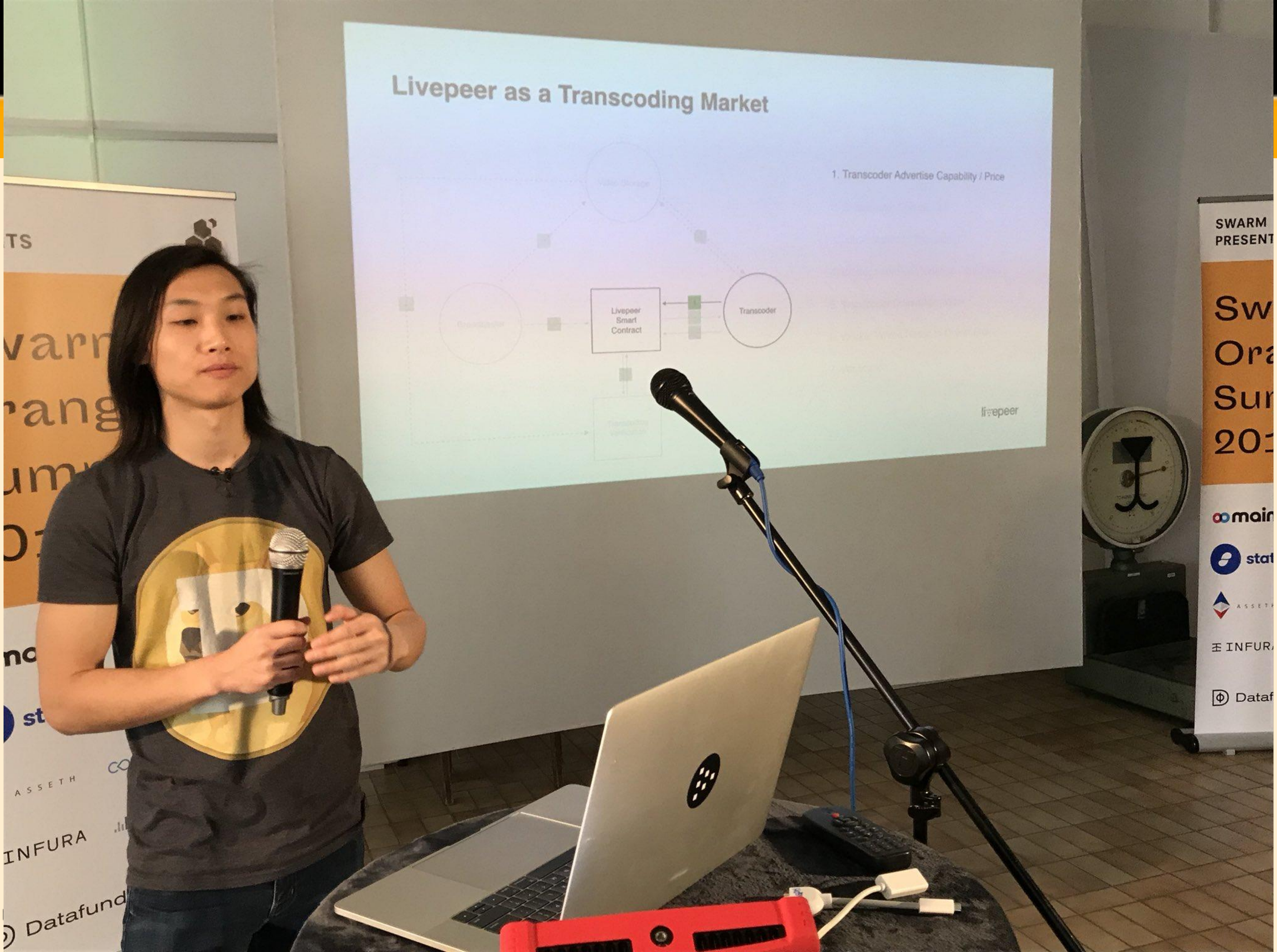
ARTIS

Advantages over Payment Channels

- No extra infrastructure needed
- No sender hot wallet needed
- No upfront deposit needed / overall reduced liquidity requirements

