BLOCKCHAINS FOR CONDITION MONITORING

Deliverable D 5.2

Final Report

Project acronym:	B4CM
Starting date:	01/12/2018
Duration (in months):	48
Call (part) identifier:	H2020-S2RJU-OC-2018
Grant agreement no:	826156
Due date of deliverable:	Month 48
Actual submission date:	16-10-2023
Responsible/Author:	John Easton (UoB)
Dissemination level:	Public
Status:	Issued

Reviewed: Yes

This project has received funding from the European Union's Horizon Europe research and innovation programme under Grant Agreement No 826156.

Document history		
Revision	*Date*	*Description*
1	16th October 2023	Initial release

Report contributors		
Name	Beneficiary Short Name	Details of contribution
John Easton	UoB	Lead author
Rahma Alzahrani	UoB	Contributing author

Table of Contents

1 Executive Summary 1
2 Abbreviations and acronyms 2
3 Background 3
4 Objective/Aim 4
5 Software Architecture and Framework 5
5.1 Platform / SDK Selection 5
5.2 Design of Architecture 6
6 Accounting Model 9
6.1 Access Agreement Model 9
6.2 Payment Process 10
6.3 Agreement Revocation 11
7 Proof of Concept 13
7.1 Deployment 13
7.2 Walkthrough 18
8 Delivery of Planned Objectives 21
9 Conclusions and Recommendations 23
10 References 25

1 Executive Summary

The pursuit of higher quality services in the railway sector is a continuous process, and the availability in recent years of affordable, reliable, digitally enabled additions to traditionally mechanical-based infrastructure systems has provided a fruitful avenue for advancement. Remote Condition Monitoring (RCM) systems are one example of a tool that has been widely deployed to improve the standards of maintenance, reliability, and safety across the rail network. Such systems offer particular benefits at the traditional boundaries of responsibility within the industry (e.g. the interface between the infrastructure and rolling stock) where complex physical interactions may make the cause to defects difficult to determine. Although this type of cross-interface monitoring of assets may be the most technically practical solution to many industry-wide problems, commercially they can prove complex as the business paying to install, maintain, and operate the sensing device is not the party benefitting from the data collected. As a result, it can be hard to generate business cases for the purchase, installation and operation of cross-interface monitoring systems that would have recognised industry-wide benefits.

This document summarises the findings of the B4CM project, a programme of work designed to establish whether distributed ledger technologies can prove an auditable log of data exchanges between actors in a RCM context, thereby enabling a fairer attribution of costs and benefits.

The team comment on the extent to which B4CM has delivered on its specific objectives, both in terms of the capability of the technology, and the implementation potential of the current SDKs.

In conclusion, the B4CM team recommend that:

- While this work has shown that distributed ledger technologies do offer significant potential value to the rail sector, early adoption should focus on use cases that are more traditionally associated with the transfer of tokens, such as rail ticketing, in order to prove the business case and build staff capability within the industry;
- Once a sufficiently experienced staff base has been established within rail, initial industrial deployments of the technology around audit of data shadow transactions being made in other data exchange platforms, such as the GB Rail Data Marketplace, providing firm evidence of the applicability of the technology in a way that does not impact on mission critical activities. Only once this has been demonstrated, and performance at representative scale proven, will it be appropriate to move forward with live operational systems.

2 Abbreviations and acronyms

Abbreviation / Acronym	Description
B4CM	Blockchains for Condition Monitoring
IoT	Internet of Things
RCM	Remote Condition Monitoring
RSSB	Rail Safety and Standards Board
SC	Smart Contract
TTP	Trusted Third Party
UOMS	Unattended Overhead Line Equipment Monitoring System

3 Background

Over the past decade there has been a significant level of investment throughout Europe in the digitalisation of the rail network. This includes the installation of sensors on the infrastructure and vehicles, the deployment of next generation traffic management systems that allow real-time management of the system, and the provision of mobile applications for passengers and staff. Despite the wealth of new data provided by these systems, the railways are still struggling in their aspiration to be an information-led industry due to a lack of traceability of information usage, and the commercial barriers between stakeholders.

Blockchains are a disruptive technology that have the potential to accelerate the development of rail as the primary medium-distance carrier within the wider multi-modal transportation system. Directly funded by the rail industry via the EU Shift2Rail Joint Undertaking, the Blockchains for Condition Monitoring (B4CM) project will identify key use cases for the technology within the railways, deliver a blockchain-based testbed that enables the benefits of the technology to be formally evaluated, and demonstrate the value of blockchains in the attribution of data costs across organisational boundaries within the European rail sector.

The overall aim of the B4CM project is to develop and deliver a blockchain-based testbed for the attribution of data costs across organisational boundaries, and to demonstrate the operation of the framework and in the context of the European Rail Industry, enabling future developers to extend the tools produced based on a known working configuration.

B4CM has the following research and training objectives:

Objective 1: To identify and develop use cases that support the application of blockchain in the railway sector;

Objective 2: To develop an implementable blockchain framework for the attribution of data costs in systems crossing organisational boundaries;

Objective 3: To evaluate mechanisms for the incorporation of the developed blockchain framework into the financial processes of the European rail sector;

Objective 4: To develop a testbed, demonstrating the operation of the framework in the context of rail sector, enabling future developers to extend the tools produced based on a known working configuration;

Objective 5: To disseminate the findings of the project and the lessons learned to influence best practice in innovation and technology uptake in a key and evolving field within the European rail sector;

Objective 6: To support the development of a researcher in gaining a PhD and thus generating a skilled specialist valuable to the European rail sector.

This document, the final report on the B4CM project, summarises the key project outcomes and makes recommendations for the future.

4 Objective/Aim

This document forms part of Work Package 5 of the B4CM project and has been prepared to summarise the key findings of the project, comment on the success of the project, and report recommendations for the future.

Although originally devised under the Shift2Rail programme, and intended to contribute to TD 3.7, 3.8, 4.6, 5.0 of that activity, the content of this document can also be seen as a direct contribution to the objectives of FP3 (IAM4RAIL) under the current Europe’s Rail programme, focussing on the holistic and integrated asset management for Europe’s rail system.

5 Software Architecture and Framework

In Deliverable D1.1 the B4CM team introduced a concept for a blockchain based framework that would enable the attribution of costs to partners in cross-industry remote condition monitoring systems. The core framework was made available via the project GitHub.

5.1 Platform / SDK Selection

In order to ensure the industrial relevance of the outcomes, from the inception of the project the B4CM team have based their decision-making around case studies of existing cross-interface Remote Condition Monitoring (RCM) systems. Specifically, these use cases would focus on two of the four “quadrants” typically seen in rail RCM systems; sensors mounted on the infrastructure monitoring the vehicles, and sensor mounted on the vehicles monitoring the infrastructure. Building on work performed by the Rail Safety and Standards Board (RSSB) under their project T857[1], [2], the use cases initially selected were axle bearing monitoring (RailBAM), and unattended overhead line monitoring (UOMS). Stakeholder analysis was performed for each (Figure 1), and this was aligned with the wider industry structure to establish a set of core roles and requirements for the framework (e.g. knowledge of the identity of participants, the need for automation of contracts etc.).

