Team Big Time Concrete Architecture Report

https://www.youtube.com/watch?v=Z2lln3Xqd_E

Team Member

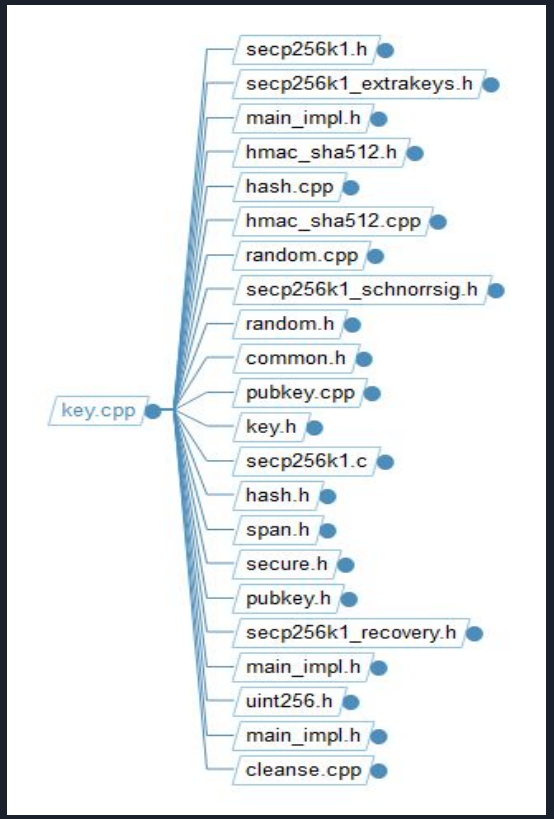
- Lucas Wong- Team Lead
- Azeem Khan- Presenter
- Yannik Brunzema
- Ben Tomkinson- Presenter
- Kenny Wong
- Oscar San

Overview

- Top level concrete architecture
  - Recap conceptual architecture
  - Derivation Process
  - Concrete architecture
  - Reflection analysis
- Subsystem Analysis
  - Inner Architecture
  - Connections
  - Reflection Analysis
- Sequence Diagrams
- Lessons Learned

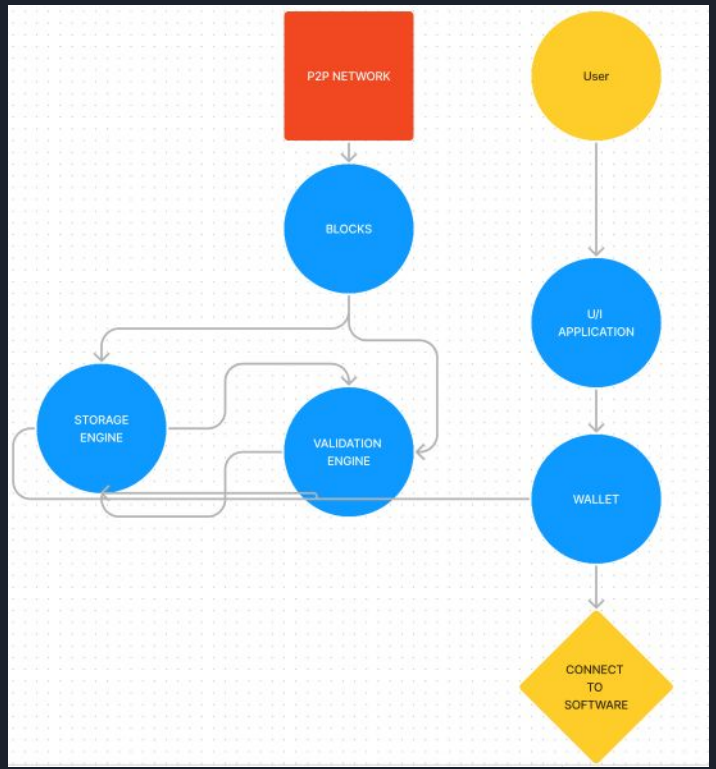
Top Level Architecture: Derivation Process

- Concrete architecture derived through source code analysis
- P2P network also analyzed for insights
- Key.cpp used as example to illustrate dependencies
- Key.cpp depends on hash.cpp and other files
- Dependency implies connection between wallet and security components