Not better for all, but for some (most?) use cases

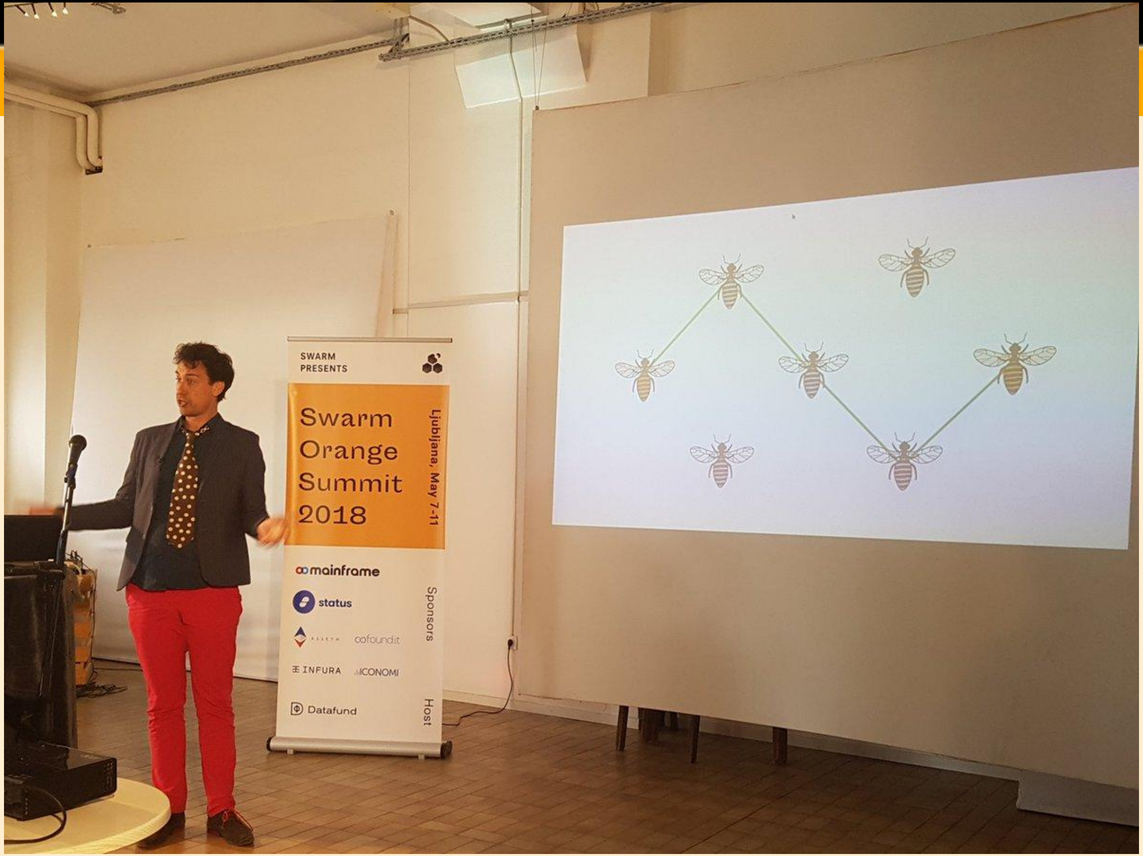
Agenda
1. Intro to Swear and Swindle
2. Walkthrough of the contract
3. Issues and open problems
SWARM PRESENTS
Swarm Orange Summit 2018
Ljubljana, May 7-11
mainframe
status
ASSETH
cofound.it
INFURA
ICONOMI
Sponsors
Datafund
Host



Livepeer as a Transcoding Market

1. Transcoder Advertise Capability / Price

WHISPER
• Envelopes
• Messages
• Filters
• Topics
SWARM PRESENTS
Swarm Orange Summit 2018
Ljubljana, May 7-11
mainframe
status
Sponsors
INFURA
Datafund
Host


SWARM PRESENTS
Swarm Orange Summit 2018
Ljubljana, May 7-11
mainframe
status
Sponsors
cofound.it
INFURA
ICONOMI
Datafund
Host

When lambo?
SWARM PRESENTS
Swarm Orange Summit 2018
mainframe
status
INFURA
Datafund

swarm

Part 2:
SWAP

Disclaimer

The following contracts are purely for **experimental** purposes

They are not intended to ever be deployed to a real network

They contain many hacks and security issues (you WILL lose ether!)