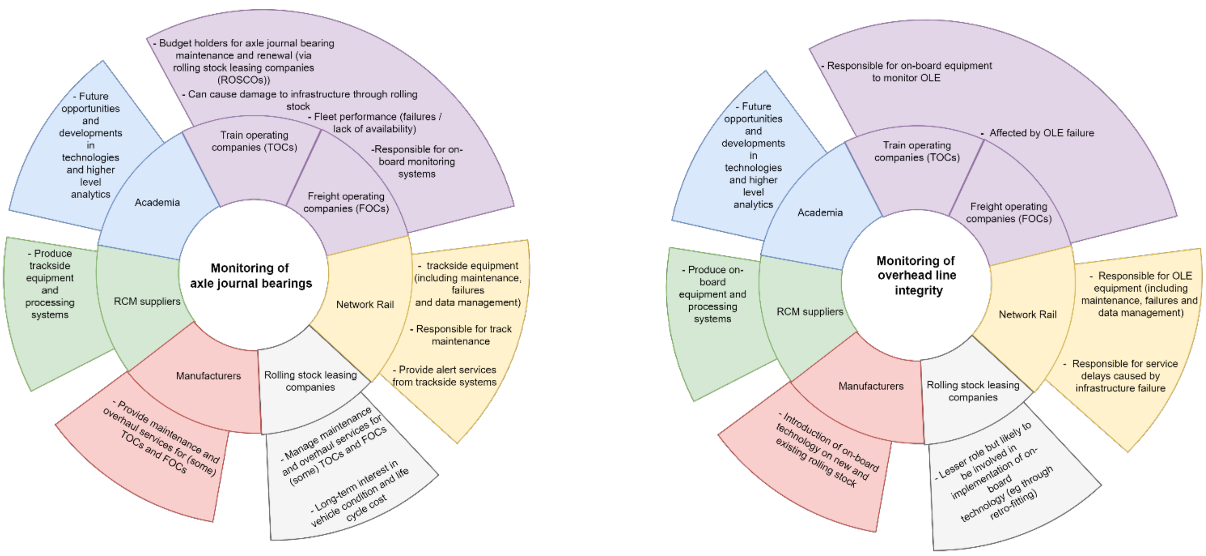

Figure 1: Main stakeholders in selected use cases

Based of the requirements extracted from the use cases, the B4CM team then compared and contrasted the functionality of four of the most widely used distributed ledger platforms / SDKs currently available for use by developers. Of these, a need for maturity in the solution meant that three of the reviewed platforms were drawn from the Hyperledger family of products (as these are collectively supported by an industrial-backed foundation), with the remaining product, Ethereum, being an established actor in the field over many years. The results of the comparison are summarised in Table 1, and led to the selection of Hyperledger Fabric as the SDK of choice for the project.

Table 1: Comparison of functionality between distributed ledger platforms

Criteria	Ethereum	Hyperledger Fabric	Hyperledger Sawtooth	Hyperledger Iroha
Supports smart contracts	✔	✔	✔	✔
Consensus algorithm modularity	✕	✔	✔	✕
Built-in components for managing identity	✕	✔	✕	✔
Support for payment in fiat currency	✕	✔	✔	✔
Proficient in maintaining different privacy levels between users	✕	✔	✔	✔

5.2 Design of Architecture

Building on the choice of platform, the team moved forward with developing the initial high-level architecture for the framework (Figure 2). It was clear from an early stage in the work, that in order to deliver the required levels of performance the amount of data stored within the chain itself would have to be kept to a minimum, and as a result off-chain storage would be required for the data being exchanged. As a result, the high-level architecture was supported by a layered view of the interactions between the logical structures to be deployed within the blockchain network, including the interaction with the external store (Figure 3).

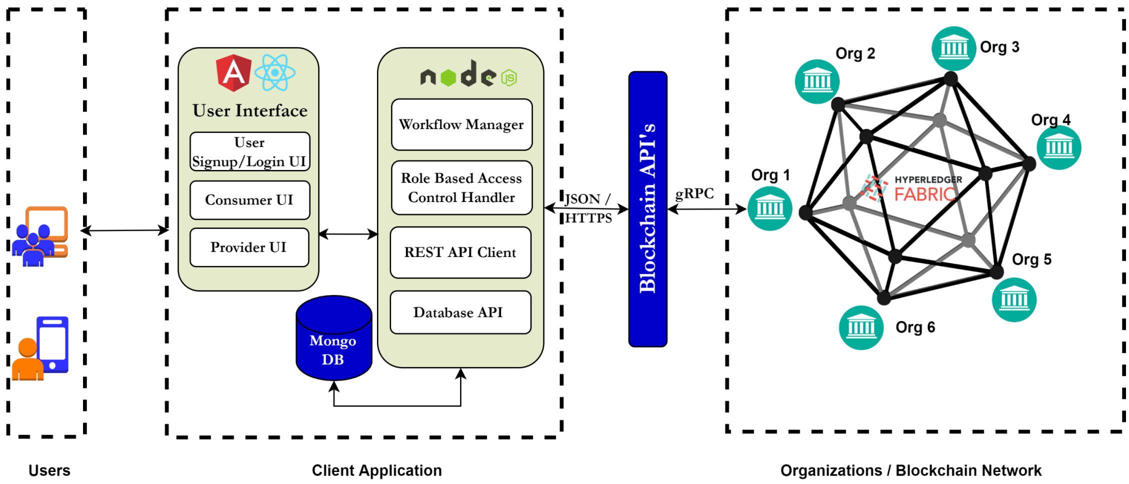

Figure 2: Initial high-level architecture of the B4CM framework

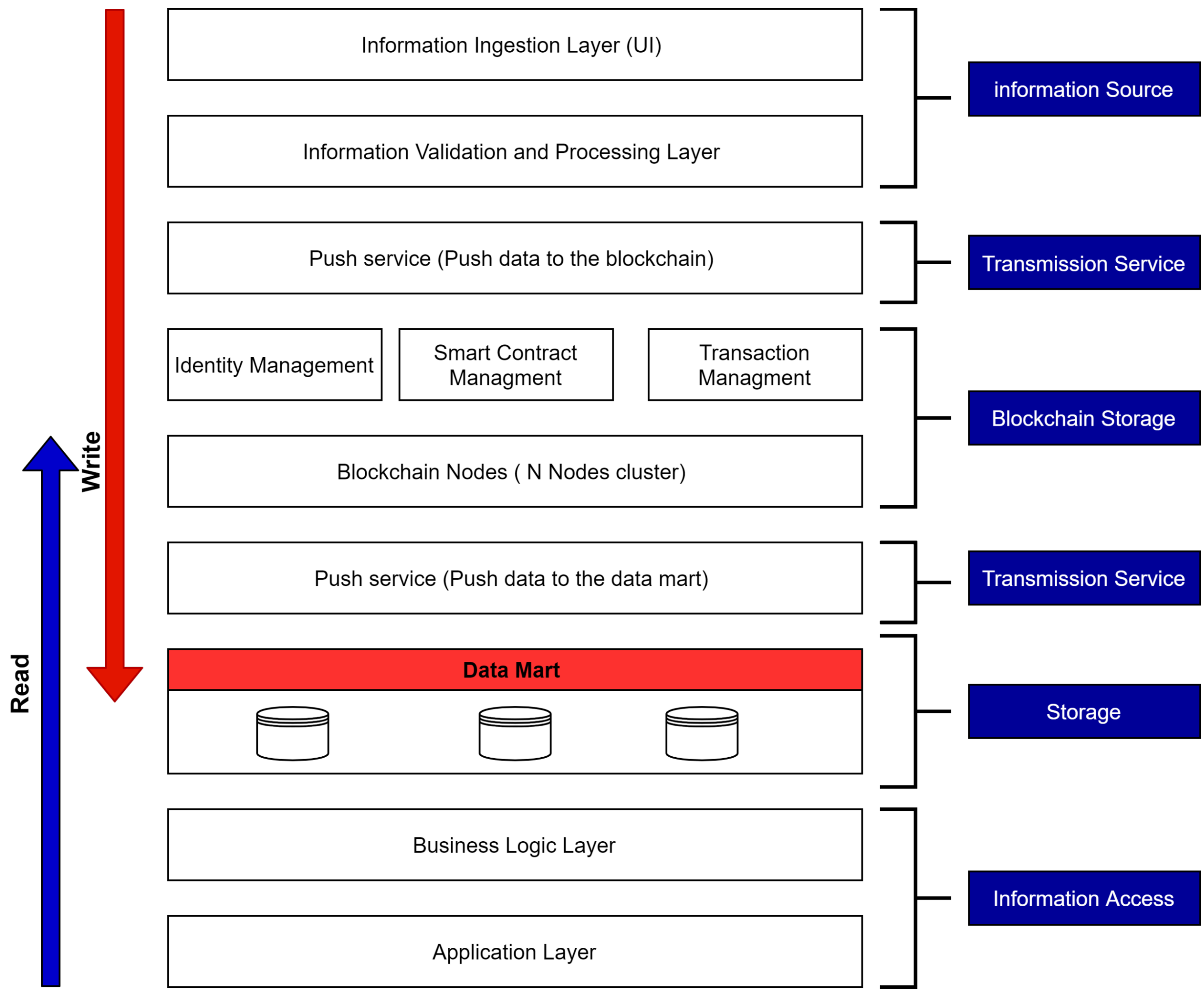

Figure 3: Logical structure of the blockchain network's functional layers

A set of core data structures / entities were established that allowed the structure of a transaction to be captured on the blockchain, their interactions are shown in Figure 4 and were further expanded on in Deliverable D2.1 (see Section 6.1 of this document for a summary).

