Research and Development in Blockchain

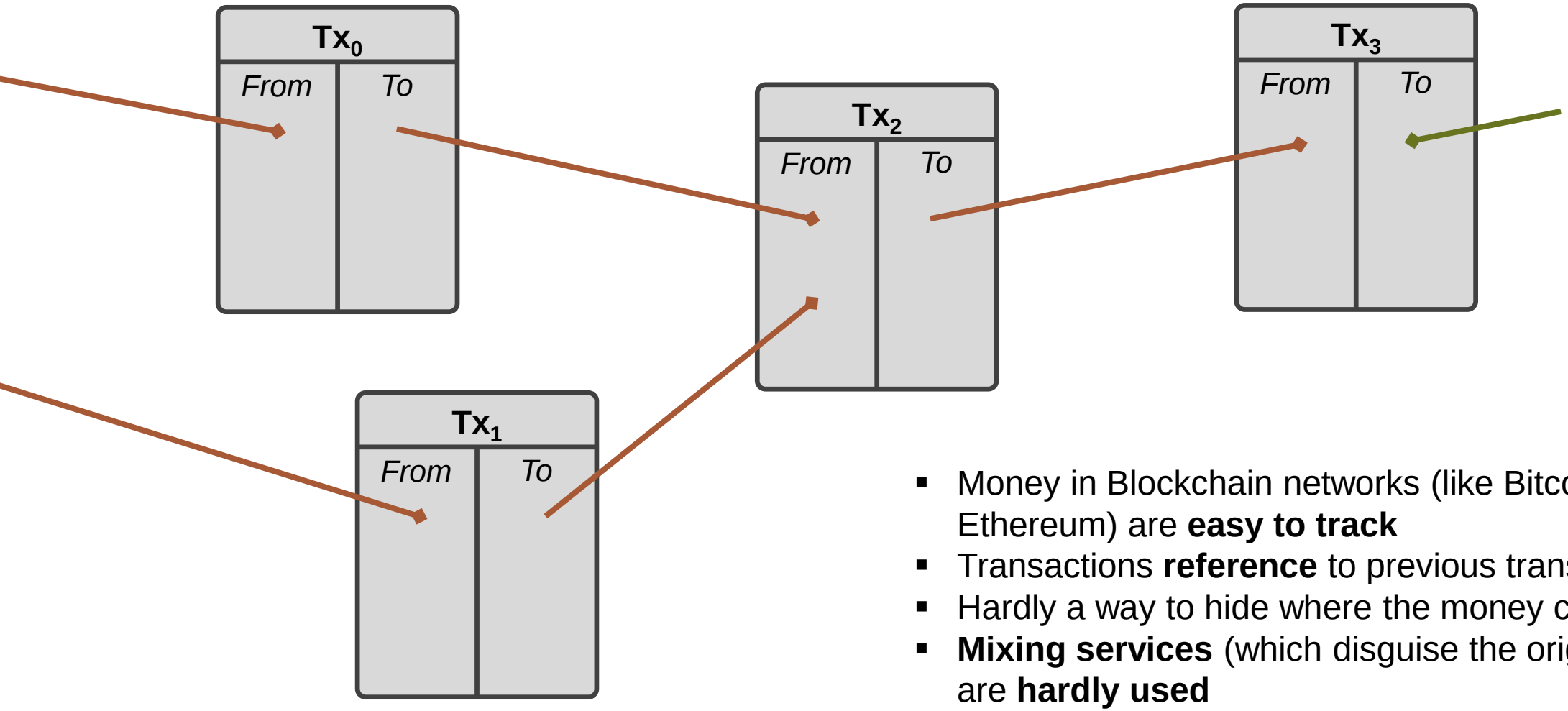
Ulrich Gallersdörfer, M.Sc. – 2018, Sebis-Day

Chair of Software Engineering for Business Information Systems (sebis)
Faculty of Informatics
Technische Universität München
wwwmatthes.in.tum.de

Outline

1. Analysis of Cross-Blockchain Transactions
2. Design Patterns in Solidity
3. Current (and future) Research

Transactions in Blockchains are easy to track



- Money in Blockchain networks (like Bitcoin or Ethereum) are **easy to track**
- Transactions **reference** to previous transactions
- Hardly a way to hide where the money came from
- **Mixing services** (which disguise the origin of coins) are **hardly used**

➔ Scam and theft still a dominant problem without a possibility to efficiently track the money

Coins are exchanged to other currencies

Why are we not able to efficiently track the money? ➔ Currencies are changed into other currencies.

Trading Platforms

BITTREX GDAX BITFINEX POLONIEX

Instant Cryptocurrency Exchange

ShapeShift Changelly

Trading platforms are not used, as an exchange is time consuming and requires user registration.
Instant Cryptocurrency Exchanges allow a **direct exchange** of two currencies.

