Implementation Aspect: IIoT and Blockchain

An Industrial Internet Consortium White Paper

Version 1.0

2020-07-22

Blockchain is an emerging distributed ledger technology, usually used to support tamper-proof recording of events and transactions, such as the management of digital currencies like Bitcoin. The underlying concept is based on strong cryptographic principles to implement a secure distributed ledger. Contractual business logic between stakeholders can be automated with smart contracts running independently on the blockchain.

When to Use Blockchain in IIoT Solutions

When looking to use blockchain in an Industrial Internet of Things (IIoT) project, we must understand both potential benefits and risks. There are many unsolved problems such as scalability and performance that need to be addressed so blockchain technology can evolve quickly. Due to this state of development, the risk from a project manager´s point of view is increased. Special care must therefore be taken when selecting blockchain technology and designing blockchain-based IIoT-solutions.

On the upside, blockchain has the potential to solve many of the problems faced during the lifecycle of IIoT-enabled assets, including provisioning, usage tracing and asset decommissioning. Blockchain can enable tamper-proof chain of custody, trace important events and structural changes in a heterogeneous IIoT ecosystem. This makes an evaluation of the ecosystem complexity and trust level necessary to justify blockchain relevance.

Other technologies, such as artificial intelligence, additive manufacturing, and virtual and augmented reality, need to be integrated to support project managers driving design decisions and to overcome the complexity in IIoT.

This table provides an overview of typical IIoT use cases and how blockchain can be used.

IIoT Use Case	Blockchain Applications
Device Monitoring	e.g., logging of SLA violations
Device Analytics	e.g., logging of analytics results / forecasts
Edge Autonomy	e.g., logging of fault protocols (e.g., accident)
New IIoT-based Services	e.g., truck odometer fraud prevention

Distribution and Ownership of Data

When defining blockchain-based IIoT solutions, solution architects must decide how to manage the distribution and ownership of data in the system.

Network reliability, bandwidth and latency as well as functional or solution usage attributes have to be designed with the system’s performance and trustworthiness in mind. This already complex problem is complicated further with blockchain in the picture, as blockchain-related implications need to be considered on all layers of the solution architecture. For example, a key question is how to manage the wallet of each user, which contains the public and private key pair required to create and access respective user data in the blockchain. Consequently, who owns and controls these wallets also becomes a key question.

There are three possible patterns for the distribution of data in a blockchain-based IIoT solution. In this section, we will describe them while using the three-tier architecture pattern from IIC's *Industrial Internet Reference Architecture (IIRA)* as a guiding principle. To illustrate the three patterns, we will use a simple example in which a truck is equipped with temperature and shock sensors. If certain temperature or acceleration values are exceeded, the transportation company will be liable for the violation of the agreed upon Service Level Agreements (SLAs).

Platform-Controlled Wallet

The first architecture pattern assumes that all data and control flows are managed centrally by the platform tier. As shown in Figure 1, the central platform makes all the decisions and controls the wallets. For example, the platform monitors the data coming in from the truck in the field and logs it in the time-series database of the platform. So as not to overload the blockchain, only significant events (e.g., SLA violations) are logged. Data saved on the blockchain or in the database is decided by the platform.

This pattern has the following benefits:

- full control over data, control flows and wallets in the platform. This means that the implementation will be easier to create and maintain,
- no problems with remote management of wallets and
- no investment in specialized hardware.

However, a central disadvantage of this approach is that it requires all parties involved (including the customer who is relying on the SLAs) to trust the link from the truck to the platform. Moreover, the data access rights remain with the platform provider, which needs high data security and privacy means to ensure trustworthiness with the end-user. This might not always be a given. In this case, this pattern is not an option.

