securosys

SICHERE AUFBEWAHRUNG VON TOKENISIERTEN WERTEN UND KRYPTO-WÄHRUNGEN

MARCEL DASEN, VP ENGINEERING

AGENDA

01 / The blockchain hype cycle

02 / Tokenizing assets

03 / Digital assets, crypto currencies, smart contracts

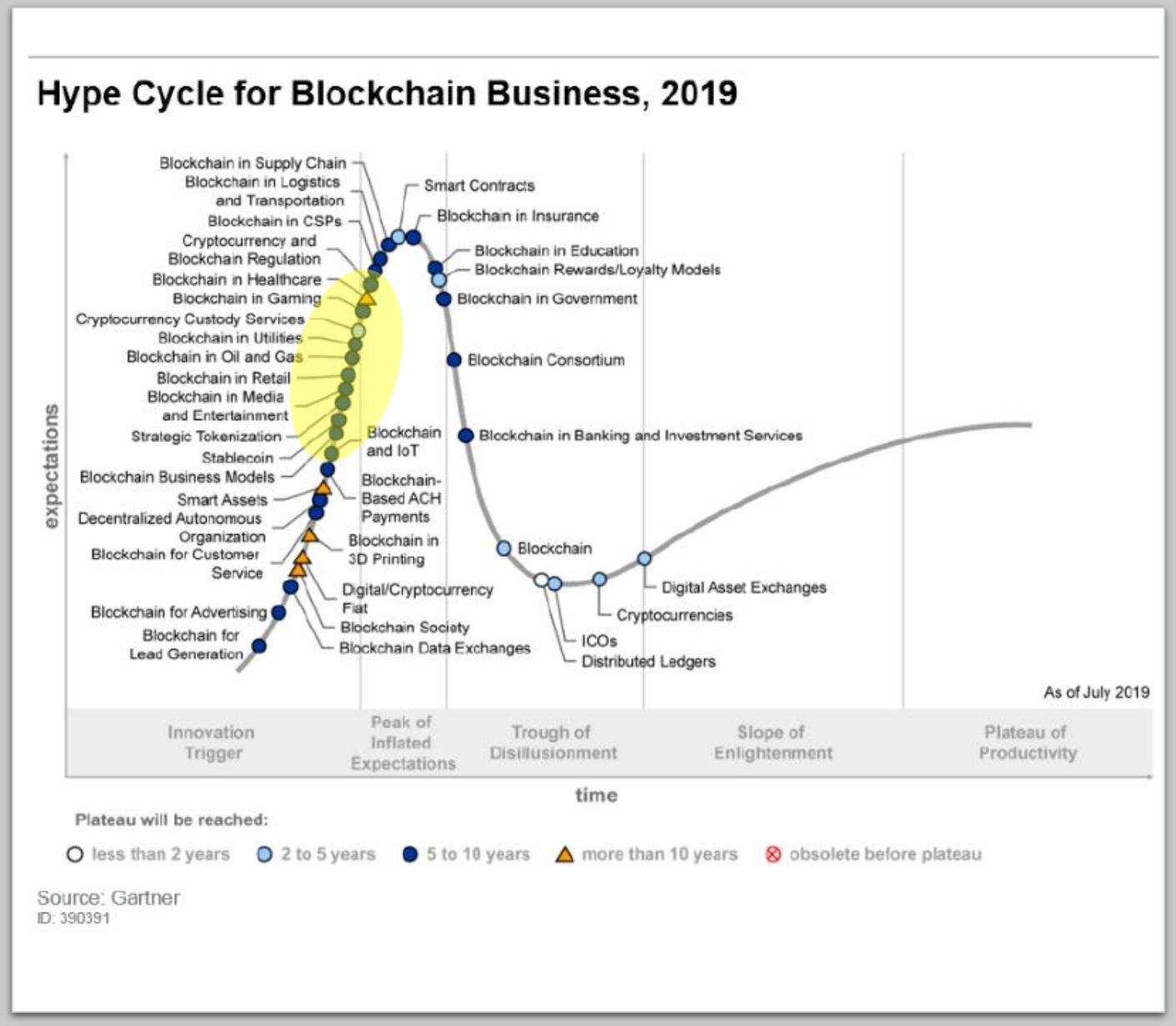
04 / “Asset storage” on the blockchain & securing access to the assets

