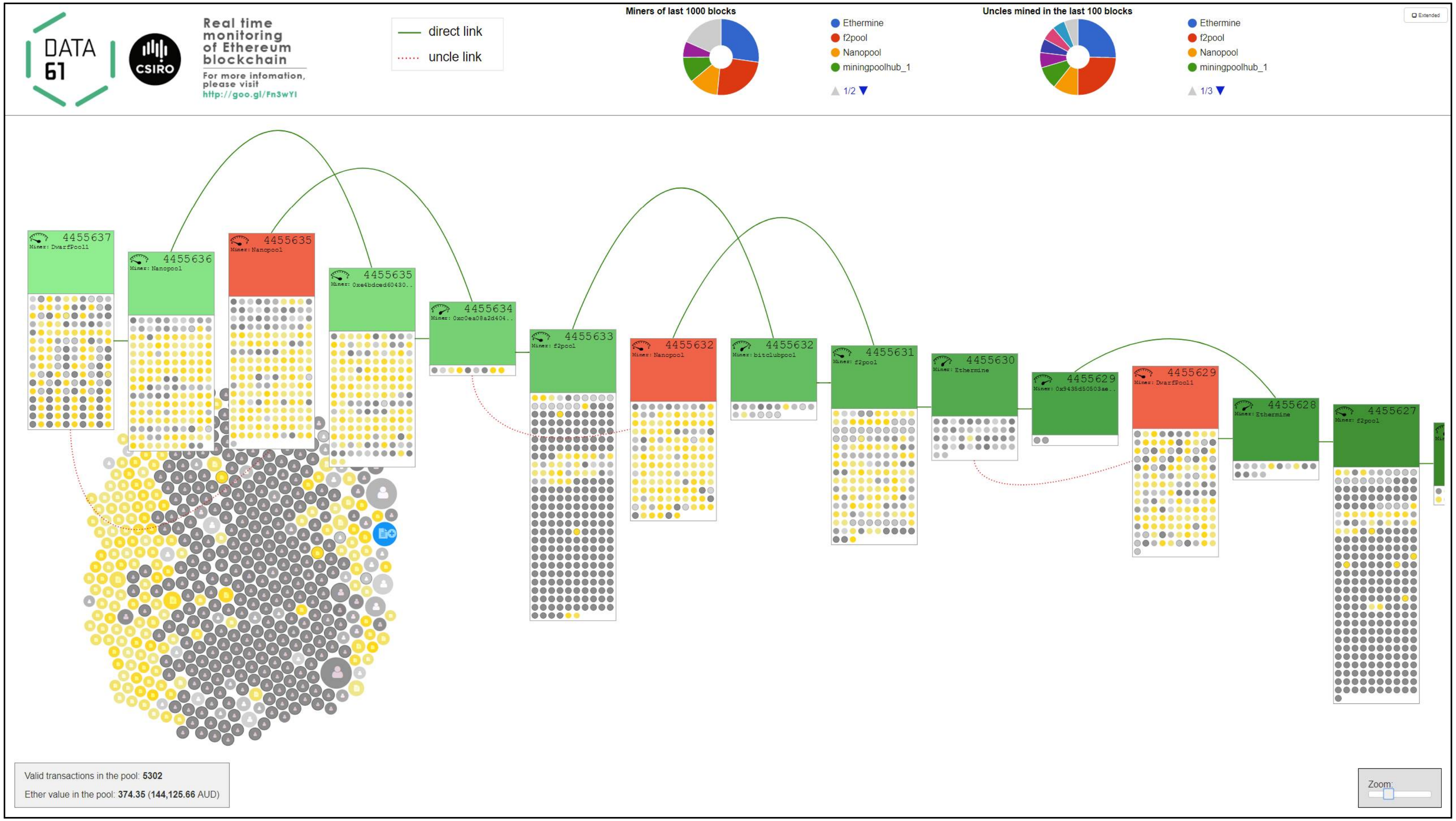
Software Architecture and Engineering for Blockchain-based Applications

Ingo Weber | Principal Research Scientist & Team Leader
ingo.weber@data61.csiro.au

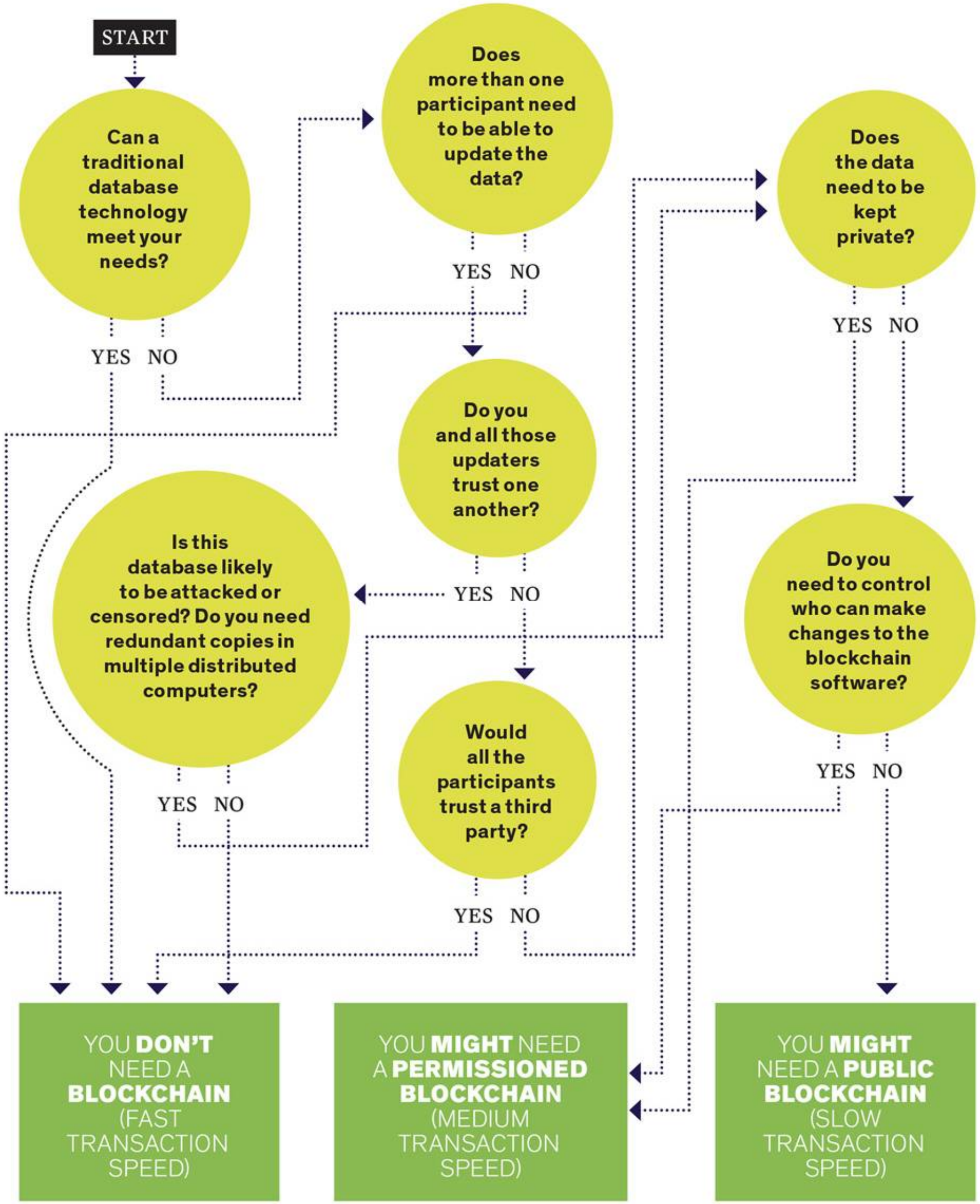
Conj. Assoc. Professor, UNSW Australia | Adj. Assoc. Professor, Swinburne University

www.csiro.au

What does a Blockchain look like?
http://ethviewer.live



What blockchain do you need (if any)?

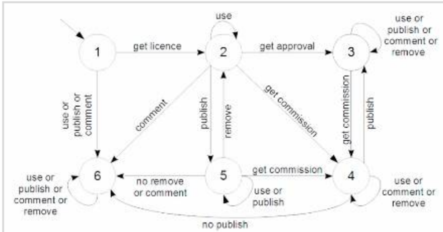


Source: IEEE Spectrum Oct 2017

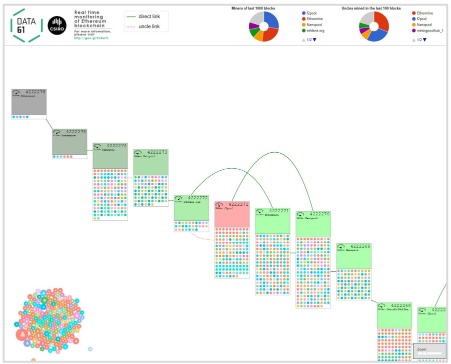
Blockchain Research at Data61

Blockchain Research at Data61

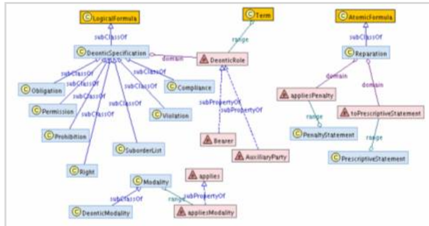
- Designing Systems with Blockchain
  - Design Trade-offs
  - Model-driven development
  - Governance and risk management
- Trustworthy Blockchain Systems
  - Formal



