Slido Q and A from Week 2

An Offtopic Lawrence, I was doing some basic research last weekend around upgradeable smart contracts, I was wondering what are your thoughts about them?

```
 Upgradable contracts are great since you can include new features to your contract or even bug fixes in case it happens. However this indeed goes against code immutability, but it's an important feature nonetheless.
```

Are topics like smart pointers, concurrency required to excel while coding zk apps

```
 Well, understanding the concepts are certainly beneficial when coding for zkSNARKs apps.  But remember that the specific requirements can depend from the nature of the zkSNARK app that you are working on.
```

Did I understand well that Mina blockchain has possibility of keeping the state of the blockchain always the same size? If yes, how is this achieved?

```
 Yes, it's because of the recursive zk-SNARKs
```

For the `mut` reference we cannot have 2 mut references to avoid writing in the same time got it. But the reference AND the owner can write in the same time?

```
 The owner and a mutable reference cannot mutate a value at the same time. There is a data race when two or more pointers access the same memory location at the same time. https://doc.rust-lang.org/book/ch04-02-references-and-borrowing.html?highlight=data%20race#mutable-references
```

How can I skip to more hands-on coding? I know already Rust and blockchain, and want to learn ZKP math and Rust-based tools/libs.

```
 We only have a short introduction in Rust and some exercises, however the upcoming materials will be focused more on zkEVMs.
```

how's wraparound implemented?

```
 The Some(value) is an enum variant which is part of the
Option<t> enum that can be Some or None. This is useful when
handling values without using the null references.
https://doc.rust-lang.org/book/ch06-01-defining-an-enum.html?
highlight=option#the-option-enum-and-its-advantages-over-
null-values
```

I have a silly question, thinking of solidity/vyper for smart contracts, what would be use rust for ?? (so I can start reading about 😊)

```
 In the last couple of year Rust has become extremally
popular because of it's efficiency and type safety and a lot
of ZK technologies such as Starknet or Aztec are Rust based.
```

I thought ZkSync was a generalised L2? What specific application is it build for?

```
 Indeed it's a generalized L2 Solution.
https://www.alchemy.com/ecosystem/zksync
```

in the above g(a) loop example error . will that be caught in compilation ? or runtime

```
 During compilation. Rust’s borrow checker ensures that
references to an object do not outlive the object itself.
```

Is data availability literally just about where the data of the transactions is stored? So is it only for calls that retrieve data, and not state changes?

```
 It includes both calls that retrieve data and state changes.
```

Is match like switch in other languages?

```
 Somewhat like that, although it has different rules.
https://doc.rust-lang.org/book/ch18-00-patterns.html
```

Polygon PoS was missing from the l2 comparison table - is it considered a "side chain" rather than an L2? Can you elaborate on the difference?

```
 Yes, Polygon PoS is considered a combination of a sidechain
and L2. Maybe the following article will give you an overview
on the differences: https://polygon.technology/blog/layer-2-
demystified-how-polygon-scales-ethereum
```

so we use Rust because of the memory handling in the EVM?

```
 Yes, that's correct. You can check Aztec/Starknet for more
implementations in Rust, though memory managment in Rust is
not the same as memory management in the EVM.
```

talking about derivation: can the L2 automatically confirm the execution on the L1?

```
 L2 can refer to L1, but Laurence is discussing about L2 at
this point.
```

where are the extra rust materials from last week?

```
 Discord channel in the read me section.
https://discord.com/channels/705799923014041651/1166011505540
604027/1177196199535726673
```

Where would I need rust primarily in zk? Developing the protocol OR building DAPPs on it?

```
 In the last couple of year Rust has become extremally
popular because of it's efficiency and type safety and a lot
of ZK technologies such as Starknet or Aztec are Rust based.
```

why use s = String::from"("hello") rather than just s = "hello"?

```
 Your version will be a &str rather than a String.
https://doc.rust-lang.org/book/ch04-03-slices.html?
highlight=%26str#string-slices
```

Can you please talk a bit more on the L2s from trust model and assumptions, upgradeability of the components, etc. I think in a multi-chain world, they matter.

```
 Sure! We will go through these topics in the upcoming
lessons.
```

how is ownership handled for vec values? for Some(&1), 1 is an integer. Is it something about heap storage

```
 a vec owns the values it contains which means when the vec
is dropped all the elements are dropped as well. For your
```

```
example 1 is indeed an integer, but &1 is a reference to an integer.
```

How is the y in the inner scope equal to 5?

```
 Because when you are going in the match function, the first arm is not valid, and then it goes to the second arm which is valid Some(y) => ... This is because we have set y = 10 and this validates the Some(y) option.
```

I am interested in your opinion about unsafe rust.

```
 Good question, I believe it all comes down to the developer. I would advise for developers that are starting to learn Rust to not use it. However, with more experience in Rust, the best way to go around it is to keep the unsafe blocks small.
```

if bridges are "for asset transfer", any asset transfer from layer2 to layer 1 should be done through them? I'm correct?

```
 Yes, typically moving assets from L1 to L2 and L2 to L1 are through bridges which facilitate this transfer.
```

Is T, or type, the only kind of generic, or are there others?

```
 "Generics are not limited to type parameters like T. You can define a function that uses generics, or you can define structs and enums that uses generics, impl blocks, trait definitios etc. https://doc.rust-lang.org/book/ch10-00-generics.html"
```

let mut v1_iter = v1.iter(); Why don't we need to use v1.mut_iter() or so?

```
 To use v1.mut_iter(), v1 will need to be mut, which in our example is not.
```

So the only difference between an array and a vector is that the array is fixed size and the vector dynamic (can change size)?

```
 Yes, that's correct. However there are other differences such as an array are stored on the stack vs vectors which
```

keeps all the elements in an array allocated on the heap. https://hashrust.com/blog/arrays-vectors-and-slices-in-rust/

Sorry, I'm still not clear on the difference between let num = 4, and let num = Some (4), and why we would need to use the second version?