Finally, all of these elements were pulled together into the core B4CM framework, which was presented in the deliverable as a set of concept interfaces, and used as the basis for the Proof of Concept demonstrator that would later be presented as part of project Deliverable D3.1 (see Section 7.2 of this document for details)

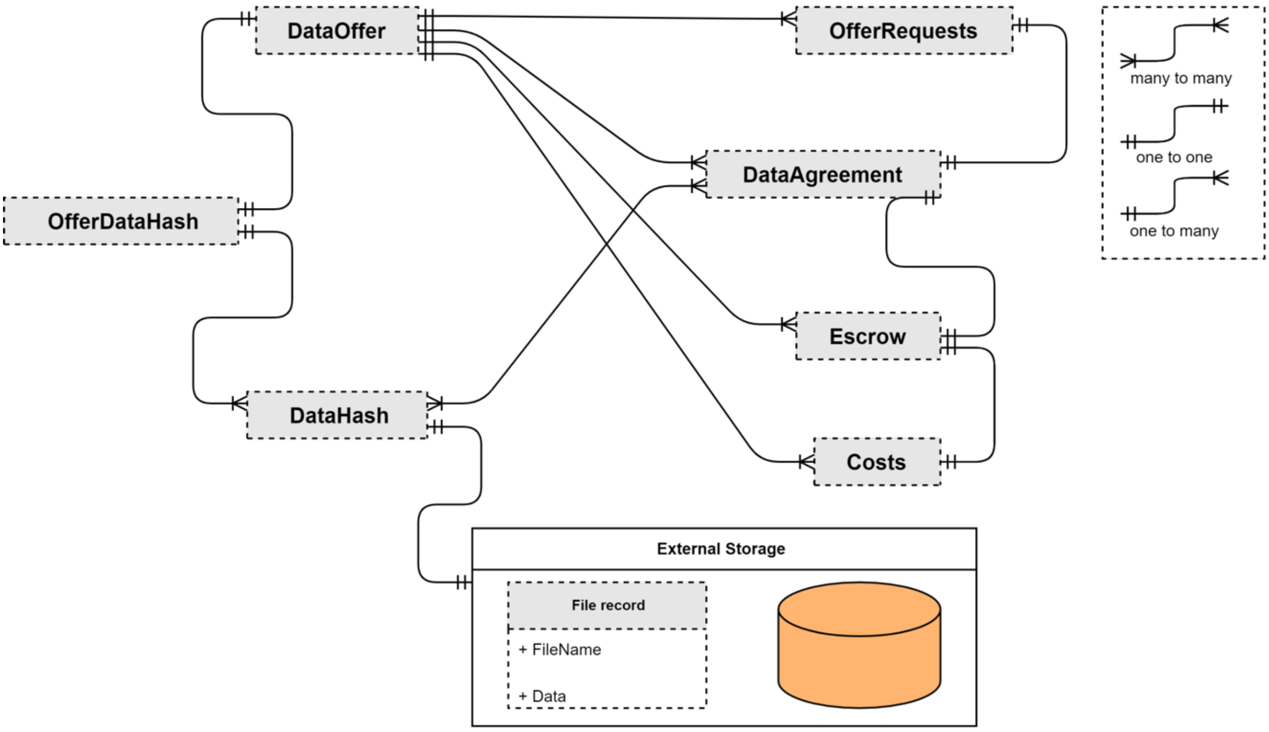

Figure 4: Relationships between the chaincode entities and the external storage used alongside the core ledger

6 Accounting Model

In Deliverable D2.1 the B4CM team proposed an accounting model to accompany the framework introduced in Deliverable D1.1.

6.1 Access Agreement Model

At the core of the accounting model is the concept of a data access agreement. Agreement over, and preparation of, these commercial documents has been a cause of significant delay in many cross-interface RCM projects within the rail industry in recent years. While some activities, most notably the template commercial agreements provided by RSSB's T1010 project[3], have attempted to resolve this issue the need to manage complex sets of agreements by hand still limits the rail industry's ability to adequately exploit its data resources.

B4CM's use of a shared ledger allows for the possibility of the rapid establishment of short-term, low value (and thereby limited risk), agreements around each data transaction in the form of Smart Contracts (SC). In order to implement the access agreement models, the B4CM team included specific provisions for them within the blockchain ledger; two new records, "DataAgreement" and "Escrow", were created that would be automatically generated by SC and appended to the ledger each time a new data access request is made by a consumer. The DataAgreement held information on the new agreement between the data consumer and data provider, including the data offered by the provider, the unit price, and the period of validity. The Escrow record formed the basis for enforcement of access to the data and exchange of payment on release, and as such was primarily suited to the management of transfers of static or semi-static datasets (historical corpuses of monitoring data, reference data on the infrastructure, asset information etc.).

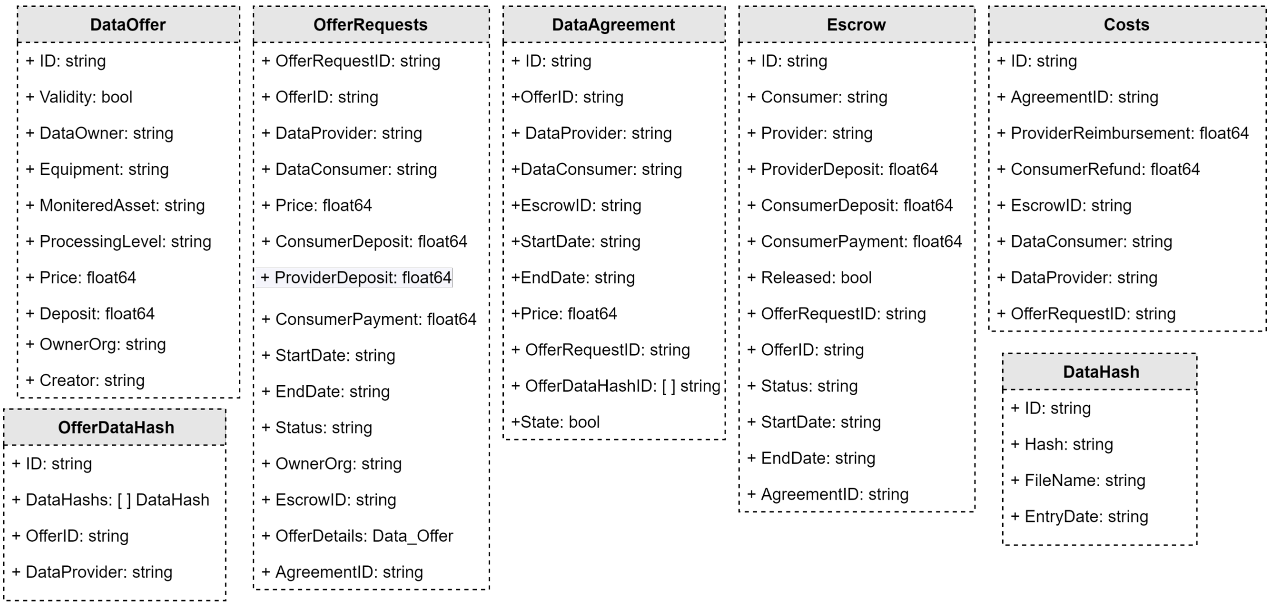

Figure 5: Data structure for the lodging of commercial agreements on the blockchain

In real-time monitoring, when highly dynamic data exchanges existed for very short periods in (near) real time between Internet of things (IoT) devices or cloud data lakes, a full Escrow process may not be practical or applicable; in these cases, data requests would be processed subtlety differently, using predefined access control lists that permitted access to the chain / trading facility by legitimate devices only. The DataProvider and DataConsumer attributes in the OfferRequests record would, in these situations, then represent IoT device / cloud platform IDs. Access control would be implemented within the deployed smart contract to grant access restrictions as per specifications in the network configuration for the partner organisation. Exchanges would be limited in size and value, making payments very small and frequent; it should be noted that this scenario would not be practical in the general case (as there would be a huge overhead for larger datasets), it did, however, apply well the microtransaction scenario and required minimal changes to the Escrow based mechanism already defined.

