

APRIL 2019

FSBC Working Paper

# Entscheidungshilfe für den Einsatz von Blockchain-Technologien in Unternehmen: Vier Frameworks im Vergleich

Daniel Höfelmann, Philipp Sandner

**Mit diesem Paper sollen Entscheider bei der Auswahl einer passenden Blockchain bzw. Distributed Ledger Technology (DLT) unterstützt werden. Hierzu wurden Ethereum, Hyperledger Fabric, R3 Corda und Quasar/Stellar für den Einsatz in einem zugangsbeschränkten („permissioned“) Szenario untersucht und nach verschiedenen Punkten wie z.B. Performance-Werten, Kosteneffizienz und Sicherheitskriterien bewertet.**

## Zielsetzung

Die Begriffe Distributed Ledger Technology (DLT) und Blockchain werden oftmals synonym verwendet. Um Unternehmen bei der Auswahl einer passenden Technologie zu unterstützen, wurden vier verschiedene DLT für den Einsatz in einem zugangsbeschränkten („permissioned“) Szenario untersucht. Die vier untersuchten DLTs sind Corda, Hyperledger Fabric, Ethereum sowie die weniger populäre Lösung Quasar. Da Quasar auf unterschiedlichen Infrastrukturen aufsetzen kann, wurde für diese Betrachtung die Kombination mit Stellar gewählt.

Frankfurt School Blockchain Center  
[www.fs-blockchain.de](http://www.fs-blockchain.de)  
[contact@fs-blockchain.de](mailto:contact@fs-blockchain.de)

Follow us  
[www.twitter.com/fsblockchain](https://www.twitter.com/fsblockchain)  
[www.facebook.de/fsblockchain](https://www.facebook.de/fsblockchain)

Frankfurt School of  
Finance & Management gGmbH  
Adickesallee 32-34  
60322 Frankfurt am Main  
Germany

Ziel ist es, Entscheidungsträger und Projektleiter bei der Auswahl einer geeigneten Distributed Ledger Technology zu unterstützen, indem Vor- und Nachteile der verschiedenen Systeme veranschaulicht werden. Bewertet wurden diese nachfolgenden Kriterien: Benutzerfreundlichkeit der Installation, Effizienz und Performance, Kosteneffizienz, Release-Fähigkeit und Aktualität, Sicherheit sowie Verwaltung.

Die hier dargestellten Interpretationen, Analysen und Schlussfolgerungen spiegeln die Meinung der Autoren wider.

## **Begrifflichkeiten und Rahmenbedingungen**

Der Begriff DLT bezeichnet in diesem Paper alle Systeme, die theoretisch vielen oder allen wirtschaftlichen Akteuren eine gleichzeitige und vollständige Einsicht in Transaktionen aller Art ermöglichen. Nach dieser Definition ist die Blockchain nur eine Form der DLT, was neben Ethereum die Berücksichtigung weiterer DLTs wie z.B. Hyperledger Fabric, R3 Corda oder Quasar/Stellar ermöglicht.

In diesem Paper wird davon ausgegangen, dass der Einsatz der geprüften DLTs in einem permissioned Network erfolgt, d.h. in einem Umfeld, in dem der Zugang zu dem System für einige oder fast alle Akteure beschränkt bzw. beschränkbar ist.

Wird der Zugang eines Systems beschränkt und unterliegt der Kontrolle Einzelner bzw. einer beschränkten Anzahl von Akteuren, kann nicht per se von dem Erhalt der Attribute einer öffentlichen Blockchain ausgegangen werden (insbesondere seien hier die Unveränderlichkeit der Historie und die hohe Sicherheit gegen viele Angriffsvektoren genannt).

Diese Attribute sind zumindest in der Anlage der Blockchain stark mit der Verteilung des Systems einander nicht vertrauender wirtschaftlicher Akteure ausgelegt, die alle dem System autark beitreten können und so nicht kontrolliert werden („permissionless“).

Bei permissioned Systemen ist daher je nach Konfiguration des Systems zu prüfen, welche der Attribute ggf. entfallen oder gefährdet sein können, sowie

ob nicht eine Architektur wie eine herkömmliche Server-Client-Beziehung wirtschaftlich sinnvoller ist.