securosys

01 / THE BLOCKCHAIN HYPE CYCLE

HYPE CYCLE FOR BLOCKCHAIN 2019

BLOCKCHAIN HSM PROTECT TOKENIZED ASSETS AND PERMISSIONED BLOCKCHAINS



securosys

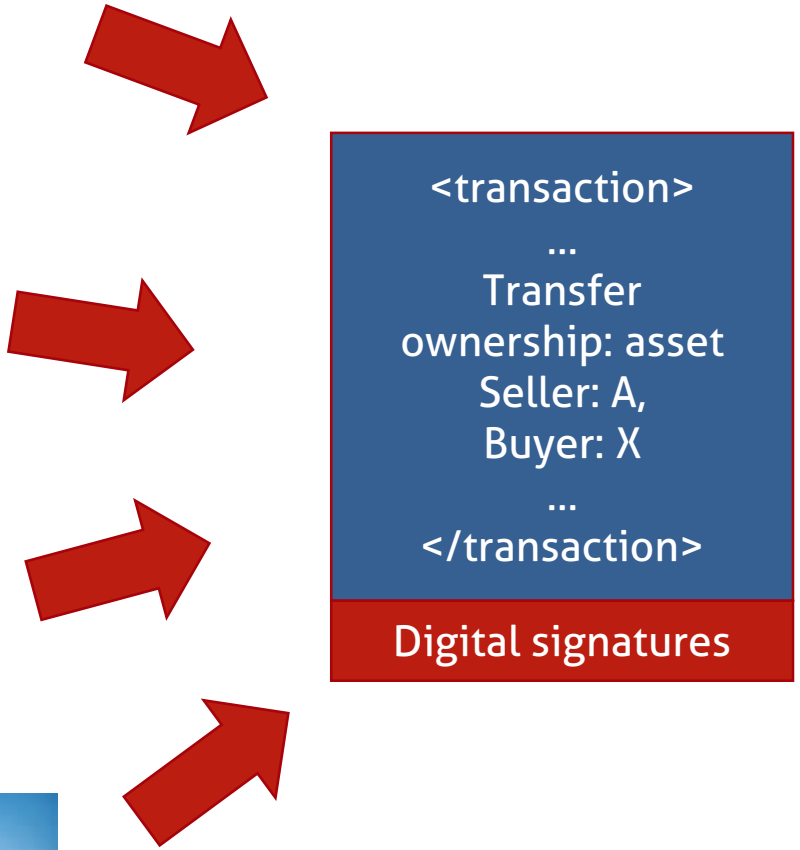
02 / TOKENIZING ASSETS

DIGITAL ASSETS REVOLUTION

- Digital automation of transfer of ownership for any kind of asset
  - Fractional ownership
  - Tokenization of physical goods
  - Public registries
  - Proof of origin (certificate of origin)

TOKENIZED ASSETS

- A tokenized asset is an…
  - immutable
  - digital representation
- …of a real asset



securosys

03 / DIGITAL ASSETS, CRYPTO CURRENCIES AND SMART CONTRACTS

DIGITAL ASSETS, SMART CONTRACTS AND CRYPTO CURRENCIES

Digital asset

<transaction>
...
Transfer
ownership: asset
Source: A,B,C
Dest: X (Y,Z)
...
</transaction>

Digital signatures

Smart contract

<Contract>
...
If (condition) then
execute ...
...
</contract>

Digital signatures

Cryptocurrency

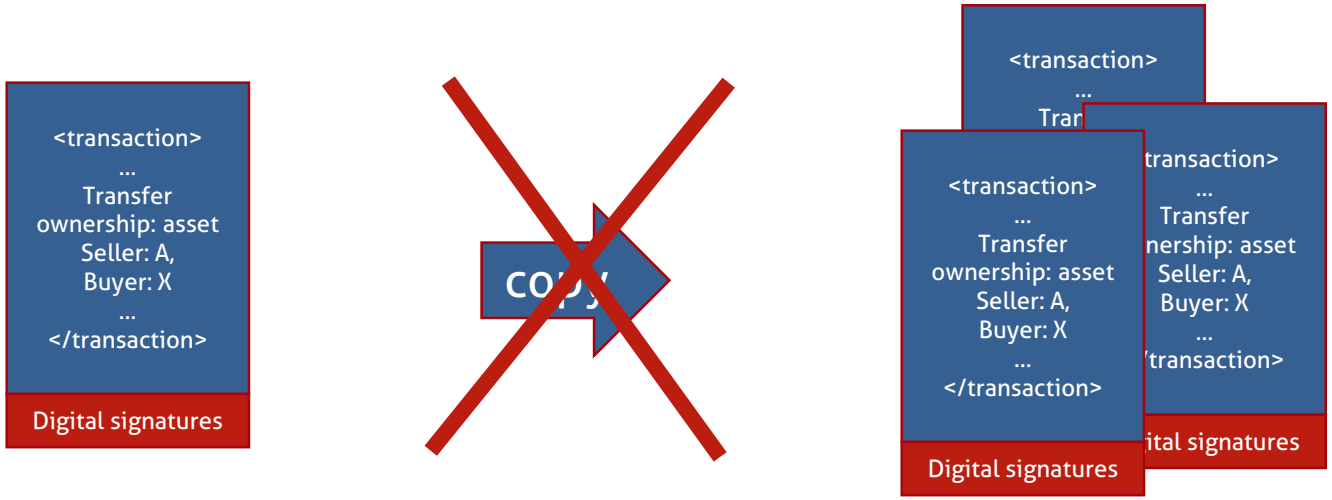
<transaction>
...
pay amount
Source: A
Dest: B