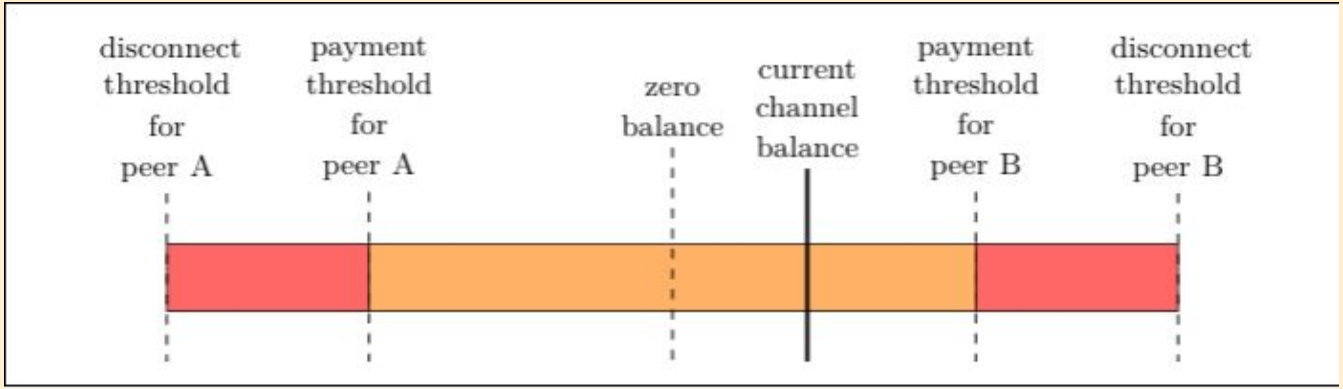
Real contracts will be rewritten from scratch once the design is clear

SWAP

Swarm nodes exchange services.

Service costs cause “channel imbalance”

Once payment threshold is reached, a cheque returns channel to balanced state


disconnect threshold for peer A
payment threshold for peer A
zero balance
current channel balance
payment threshold for peer B
disconnect threshold for peer B