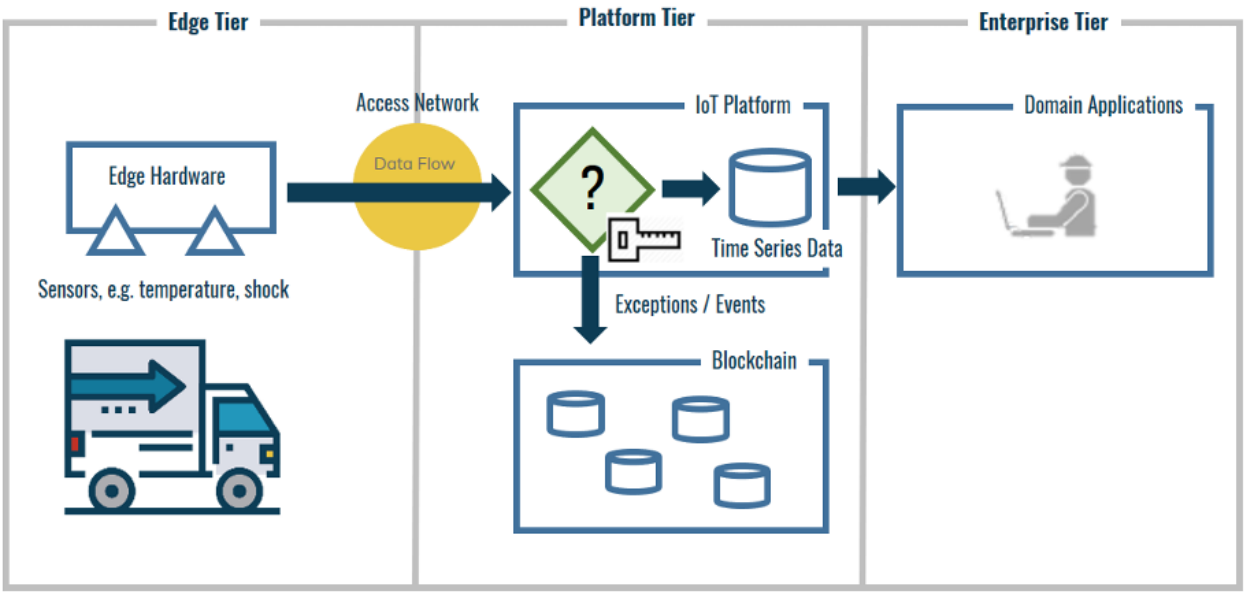

Figure 1: Platform-Controlled Blockchain

ASSET-CONTROLLED WALLET

The next pattern assumes that every asset has an embedded wallet to sign and access its data on the blockchain or provide access to the data. For example, *Trusted Platform Module (TPM)* technology can be used to implement a specialized hardware that cannot be tampered with in the field. It provides a secure storage of the corresponding wallet. As shown in Figure 2, the software running on this hardware is directly deployed on the asset and it will decide which data to write to the blockchain. Beforehand, it signs the data with the private key of its wallet.

The key advantage of this solution is a completely tamper-proof link directly from the sensor to the blockchain.

However, there are some disadvantages, which include:

- potentially costly custom hardware,
- potentially higher development and maintenance costs, due to the fully distributed nature of the system and
- trust in the TPM provider is required.

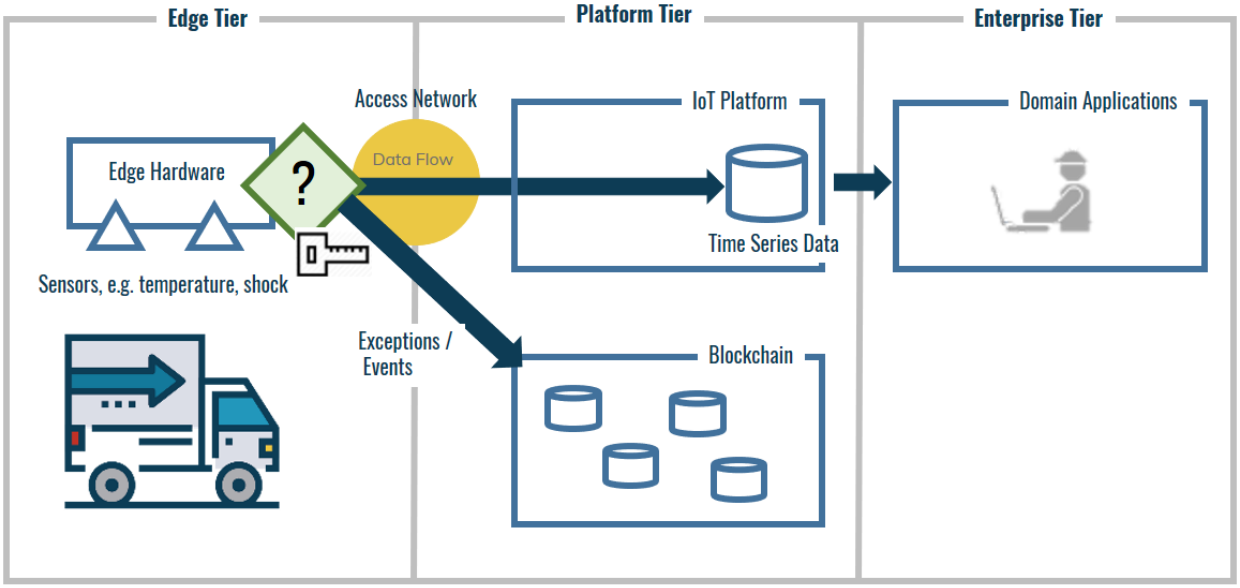

Figure 2: Asset-Controlled Blockchain

Smart Contracts Enhancement

Finally, smart contracts enhance the demonstrated architecture patterns. A smart contract implements business logic directly embedded into the blockchain. This means that—due to the distributed and cryptographic nature of blockchain—both the logged data and the business logic are tamper-proof. It will only be executed if the maturity of distributed nodes in the blockchain agree about the outcome of a decision. A smart contract enables an independent execution of business logic between stakeholders. Moreover, the agreed-upon outcome builds trusted input for further procedures.