Andererseits gilt, dass die Beschränkungen von DLT-Systemen auch nützlich sein können: Vertrauen sich etwa Tochtergesellschaften nicht, können sie über die Nutzung einer DLT eine gemeinsame Vertrauensbasis herstellen. Auch im Supply Chain Management kann eine solche Lösung Vorteile bieten. NGOs, öffentliche Verwaltung und andere Institutionen könnten zulassungsbeschränkte Systeme sinnhaft und zum Vorteil aller einsetzen. Es kommt auf die konkrete Ausgestaltung der Beschränkungen, Kontroll- und vor allem Steuerungs-Mechanismen an.

## Distributed Ledger Technologies im Vergleich

**Ethereum – Die Allround-Blockchain.** Ethereum wurde als zulassungsfreie, öffentliche Blockchain entwickelt, in der jeder „Smart Contract“ in Verbindung mit dezentralen Anwendungen (dApps) programmiert werden kann. Hierfür wird eine virtuelle Maschine (VM) auf der Blockchain bereitgestellt, für deren Nutzung, je nach benötigtem Aufwand für die Ausführung des Programmiercodes, eine Gebühr bezahlt werden muss.

Die meist genutzte Programmiersprache für Ethereum ist Solidity. Mit ihr können Smart Contracts geschrieben und auf der Ethereum-Blockchain ausgeführt werden. Die Programmiersprache basiert auf C++, Python und Java Script, und ist auf der Ethereum Virtual Machine (EVM) implementiert. Der große Vorteil von Solidity liegt in ihrer Verständlichkeit und einfachen Umsetzung.

Es besteht zudem auch eine Go-Implementierung zum Schreiben von Ethereum Smart Contracts, jedoch sind in Go geschriebene Programme nicht direkt auf der EVM implementierbar. Programmierer müssen sogenannte Compiler schreiben, um Smart Contracts von Go in EVM-Bytecode umwandeln zu können.

In der offenen und zulassungsfreien Version von Ethereum kann jeder Entwickler eine Anwendung erstellen und mit der Blockchain interagieren.

Ebenso können Benutzer von dApps auf die Blockchain zugreifen. Ein weiterer Akteur sind die Miner. Diese erstellen Blöcke, in denen Benutzertransaktionen zusammengefasst und mittels einer Prüfsumme (Hash) abgesichert werden. Die Node-Operatoren validieren diese Blöcke und bestätigen sie.

### **Hyperledger Fabric – Das Framework der Linux Foundation.**

Hyperledger Fabric ist ein Blockchain-Framework der Linux Foundation. Insgesamt sind im Projekt „Hyperledger“ etliche DLT Infrastrukturen und Projekte von der Linux Foundation zusammengefasst worden.

Hyperledger Fabric, im Folgenden nur als Fabric bezeichnet, ist ein zugangsbeschränktes Ledger. Die Architektur von Fabric basiert auf mehreren Ledgern, deren Operationen voneinander unabhängig sind. Dennoch gibt es ein Adressierungssystem, das es einer Transaktion eines Ledgers erlaubt, die Transaktionen und Smart Contracts eines anderen Ledgers zu sehen und zu adressieren. Fabric bietet eine erweiterbare und modulare Architektur, die für verschiedene Bereiche eingesetzt werden kann und somit unabhängig von einem bestimmten Anwendungsbereich ist.

Programme in Fabric wurden ursprünglich als Chaincode bezeichnet, heute ist auch hier der Begriff Smart Contract gebräuchlich. Fabric unterstützt das Schreiben von Chaincodes in Go und Java. Die Chaincodes werden schließlich in einem Docker-Container ausgeführt.

Die Zugangsbeschränkung von Fabric sowie die validierenden und nicht-validierenden Nodes werden durch den oder die Betreiber des Netzwerkes festgelegt. Je nach Bedarf kann der Betreiber des Netzwerkes unterschiedliche Zugangsrechte an Nutzer verteilen, um die erforderlichen Transaktionen im Rahmen des Netzwerkes durchführen zu können.