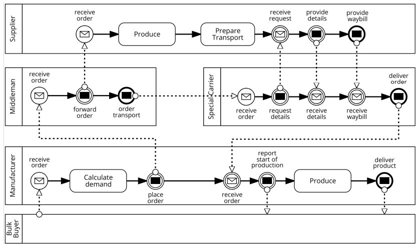
  - Empirical



- Defining and Using Smart Contracts
  - As Legal Contracts

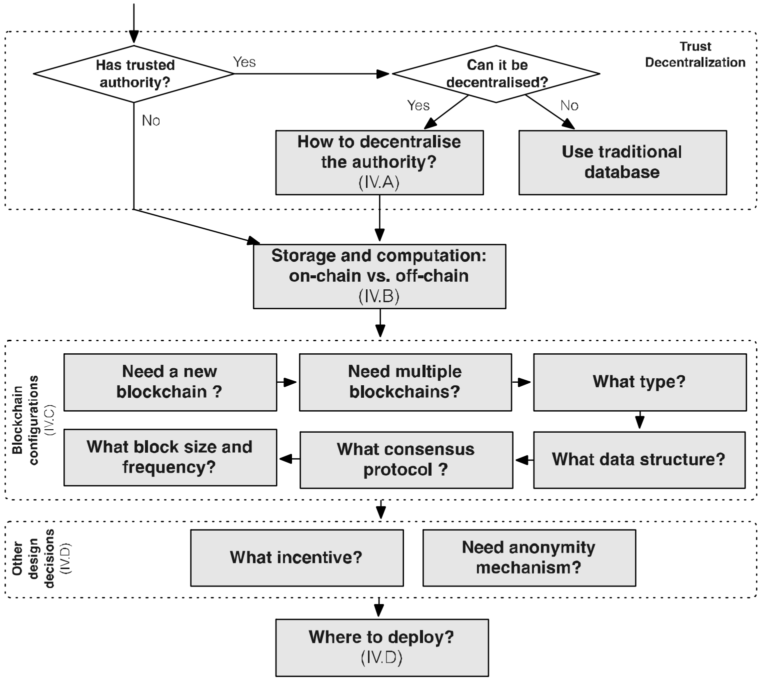


  - Business Process



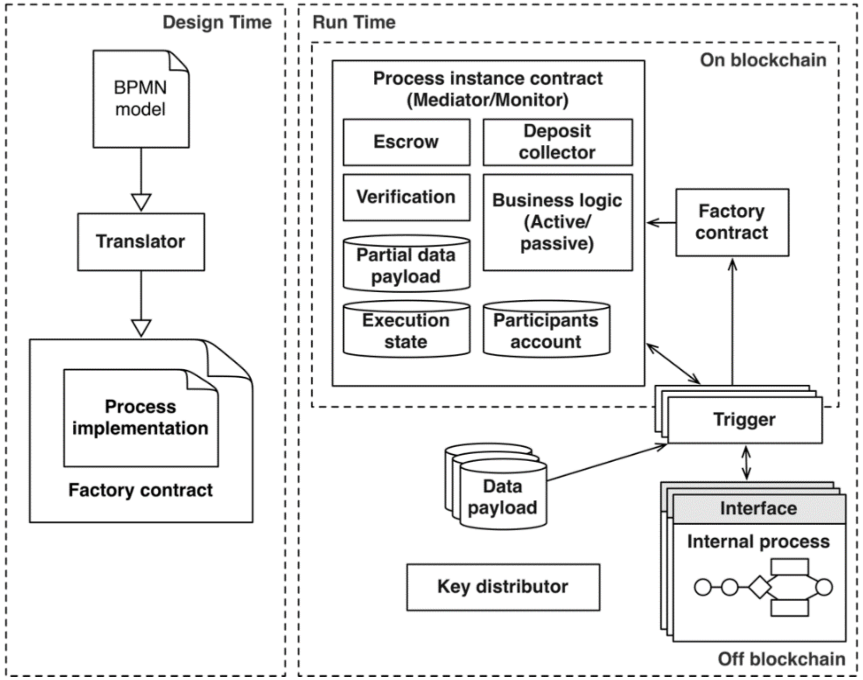
Designing Systems with Blockchain

- Design Process
  - **A taxonomy of blockchain-based systems for architecture design,** X. Xu, I. Weber, M. Staples et al., ICSA2017.
  - **The blockchain as a software connector,** X. Xu, C. Pautasso, L. Zhu et al., WICSA2016.
- Quality Analysis
  - **Quantifying the cost of distrust: Comparing blockchain and cloud services for business process execution.** P. Rimba, A. B. Tran, I. Weber et al., accepted by Scalable Computing and Communications (SCAC) journal, 2017
  - **Comparing blockchain and cloud services for business process execution,** P. Rimba, A. B. Tran, I. Weber et al., ICSA2017.
  - **Predicting latency of blockchain-based systems using architectural modelling and simulation**, R. Yasaweerasinghelage, M. Staples and I. Weber, ICSA2017.
- Model-Driven
  - **Regerator: a Registry Generator for Blockchain,** A. B. Tran, X. Xu, I. Weber, CAISE2017.
  - From business process models, see next slide
- Integration with other systems
  - **EthDrive: A Peer-to-Peer Data Storage with Provenance,** X. L. Yu, X. Xu, B. Liu, CAISE2017.
- Governance and risk management
  - **Risks and Opportunities for Systems Using Blockchain and Smart Contracts,** Treasury report



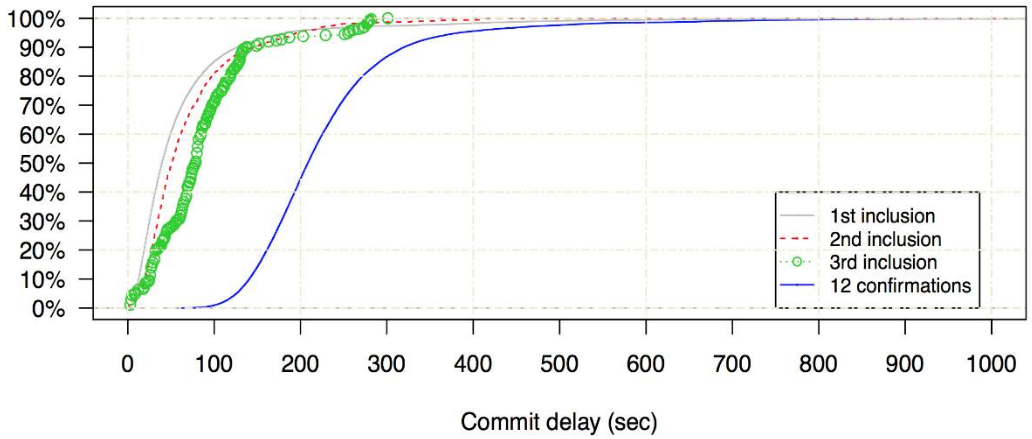
Defining and Using Smart Contracts

- Business Process
  - **Untrusted business process monitoring and execution using blockchain**,
    I. Weber, X. Xu, R. Riveret et al., BPM2016
  - **Optimized Execution of Business Processes on Blockchain**,
    L. García-Bañuelos, A. Ponomarev, M. Dumas, Ingo Weber, BPM2017
  - **Caterpillar: A blockchain-based business process management system,**
    O. López-Pintado, L. García-Bañuelos, M. Dumas, and I. Weber, BPM2017 Demo
  - **Runtime verification for business processes utilizing the Bitcoin blockchain,**
    C. Prybila, S. Schulte, C. Hochreiner, and I. Weber, Future Generation Computer Systems (FGCS), accepted August 2017
- Legal vs. Smart Contracts
  - **Evaluation of Logic-Based Smart Contracts for Blockchain Systems**,
    F. Idelberg, G. Governatori, R. Riveret et al., RuleML2016



Trustworthy Blockchain Systems

- Formal
  - **The Blockchain Anomaly,** C. Natoli, V. Gramoli, *NCA*2016
  - **On the Danger of Private Blockchains,** V. Gramoli, *DCCL*2016
  - **(Leader/Randomization/Signature)-free Byzantine Consensus for Consortium Blockchains,** T. Crain, V. Gramoli, M. Larrea, M. Raynal, arXiv:1702.03068, 2016
- Empirical
  - **On availability for blockchain-based systems,** I. Weber, V. Gramoli et al., *SRDS 2017*
  - **New kids on the block: an analysis of modern blockchains**, L. Anderson, R. Holz, A. Ponomarev et al., arXiv:1606:06530, 2016



Projects with Australian Treasury

- Funded by Australian National Innovation Science Agenda
- Two reports, launched 6 June 2017
- See www.data61.csiro.au/blockchain


CSIRO's Data61 and Treasury join forces to examine the blockchain

Blockchain expected to change the way Australia's economy operates

Jennifer O'Brien (CIO)
04 May, 2016 12:16

- **DLT Foresight**
  - What might plausibly happen, across society & economy?
- **Technical Risks & Opportunities**
  - How do needs in various use cases fit blockchain's capabilities?


BUSINESS INSIDER AUSTRALIA

Tech Insider | Money & Markets | Briefing | Ideas

Recently on LIFEHACKER

How To Tell If A Photo Has Been Doctored

BRIEFING

The CSIRO believes blockchain technology will change Australian industry

CHRIS PASH
JUN 7, 2017, 9:06 AM


Today's Paper | Markets Data

FINANCIAL REVIEW

Home | News | Business | Markets | Street Talk | Real Estate | Opinion | Technology | Personal Financ

Can IT move at the speed of DevOps?

