Towards An Analysis of Network Partitioning Prevention for Distributed Ledgers and Blockchains

Hyperledger Fabric Architecture WG - December 2nd 2020

DB Systel GmbH | Dr. Michael Kuperberg | Blockchain and DLT Solutions | 2020-12-02

D_BLOCKCHAIN

DB Systel
Digital bewegen. Gemeinsam.

Disclaimer / Legal Notice

All statements in this slide deck are to be considered as non-binding information.
They reflect the opinion of the presenter and are in no way statements made on behalf of the Deutsche Bahn AG, DB Systel GmbH or any of the companies which are part of the DB holding.

Any mention of a product or of a trademark is not to be understood as a recommendation / endorsement / valuation or as a judgement. Trademarks are owned by the respective holders.

No guarantee is given that the provided information is correct, complete or up-to-date.
Distribution or copying of contents from this presentation (or parts thereof) is only allowed based on written permission from the author and copyright holder: Michael.Kuperberg@deutschebahn.com

Alle Darstellungen in diesem Foliensatz sind als unverbindliche Information zu verstehen. Sie geben lediglich die Meinung des Autors wieder und sind nicht als Aussage im Namen der Deutschen Bahn AG, der DB Systel GmbH oder eines Unternehmens der DB-Holding zu verstehen.

Die Erwähnung eines Produktes oder einer Handelsmarke ist nicht als Empfehlung oder als Bewertung zu verstehen. Die jeweiligen Trademarks gehören ihren Eigentümern.

Für Korrektheit, Aktualität und Vollständigkeit der Information wird keine Gewähr übernommen. Vervielfältigung oder Reproduktion von (Teil-)Inhalten oder vom Vortrag ausschließlich mit schriftlicher Genehmigung des Autors und Copyright-Inhabers: michael.kuperberg@deutschebahn.com.

Deutsche Bahn AG / DB Systel GmbH ⟷ Blockchains/DLTs

DB Systel GmbH:
- full-service IT provider with >1 billion € revenue
- 4400+ employees (Frankfurt, Berlin, Erfurt, UK)

Since 2016: pursuing the "Blockchains & Distributed Ledger Technologies" topic in a systematic way

Why do we deal with blockchains?
- Today's public transportation is a network of providers with many opportunities in backend integration
- BRCS: Blockchain-based Rail Control System
- SSI: Self-Sovereign Identity

Problem Statement and Research Goals

Recall the CAP theorem: it is not possible to achieve more than two out of three following qualities at the same time: Consistency, Availability and Tolerance of Network Partitioning.

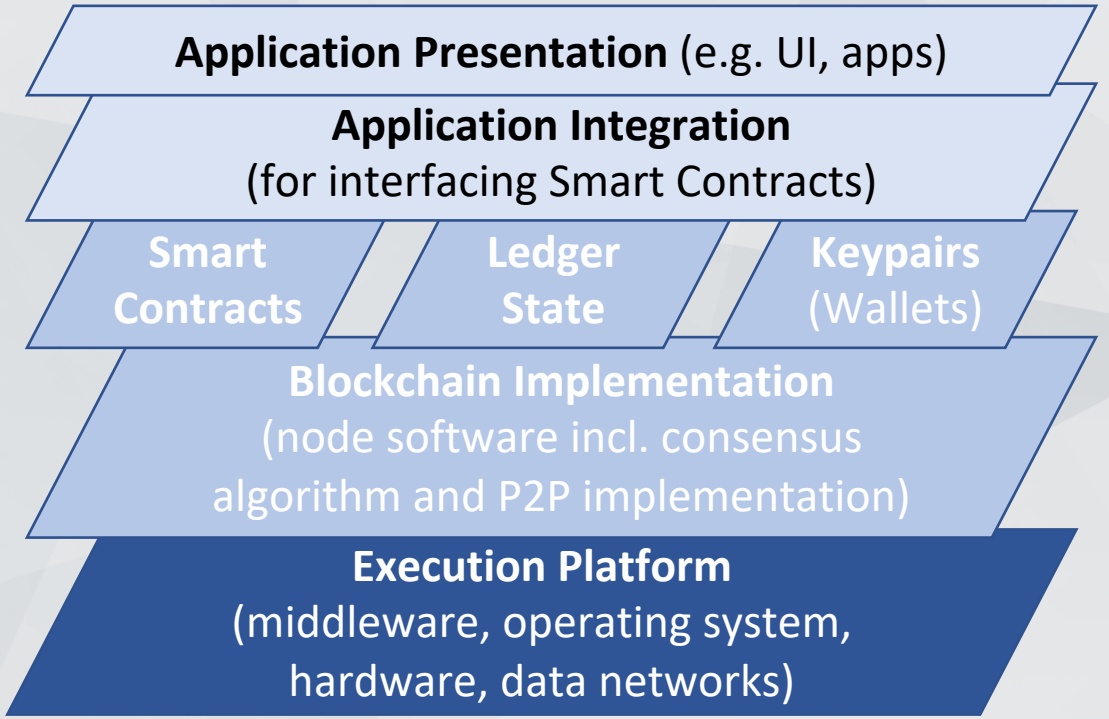
Network partitioning: two (or more) groups of nodes become fully isolated from each other
→ each group can see itself as a complete, autonomous and fully-functioning network
→ each group establishes its own consensus about data state - without being aware of the other groups
From a global perspective, this leads to **"multiple truths"** → high chance of conflicts and contradictions