6.2 Payment Process

Fulfilment of the accounting model is based in the idea that payments on any trading site may be realised using post-paid or pre-paid models. The post-paid model requires the provider to place trust in the consumer (buyer) that the payment will be made as agreed after the data is delivered. The pre-paid model requires that the consumer places trust in the provider that the data will be delivered once the payment has been made as agreed. Neither model guarantees both consumer and provider satisfaction, and both bear some risk if the other party breaches the terms of the agreement. There is also a requirement for a Trusted Third Party (TTP) to provide both the provider and the consumer with an escrow service. An example of the sequence model for such an exchange is shown in Figure 6.

For large transfers of pre-existing data (i.e. historical data products) the B4CM framework uses Smart Contracts to provide the Escrow functions within the data exchange workflow, removing the need for a TTP and enabling the finances to be automatically released after the consumer has confirmed data delivery.

The complete payment process flowed as follows:

- An escrow SC will be initiated once the consumer responds to a published offer. The escrow details the offer being responded to, and triggers payment of the corresponding charge and deposit by the consumer. On receipt, the SC will then direct the request to the provider.
- On receiving the request, the provider will check if the payment and deposit detailed in the escrow are matched with their offer. Then, in order to lock up the escrow, the provider must pay their deposit, which may not be less than the deposit of the consumer. If the provider determines that the size of the payment or the deposit does not match with the terms of their offer, the provider can reject the request and the consumer will get back their payment.
- The process of locking the escrow will trigger a SC to initiate an agreement, in which the period over which the consumer has access to the provider’s data is specified.
- The cost of data consumption will be monitored via the SC when the escrow is released. The escrow will be released automatically if either of the two states below are realised:
    - The agreement’s expiry date is reached; or

- The agreement is revoked.

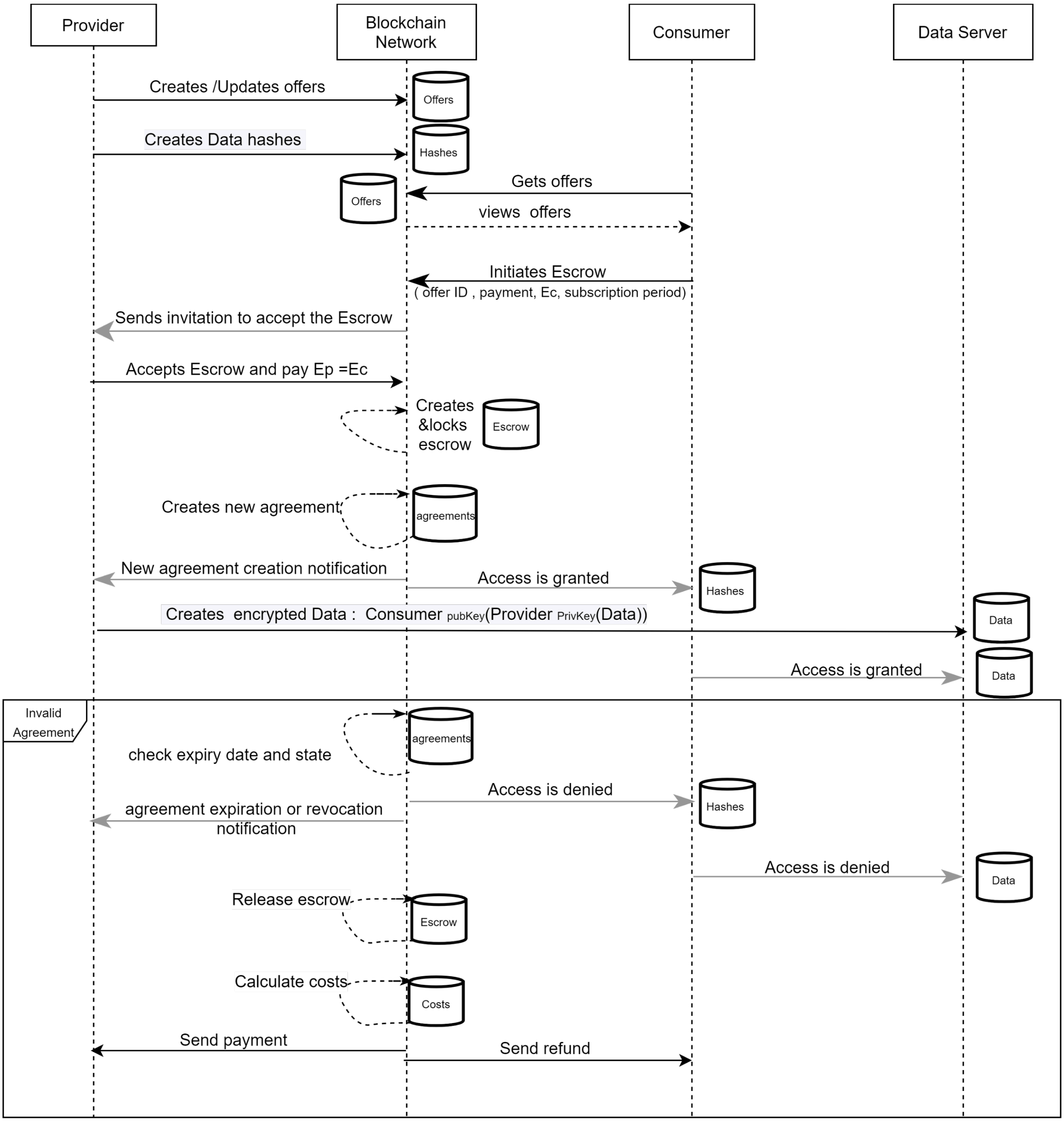

Figure 6: Data access agreement sequence model

6.3 Agreement Revocation

One of the advantages of the use of an Escrow in the data exchange process is that it provided an opportunity for both the data provider and the data consumer to either withdraw from the agreement before completion, or to respond in the event of fraudulent behaviour on the part of the other participant in the data exchange. The B4CM team identified a set of six scenarios that may result from such a situation and provided formalised responses for each within the proposed accounting model. A key element of deciding whether claims issued within the framework are genuine is linked to evidencing any alleged fraudulent behaviour based on the data, and the combination of the immutable record on the ledger, along with the transactional nature of the SC mean that the blockchain is an ideal environment for testing this.

Figure 7 shows an example of one of the modelled revocation scenarios; in this case the data consumer had received the requested data as agreed but raised a genuine complaint about the latency in providing the appropriate hashes (necessary to validate the data) to the network on the part of the provider. As illustrated in Figure 7, the deployed SC would evaluate this claim by checking the time of appended hash values on the chain, using the block's timestamp. As the consumer's claim was genuine, the agreement would then be revoked, triggering the calculation of net costs for only the time the agreement was fully in force (i.e. the time after the hashes had been published, not the full time since initial delivery).

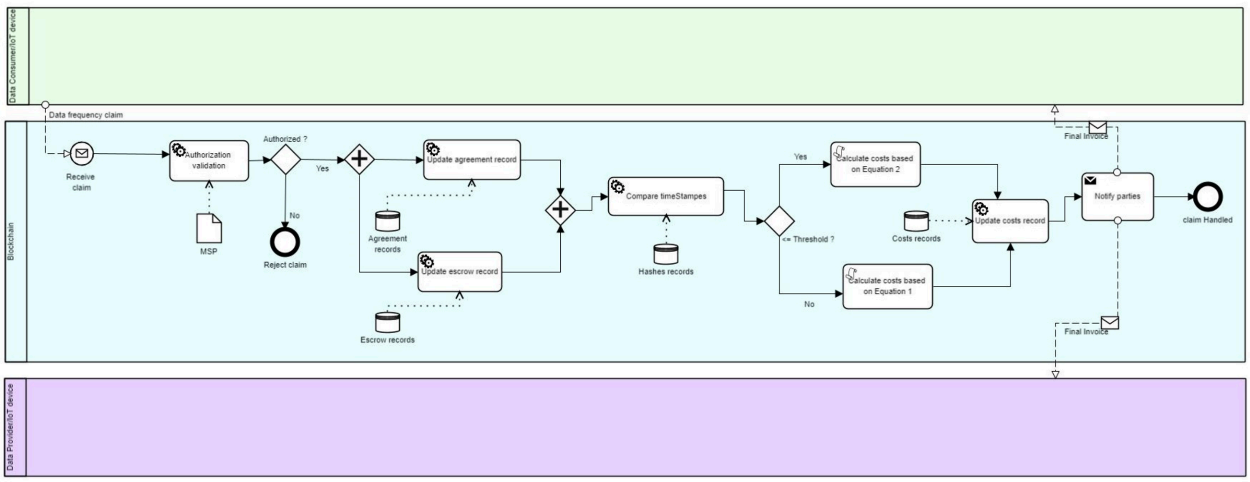

Figure 7: Example revocation of exchange agreement based on late delivery

7 Proof of Concept

The proof of concept deployment was developed around two realistic industrial case studies, one in which fixed monitoring equipment mounted on the infrastructure was monitoring the vehicles (axle bearing monitoring), and one in which sensors on the vehicles were monitoring the infrastructure (inertial measurement of track geometry). The use cases were chosen so real-world data, gathered by the respective monitoring systems, could be managed and replayed within the proof of concept giving a representative perspective on it's likely performance "in the wild".