Home / Technology

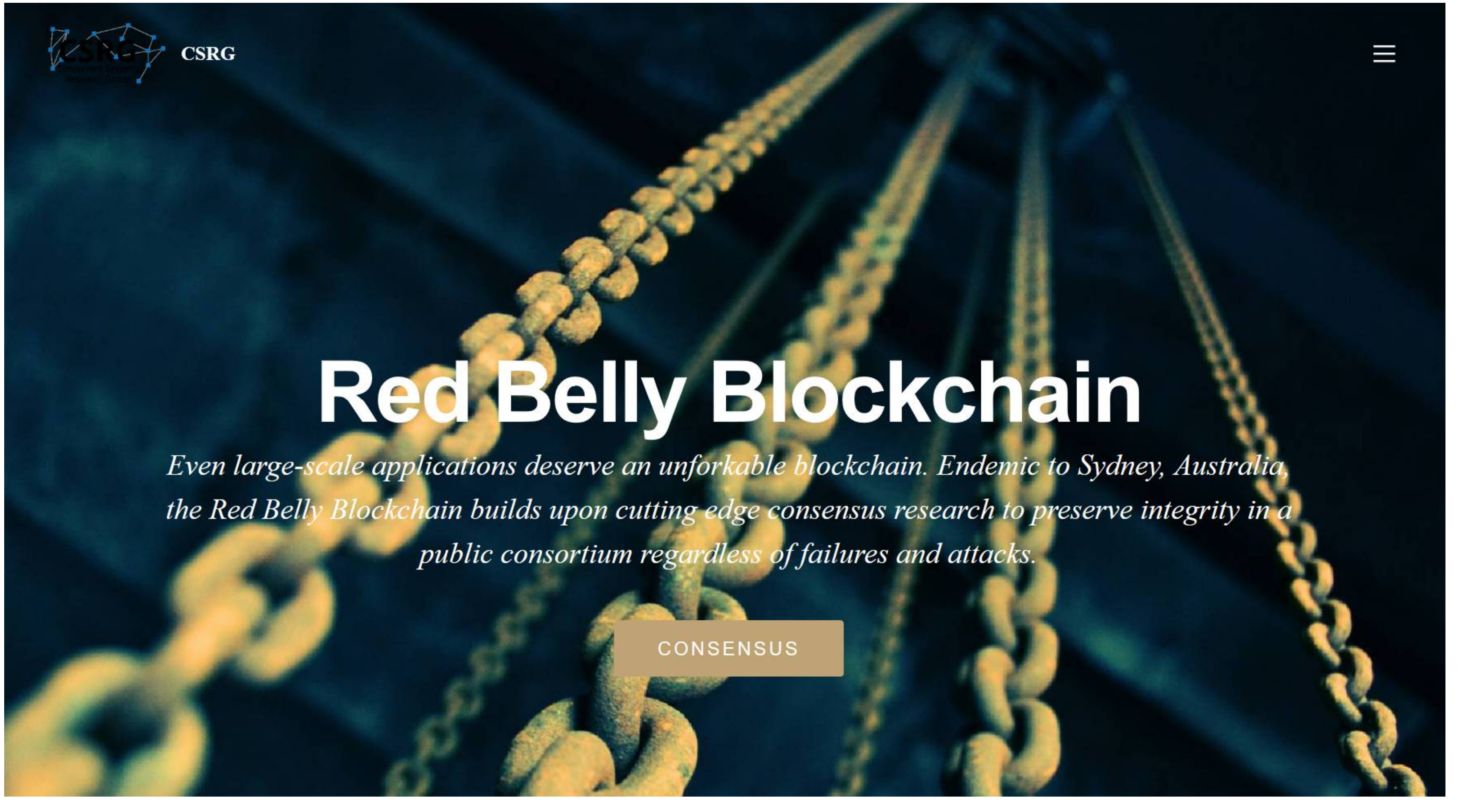
Jun 7 2017 at 12:01 AM | Updated Jun 7 2017 at 12:01 AM

Save Article | Print

Data61 reports blockchain will have a profound impact on the economy

Red Belly Blockchain (with USyd)

- New technology particularly for private / consortium blockchain
- In lab experiment achieved 660,000 tps with 300 nodes in one DC, and > 50,000 tps with globally distributed nodes



Architecting Applications on Blockchain

A taxonomy of blockchain-based systems for architecture design, X. Xu, I. Weber, M. Staples et al., ICSA2017.

Comparing blockchain and cloud services for business process execution, P. Rimba, A. B. Tran, I. Weber et al., ICSA2017, short paper.

Predicting latency of blockchain-based systems using architectural modelling and simulation, R. Yasaweerasinghelage, M. Staples and I. Weber, ICSA2017, short paper.

On availability for blockchain-based systems, I. Weber, V. Gramoli et al., SRDS 2017

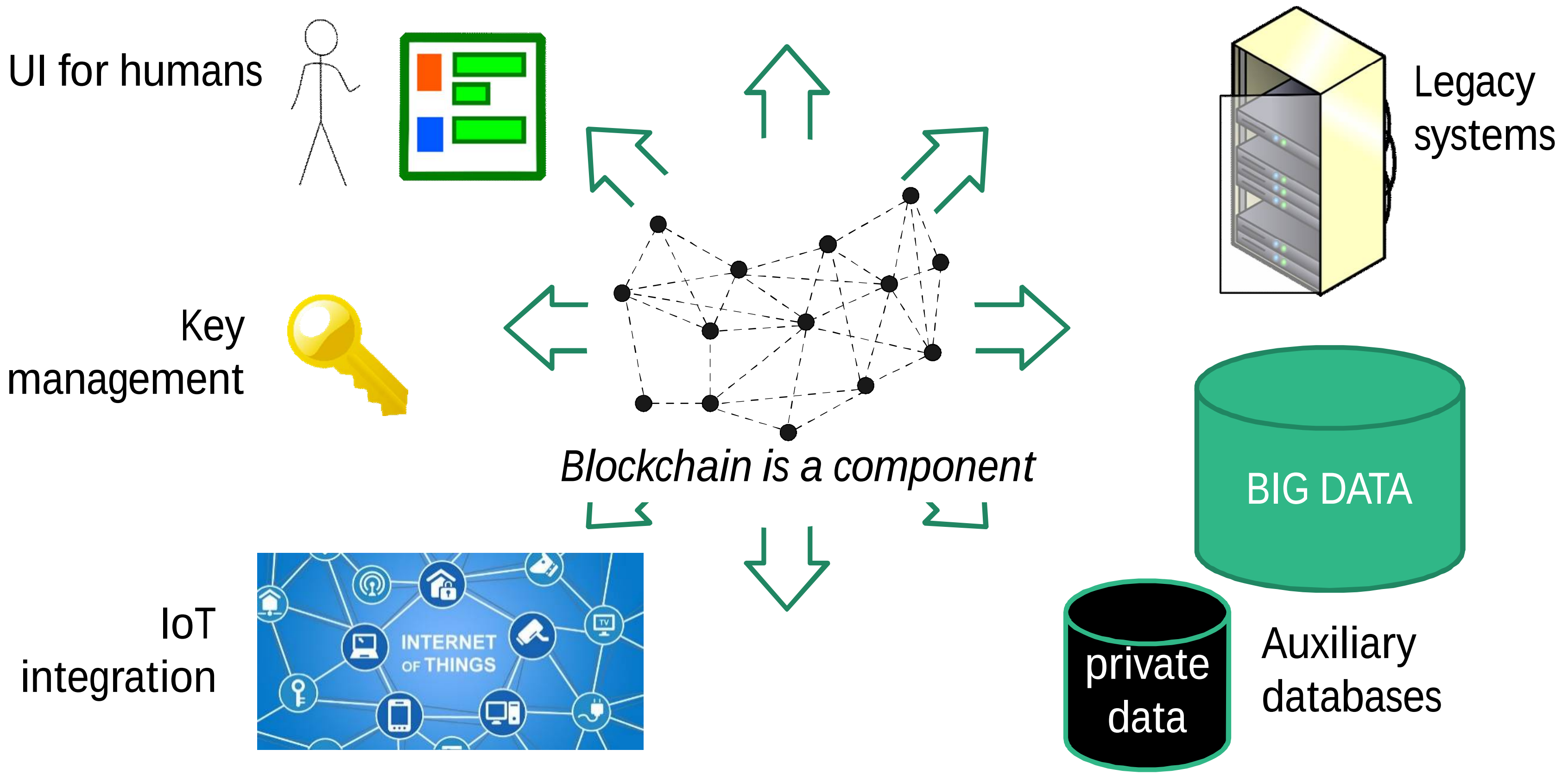
Overview

- Many **interesting applications** for Blockchain
  - Basically of interest in most lack-of-trust settings where a distributed application can coordinate multiple parties
  - Examples:
    - Supply chains
    - Handling of titles, e.g., land, water, vehicles
    - Identity
  - Many startups and initiatives from enterprises / governments
- ... but also many **challenges**
  - When to use blockchain
  - Trade-offs in architecture
    - Downsides: cost, latency, confidentiality
    - What to handle on-chain, what off-chain?