Der zugangsbeschränkte Charakter von Fabric ergibt sich aus dem Bedürfnis der Nutzer nach Privatsphäre. Die Privatsphäre gilt allerdings nicht gegenüber Regulierungsbehörden, diese besitzen die Möglichkeit zur Identifizierung und Überprüfung. Die Verschlüsselung z.B. der Identität erfolgt daher so, dass sie vor anderen unerwünschten Teilnehmern verborgen bleibt, aber von z.B. Regulatoren eingesehen werden kann.

Fabric benötigt vor allen Transaktionen ein kryptografisches Zertifikat, das die vertraulichen Daten eines Benutzers beinhaltet und im Netzwerk registriert ist. Aus jeder Identität kann das Protokoll Sicherheitsschlüssel generieren, mit denen die Mitglieder im Netzwerk Transaktionen durchführen können. Die Identitäten der Transaktionspartner sind verborgen, um die Privatsphäre innerhalb des Netzwerks zu sichern.

Innerhalb von Fabric entsteht die Vertraulichkeit der Inhalte dadurch, dass Transaktionen so verschlüsselt werden, dass nur die Beteiligten sie entschlüsseln und ausführen können. Fabric bietet mit „Channels“ eine Lösung an: Bestimmte Benutzer kommunizieren in einem Sub-Netzwerk, auf das nur berechnete Benutzer Zugriff haben. Darüber hinaus kann eine (durch einen oder mehrere Smart Contracts realisierte) Geschäftslogik auch kryptografisch gesichert werden (wenn die Vertraulichkeit von ihren Stakeholdern verlangt wird), sodass sie erst zu einem bestimmten Zeitpunkt geladen und entschlüsselt wird.

**R3 Corda – Die „Blockchain“ für die Finanzindustrie.** Corda ist ein globales logisches Ledger von R3, in dem alle teilnehmenden Wirtschaftsakteure miteinander interagieren. Es ermöglicht den Parteien, Vereinbarungen untereinander sicher, konsistent, zuverlässig, privat und verbindlich zu erfassen und zu verwalten. Das Wort global im globalen logischen Ledger bedeutet, dass jeder Teilnehmer nur die ihn betreffenden Daten sieht. Der logische Teil bezieht sich auf die physische Implementierung, die unterschiedlich zusammengesetzt sein kann. Im Gegensatz zu Fabric und Ethereum wurde Corda ausschließlich für die Finanzindustrie entwickelt.

Der Code in Corda wird mit Kotlin geschrieben, einer Programmiersprache von JetBrains, die auf JVM und Javascript basiert. Kotlin hat einen hohen Integrationsgrad, wodurch die JVM jedes verwandte Programmierparadigma verwenden kann.

Die Erfassung und Abwicklung von Finanzvereinbarungen beinhalten drei große Punkte: Erstens werden die vom System verwalteten Aufzeichnungen nur denjenigen Akteuren zugänglich gemacht, die ein legitimes Interesse an den von ihnen verwalteten Vermögenswerten und Vereinbarungen haben – dies ähnelt der Channel-Lösung von Fabric. Zweitens wird die Ausführung

von Vereinbarungen, die vom System verwaltet werden, in einem Computercode beschrieben, der explizit auf eine übergreifende Rechtsprosa verweist und die Legitimität der Vereinbarung sichert. Und letztlich müssen Teile des Systems offen sein (Open Source, offene Entwicklungsprozesse und offene technologische Industriestandards), um eine breite Akzeptanz in der Finanzwelt zu erreichen.

Die Modularität und Interoperabilität von Corda ermöglicht es außerdem Unternehmen, bereits bestehende Systeme, wie z.B. Datenbanken, in das Corda-Netzwerk zu integrieren.

**Quasar/Stellar – Das konsortiale Blockchain-Cash-System.** Quasar ist ein zugangsbeschränktes, DLT-basiertes, elektronisches Kassensystem mit integrierten Regeln. Diese Regeln dienen der Erfüllung von gesetzlichen und regulierungsrelevanten Richtlinien. Quasar ermöglicht sofortige und irreversible digitale Bezahlung zwischen Unternehmen, Personen und Geräten im Internet of Things. Quasar basiert auf dem „Mehrzweck-Wallet-Ausgabemodell“ der Firma Quantoz und kann für viele Anwendungsfälle eingesetzt werden, wie z.B. die Erweiterung von Legacy-Systemen für Finanzdienstleistungen.