The relationship between the assets and monitoring equipment involved directly in the case study, the wider actors and ecosystem of the railway, and the blockchain network (in the form of the deployed framework) was summarised in the context of the diagram shown in Figure 8.

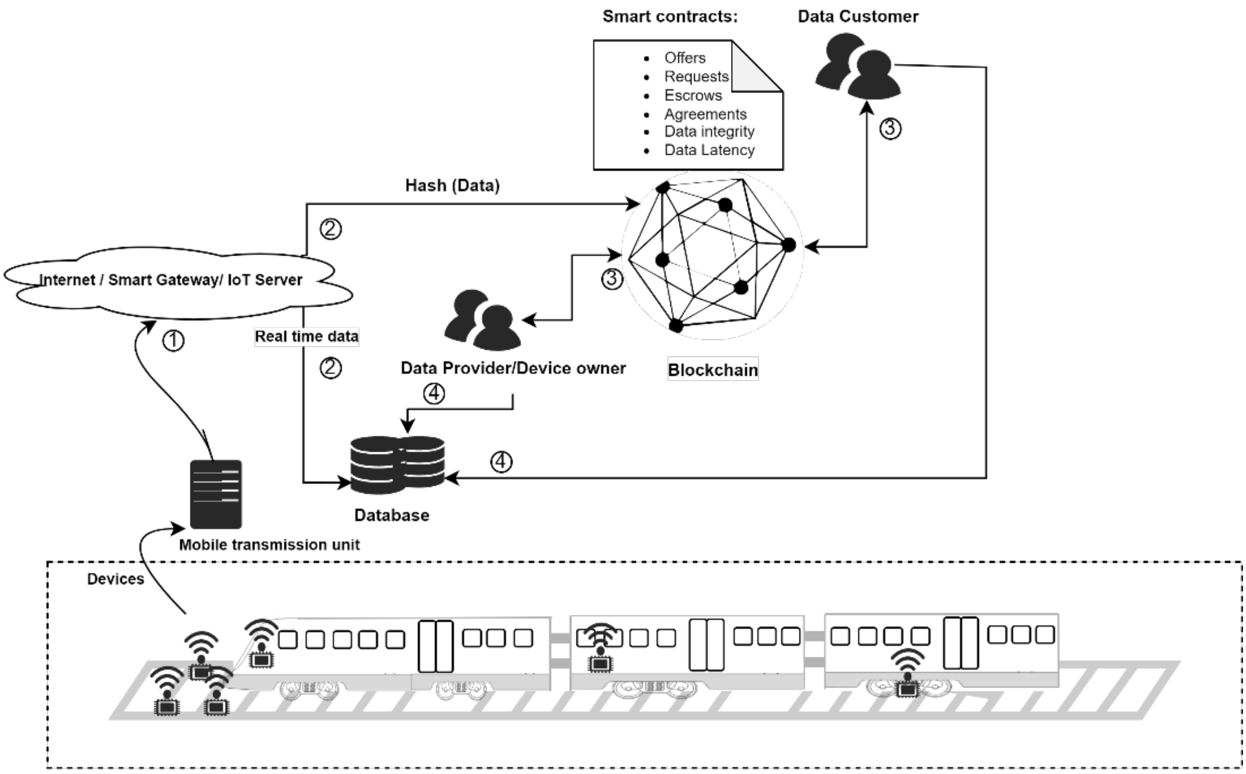

Figure 8: Conceptual view of deployed framework within wider monitoring ecosystem

7.1 Deployment

The deployed platform consisted of three main parts, the Blockchain Network, the IoT device simulator, and the client application. The development stack for the blockchain network was as shown in Table 2. The operating system used for development was Ubuntu Linux 18.04.4 LTS running on an Intel Core i7-8650@1.90 GHz with 15.5 GB of memory. The virtual environment hosting the framework was docker version 19.03.8 while the docker images and containers were configured and orchestrated by docker-composer version 1.23.2. The Hyperledger Fabric (V 2.2)

SDK was used to create the blockchain. Smart contracts were coded using Golang, with state information held in Couch DB, a choice that was made to enable complex queries against the data held on the chain. In Table 3, the development tools used to implement the simulated IoT platform are listed. The IDE used to develop the simulation platform was Spring Tool Suite (STS), a free Eclipse-based development environment for developing web applications. The interface between the sensor device simulators and the server was provided by Redis, and this was linked to the blockchain network using a standard HTTP REST webservice. The interface between the sensors and the blockchain can be seen in Figure 9, while the relationship to the overall architecture of the proof of concept is shown in Figure 10.

Table 2: Deployment environment (blockchain framework)

Component	Description
CPU	Intel Core i7-8650 @ 1.90 GHz
Memory	15.5GB
Operating System	Ubuntu Linux 18.04.4 LTS
Docker Engine	Version 19.03.8
Docker Compose	Version 1.23.2
Node	Version 14.15.4
Hyperledger Fabric	Version 2.2
IDE	Visual Studio Code
DBMS	Couch DB, MongoDB
Programming Language	Node.js, GoLang

Table 3: Deployment environment (IoT simulator)

Component	Description
Storage server / Gateway	Redis
Memory	15.5GB
Operating System	Ubuntu Linux 18.04.4 LTS
IDE	STS
Transmission Protocol	HTTP
Programming Language	JAVA

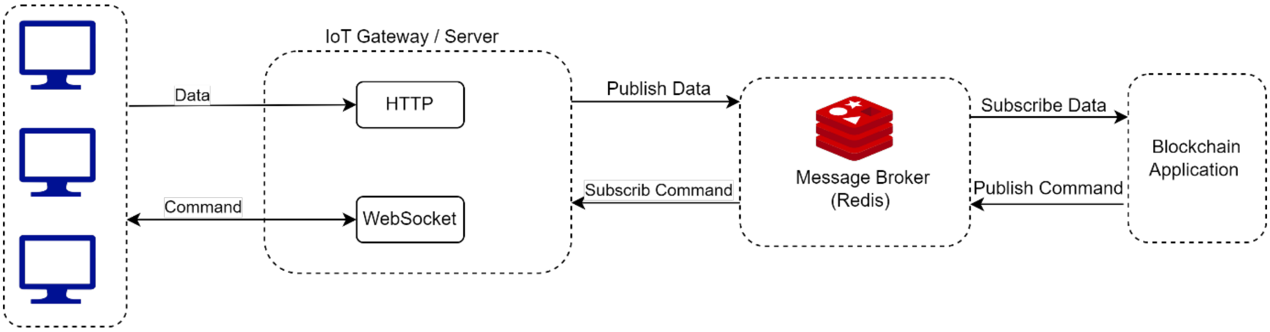

Figure 9: Architectural diagram showing the interface between the simulated sensors and the blockchain via Redis