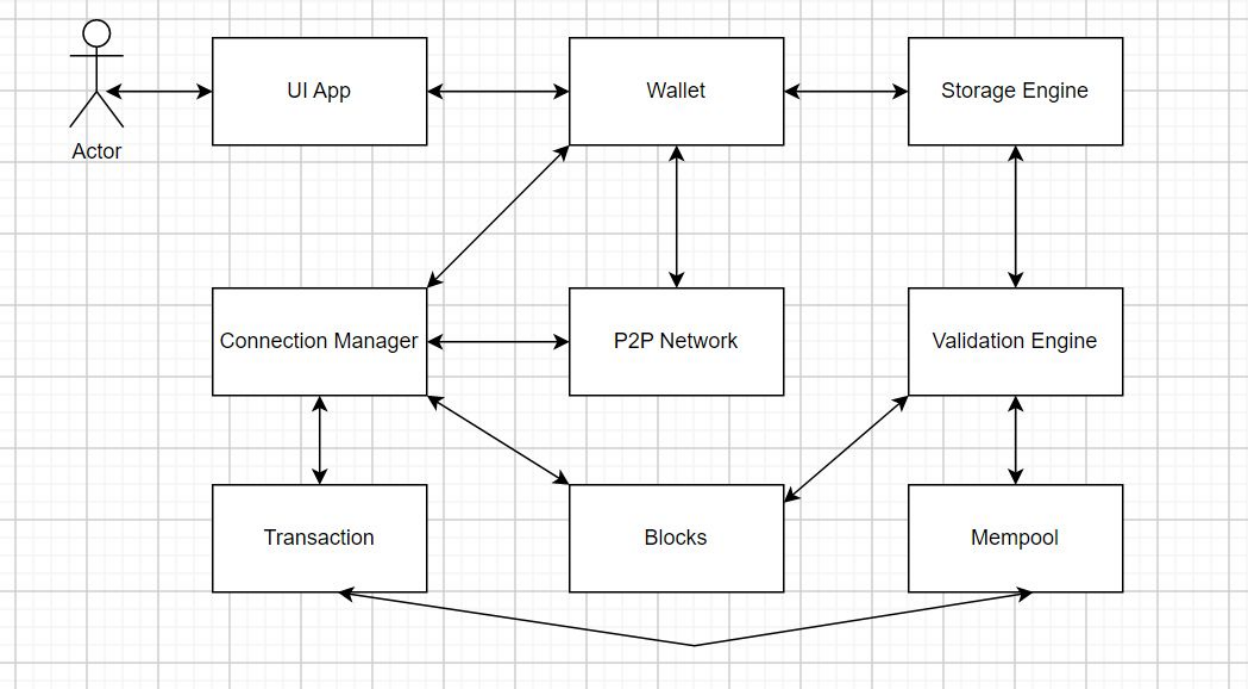


Top Level Architecture: Recap of Conceptual Architecture

- User interacts with the UI and connects to the wallet
- Wallet connects to the Bitcoin Core software
- Wallet obtains core functionality through communication with the storage and validation engines
- Storage and validation engines are dependent on each other
- Engines are closely connected to the P2P network
- Wallet indirectly interacts with the network's branching components
- Interaction between the wallet and network is essential to the software's architecture



Top Level Architecture: Concrete Architecture



Top Level Architecture: Reflection Analysis

- New nodes observed: transaction, mempool, and connection manager.
- Wallet dependencies differ from conceptual to concrete architecture.
- Block node interaction changes in concrete architecture.
- Importance of transaction node not mentioned in conceptual architecture.