</transaction>

Digital signatures

digital assets

PROBLEM: COPY PROTECTION



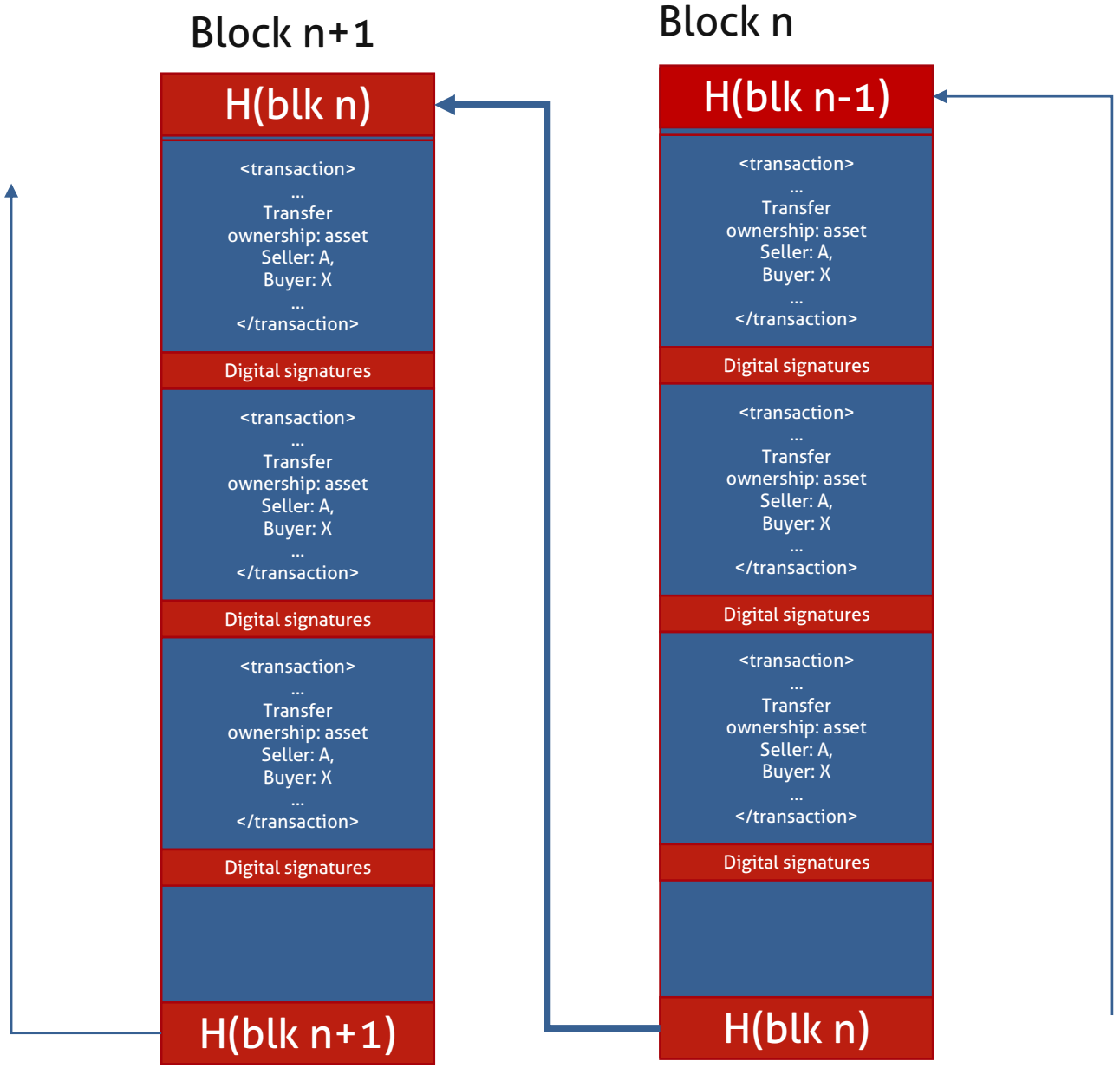
Store on immutable data structure:

→ Blockchain

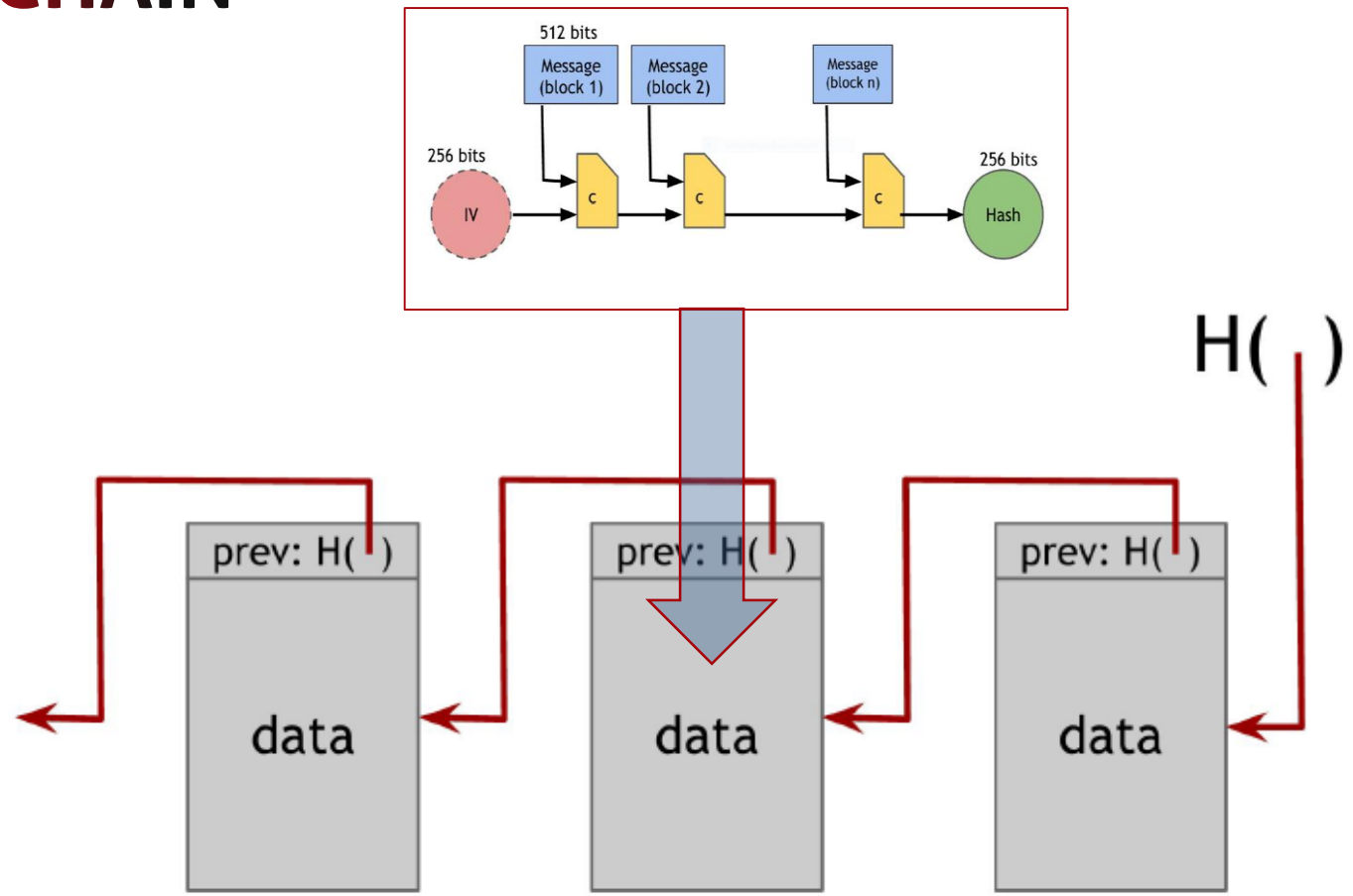
securosys

04/ „ASSET STORAGE“ ON THE BLOCKCHAIN AND HOW TO SECURE ACCESS TO THE ASSETS

COMBINE TRANSACTION IN BLOCKS



STORAGE OF BLOCKS IN A CHAIN



Through chaining of blocks with hash H() the blocks cannot be altered

PROBLEM: BLOCKCHAINS CAN BE COPIED

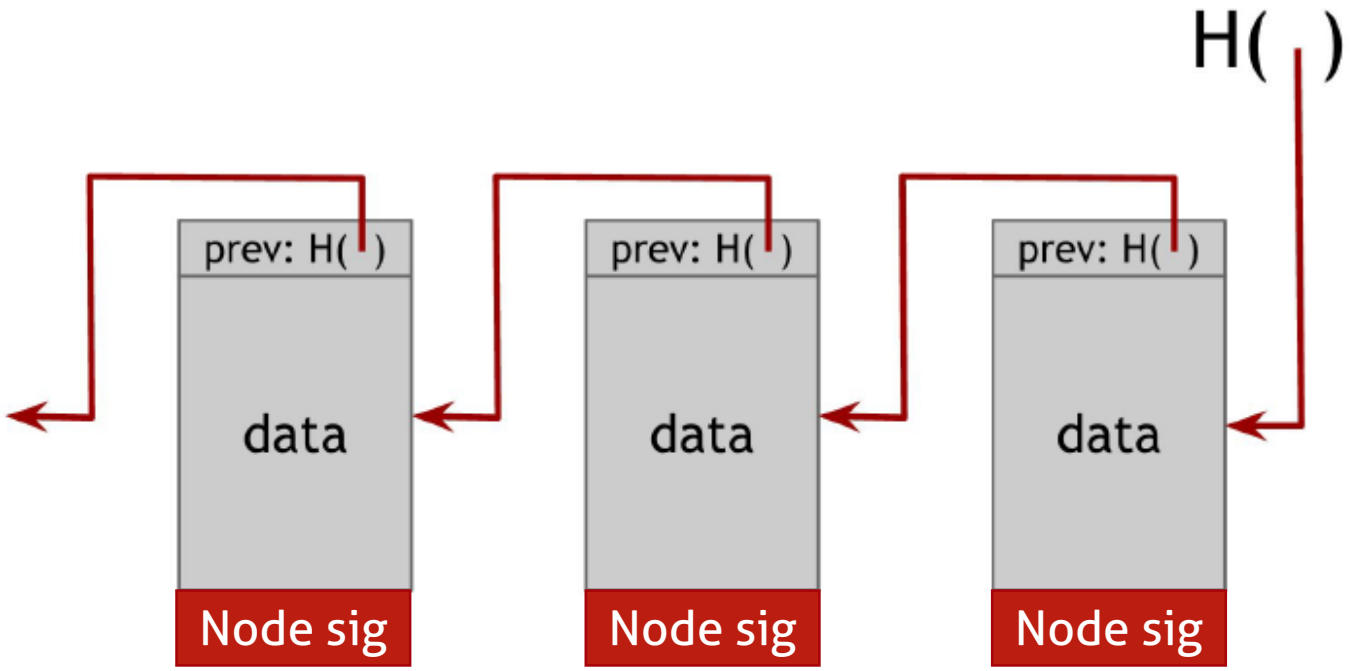
- **It's a feature not a problem!**
  - Everybody can keep his own copy as proof
  - Multiple copies = redundancy = increased fault protection
  - Maintaining can be distributed (DLT)

But there is a new problem: If copied chains are amended locally, which amendment is the "right" one?

DISTRIBUTED LEDGER TECHNOLOGY (DLT)

- Blocks are amended at multiple locations (nodes)