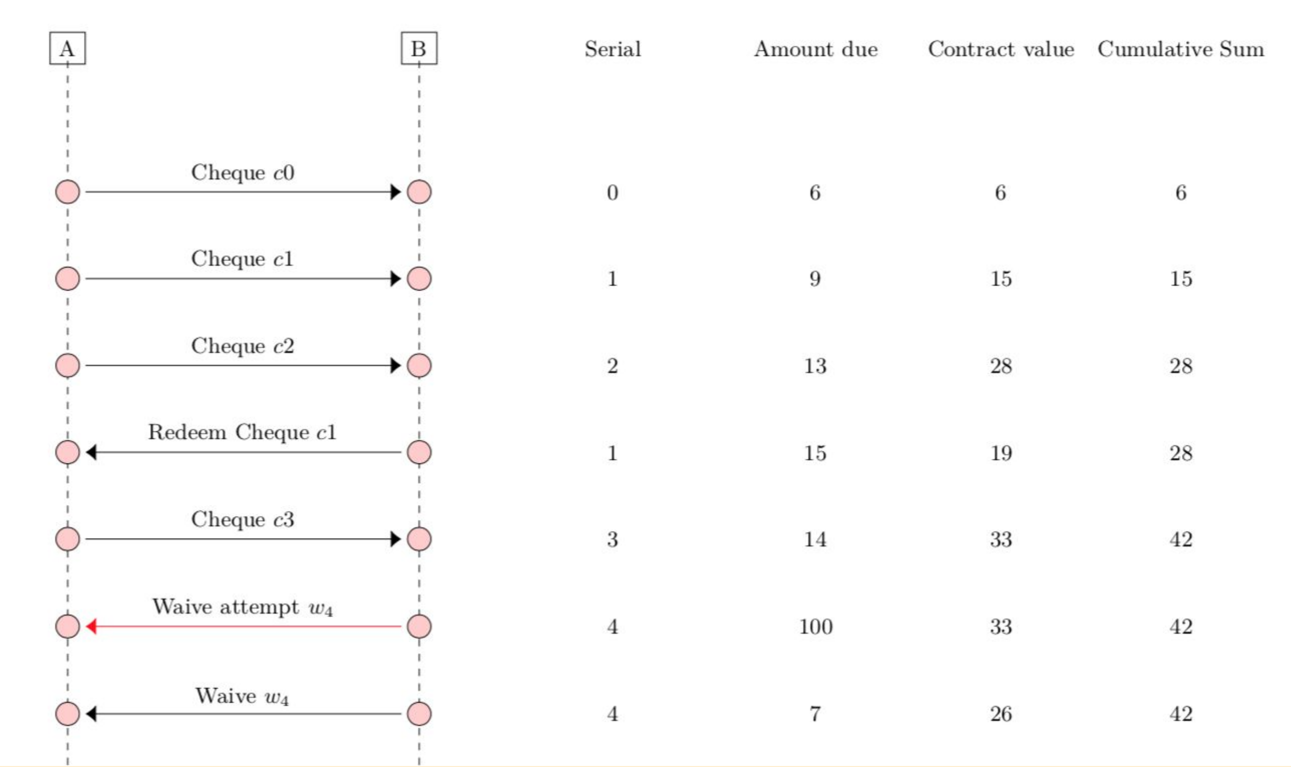
swarm

Cheques

Cheques have **cumulative** values

Not every cheque needs to be cashed

Cheques can also decrease in value if both parties agree



	Serial	Amount due	Contract value	Cumulative Sum
Cheque $c0$	0	6	6	6
Cheque $c1$	1	9	15	15
Cheque $c2$	2	13	28	28
Redeem Cheque $c1$	1	15	19	28
Cheque $c3$	3	14	33	42
Waive attempt w_4	4	100	33	42
Waive w_4	4	7	26	42

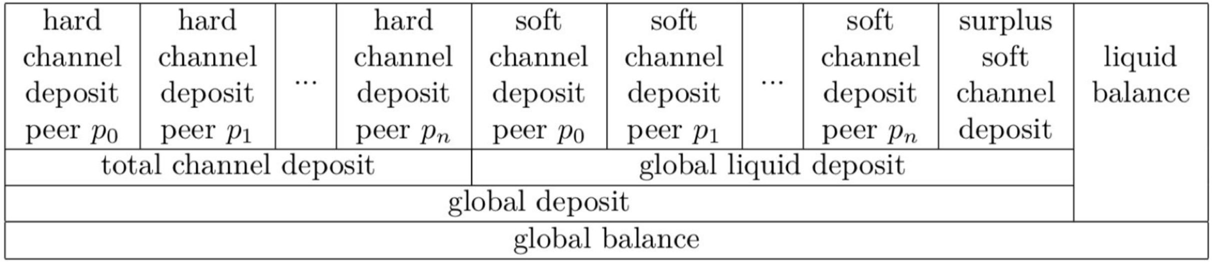
Hard Deposits

Hard Deposits guarantee solvency for a specific **beneficiary**

The ether are **locked** for any other usage

There has to be a way to decrease the deposit

This involves a **timeout**



hard channel deposit peer p_0	hard channel deposit peer p_1	…	hard channel deposit peer p_n	soft channel deposit peer p_0	soft channel deposit peer p_1	…	soft channel deposit peer p_n	surplus soft channel deposit	liquid balance
total channel deposit				global liquid deposit					
global deposit									
global balance									

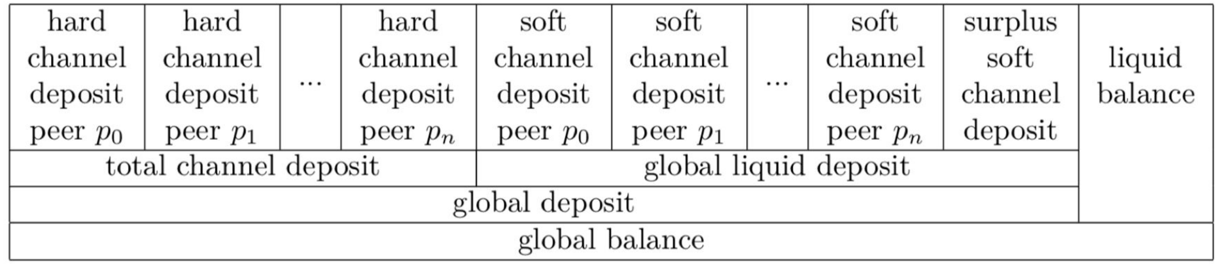
Figure 5: Chequebook balances and deposits.