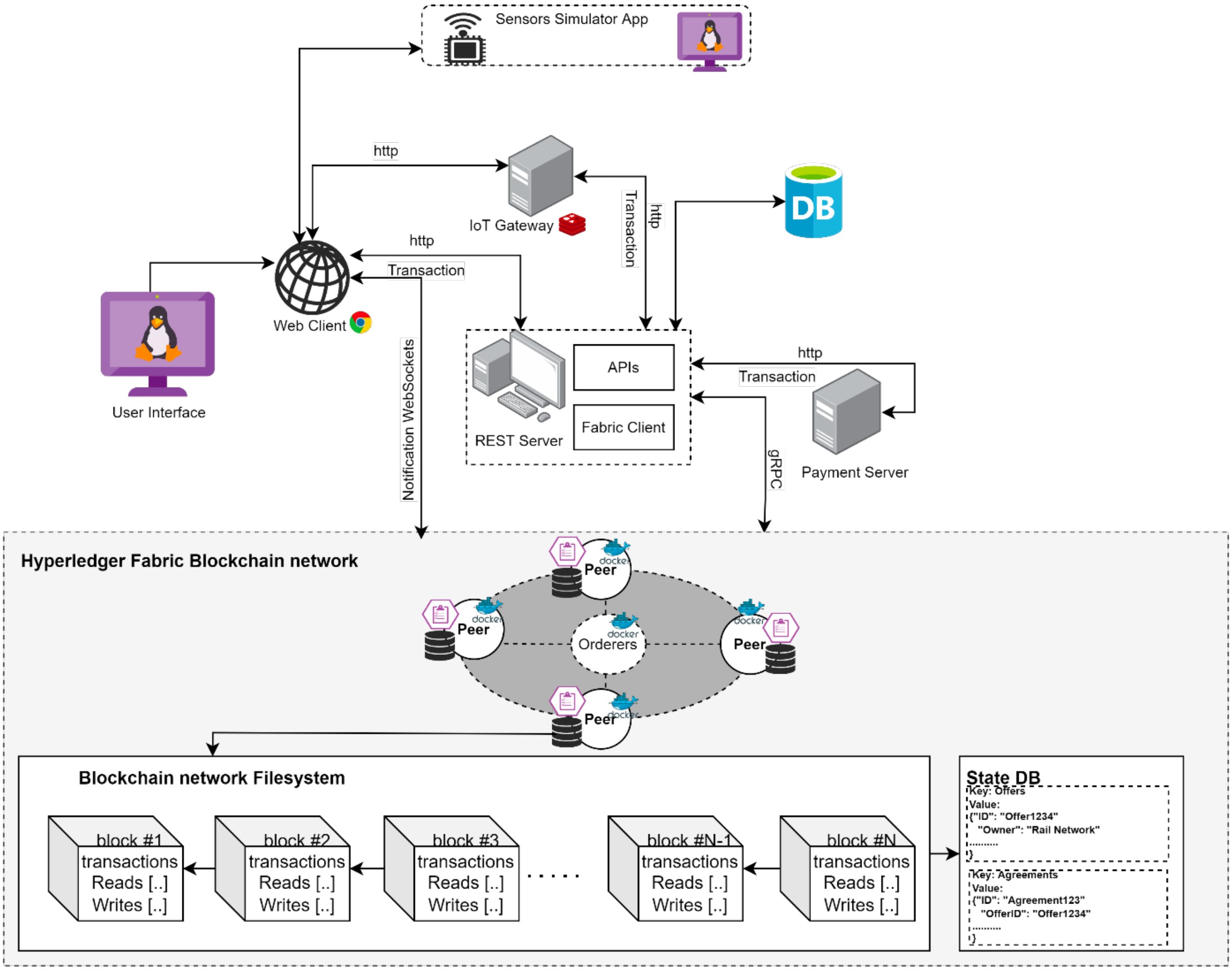

Figure 10: Architectural diagram for the proof of concept

With live access to the real data collection systems clearly impractical for a proof of concept, a simulator was implemented that enabled data recorded by the measurement systems that were the subjects of the use cases to be replayed into the framework in pseudo real-time based on the timestamps from the original measurements (Figure 11).

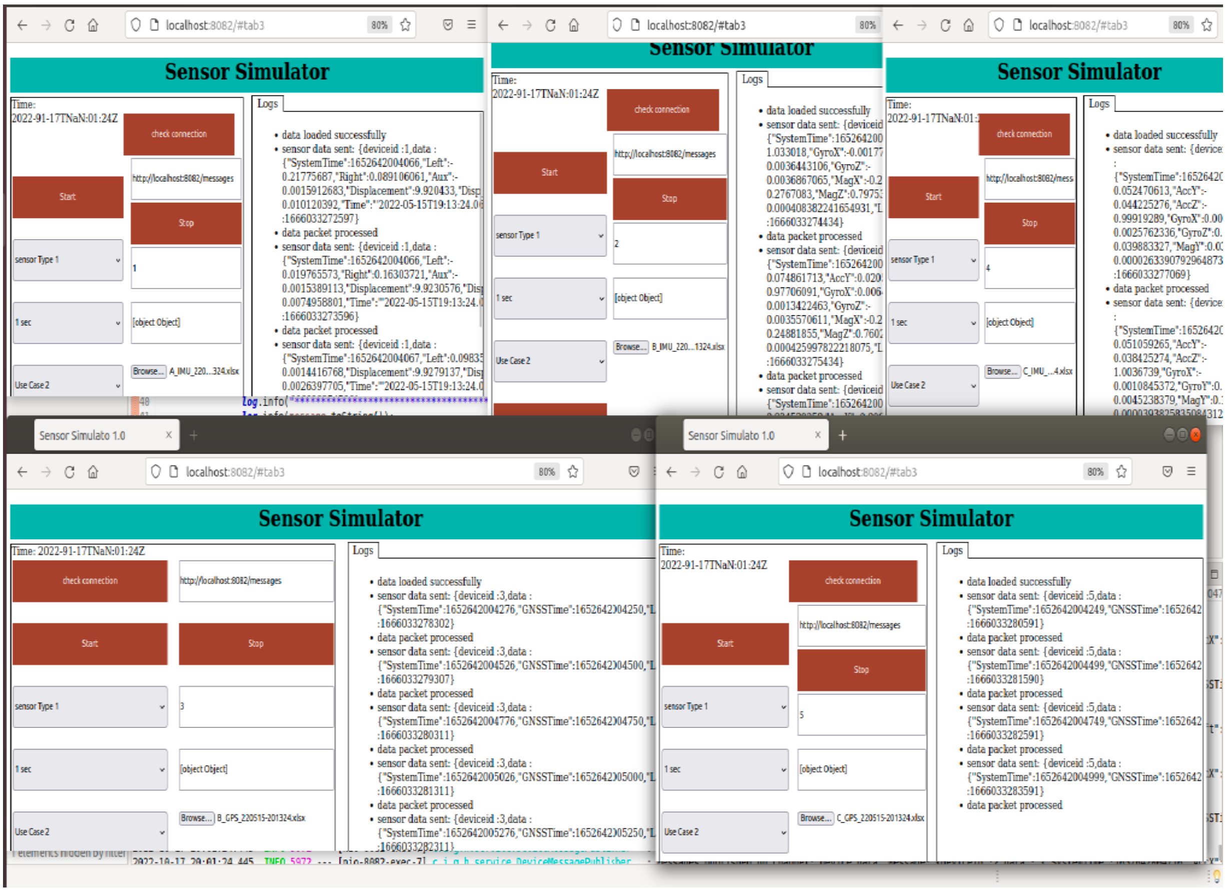

Figure 11: Sensor simulator replaying data for the second use case

The proof of concept implemented the full accounting model originally proposed in Deliverable 2.1, although some updates to the flow of transactions were made, primarily in response to review comments from that document (e.g. support for micropayments). This was particularly relevant in the case of live sensor data being requested in real time. Despite this, the core workflow of a transaction (Figure 12) is easily recognisable as that presented in Section 6.2, the creation of data offers by a data provider, the generation of an offer inquiry in response by one or more data consumers, the data transmission and Escrow process, and finally the generation and validation of agreements by all parties in the exchange.