1. Is it possible to **detect** a transaction for a instant cryptocurrency exchange?

2. Is it possible to **track the flow of money** across **different currencies**?

We trace cross-blockchain transactions

Data usage

- **Transaction data** from two Blockchains
- **Exchange Rates** based on prices, exchange fees & transaction fees
- **Exchange Duration** based on Timestamps
- Known **Exchange Addresses**

Findings

- **Very high volume (200 Mio. / month)**
- **High detection rate** of exchange transactions **(92%)**
- **Correct matching very hard** due to
  - Many possible exchange pairs
  - Over 30 cryptocurrencies
  - Very small transaction amounts
- However, high volume tx traceable

Outline

1. Analysis of Cross-Blockchain Transactions
2. Design Patterns in Solidity
3. Current (and future) Research

Smart Contract Software Engineering is hard

Contract-based language

Notion of money

One single bug can be fatal

High computational costs

We collected 14 different patterns

Behavioral

1. Guard Check
2. State Machine
3. Oracle
4. Randomness

Security

5. Access Restriction / Ownable
6. Check Effects Interactions
7. Secure Ether Transfer
8. Pull over Push
9. Emergency Stop

Upgradeability

10. Proxy Delegate
11. Eternal Storage

Economic

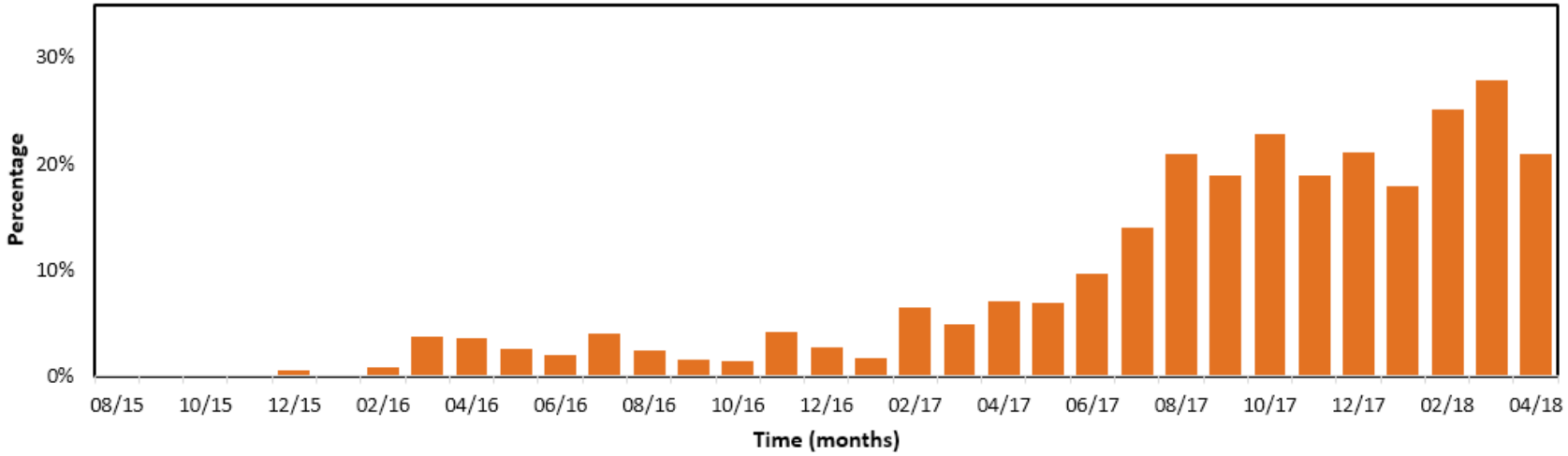
12. String Equality Comparison
13. Tight Variable Packing
14. Memory Array Building

We measure the usage of these patterns

With bytecode analysis, we are able to measure common standards.

Ownable Pattern

- Increasing usage
- Doubled usage from 10% to over 20% in only two months
- >20% of contracts



Patterns are heavily used in Smart Contract development community!

Outline

1. Analysis of Cross-Blockchain Transactions
2. Design Patterns in Solidity
3. Current (and future) Research

Data Analytics in Blockchain

Public Blockchains offer rich data sets

➔ Data is highly diverse: Transaction Data, User accounts, Smart Contracts (95% Bytecode, 5% Source Code), Flow of money

These Data allows for various analytics

➔ Usage of Software Patterns (current project)
➔ Trends in Blockchain
➔ Meta-services (Origin of Money, Taxes, …)

➔ If data analytics is interesting for you, talk to me.

	Ethereum	Bitcoin
Size	100 GB	180 GB
Growth	8Gb/Month	4Gb/Month
Tx	309 Mio.	340 Mio.
Contracts	5,6 Mio.	---

M.Sc.

Ulrich Gallersdörfer
Wissenschaftlicher Mitarbeiter

Technische Universität München
Faculty of Informatics
Chair of Software Engineering for Business Information Systems

Boltzmannstraße 3
85748 Garching bei München

Tel +49.89.289.17137
Fax +49.89.289.17136

ulrich.gallersdoerfer@tum.de
wwwmatthes.in.tum.de