Soft Deposits

Soft Deposits are an on-chain guarantee of solvency for a group of people and off-chain for individuals

An allocation table is periodically shared with all participants using

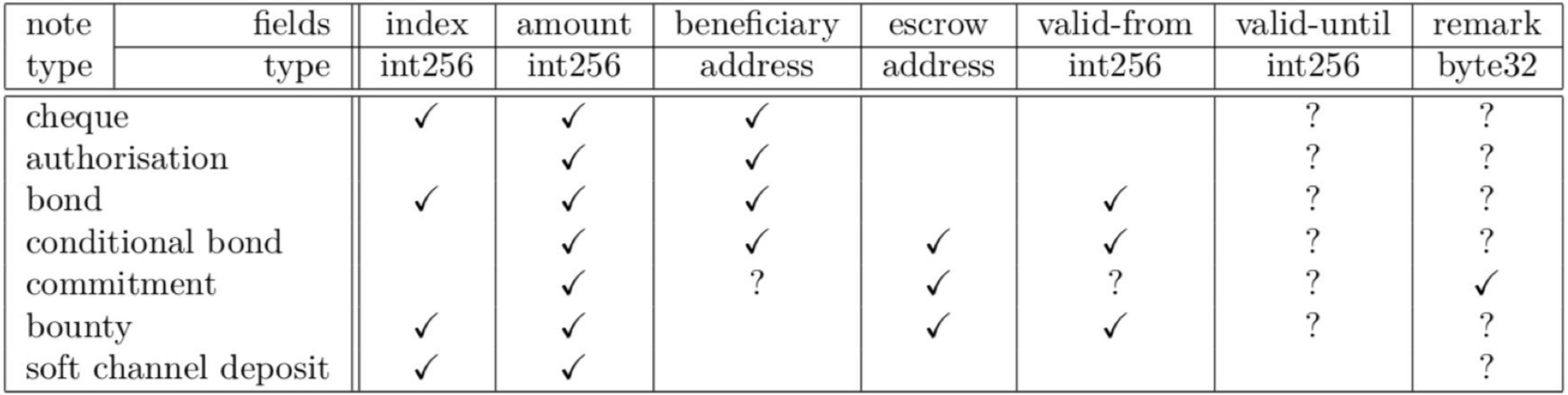
Mutable Resource Updates



<table>
<tr><td>hard channel deposit peer p_0</td><td>hard channel deposit peer p_1</td><td>…</td><td>hard channel deposit peer p_n</td><td>soft channel deposit peer p_0</td><td>soft channel deposit peer p_1</td><td>…</td><td>soft channel deposit peer p_n</td><td>surplus soft channel deposit</td><td rowspan="3">liquid balance</td></tr>
<tr><td colspan="4">total channel deposit</td><td colspan="5">global liquid deposit</td></tr>
<tr><td colspan="9">global deposit</td></tr>
<tr><td colspan="10">global balance</td></tr>
</table>

Figure 5: Chequebook balances and deposits.

Promissory Notes



note	fields	index	amount	beneficiary	escrow	valid-from	valid-until	remark
type	type	int256	int256	address	address	int256	int256	byte32
cheque		✓	✓	✓			?	?
authorisation			✓	✓			?	?
bond		✓	✓	✓		✓	?	?
conditional bond			✓	✓	✓	✓	?	?
commitment			✓	?	✓	?	?	✓
bounty		✓	✓		✓	✓	?	?
soft channel deposit		✓	✓					?

Figure 8: Taxonomy of promissory notes: '✓' indicates a mandatory field, '?' indicates optional field. Types show the corresponding solidity type to encode in the ABI.

swarm

Current status

Cheques and Hard Deposits are implemented

Promissory notes exist but not all fields work properly (index)

Test suite is still incomplete (especially with notes)

Security has been in the background for now

Invoice mechanism highly experimental

Soft Deposits not implemented at all

Part 3:
Swear & Swindle

Swear

Swap is for paying services in real time

Swear is for service that need to occur in the future

Basically simulates a courtroom

Witnesses are contracts verifying proovable evidence

Swear & Swindle

Basic flow:

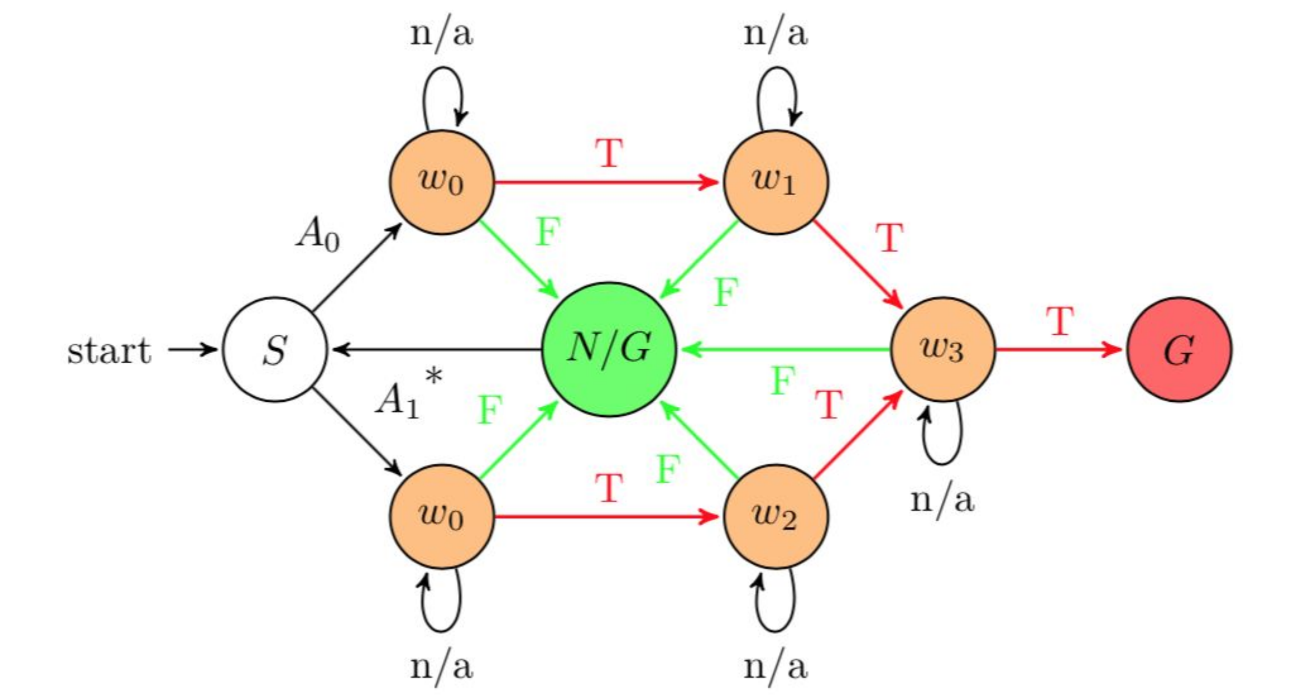
- Service provider puts up a deposit before providing the service
- If the user is not satisfied a trial can be started
- If the trial ends with a GUILTY verdict, the provider loses the deposit
- Otherwise the deposit is returned at the end

Swindle Trial

Trials are a state machine (implemented in a contract)

At every state a **witness** is called and presented with evidence

Outcome determines the next state



swarm

Witness Interface

A **witness** is contract implementing a certain interface

```
function testimonyFor(address owner, address beneficiary, bytes32 noteId)
public view returns (TestimonyStatus);
```

On-chain Swear

1. Service provider posts deposit to Swear contract
2. Plaintiff can open a trial on-chain
3. Swindle handles the trial
4. If the verdict is GUILTY deposit goes to the plaintiff
5. Otherwise the provider can withdraw after the timeout

Off-chain Swear

1. Service provider signs a SWAP promissory note
   a. Remark encodes the trial rules contract and some payload
   b. validUntil is the timeout for the service
   c. Swear is the escrow witness (and implements the Witness interface)
2. If there is no dispute, there is no on-chain activity
3. In case of a dispute the plaintiff can submit the note to Swear
4. Swindle handles the trial
5. If the verdict is GUILTY deposit Swear will allow the note to be used

swarm

Oracle Trial

A simple test trial of 2 Oracle Witnesses

- answer can be controlled by owner
- meant to be used in testing

Both oracles need to accept the evidence for a GUILTY verdict

Uses the on-chain mechanism

Hash Trial

A simple test trial of 1 Hash Witness

- NOT GUILTY if the preimage of a hash can be presented
- GUILTY if timeout

Basically a very primitive form of chunk insurance

(compatible with POC-2 and POC-3)

Uses the off-chain mechanism with SWAP

ENS Mirror Trial

A simple test trial involving contract interaction

- Provider promises to mirror ENS record
- If the ENS record is not updated in time, a trial can be started
- Not yet compatible with the new code

Code

All the code can be found at

github.com/ethersphere/swap-swear-and-swindle/tree/rewrite

master branch (not default!)

There is also documentation!

More (Solidity and Go) developers for sw3 needed!

swarm

The End