Our work – AAP team

- Architecting applications on Blockchain:
  - Taxonomy and design process
  - “Cost of Distrust”: how much more expensive is blockchain?
  - Availability analysis from viewpoint of DApss
  - Latency: simulation under changes
- Model-driven development of smart contracts
  - Business process execution
  - Model-based generation of registries and UIs

Blockchains are Not Stand-Alone Systems

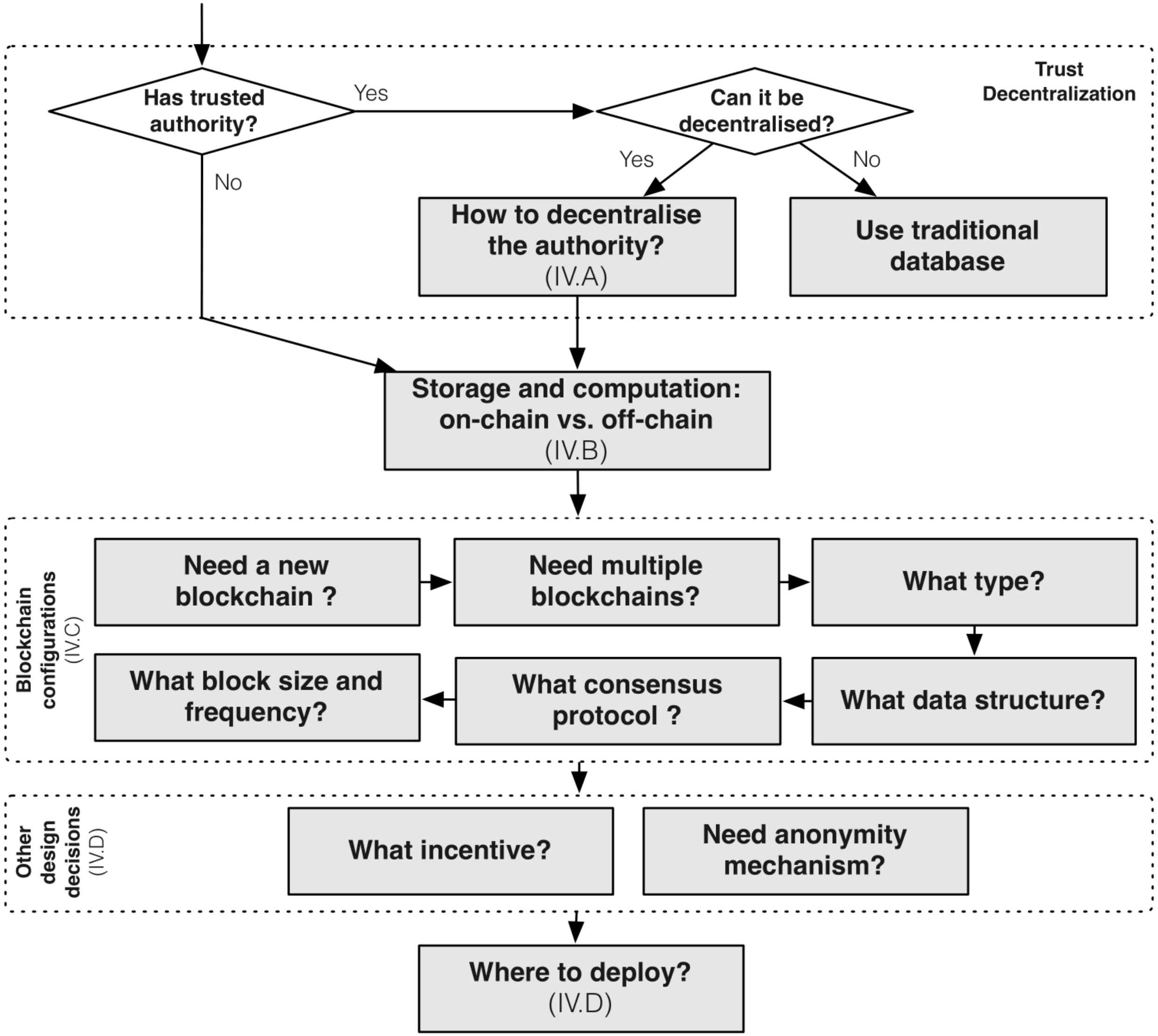


Non-Functional Trade-Offs

- Compared to conventional database & script engines, blockchains have:
  - (-) Confidentiality, Privacy
  - (+) Integrity, Non-repudiation
  - (+ read/ - write) Availability
  - (-) Modifiability
  - (-) Throughput / Scalability / Big Data
  - (+ read/ - write) Latency

Security: combination of CIA properties

Design process



Taxonomy

Blockchain-related design decisions regarding (de)centralisation, with an indication of their relative impact on quality properties

Legend: ⊕: Less favourable, ⊕⊕: Neutral, ⊕⊕⊕: More favourable

Design Decision	Option	Impact: Fundamental properties	Impact: Cost efficiency	Impact: Performance	Impact: #Failure points
Fully Centralised	Services with a single provider (*e.g.*, governments, courts)	⊕	⊕⊕⊕	⊕⊕⊕	1
	Services with alternative providers (*e.g.*, banking, online payments, cloud services)				
Partially Centralised & Partially Decentralised	Permissioned blockchain with permissions for fine-grained operations on the transaction level (*e.g.*, permission to create assets)		⊕⊕	⊕⊕	*
	Permissioned blockchain with permissi[illegible] permission-less normal nodes				
Fully Decentralised	Permission-less blockchain	⊕⊕⊕	⊕	⊕	Majority (nodes, power, stake)

		Fundamental properties	Cost efficiency	Performance	#Failure points
Verifier	Single verifier trusted by the network (external verifier signs valid transactions; internal verifier uses previously-injected external state)	⊕⊕	⊕⊕	⊕⊕	1
	M-of-N verifier trusted by the network	⊕⊕⊕	⊕	⊕	M
	Ad hoc verifier trusted by the participants involved	⊕	⊕⊕⊕	⊕⊕	1 (per ad hoc choice)

Also consider skills available for a specific platform

Taxonomy

Blockchain-related design decisions regarding storage and computation, with an indication of their relative impact on quality properties

Design Decision		Option	Impact			
			Fundamental properties	Cost efficiency	Performance	Flexibility
Item data	On-chain	Embedded in transaction (Bitcoin)	⊕⊕⊕⊕	⊕	⊕	⊕⊕
		Embedded in transaction (Public Ethereum)		⊕⊕⊕⊕	⊕	⊕⊕⊕
		Smart contract variable (Public Ethereum)		⊕⊕	⊕⊕⊕	⊕
		Smart contract log event (Public Ethereum)		⊕⊕⊕	⊕⊕	⊕⊕
	Off-chain	Private / Third party cloud	⊕	~KB Negligible	⊕⊕⊕⊕	⊕⊕⊕⊕
		Peer-to-Peer system		⊕⊕⊕⊕	⊕⊕⊕	⊕⊕⊕
Item collection	On-chain	Smart contract	⊕⊕⊕⊕	⊕⊕⊕⊕ (public)	⊕⊕⊕⊕	⊕
		Separate chain		⊕ (public)	⊕	⊕⊕⊕⊕
Computation	On-chain	Transaction constraints	⊕⊕⊕⊕	⊕	⊕	⊕
		Smart contract				
	Off-chain	Private / Third party cloud	⊕	⊕⊕⊕⊕	⊕⊕⊕⊕	⊕⊕⊕⊕

Taxonomy

Blockchain-related design decisions regarding blockchain configuration

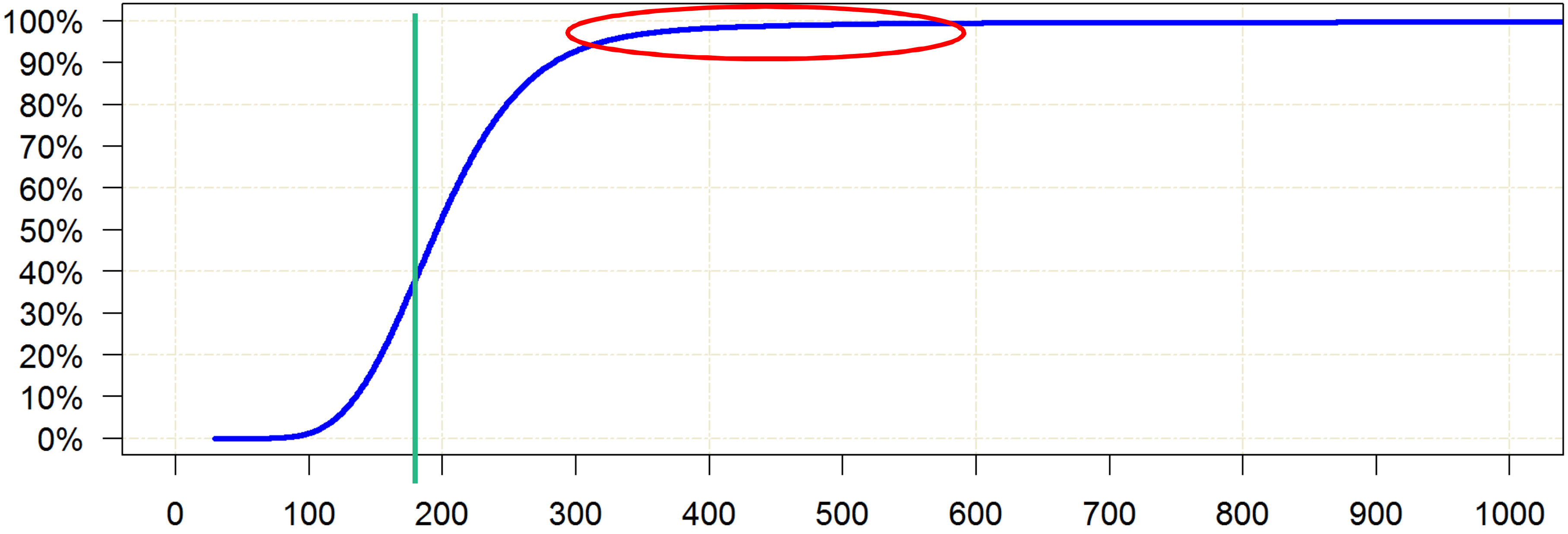
Design Decision		Option	Impact			
			Fundamental properties	Cost efficiency	Performance	Flexibility
Blockchain scope		Public blockchain	⊕⊕⊕	⊕	⊕	⊕
		Consortium/community blockchain	⊕⊕	⊕⊕	⊕⊕	⊕⊕
		Private blockchain	⊕	⊕⊕⊕	⊕⊕⊕	⊕⊕⊕
Data structure		Blockchain	⊕⊕⊕	⊕	⊕	⊕
		GHOST	⊕⊕	⊕⊕	⊕⊕	⊕
		BlockDAG	⊕	⊕⊕⊕	⊕⊕⊕	⊕⊕⊕
		Segregated witness	⊕⊕⊕	⊕⊕	⊕	⊕
Consensus Protocol	Security-wise	Proof-of-work	⊕⊕⊕	⊕	⊕	⊕
		Proof-of-retrievability	⊕⊕⊕	⊕	⊕	⊕
		Proof-of-stake	⊕⊕	⊕⊕	⊕⊕	⊕⊕⊕
		BFT (Byzantine Fault Tolerance)	⊕	⊕⊕⊕	⊕⊕⊕	⊕
	Scalability-wise	Bitcoin-NG	⊕⊕⊕	⊕	⊕	⊕
		Off-chain transaction protocol	⊕	⊕⊕⊕	⊕⊕	⊕⊕⊕
		Mini-blockchain	⊕⊕	⊕⊕	⊕	⊕⊕
Protocol Configuration	Security-wise	X-block confirmation	⊕	⊕	⊕	⊕⊕⊕
		Checkpointing	⊕⊕⊕	⊕⊕⊕	⊕⊕⊕	⊕
	Scalability-wise	Original block size and frequency	⊕⊕⊕	n/a	⊕	n/a
		Increase block size / Decrease mining time	⊕	n/a	⊕⊕⊕	n/a
New blockchain	Security-wise	Merged mining	⊕⊕⊕	⊕⊕	⊕	⊕
		Hook popular blockchain at transaction level	⊕⊕	⊕	⊕⊕	⊕⊕⊕
		Proof-of-burn	⊕	⊕	⊕⊕⊕	⊕⊕
	Scalability-wise	Side-chains	⊕⊕⊕	⊕	⊕	⊕
		Multiple private blockchains	⊕	⊕⊕⊕	⊕⊕⊕	⊕⊕⊕