Der Code in Quasar wird mit C++ geschrieben. Die Verwendung offener Standards und ein verteiltes System von Nodes, an die sich alle Geräte frei anschließen können, ermöglicht es jedem Entwickler neue Zahlungsanwendungen in Quasar zu entwickeln.

Die Integration von Wallets, Tools, Geräten und Dienstleistungen von Drittanbietern führt zu einem schnellen Akzeptanzanstieg des bei Entwicklern und Anwendern. Entwicklungen von Drittanbietern werden durch offene APIs eingebunden und bieten Systembetreibern einen schnellen Weg zur Entwicklung neuer Produkte.

Quasar ist als Netzwerk von Servern (Nodes) an mehreren Standorten konzipiert, die Stellar als ein verteiltes Ledger betreiben. Dieses Ledger zeichnet jeden Vorgang im System auf. „White-listed Entities“ (z.B. teilnehmende Banken) können Nodes betreiben. Diese Nodes kommunizieren miteinander, um Transaktionen zu verifizieren und das

Ledger zu synchronisieren. Das Ledger erfasst Geld als Guthaben, das vom Systembetreiber (ausstellende Behörde oder Bank) ausgegeben wird. Dieser Systembetreiber fungiert als Brücke zwischen dem traditionellen Bankkonto und dem Quasar-Netzwerk.













## Die Analyse

Zur Analyse der vier DLTs wurden die folgenden sechs Kategorien gewählt: Benutzerfreundlichkeit der Installation, Effizienz und Performance, Kosteneffizienz, Release-Fähigkeit und Aktualität, Sicherheit sowie Verwaltung.

Diesen Kategorien wurden insgesamt 38 Kriterien zugeordnet. Die vier DLTs wurden hinsichtlich dieser Kriterien geprüft und auf einer 5-stufigen Skala von „negativ“ über „neutral“ bis „positiv“ bewertet. Die nachfolgenden Tabellen beinhalten ausschließlich die Kriterien und deren Bewertung anhand einer erweiterten Ampel-Skala.

Tabelle 1

### DLT im Vergleich: Installation – Benutzerfreundlichkeit

Kriterium	Ethereum	Hyperledger Fabric	R3 Corda	Quantoz Quasar
Installation				
Programmiersprachen (Plattform)	GoLang, C++, Java	GoLang, Java	Java	C++
Programmiersprachen (Smart Contracts)	Solidity	Go, Java	Kotlin	QuBic
Modularität				
Dokumentation				

**Legende:**  Positiv  Eher positiv  Neutral  Eher negativ  Negativ  Nicht bewertbar

Für den installierenden Anwender (siehe Tabelle 1) bietet Ethereum eine gute Dokumentation auf Github, dazu zahlreiche ebenso gut dokumentierte Forks und Docker-Pakete, die einem erfahrenen Entwickler die Installation sehr einfach ermöglichen. Ebenso sollte es sich mit Fabric und Corda verhalten, allerdings schien während der Analyse bei Fabric wenig Aktivität vorhanden, was insbesondere die Commits (Einreichungen von Code) angingen.

Corda ist auf Github nur sehr kurz angerissen. Es wird darauf verwiesen, dass der Code nicht eigenständig verändert werden sollte. Stattdessen sollen Änderungsvorschläge als Anträge eingereicht werden, um die Qualität der Plattform zu wahren.

Quasar ist nicht öffentlich dokumentiert bzw. es war nicht möglich, eine öffentliche Dokumentation einzusehen und entsprechend zu prüfen. Stellar hingegen ist auf der Seite der Stellar Stiftung und auf Github gut beschrieben, deshalb wurde das Kriterium Dokumentation als neutral gewertet.

Hinsichtlich der Modularität punktet insbesondere Fabric. Fabric ist nur eins von elf Hyperledger-Projekten; andere Module sind Lösungen wie „Burrow“, welches die Implementierung von Ethereum-basierten Solidity Smart Contracts ermöglichen soll oder „Indy“, das sich ausschließlich mit der Lösung von digitalen Identitäten auf Hyperledger konzentriert.