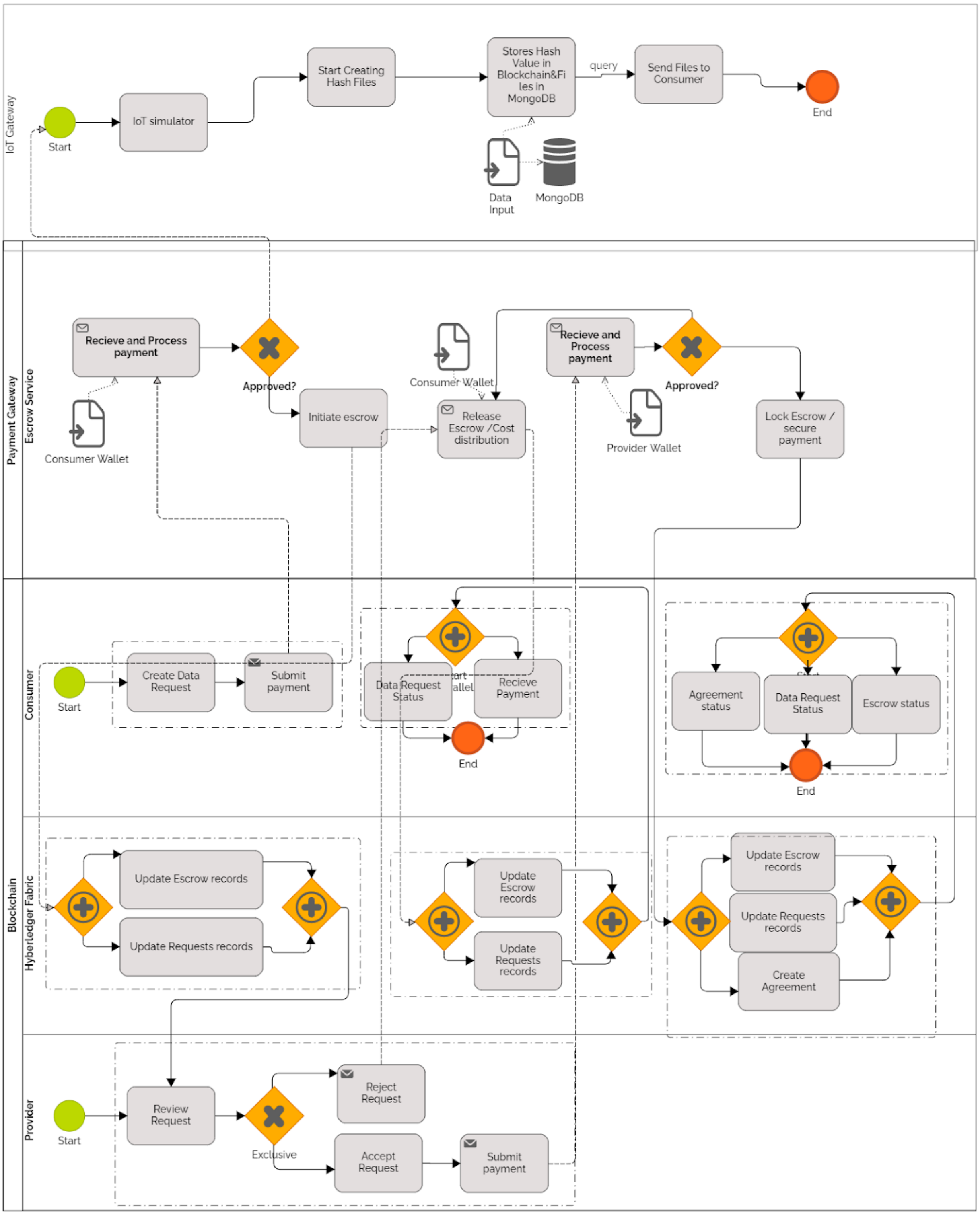

Figure 12: Flow of a generic transaction within the proof of concept

The development test harness for a third-party payment service was integrated into the proof of concept to demonstrate how account payments could be resolved within the mechanism of the data exchange Escrow (Figure 13).

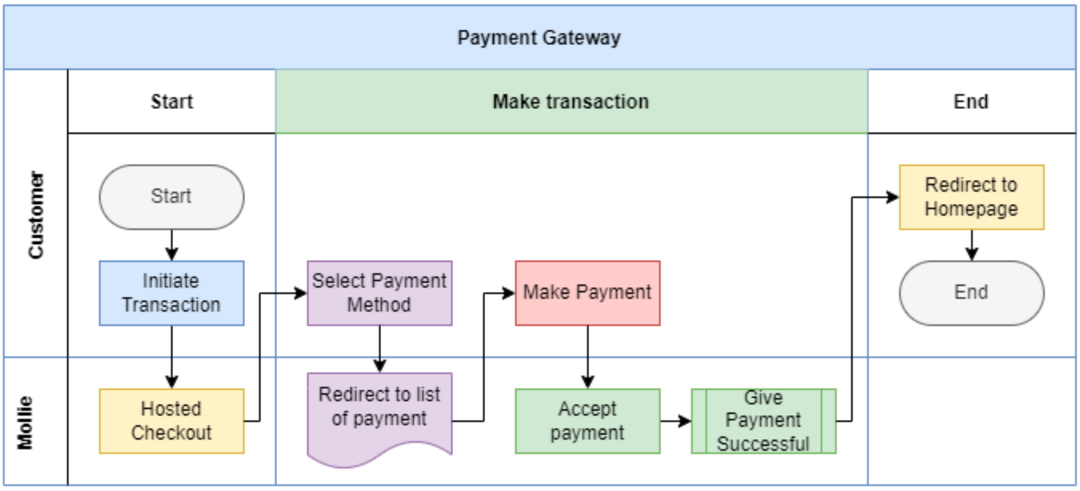

Figure 13: Process for resolving payments via a third-party card payment service

7.2 Walkthrough

Walkthroughs of the proof of concept were provided to evidence the delivery of this element of the work; these were in two forms, a set of screenshots that provided a convenient mechanism for reporting within the project deliverable document, and walkthrough videos that gave a more complete understanding of the system than is achievable in a set of static images.

A video introducing the use cases is available at https://youtu.be/0Du9G1q9YIc and a second providing the walkthrough of the proof of concept application is available at https://youtu.be/4zxwuRMODeo both were live at the time of project completion and will be maintained for as long as is practical beyond the lifetime of the project.

The walkthrough content covered the range of functionality required to complete a transaction within the system, from the generation of an offer by a data provider (Figure 14) and raising of a request (Figure 15), through to the completion of Escrow (Figure 16) and cost calculation ahead of release (Figure 17).