Cost of Distrust

- RQ: How much more expensive is blockchain over Cloud services?
  - Lens: business process execution
  - AWS SWF vs. Ethereum public blockchain
    - In both cases: pay per instruction
  - Experiments on two use cases:
    - Incident management (literature)
      - 32 instances on public Ethereum vs. 1000 runs on SWF
    - Invoicing (industry, 5316 log traces, 65K events)
      - Full log replayed on SWF and private Ethereum
- Result:
  - 2 orders of magnitude more expensive to use blockchain
    - exchange rate: 1 Ether = US$ 11.32 (26/8/2016)
  - ~US$ 0.35 per process instance on public blockchain
    - outweighed by cost of escrow (if needed) for about US$ 10 of value

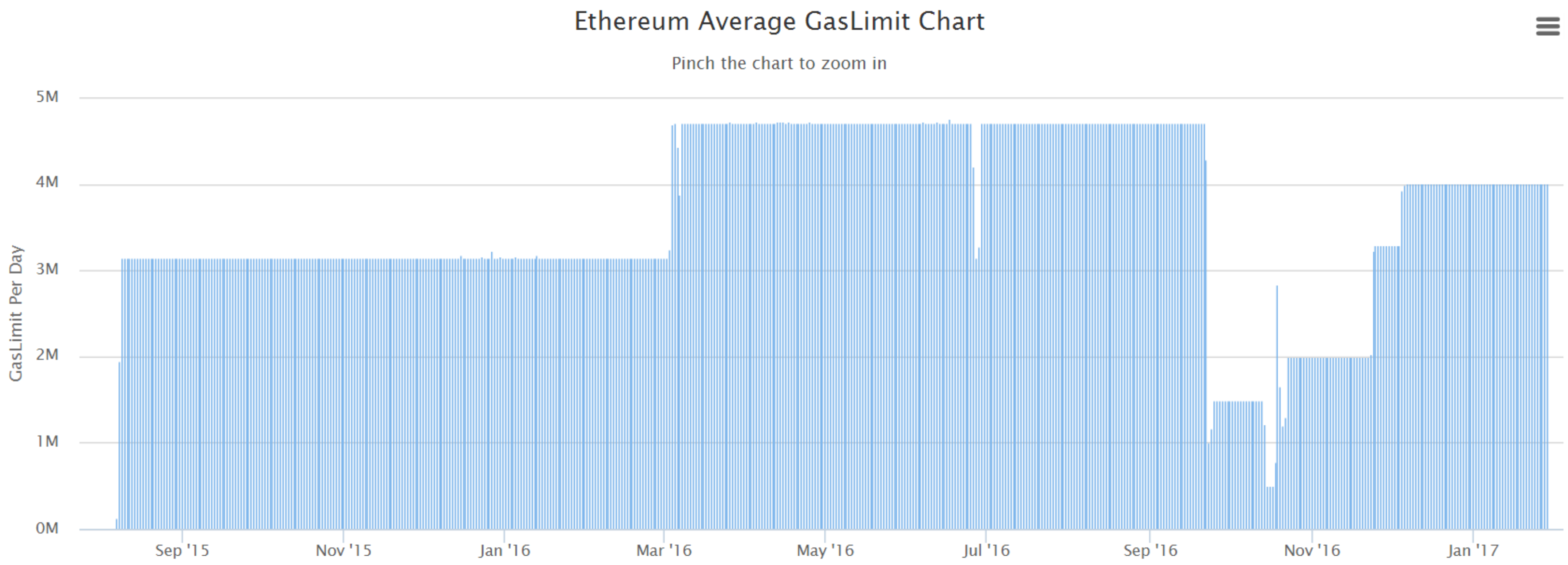
Availability

- Availability: the system‘s readiness for correct service
  - Blockchain as component: read: +++ ; write: ?
- Ethereum: 12 blocks confirmation is accepted, said to take 3 mins
  - Measurements:



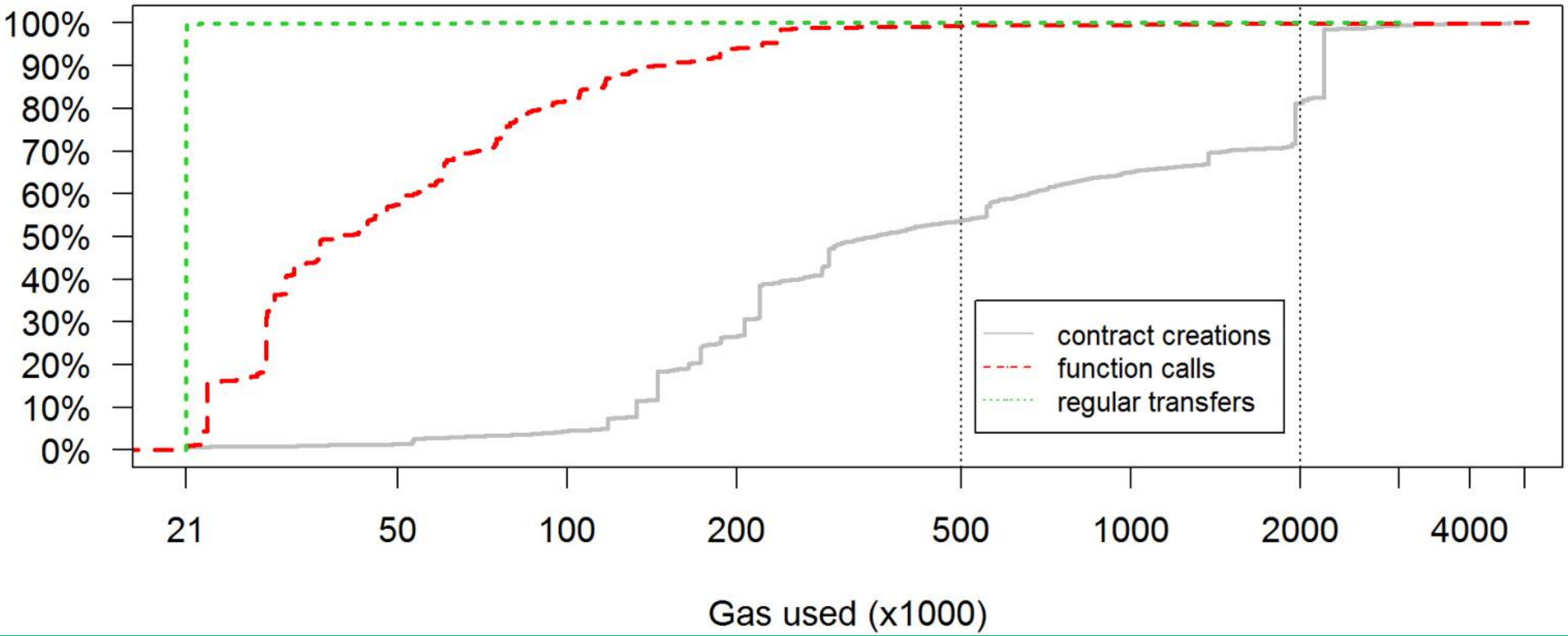
Availability

- Potential issue: block gas limit
  - Gas limit is set by miners through “voting”
  - The sum of Gas of all transactions in a block must be less than the limit
- Response to DDoS attack: lower block gas limit



Availability

- Potential issue: block gas limit
  - Gas limit is set by miners through “voting”
  - The sum of Gas of all transactions in a block must be less than the limit
- Response to DDoS attack: lower block gas limit
- Who would be affected by that?