Bei Fabric wurde von Anfang an das Designziel eines möglichst modularen Aufbaus verfolgt, sodass viele Funktionsebenen von vornherein sauber getrennt sind.

Bei der Effizienz- und Performance-Betrachtung (siehe Tabelle 2) ist zu beachten, dass bei einem beschränkten und ggf. sogar von einem Unternehmen kontrolliertem Set-up einige Konfigurationen, wie etwa der Proof-Algorithmus, so angepasst werden könnten, dass z.B. Ethereum auch performanter arbeiten könnte. In der Betrachtung wurde dies soweit möglich berücksichtigt, dabei jedoch auch der grundsätzliche Zeitaufwand für ein Konsensverfahren in Betracht gezogen. Dabei ist darauf hinzuweisen, dass Ethereum schon sehr früh über die Verkürzung der notwendigen Daten auf der Blockchain nachgedacht hat und dies auch umsetzt.



Tabelle 2

## DLT im Vergleich: Effizienz/Performance

Kriterium	Ethereum	Hyperledger Fabric	R3 Corda	Quantoz Quasar
Datenspeicherung	●	●	●	●
Kürzen von Blocks und StateDB	●	●	●	●
Datenvolumen pro Sekunde	●	●	●	●
Multi vs. Single Threading (DLT)	●	●	●	●
Multi vs. Single Threading (Validation)	●	●	●	●
Durchschnittliche Transaktionsbestätigungsszeit	●	●	●	●
Hardware	●	●	●	●

Quasar/Stellar, Corda und Fabric sind Ethereum in diesem Punkt ebenbürtig. Lediglich bei den möglichen Transaktionsvolumen steht Ethereum allen nach. In der Standard-Konfiguration mit Proof-of-Work gibt es natürliche Limitierungen bei der Menge an Transaktionen und diese können auch in anderen Konfigurationen auftreten. Quasar/Stellar, Corda und Fabric nennen hier höhere Werte, allerdings sind bei Corda und Fabric auf Grundlage der Recherche diese nur in sehr günstig gewählten und nur von den jeweiligen Herstellern/Konsortien durchgeführten Tests dokumentiert worden. Eine Prüfung und/oder Bestätigung hoher Transaktionsvolumen durch Dritte war für alle drei Systeme nicht zugänglich. Insofern können hier Quasar/Stellar, Corda und Fabric keine positiven Werte attestiert werden.

In Sachen Transaktionsgebühren in Tabelle 3 schneiden alle DLT positiv ab, da bei allen die Gebühren frei gestaltbar sind. Die potenziellen Wartungskosten für Applikationen auf Ethereum und Fabric werden als eher positiv bewertet. In beiden Fällen kann aus einer großen Menge externer Entwickler gewählt werden, was die Preisfindung eher positiv beeinflusst. Bei

Corda und Quasar/Stellar ist jedoch zumindest unklar, wie ausgereift der Markt externer Entwickler mit einschlägiger Erfahrung ist.

Tabelle 3

### DLT im Vergleich: Kosteneffizienz





























Kriterium	Ethereum	Hyperledger Fabric	R3 Corda	Quantoz Quasar
Transaktionsgebühren				
Wartungskosten				

Tabelle 4

### DLT im Vergleich: Release-Fähigkeit/Aktualität

Kriterium	Ethereum	Hyperledger Fabric	R3 Corda	Quantoz Quasar
Hersteller				
Reifegrad der DLT				
Updates				
Upgrade-Fähigkeit				
Community				

Die Release-Fähigkeit und Aktualität in Tabelle 4 zeigt klar, dass Ethereum die meisten Entwickler, den höchsten Reifegrad hinsichtlich Überprüfung durch Dritte und die größte Community aufweist. Quasar/Stellar, Corda und Fabric können dafür jedoch im Bereich Upgradebarkeit punkten, da das Installieren von Updates bei diesen leichter gestaltet ist als bei Ethereum.