Subsystem Analysis: Inner Architecture

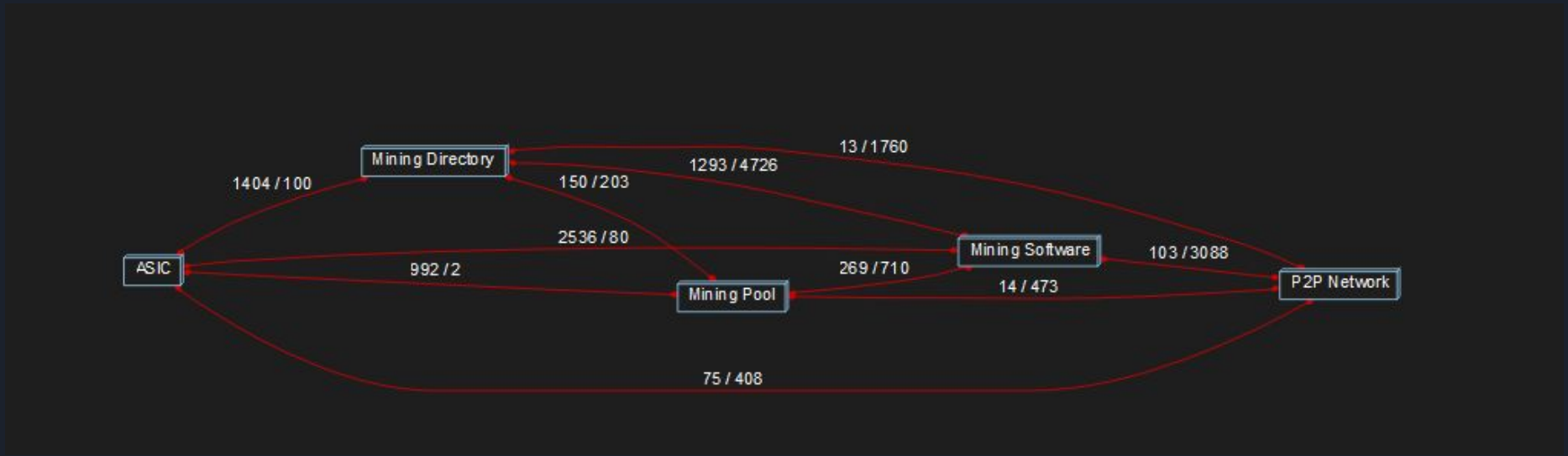
Conceptual Architecture

- Proof of work
- Block Validation
- Block Propagation

Concrete Architecture

- Mining code
- Block and Transaction Validation
- Mempool
- P2P Network

Subsystem Analysis: Connections