- A **consensus algorithm** guarantees that only one consensus state prevails
  - Permissioned (a distributed DB like algorithm, typically using a **digitally signed** state variable) – examples: Hyperledger, Corda, …
  - Proof of work The first to solve a puzzle – example: BTC
  - Proof of stake Proof that you are willing "to pay" – example
  - … various creative ideas …

EXAMPLE PERMISSIONED BLOCK CHAIN



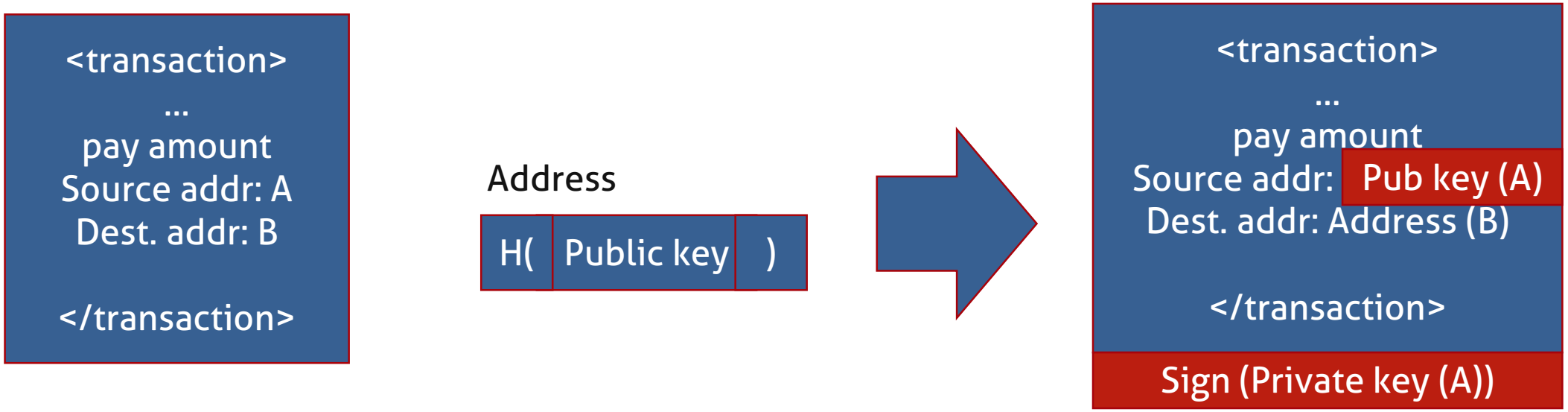
SECURITY OF DIGITAL ASSETS

- Storage “public” and unalterable on blockchain
- Blockchain can be copied; thus, system is reliable
- Consensus on transaction can be distributed
  - no central trusted authority needed (but possible)
- Transaction validation by digital signing
  - Need for reliable storage of private signature keys