Note that this pattern can be combined with either of the first two patterns. For simplicity, Figure 3 shows the use of smart contracts with an asset-controlled wallet.

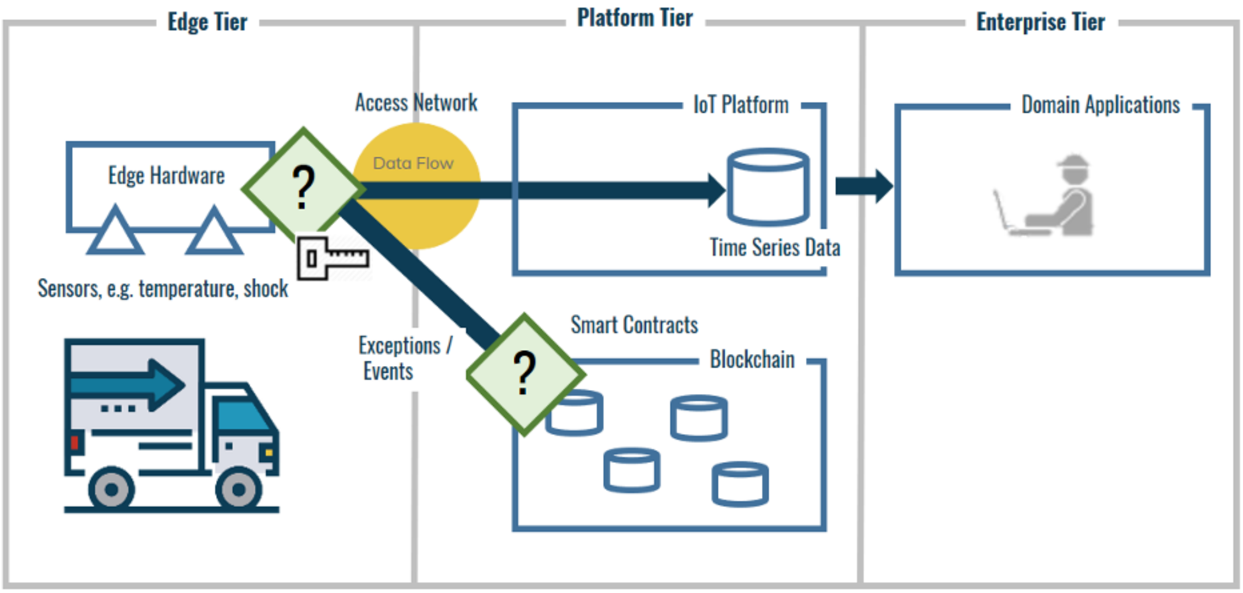

Figure 3: Smart Contracts

Conclusions and Outlook

Current project managers face high technological complexity when it comes to solution design. A relevant component in the design of distributed systems is the emerging blockchain technology. Due to the distribution of functionalities and a missing intermediary, different design and implementation principles need to be considered, as well as further technical challenges.

Blockchain guarantees trust between the stakeholders of a distributed system as an immutable, transparent, anonymous and tamper-free data structure. Despite the benefits of blockchain, the risks of the evolving technology need to be considered while designing blockchain-based IIoT solutions.

This document described three design patterns for the distribution and ownership of data in a blockchain-based IIoT solution while using the three-tier architecture pattern from IIC's IIRA reference architecture. The patterns are illustrated by a simple practical example to emphasize the patterns usage.

After the decision is made to use blockchain in an IIoT solution, the illustrated patterns give design alternatives and support the design decision in accordance with the requirements on data distribution. Based on the chosen pattern, the trustworthiness of the IIoT solution can be evaluated by the pattern's characteristics. Furthermore, transparency of data ownership and access is created that supports the project manager's decision.

Finally, the chosen pattern provides the blueprint for a more detailed architecture and implementation design by the solution architect. This tool provides guidance for project managers that are preparing an IIoT project.

AUTHORS AND LEGAL NOTICE

This document is a work product of the Industrial Internet Consortium Industrial Distributed Ledger Task Group, co-chaired by Mike McBride (Futurewei) and Xinxin Fan (IoTeX).

Authors: The following persons contributed substantial written content to this document: Dirk Slama (Bosch Software Innovations) and Daniel Burkhardt (Ferdinand-Steinbeis-Institute).

Technical Editor: Stephen Mellor (IIC staff) oversaw the process of organizing the contributions of the above Authors into an integrated document.