Tabelle 5

**DLT im Vergleich: Sicherheit**

Kriterium	Ethereum	Hyperledger Fabric	R3 Corda	Quantoz Quasar
Betriebliche Belastbarkeit				
Datenschutz bei Transaktionen				
Regulierungen				
GDPR				
Informationsvertraulichkeit				
Informationsverfügbarkeit				
Authentizität von Identitäten				
Authentizität der Informationen				
System-Integrität				
Daten-Integrität				
Partitionierungsangriffe				
Denial of Service / Distributed Denial of Service Attacks				

Im Bereich Sicherheit punktet Ethereum vor allem mit der Belastbarkeit durch die Existenz als öffentliches System (siehe Tabelle 5). Durch die offenen Quellcodes ist die öffentliche Blockchain einer stetigen Untersuchung durch Angriffe Dritter ausgesetzt, zumal deren Gesamtwert im Bereich von Milliarden US Dollar liegt. Hier können die nicht öffentlich zugänglichen Systeme wie Quasar/Stellar, Corda und Fabric logischerweise nicht dieselbe Belastbarkeit der Sicherheitsaspekte vorweisen. Aspekte des Datenschutzes – insbesondere die der Datenschutzgrundverordnung – bleiben bei allen untersuchten DLT aktuell ungeklärt. Es ist noch nicht abschließend geklärt, ob die Schlüssel der asymmetrischen Verschlüsselung (private/öffentliche Schlüsselpaare) und Prüfsummen (Hashes) persönliche Daten

repräsentieren. Sollte dies zukünftig der Fall sein, ist die Nutzung der asymmetrischen Verschlüsselung grundsätzlich herausfordernd. Es obliegt hauptsächlich den Entwicklern/Architekten jeder Applikation, die geltenden Datenschutz-bestimmungen hinsichtlich der Anwendung zu gewährleisten.

Im Bereich Verwaltung (Tabelle 6) sind Quasar/Stellar, Fabric und Corda gegenüber Ethereum überlegen. Das liegt vor allem daran, dass die drei Systeme per se für Business-Infrastrukturen ausgelegt sind und daher von ihrer Architektur ein höheres Maß an Interoperabilität und Features, wie Testbarkeit und Logging, mit sich bringen. Nichtsdestotrotz lassen sich durch die Anpassbarkeit einer permissioned Blockchain auch diese Features in Ethereum integrieren. Einzig und alleine der initiale Aufwand ist dabei höher bzw. dieser kann wesentlich höher sein.

Tabelle 6

### DLT im Vergleich: Verwaltung

Kriterium	Ethereum	Hyperledger Fabric	R3 Corda	Quantoz Quasar
Interoperabilität				
Zugriffskontrolle				
Anpassbarkeit				
Integration von Drittanbietern				
Prozess der Benutzerregistrierung				
Testbarkeit				
Logging				

### Fazit

Ethereum ist die erste weitverbreitete Blockchain mit einer virtuellen Maschine und weltweit die DLT mit den meisten Entwicklern, Smart

Contracts und Anwendungen. Dadurch bietet Ethereum auch mit Abstand die meiste Dokumentation und Kritik durch Dritte.

Auf Ethereum haben einige Smart Contracts sowie deren ausgelagerte Bibliotheken in der Vergangenheit signifikante Sicherheitslücken aufgewiesen (bekannt sind der Hack des Smart Contracts „The DAO“, der „Parity Bug“ und ebenfalls von Parity die „I accidentally killed it“-Sicherheitslücke in der Bibliothek einer Multisig Wallet). Obwohl diese Hacks nicht die Blockchain selbst betreffen, zeigen sie, dass deren Komplexität viele der heutigen Ethereum-Entwickler überfordern kann.

Insgesamt scheint aber Ethereum kurz- und mittelfristig die höchste Investitionssicherheit zu bieten: Ethereum ist Open Source, besitzt eine große Community sowie gleichzeitig eine hohe globale Verteilung. So kann die Investition in signifikante Geschäftsprozesse auf einer selbst oder gemeinsam mit anderen Akteuren kontrollierten Ethereum-Blockchain mit der höchsten Wahrscheinlichkeit lange genutzt werden. Abhängig vom Anwendungsfall sollte jedoch ein Augenmerk auf den Faktor Performance (Datenvolumen pro Sekunde) gelegt werden.