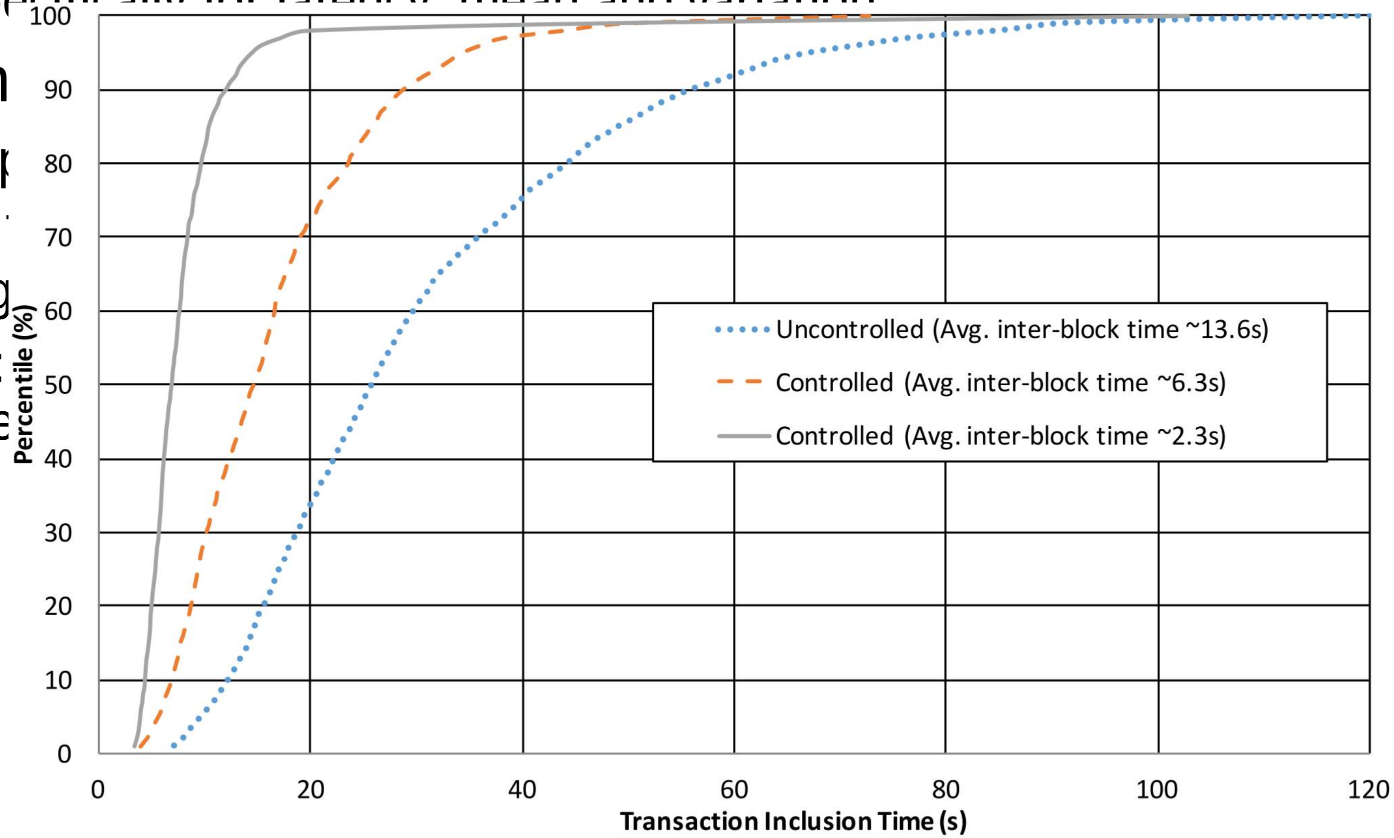


Latency simulation

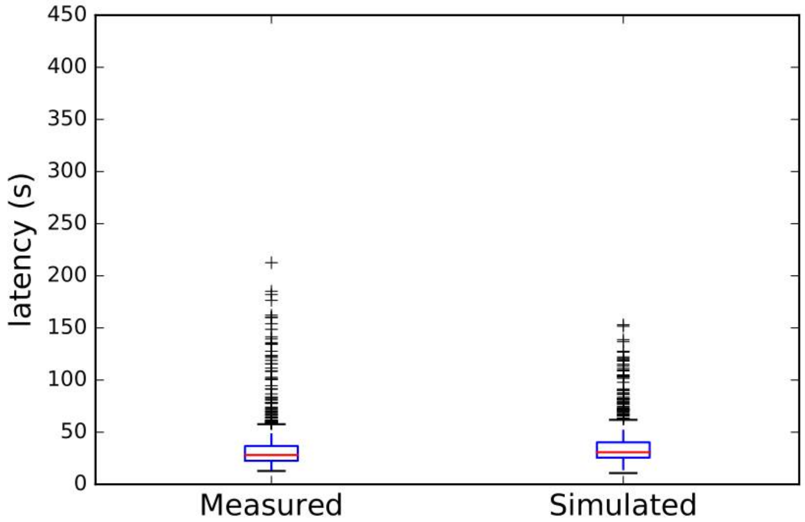
- Goal: predict latency for blockchain-based application before building it
  - Challenge specifically for latency: mean and variation
- Means: Architecture performance modeling
  - Paladio Component Models with individual latency distributions + connections + probability of branching
  - Allows changing the models for *What-If analysis*
  - For instance: change inter-block time on private blockchain – what does that mean for overall application latency?

Latency simulation

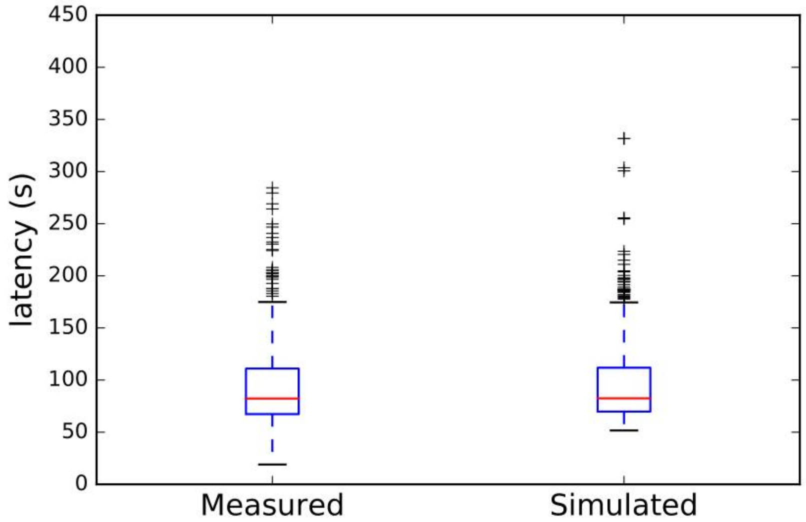
- Goal: predict latency for blockchain-based application before building it
  - Challenge specifically for latency: mean and variation
- Means: Arch
  - Paladio Comp
    connections
  - Allows chang
  - For instance:
    mean for ove



Latency: what if we change required number of confirmation blocks?

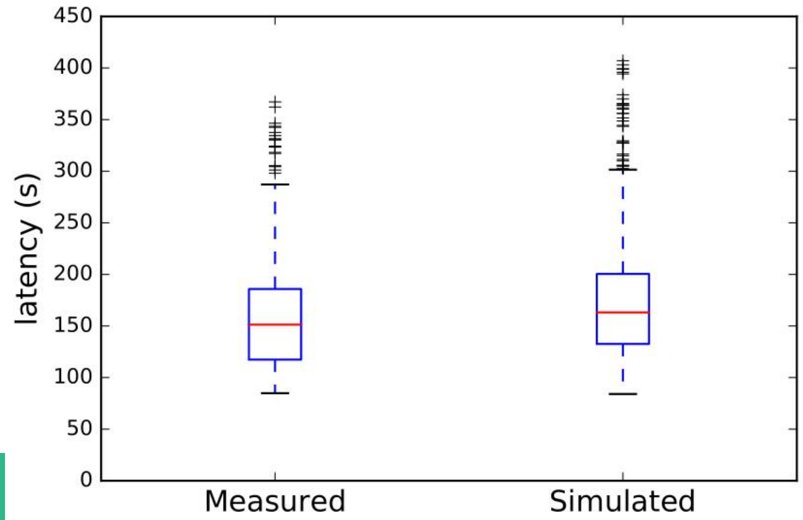


1 block



6 blocks

12 blocks



Using Smart Contracts for Business Process Monitoring and Execution

Ingo Weber, Sherry Xu, Regis Riveret, Guido Governatori, Alexander Ponomarev and Jan Mendling
Untrusted business process monitoring and execution using blockchain. BPM 2016

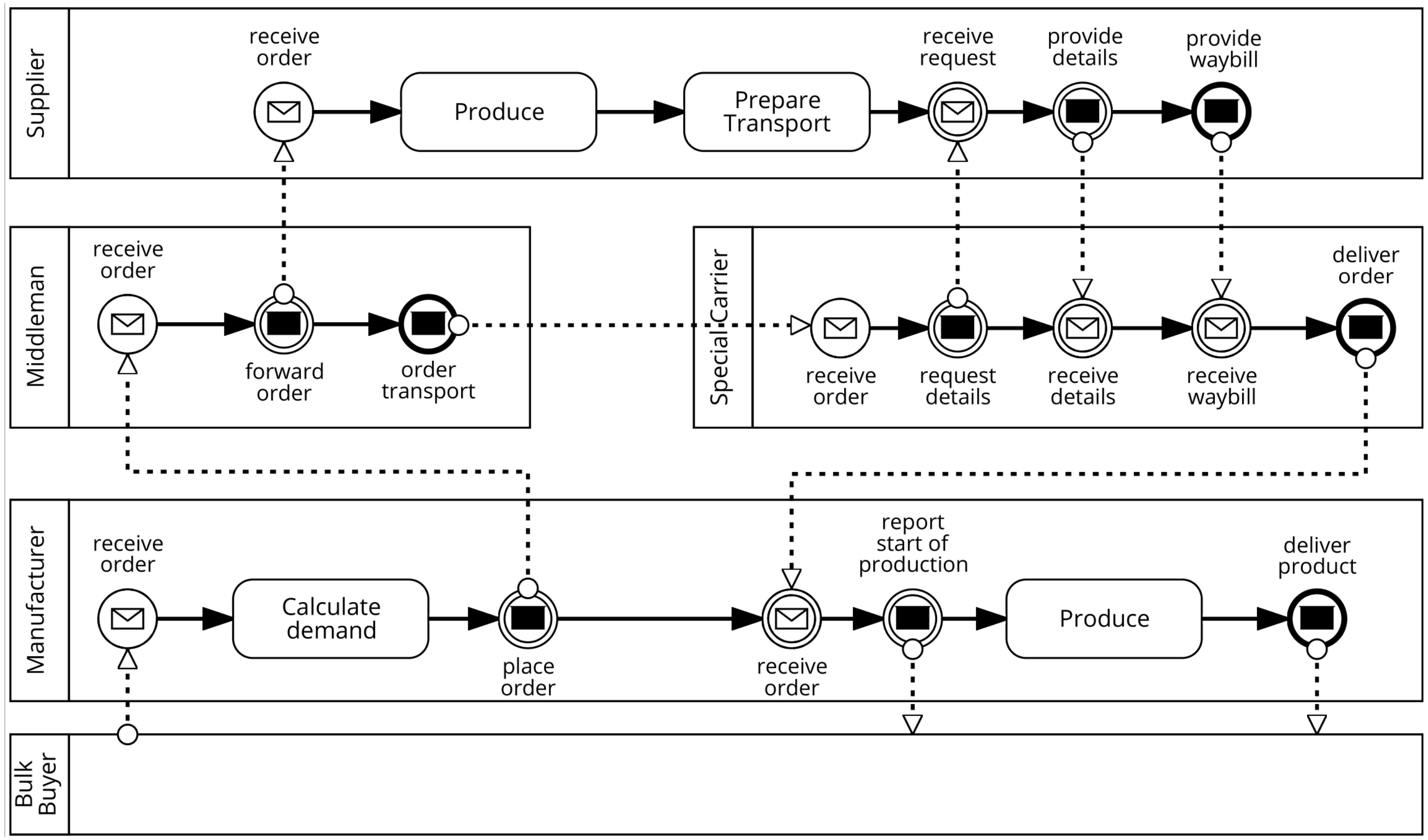
Luciano García-Bañuelos, Alexander Ponomarev, Marlon Dumas, and Ingo Weber
Optimized Execution of Business Processes on Blockchain. BPM 2017