WHY TOKENIZING REAL WORLD ASSETS

- Easier to process than physical goods
- Easier to transfer ownership
  - Clearing & Settlement
- Transparency: The history is on the blockchain
- No intermediaries or trusted 3rd parties needed for trades
  - but trust in algorithms
  - but trust in proper execution of algorithms (TEE or SEE)

THE TRANSACTION PROCESS



TRANSACTION BASICS: DIGITAL SIGNATURES

- Three methods required
  - Key generation method: (sk, pk) := generateKeys(keysize)
  - Sign method: sig := sign(sk , message)
  - Verify method: isValid := verify(pk , message , sig)

- Practical concerns
  - Keep sk secret
  - Use addresses (for PQC concern)

STORING OF TOKENIZED ASSETS = STORING OF DIGITAL KEYS

Tokenization requires a wealth of digital keys

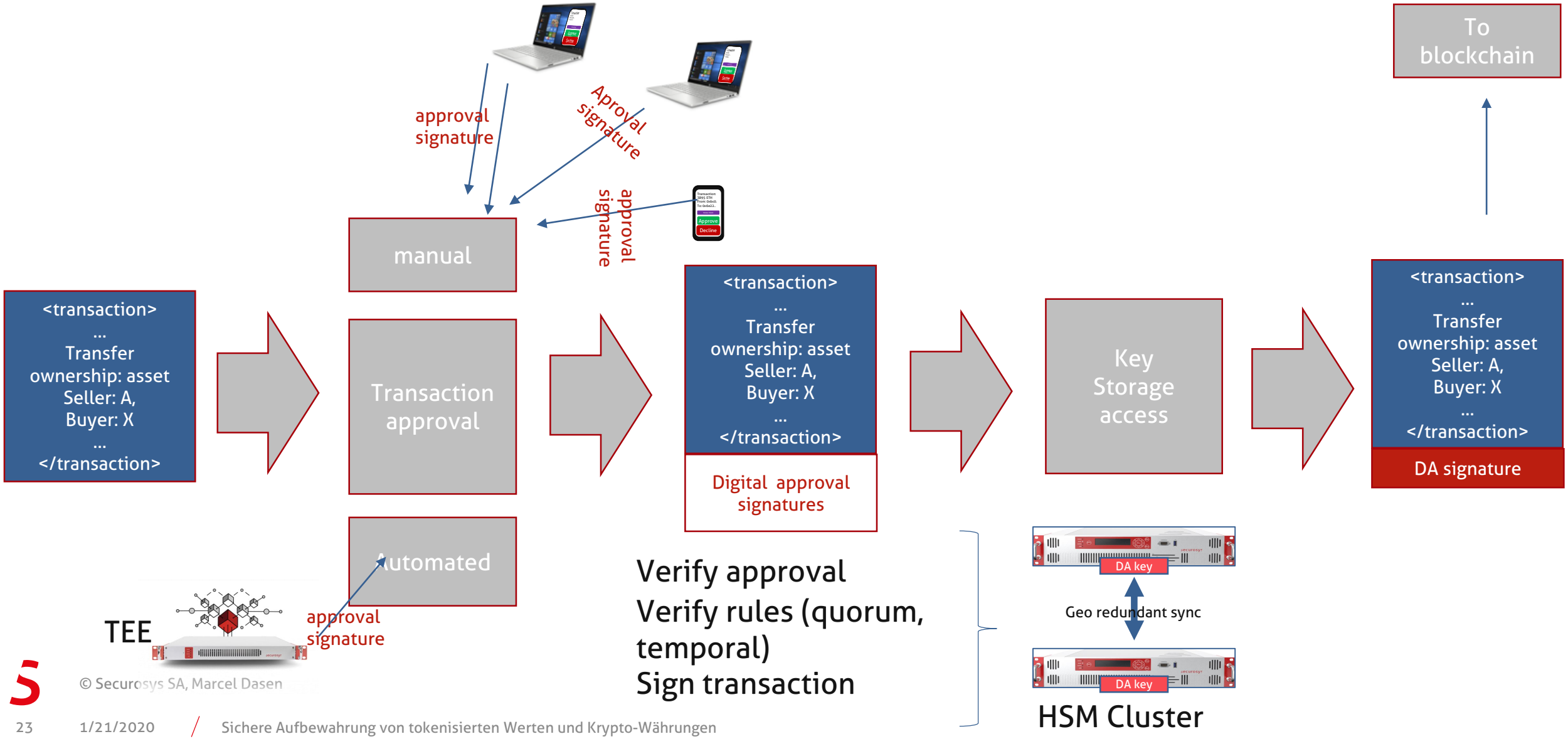
- Billions of asset keys
- Millions of user keys
- Ten thousands of TEE keys
- Thousands of node keys