The second version creates an Option, which is a way to handle possible absences/presence of a value. https://doc.rust-lang.org/book/ch06-01-defining-an-enum.html?highlight=option#the-option-enum-and-its-advantages-over-null-values

Sorry, probably a stupid question, but what is the trait here? Summary?

Yes, the trait that we defined is called Summary. A trait is a collection of methods that you can define.https://doc.rust-lang.org/book/ch10-02-traits.html

What data does "data availability layer" contain? I assume smart contracts bytecode and state variables but what about event logs? Could you give some example?

Includes the transaction details, state variables, event logs, block metadata and SC bytecode.

what is L3?

Layer 3 developed on Layer 2. Madara is one example https://github.com/keep-starknet-strange/madara https://book.starknet.io/ch03-05-layer-3.html

What is the advantage of scope of a variables in this manner in rust?

Some examples that I can think of are the following: memory safety and preventing ownership/borrowing issues.

What's the difference between iterating with "for x in v" and "for x in &v" in Rust?

one is borrowed (&v) and the other one is moved https://doc.rust-lang.org/book/ch04-02-references-and-borrowing.html?highlight=borrow#references-and-borrowing

Why we need bridges in OP? The prover contract should do the job, right?

```
 It's two different things. The bridge facilitates transfer of funds between L1/L2 while the prover contract handles the submission of transaction batches and the fraud-proving scheme.
```

Do rollups have validators?

```
 Those would be known as Sequencers.
```

Does EIP-4844 hurt data availability since blobs will be deleted

```
 Their deletion doesn't impact the execution of transactions/contract on Ethereum.they are designed to be around for as long as they are needed
```

For the Layer 2 they don't have a consensus mechanism (BFT, nakamato,etc....) this is just sending the information through the sequencer on the L1?

```
 L2s rely on the security and consensus mechanism of L1 (Ethereum).
```

how about the bridge? if it is updatable, wouldn't that negate roll-up

```
 Bridges and Rollups serve different roles and do not negate each other.
```

How are funds moved between a rollup bridge contract and L1 bridge contract?

```
 We will talk about them in upcoming lessons, but for more information check the following article https://ethereum.org/en/bridges/
```

What about the cross chain communication ?

```
 L1-L2 communication is possible, we will explore some of it in week 3
```

What are your thoughts on the Espresso Sequencer?

```
 we'll take a look at it tomorrow
```

What exactly is stored on the off chain database? Also, doesn't an off chain database imply that it is not a blockchain?

```
off chain generally means not layer 1
```

What is the actual difficulty with decentralizing the sequencers? Isn't it's only job ordering transactions?

```
Good question, I know that most of the L2 are already working on decentralizing the sequencers. As for difficulties, afaik, since the technology is new, some challenges would be mostly around the design to ensure security, efficiency and fairness.
```

When a token minted on L2, how can it have the same functionality as the L1 token, if its generalized?

```
You would need to create the same token contract on L1 as well if you want to have functionality on L1 and L2.
```

When we talk about forced withdrawals are we only talking about withdrawing the native token of the L1?. Or could they be of other tokens / ERC20s?

I believe it could be other ERC20s if it is supported.

where is the challenge period of fraud proof configured

```
Usually in the SC that manages the rollup.
```

Will we have another session with Patrick? 😃

```
I don't believe so.
```

Can you provide more concrete examples of the different setups and why they can be so different?

```
STARKS setups don't have the randomness that it's needed as a secret and agree on a hash function and we refer to them as transparent setups. SNARKS on the other hand they need a trusted setup with public inputs and some private inputs (witness in our case).
```

Have we previously talked about public parameters and setup phase? I don't remember them and am a bit confused

```
It will be this lesson :)
```

In P(pp, x, w) , where is x coming from? what is w?

```
w - witness, x - public input x - a public input for the proof and w is the witness a private input
```

Is it circuit size depends on the witness size?

```
No it doesn't matter.
```

Is Taiko an example of the Base Layer Rollup type?

```
Don't think so. It's more of a general-purpose ZK-rollup.
```

Is this process program -> arithmetic circuits same for all snarks and starks?

```
I believe so! They need to be converted to a boolean circuit and then to an arithmetic circuit or equivalent using arithmetic encoding.
```

So am I correct in thinking that the sequencer only sends forced tx's to the L1 when something has gone wrong with the L2?

```
Correct. If the sequencer would have been malicious, this would be a way to force the transaction to be accepted.
```

What are the open problems related to zk in general which community is currently trying to solve.

```
afaik, I would say decentralisation, privacy and data availability.
```

what does the trace mean in this note?

```
It's a record of the computation performed on your program which is then used to generate the proof.
```

What exactly is the use for Polynomial Commitment Scheme? Is it to hide the actual polynomial/ckt from the verifier?

```
Part of it and it makes the process more succinct.
```

what would be the values of each cell in a real case? I don't understand how the integers in the cells, relates to proving execution 🤔

```
 In a real situation, these could be registers within a VM, or memory location and the cells are the value held in that particular location.
```

Which zk proof system is used in ZK sync?

```
 SNARKS, but with Boojum upgrade they use STARKS.
```

Why does the setup process not have to be super secret anymore? Do some still need to be super secret?

```
 Since STARKS setup is transparent and anyone can verify the correctness of the setup and are not based on trusted setups.
```

why we need CRS

```
 I believe because the schemes are secure given that the setup has been done correctly. The values from the CRS are used in other parts of the proof process.
```