Orlenys López-Pintado, Luciano García-Bañuelos, Marlon Dumas, and Ingo Weber.
Caterpillar: A blockchain-based business process management system. BPM Demo 2017

Motivation

- Integration of business processes across organizations: a key driver of productivity gains.
- Collaborative process execution
  - Doable when there is trust – supply chains can be tightly integrated
  - Problematic when involved organizations have a **lack of trust** in each other
    → if 3+ parties should collaborate, where to execute the process that ties them together?
  - Common situation in “coopetition”

Motivation: example



Issues:

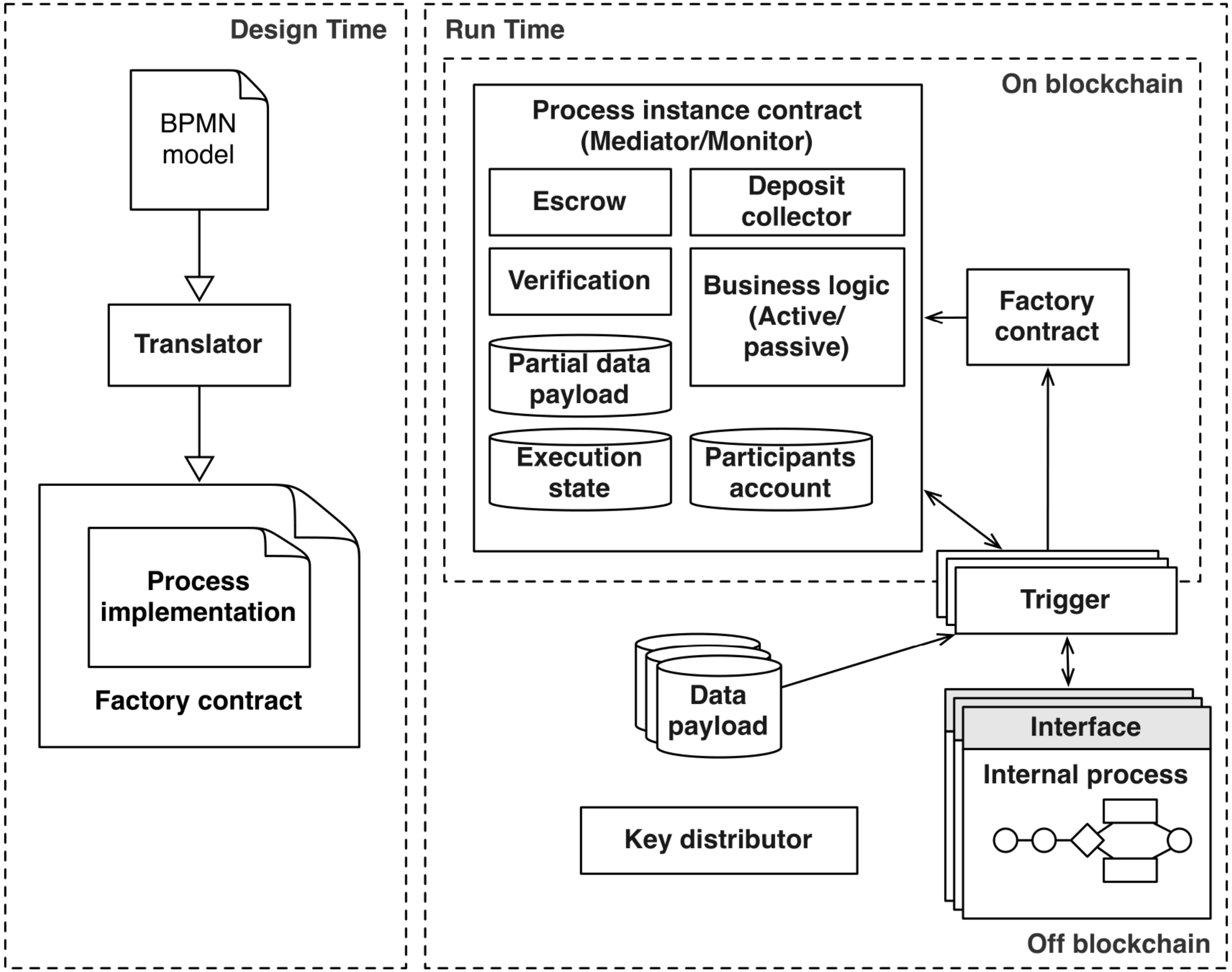
- Knowing the status, tracking correct execution
- Handling payments
- Resolving conflicts

→ ~~Trusted 3rd party~~ ?
→ Blockchain!

Approach in a nutshell

- Goal: execute collaborative business processes as smart contracts
  - Translate (enriched) BPMN to smart contract code
  - Triggers act as bridge between Enterprise world and blockchain
  - Smart contract does:
    - Independent, global process monitoring
    - Conformance checking: only expected messages are accepted, only from the respective role
    - Automatic payments & escrow
    - Data transformation
    - Encryption