- How can we leverage consensus protocols such as Proof-of-Authority, Proof-of-Stake and Proof-of-Work to **detect network partitioning situations** in enterprise-grade blockchains?
- Design recommendations for **partitioning avoidance in Proof-of-Authority** in Hyperledger Fabric
- See paper for analysis of related work

Why do it on consensus algorithm level?

- We cannot rely exclusively on the the execution platform layers (especially networking) to detect and to handle network partitioning
- **Blockchain platform implementation must be robust enough to deliver its quality promises**
- At this point, we do not yet consider all relevant algorithms - RAFT, Paxos, Tendermint, “Proof of Elapsed Time” (PoET), “Proof of Space”, “Pure Proof of Stake” etc. are left for future work

Layers of a blockchain application incl. the runtime platform



Proof-of-Work

- **At consensus level, PoW (e.g. in Ethereum) does not detect partitioning and cannot prevent it**
  - especially when combined with unpermissioned network design and probabilistic/gossip-based propagation
  - there is no consensus-level mechanism to ensure that a block reaches all active network nodes, or even a given subset thereof
  - game theory is inherently probabilistic

- Ethereum: the yellowpaper does not impose any checks to ensure freedom of data conflicts across forks
  - when a network splits, role of forks (“trunk” vs. “branch”) can develop differently

- The arguments are apply permissioned PoW implementations, unless these introduce mechanisms to maintain integrity and sufficient completeness of the network

- **Detection and prevention of network partitioning when using PoW must be performed on network level rather than on consensus level**

PoA in Hyperledger Fabric 1.4

- Multi-stage PoA: endorsements+ordering (see paper)
- Our approach start with two partitions p_1 and p_2
- 12 scenarios are possible (+permutations), see table

- The bisection scenario in case M can be avoided by preferring consistency over liveliness and imposing (for the bisection case with p = 2) the constraint

$2 \times m > a$

upon the endorsement policy of HLF PoA

- Formula also holds for a general partitioning scenario with p≥2, since the above constraint prohibits > 1 concurrent consensus findings no matter how many partitions exist

Case	Endorser nodes in p_1	Endorser nodes in p_2	Orderer nodes in p_1	Orderer nodes in p_2
A	$e_{p_1} = 0$ (none)	$e_{p_2} = a$ (all)	0	> 0
B	$e_{p_1} = 0$ (none)	$e_{p_2} = a$ (all)	> 0	0
C	$e_{p_1} = 0$ (none)	$e_{p_2} = a$ (all)	> 0	> 0
D	$0 < e_{p_1} < m$	$0 < e_{p_2} < m$	0	> 0
E	$0 < e_{p_1} < m$	$0 < e_{p_2} < m$	> 0	0
F	$0 < e_{p_1} < m$	$0 < e_{p_2} < m$	> 0	> 0
G	$m \leq e_{p_1} < a$	$0 < e_{p_2} < m$	0	> 0
H	$m \leq e_{p_1} < a$	$0 < e_{p_2} < m$	> 0	0
J	$m \leq e_{p_1} < a$	$0 < e_{p_2} < m$	> 0	> 0
K	$m \leq e_{p_1} < a$	$m \leq e_{p_2} < a$	0	> 0
L	$m \leq e_{p_1} < a$	$m \leq e_{p_2} < a$	> 0	0
M	$m \leq e_{p_1} < a$	$m \leq e_{p_2} < a$	> 0	> 0