Subsystem Analysis: Reflection Analysis

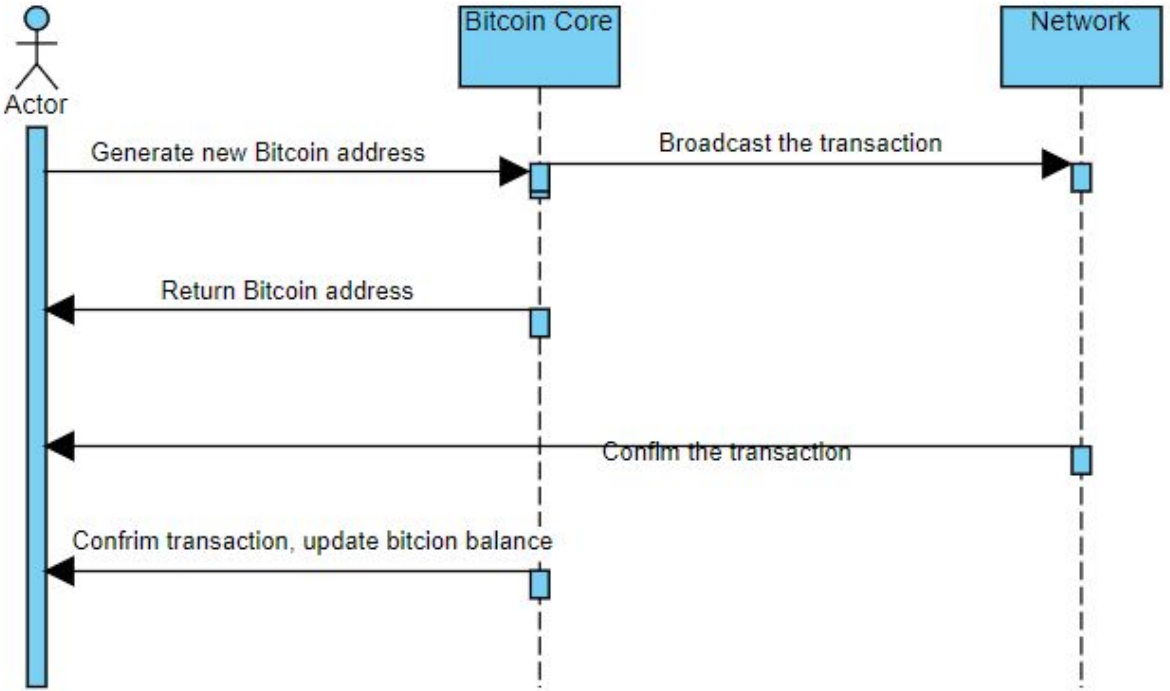
New Modules

- Mempool
- P2P
- ASIC

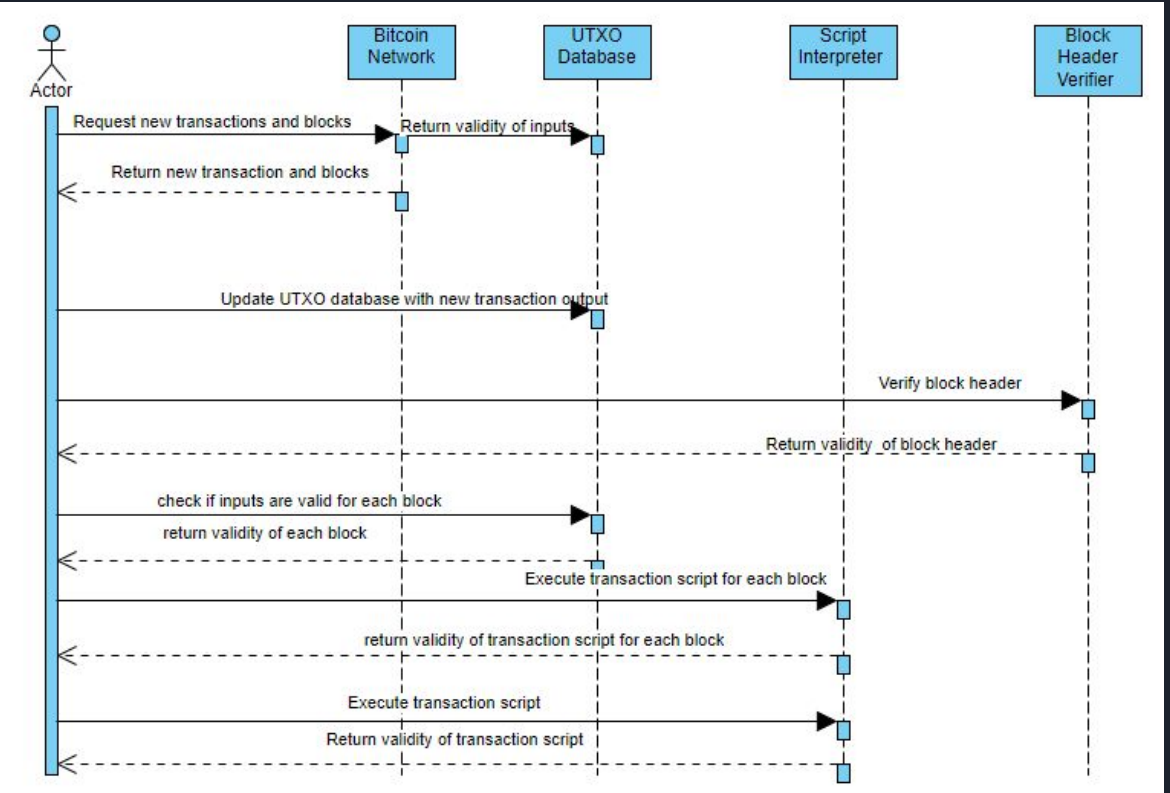
Absences/Unexpected Dependencies

- Miner code in PoW
- Block validation
- ASIC
- Pool mining

Storing Value



Block Validation



Lessons Learned & Closing Statements