ACCESS PROTECTION TO DIGITAL KEYS

- Who has access to the keys underlying the token ?
- Multiple regulation sets have to be taken into considerations
  - Rules for custodians, e.g. Banking licenses
  - Rules of beneficial ownership and custodians, auditablity
  - legal succession
  - Good business governance rules
  - Rules for risk mitigation
  - …

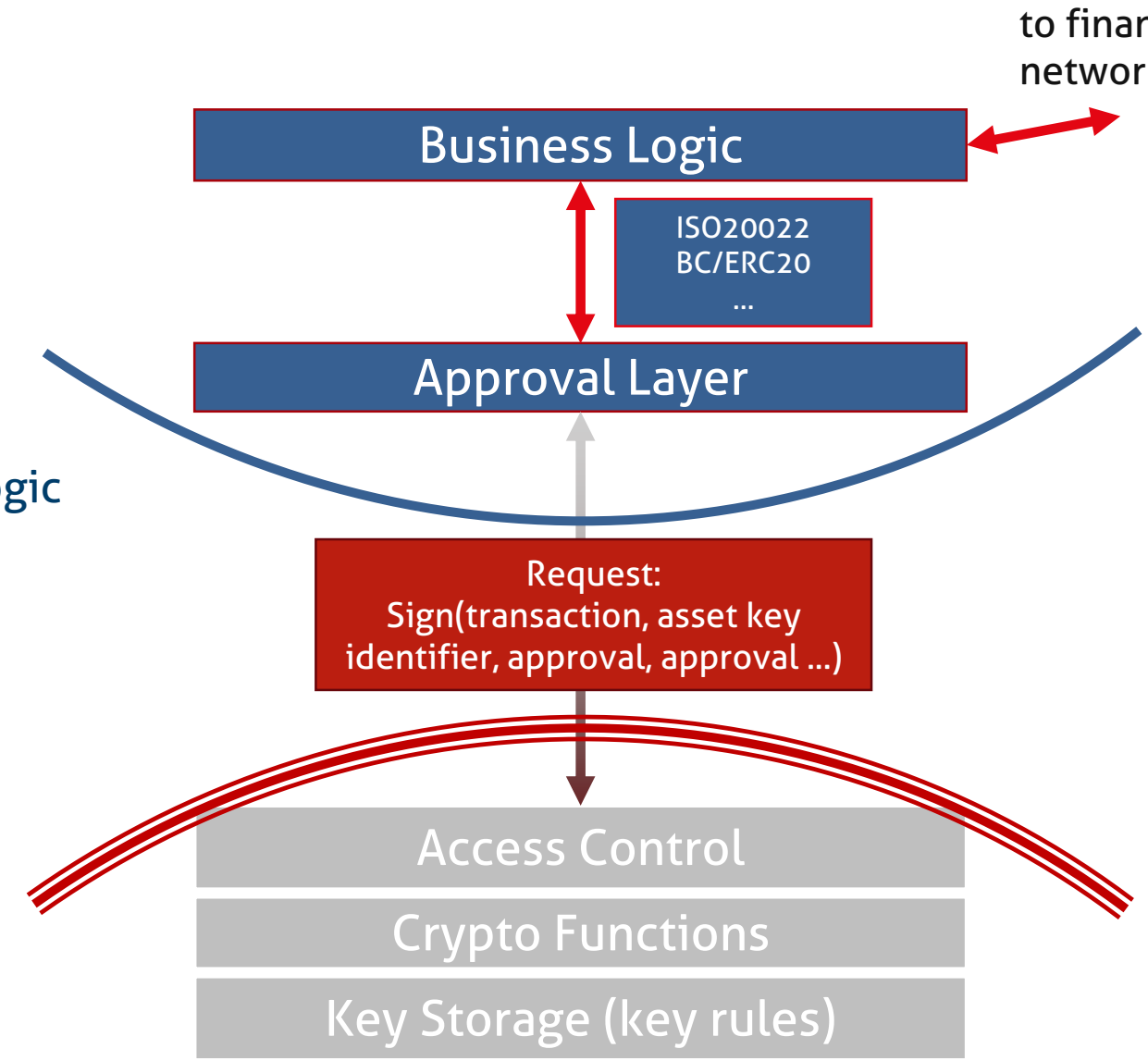
- **All rules must be algorithmically enforced and verified**