12 constellations for PoA network partitioning with *a* authorities, “at least *m* out of *a*” endorsing policy with *m* < *a*, exactly one endorsing node for each authority and two partitioning (*p* = 2) so that $e_{p2} = (a - e_{p1})$

Aura Proof-of-Authority in Parity Ethereum

- The Parity-provided PoA is based on the Aura (authority round) algorithm, i.e. *round robin*
- Limited to private or consortial networks; no concept of „organizations“ or „endorsements“

- We found no evidence on any testimonials of Aura’s behavior during network partitioning

- According to our analysis, in two independent (disconnected) network partition, two “validator rings” can emerge and thus, **concurrent truths can be established**

- Aura might detect that a certain validator is missing, but it cannot decide whether that validator is absent (dormant/shut down, which is a permitted situation), or whether it is active in a different partition
  → Aura PoA within Parity Ethereum appears **unable to detect network partitioning**

- Aura cannot be configured to follow a policy such as “every validator must see *m* out of *a* validators” (so that one could impose $2 \times m > a$ as above) → **it cannot be configured to resist network partitioning**

Conclusions

Initial foundations for partitioning analysis in the context of blockchains/DLT; role of consensus algorithms in this

- For Hyperledger Fabric, we have derived a formula to configure Proof-of-Authority consensus to avoid the creation of concurrent truths during accidental network partitioning
- We strive to devise executable and repeatable test procedures for live analysis of network partitioning scenarios, building on existing peer-to-peer network research
- We plan to extend our analysis to the situations with malicious partitioning,
  e.g. where a “Byzantine general” generates two conflicting blocks in two partitions
- We also plan to analyze specific enterprise-grade products (such as Hashgraph, Quorum, R3 Corda and others) and to perform deeper research for further consensus algorithms (such as RAFT, Paxos and others)

Thank you!

Work Email: michael.kuperberg@deutschebahn.com

Twitter of working group: https://twitter.com/DB_Blockchain

Homepage of working group: https://blockchain.deutschebahn.com

LinkedIn:
https://www.linkedin.com/in/michael-kuperberg-8a612625/

Homepage (incl. publication list & talk announcements):
https://www.michaelkuperberg.com

Dr. Michael Kuperberg
Chief Blockchain Architect

Blockchain and DLT Solutions

Michael.Kuperberg@deutschebahn.com
DB Systel GmbH
Jürgen-Ponto-Platz 1
60329 Frankfurt am Main, Germany
https://www.dbsystel.de

Related Work (1)

- Fischer et al. (1982) showed for the “all must consense” situation that involves an asynchronous system of processes that “every protocol for this problem has the possibility of nontermination, even with only one faulty process”.
- Ekparinya et al. (2019) describe an attack (based on identity/node “clones”) against Proof-of-Authority (PoA) for Ethereum. They also propose countermeasures to ensure that the network can remain live and safe.
- Singhal and Masih (2019) discuss different methods for scaling blockchain technology for automotive uses.
- Ekparinya et al. (2018) demonstrate attacks on Ethereum using network partitioning, but do not discuss how to protect against them.
- Dalui et al. (2009) deliberately introduce network partitioning to reduce the message exchange overhead - however, does not offer a solution for dealing with undesirable network partitioning.
- Buntinas (2012) describes consensus for highly parallel applications using MPI (Message Passing Interface) architectures, but does not consider network partitioning.
- Network partitioning for blockchains has been studied by Saito and Yamada (2016), but they do not
- discuss Proof-of-Authority and do not provide suggestions on choosing a partitioning-resistant consensus.
- Anjum et al. (2017) list IPFS as “highly resilient against network partitioning”, but do not consider this characteristic for the other analyzed technologies (which include Ethereum, Hashgraph etc.).