Als weitere Wahl sind momentan Fabric, Corda und Quasar/Stellar als fast gleichwertig zu sehen. Ein herausragendes Merkmal von Fabric ist die Erfahrung der Linux Foundation in der Entwicklung und Wartung von großen und komplexen Open Source-Projekten. Gerade die Tatsache, dass heute fast alle relevanten Internet Service Provider Server mit Linux-Distributionen betreiben, und nicht etwa Microsoft oder Server anderer Anbieter, ist ein starker Beweis dieser Kompetenz.

Die Beurteilung von Stellar/Quasar erweist sich insgesamt als schwierig. In der Bewertung weicht sie einerseits nur unwesentlich von Corda und Fabric ab. Andererseits sind einige Punkte nicht abschließend objektiv im Rahmen dieses Gutachtens darstellbar. Dies hat sowohl mit der Datenlage zu der Systemkombination, als auch einer bisher fehlenden Dokumentation der Leistung außerhalb einer Testumgebung zu tun.

Dies zeigt den möglicherweise größten Nachteil bei der Wahl eines proprietären Anbieters: nur dieser kann Sachverhalte und Probleme validieren und die Technologie insgesamt warten und weiterentwickeln. Da die Quasar zugrunde liegende Technologie, Stellar, auch in ihrer

öffentlichen Ausprägung nur permissioned funktioniert, ist sie hier daher nicht vollständig auf Höhe von Fabric und Corda zu sehen.

Die große Herausforderung von allen rein permissioned DLT-Lösungen ist der ggf. eintretende Verlust aller signifikanten Attribute, die insbesondere den öffentlichen Blockchains zugeschrieben werden: Unveränderlichkeit der Historie, hohe Sicherheit gegen viele Angriffsvektoren wie Sybil und Denial of Services (DoS)-Attacken sowie dem Netzwerk immanente Herausforderungen wie „Practical Byzantine Fault Tolerance“. Der tatsächliche Verlust oder Erhalt dieser Attribute in einem System ist aber nur im konkreten und hoch detailliert ausformulierten Einzelfall möglich. Tatsächlich sind diese Attribute auch bei öffentlichen Blockchains nicht automatisch gegeben, sondern müssen auch hier in einer Einzelfall-Betrachtung geprüft werden.

Sollten wesentliche Attribute verloren gehen, stellt sich die Frage, wieso nicht gut konzipierte und technologisch vollständig erprobte Lösungen genutzt werden sollten: Dies könnte eine zentral kontrollierte Plattformen mit durchdachten Rollen-/Rechte-Prinzipien sein, bei wahrscheinlich geringeren Kosten und höherer Verfügbarkeit von Entwicklern und System-Architekten und insgesamt höheren Verständnis in der gesamten Organisation.

Die Abwägung für ein konkret und detailliert beschriebenes Zielsystem ist also sehr individuell und nur dann möglich, wenn alle relevanten Aspekte bekannt sind. Des Weiteren wird empfohlen, künftige Entwicklungen der Ethereum Enterprise Alliance und des R3 Corda Konsortiums als auch konkrete Anwendungsfälle von Hyperledger Fabric und Quasar/Stellar zu beobachten und in der Entscheidungsfindung zu berücksichtigen.

**Prof. Dr. Philipp Sandner** leitet das Frankfurt School Blockchain Center. Er kann über E-Mail ([email@philipp-sandner.de](mailto:email@philipp-sandner.de)) kontaktiert werden, via LinkedIn (<https://www.linkedin.com/in/philippsandner/>) und ist auch bei Twitter aktiv (@philippsandner).

**Daniel Höfelmann**, Alumnus der Frankfurt School of Finance & Management, ist Director Innovation Management bei der Aareal Bank in Wiesbaden. Er ist erreichbar via E-Mail ([daniel.hoefelmann@aareal-bank.com](mailto:daniel.hoefelmann@aareal-bank.com)) und ebenfalls auf Twitter (@danielhoefelman) und LinkedIn (<https://www.linkedin.com/in/danielhoefelmann/>) vertreten.