SECURE TRANSACTION FLOW



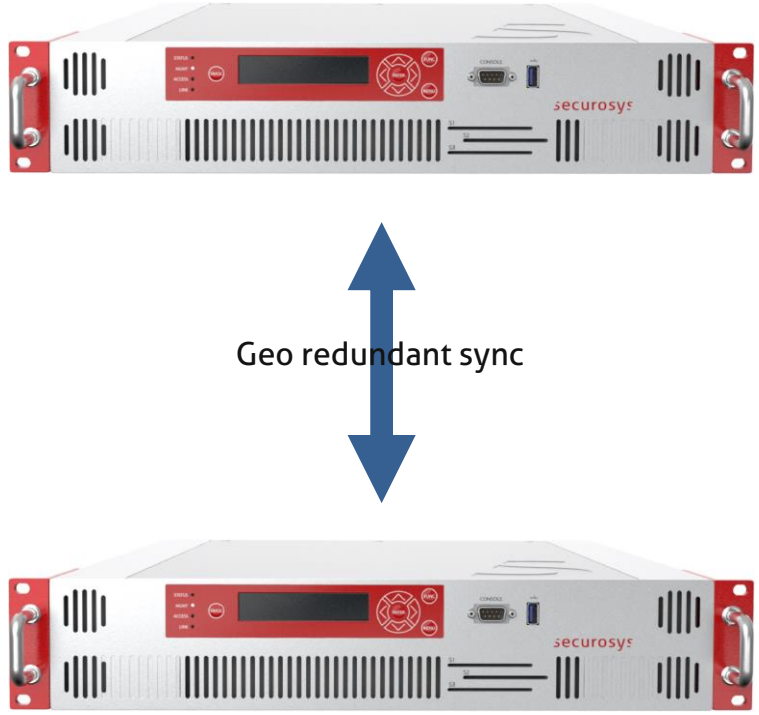
SEPARATING UNSECURE BUSINESS LOGIC FROM SECURED CRYPTO OPERATIONS

- Business Logic & Approval Layer
  - Standard IT infrastructure
  - Virtual, dockerized, cloud
- Transaction Approval
  - Transaction put together by business logic
  - Approvals get collected
  - Transaction authorization verified and signed in HSM
- Secure HSM Environment
  - Long term storage, high reliability
  - (Geo-) redundancy, high availability

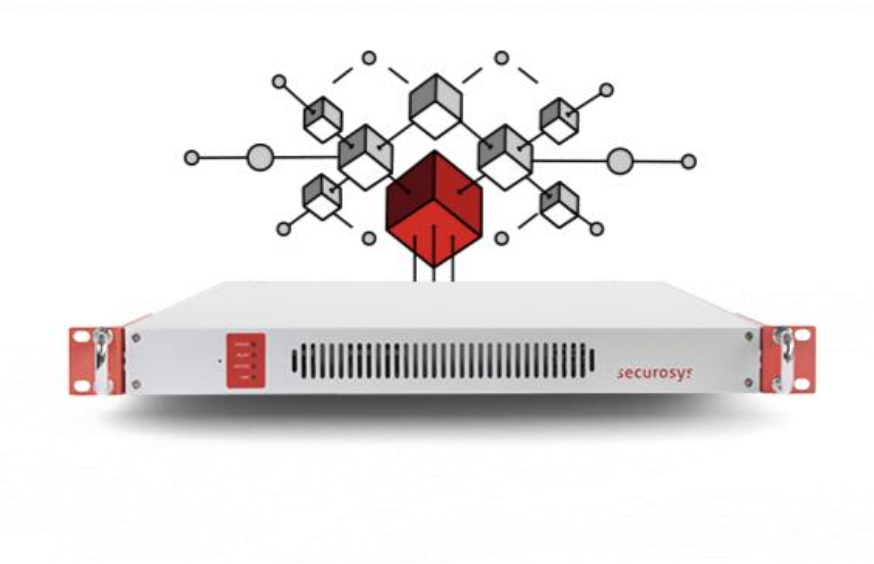


PRIMUS HSM FOR STORAGE OF BLOCKCHAIN AND ASSET KEYS

- Secure key generation
- Key derivation (BIP32,...)
- Access control to keys
- Multi approval
- Address generation (PQC)
- Audit log
- Tamper protection
- Redundancy & Reliability
- "Long" term storage



TRUST IN ALGORITHMS: TRUSTED EXECUTION ENVIRONMENT



- Asserts the "validated code" is executed
  - Verifies the digital signature of the code
  - Asserts code integrity during execution
- Asserts the signature of the input
- Returns a trustable result
  - Digitally signs the result

Provides proof that a validated code produces specific output on defined input

ZUSAMMENFASSUNG

- Technische Zuverlässigkeit
  - Redundanz / backup
  - Geographische Verteilung
  - Langzeitspeicher
- Sicherer Zugriff
  - Mehraugenprinzip -> multiple approval
  - Sichere Regelüberprüfung → Überprüfung in HSM
- Automatisierung:
  - Trusted execution environment
  - Trusted input and output

THANK YOU
for your attention.

securosys