Related Work (2)

- Wang et al. (2019) highlight the role of network partitioning for IoT-oriented blockchains and discuss the connection monopolizing Eclipse attacks, but do not discuss how consensus protocols or specific blockchain implementations are affected.
- Hu et al. (2019) discuss the threat of network partitioning as part of the security analysis for a delaytolerant payment scheme based on the Ethereum blockchain.
- Chan et al. (2018) propose a blockchain design that is built around “pipelined BFT”.
- Homoliak et al. (2019) introduce a security reference architecture for blockchains, and include network partitioning into the list of threats.
- Keshav et al. (2018) present a resilient consensus algorithm called RCanopus that “guarantees safety even in the presence of Byzantine attacks and network partitioning”.
- Li et al. (2017) deal with securing PoS blockchain protocols (explain the risks of network partitioning, but do not provide a solution).
- Yu et al. (2018) propose a novel consensus protocol (explain the risks of network partitioning, but do not provide a solution).
- P. Mahlmann and C. Schindelhauer (2005) as well as Pallavi and Prakash (2015) study partitioning prevention on the network level (independently from consensus)

References

1. M. J. Fischer, N. A. Lynch, and M. S. Paterson, “Impossibility of distributed consensus with one faulty process.” Massachusetts Inst. Of Techn., Cambridge lab for Computer Science, Tech. Rep., 1982.
2. P. Ekparinya, V. Gramoli, and G. Jourjon, “The Attack of the Clones against Proof-of-Authority,” September 2019. [Online]. Available: http://arxiv.org/abs/1902.10244
3. P. Singhal and S. Masih, “MetaAnalysis of Methods for Scaling Blockchain Technology for Automotive Uses,” 2019. [Online]. Available: http://arxiv.org/abs/1907.02602
4. P. Ekparinya, V. Gramoli, and G. Jourjon, “Double-Spending Risk Quantification in Private, Consortium and Public Ethereum Blockchains,” May 2018. [Online]. Available: https://arxiv.org/abs/1805.05004
5. M. Dalui, B. Chakraborty, and B. K. Sikdar, “Quick Consensus Through Early Disposal of Faulty Processes,” in 2009 IEEE International Conference on Systems, Man and Cybernetics, Oct 2009, pp. 1989–1994.
6. D. Buntinas, “Scalable Distributed Consensus to Support MPI Fault Tolerance,” in 2012 IEEE 26th International Parallel and Distributed Processing Symposium, May 2012, pp. 1240–1249.
7. K. Saito and H. Yamada, “What’s So Different about Blockchain? — Blockchain is a Probabilistic State Machine,” in ICDCSW 2016, June 2016, pp. 168–175.
8. A. Anjum, M. Sporny, and A. Sill, “Blockchain Standards for Compliance and Trust,” IEEE Cloud Computing, vol. 4, no. 4, pp. 84–90, July 2017.
9. X. Wang, X. Zha, W. Ni, R. P. Liu, Y. J. Guo, X. Niu, and K. Zheng, “Survey on blockchain for Internet of Things,” Computer
10. Communications, vol. 136, pp. 10 – 29, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366418306881
11. Y. Hu, A. Manzoor, P. Ekparinya, M. Liyanage, K. Thilakarathna, G. Jourjon, and A. Seneviratne, “A Delay-Tolerant Payment Scheme
12. Based on the Ethereum Blockchain,” IEEE Access, vol. 7, pp. 33 159– 33 172, 2019.
13. T.-H. H. Chan, R. Pass, and E. Shi, “PaLa: A Simple Partially Synchronous Blockchain,” IACR Cryptology ePrint Archive, vol. 2018, p. 981, 2018. [Online]. Available: https://eprint.iacr.org/2018/981.pdf
14. I. Homoliak, S. Venugopalan, Q. Hum, and P. Szalachowski, “A Security Reference Architecture for Blockchains,” April 2019. [Online]. Available: https://arxiv.org/abs/1904.06898
15. S. Keshav, W. M. Golab, B. Wong, S. Rizvi, and S. Gorbunov, “RCanopus: Making Canopus Resilient to Failures and Byzantine Faults,” October 2018. [Online]. Available: http://arxiv.org/abs/1810.09300
16. W. Li, S. Andreina, J.-M. Bohli, and G. Karame, “Securing Proof-of-Stake Blockchain Protocols,” in DPM 2017, Sept. 2017, pp. 297–315.
17. H. Yu, I. Nikolic, R. Hou, and P. Saxena, “OHIE: Blockchain Scaling Made Simple,” Nov. 2018. [Online]. Available: http://arxiv.org/abs/1811.12628
18. P. Mahlmann and C. Schindelhauer, “Peer-to-Peer Networks based on Random Transformations of Connected Regular Undirected Graphs,” in SPAA 2005, pp. 155–164.
19. R. Pallavi and G. C. B. Prakash, “A review on network partitioning in wireless sensor and actor networks,” in iCATccT 2015, Oct 2015.