Architecture

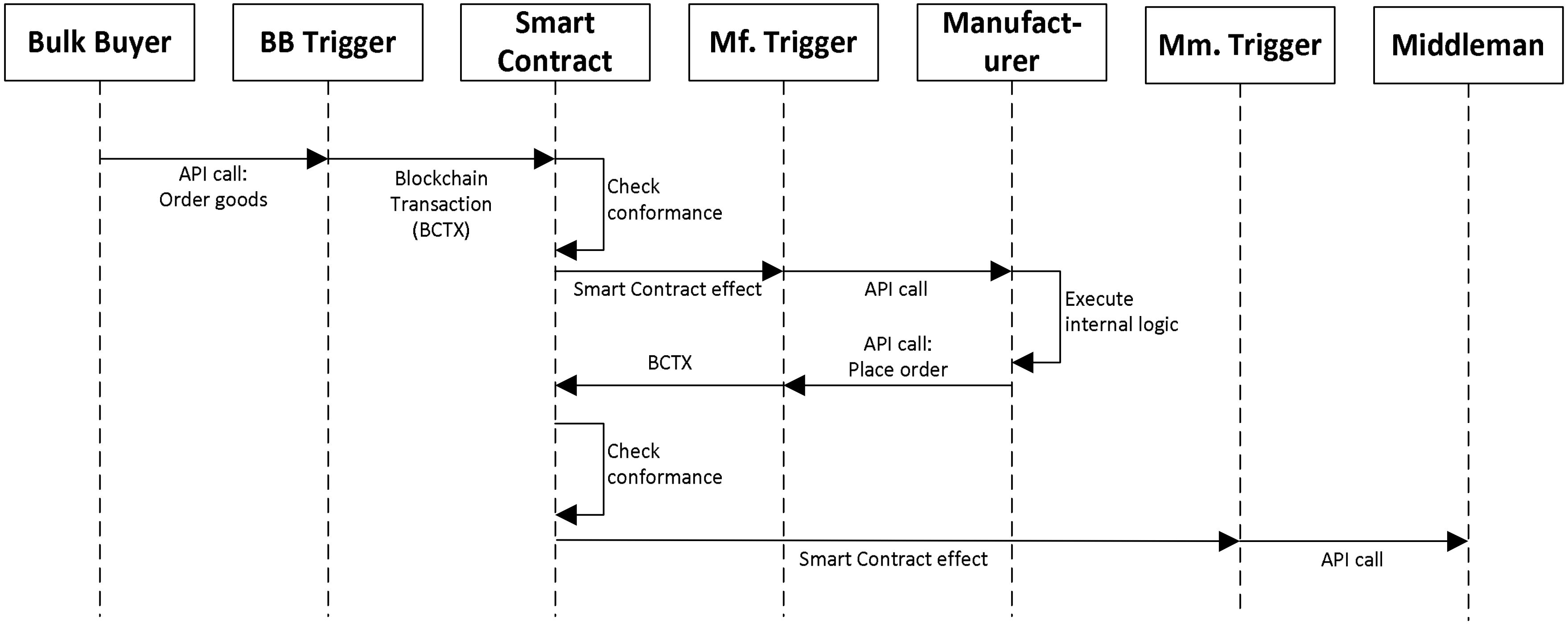


Runtime

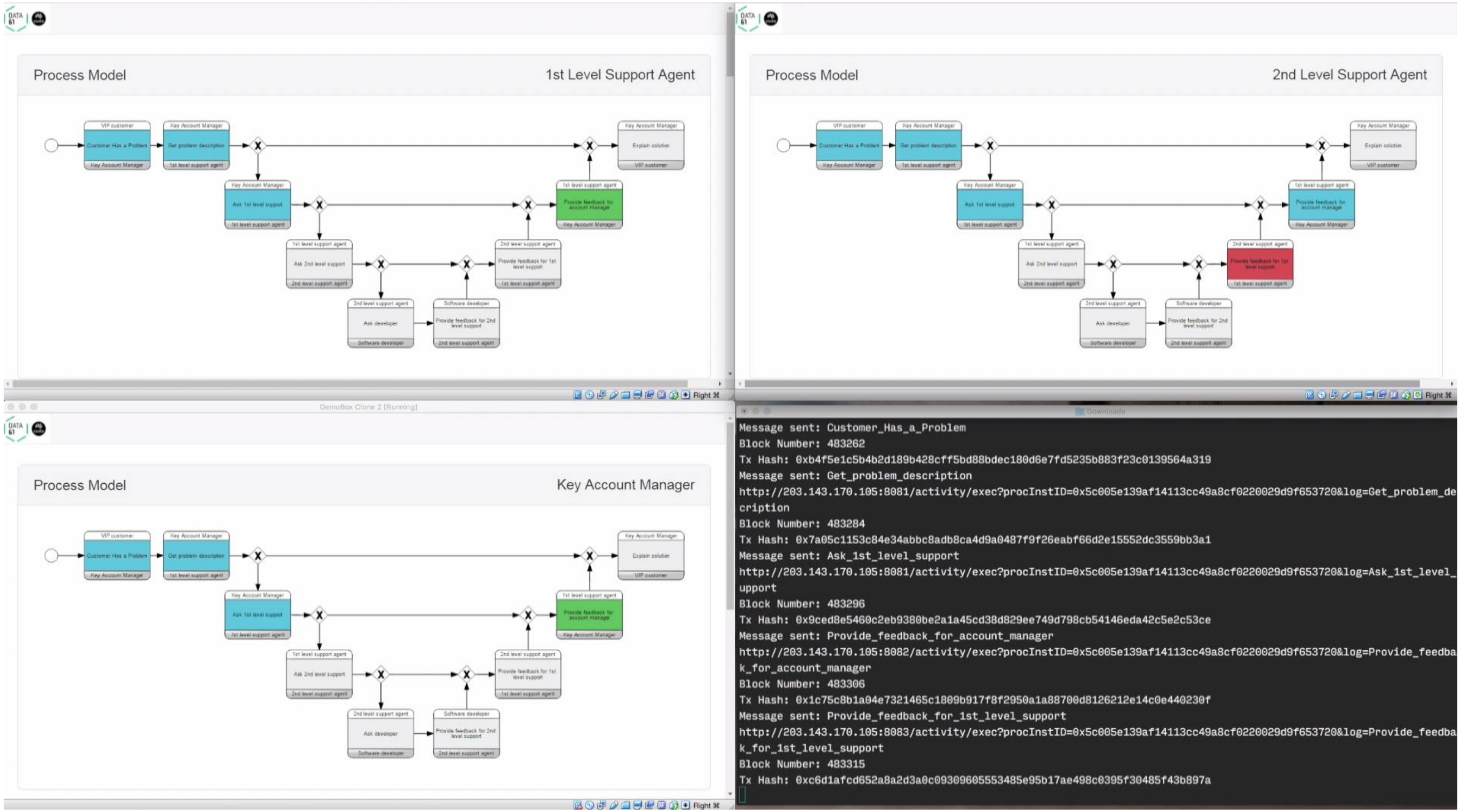
- Instantiation:
  - New *instance contract* per process instance
  - Assign accounts to roles during initialization
  - Exchange keys and create secret key for the instance
- Messaging:
  - Instead of sending direct WS calls: send through triggers & smart contract
  - Instance contract handles:
    - Global monitoring
    - Conformance checking
    - Automated payments*
    - Data transformation*

Runtime

- Instantiation:
  - New *instance contract* per process instance



Demonstration



Summary

- Architecting and developing applications on Blockchain is challenging
  - Our research:
    - Designing applications with Blockchain
      - Software Architecture methods for Blockchain-based applications
    - Empirical and formal research on Blockchains
    - How to use smart contracts
      - Model-driven development of smart contracts

- Using Blockchain for process monitoring and execution
  - Applicable in lack-of-trust settings for collaborative process execution
  - Our approach:
    - translate from process models to Solidity
    - use triggers to connect Blockchain and Enterprise systems
  - Evaluation results: latency and cost should be acceptable in many cases
  - Screencast video and more details – see papers

DATA 61

Thank you

Ingo Weber | Principal Research Scientist & Team Leader
ingo.weber@data61.csiro.au
Conj. Assoc. Professor, UNSW Australia | Adj. Assoc. Professor, Swinburne University

AAP team Blockchain research: https://research.csiro.au/data61/blockchain/

Blockchain reports: https://www.data61.csiro.au/blockchain/

www.csiro.au

CSIRO

References (1/2)

1. On availability for blockchain-based systems, Ingo Weber, Vincent Gramoli, Mark Staples, Alex Ponomarev, Ralph Holz, An Binh Tran, and Paul Rimba. In SRDS'17: IEEE International Symposium on Reliable Distributed Systems, Hong Kong, China, September 2017.
2. Quantifying the cost of distrust: Comparing blockchain and cloud services for business process execution. Paul Rimba, An Binh Tran, Ingo Weber, Mark Staples, Alexander Ponomarev, and Xiwei Xu. Scalable Computing and Communications (SCAC) journal, in print, accepted September 2017
3. Optimized execution of business processes on blockchain, Luciano García-Bañuelos, Alexander Ponomarev, Marlon Dumas, and Ingo Weber. 15th International Conference on Business Process Management (BPM'17), Barcelona, Spain, September 2017.
4. Orlenys López-Pintado, Luciano García-Bañuelos, Marlon Dumas, and Ingo Weber. Caterpillar: A blockchain-based business process management system. In BPM'17: International Conference on Business Process Management, Demo track, Barcelona, Spain, September 2017.
5. Risks and opportunities for systems using blockchain and smart contracts, Mark Staples, Shiping Chen, Sara Falamaki, Alex Ponomarev, Paul Rimba, An Binh Tran, Ingo Weber, Xiwei Xu, and Zhenjiang Zhu. Technical report, Data61, CSIRO, Sydney, Australia, June 2017.
6. Regerator: a Registry Generator for Blockchain, An Binh Tran, Xiwei Xu, Ingo Weber, Mark Staples and Paul Rimba. 29th International Conference on Advanced Information Systems Engineering (CAISE'17).
7. EthDrive: A Peer-to-Peer Data Storage with Provenance, Xiao Liang Yu, Xiwei Xu and Bin Liu. 29th International Conference on Advanced Information Systems Engineering (CAISE'17).
8. The Balance Attack or Why Forkable Blockchains are Ill-Suited for Consortium, Christopher Natoli and Vincent Gramoli. 2017 IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'17).
9. Blockchain Consensus, Tyler Crain, Vincent Gramoli, Michel Raynal, Mikel Larrea. Proceedings of AlgoTel 2017.

References (2/2)

10. A taxonomy of blockchain-based systems for architecture design, Xiwei Xu, Ingo Weber, Mark Staples, Liming Zhu, Jan Bosch, Len Bass, Cesare Pautasso and Paul Rimba. 2017 IEEE International Conference on Software Architecture (ICSA'17), Gothenburg, Sweden, April 2017.
11. Comparing blockchain and cloud services for business process execution, Paul Rimba, An Binh Tran, Ingo Weber, Mark Staples, Alexander Ponomarev and Xiwei Xu. Short paper, 2017 IEEE International Conference on Software Architecture (ICSA'17), Gothenburg, Sweden, April 2017.
12. Predicting latency of blockchain-based systems using architectural modelling and simulation, Rajitha Yasaweerasinghelage, Mark Staples and Ingo Weber. Short paper, 2017 IEEE International Conference on Software Architecture (ICSA'17), Gothenburg, Sweden, April 2017.
13. The Blockchain Anomaly, Christopher Natoli, Vincent Gramoli. Proceedings of the 15th IEEE International Symposium on Network Computing and Applications (NCA'16), IEEE Oct 2016
14. Evaluation of Logic-Based Smart Contracts for Blockchain Systems, Idelberg, Florian and Governatori, Guido and Riveret, Regis and Sartor, Giovanni. 10th International Web Rule Symposium, July, 2016
15. New kids on the block: an analysis of modern blockchains, Luke Anderson, Ralph Holz, Alexander Ponomarev, Paul Rimba, Ingo Weber. arXiv:1606:06530, 2016
16. On the Danger of Private Blockchains, Vincent Gramoli. Workshop on Distributed Cryptocurrencies and Consensus Ledgers (DCCL'16), 2016
17. Untrusted business process monitoring and execution using blockchain, Ingo Weber, Sherry Xu, Regis Riveret, Guido Governatori, Alexander Ponomarev and Jan Mendling. BPM 2016, Rio de Janeiro, Brazil , September, 2016
18. The blockchain as a software connector, Sherry Xu, Cesare Pautasso, Liming Zhu, Vincent Gramoli, Alexander Ponomarev, An Binh Tran and Shiping Chen. WICSA2016, Venice, Italy, April, 2016