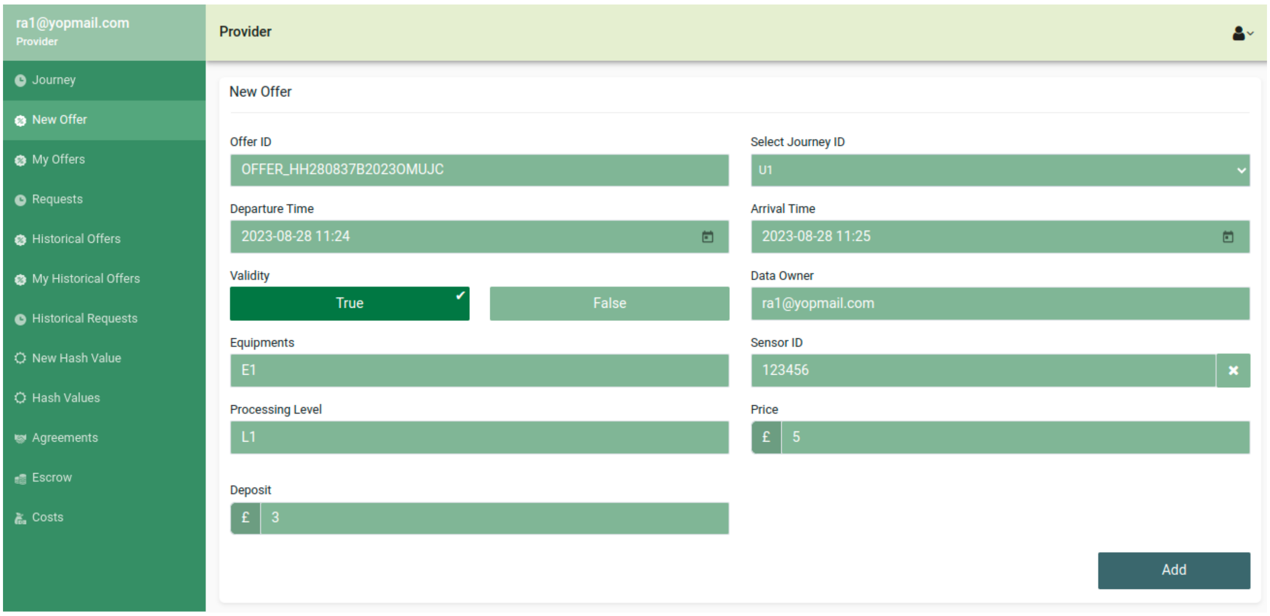

Figure 14: Creation of a data offer by a data provider

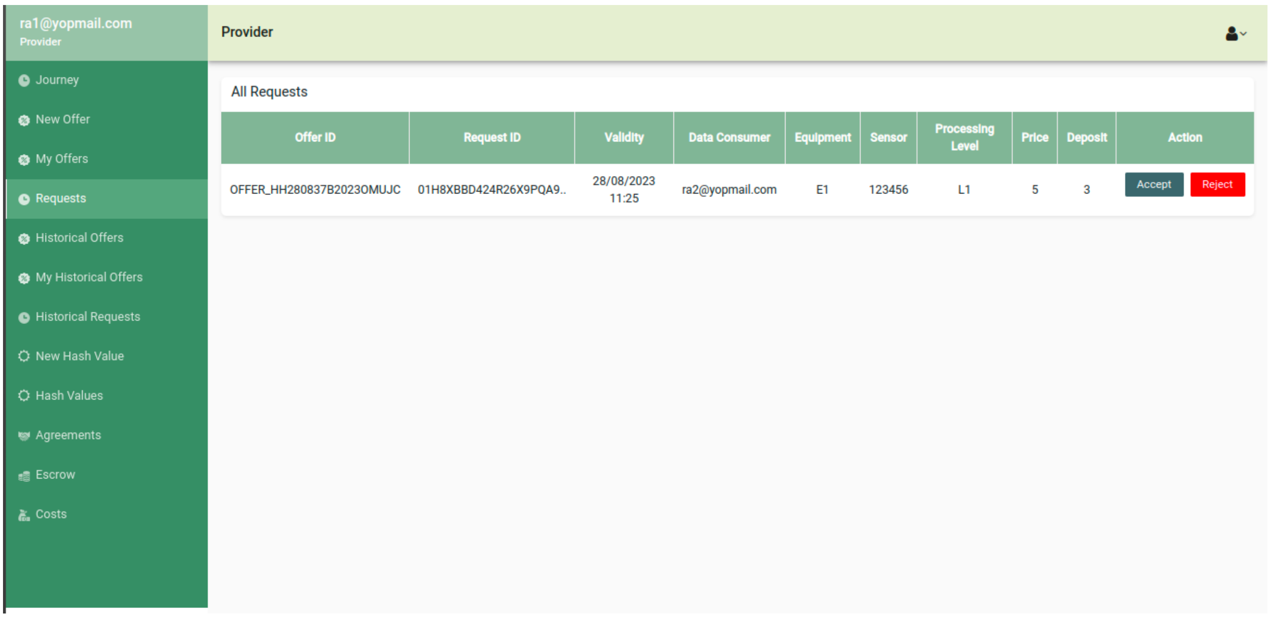

Figure 15: Provider view of a data inquiry / request issued by a consumer in response to an offer

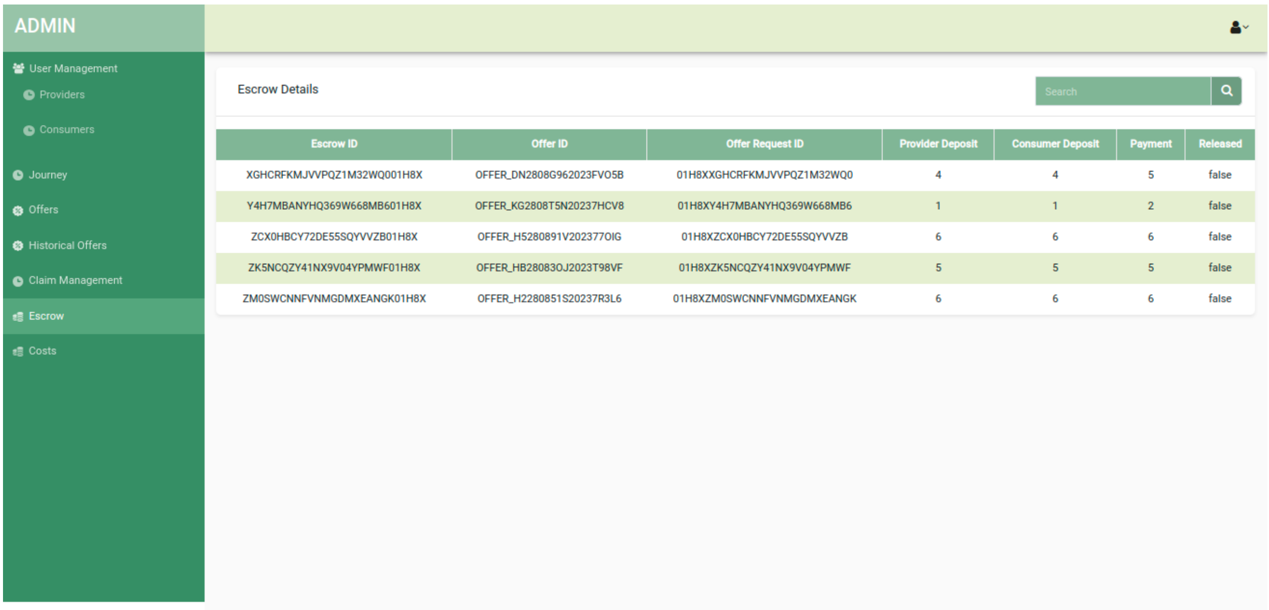

Figure 16: Review of active Escrow processes by an Admin user

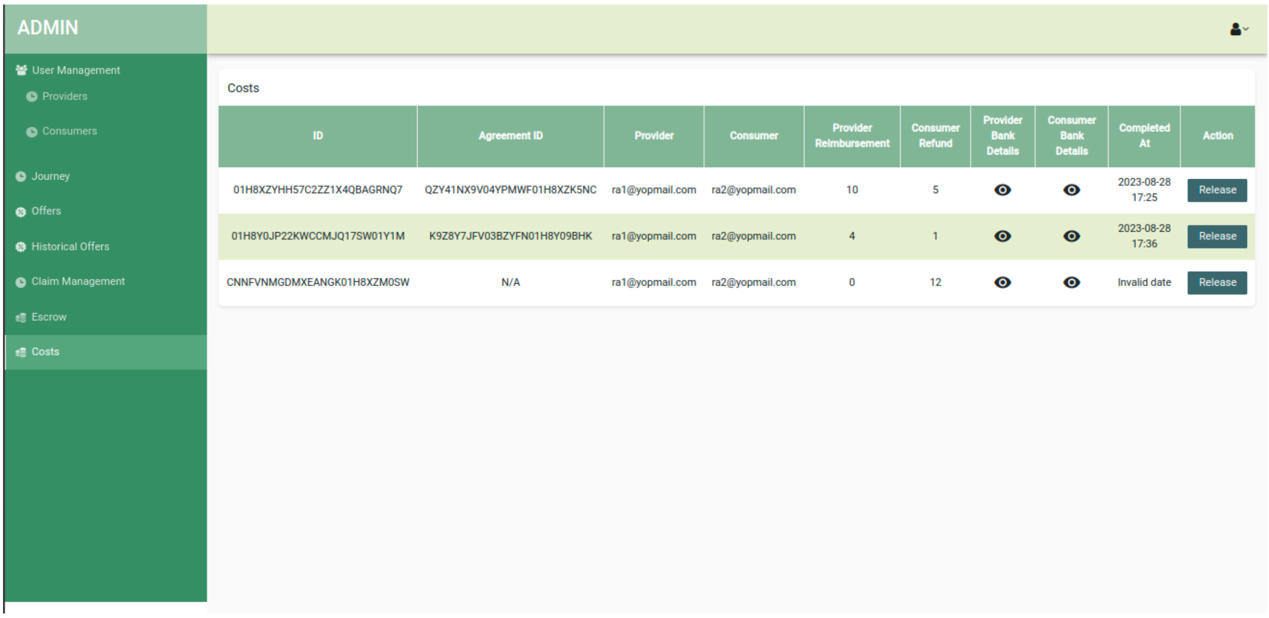

Figure 17: Record of data transaction "costs" within the network

8 Delivery of Planned Objectives

The B4CM project set out to deliver a set of six core research objectives. In this section of the report the team will revisit those targets and attempt to comment objectively on the extent to which they have been delivered.

Objective 1: To identify and develop use cases that support the application of blockchain in the railway sector

- Partially delivered: Throughout the project, the B4CM team have been working with representative industrial use cases to inform their decision making process. In Deliverable D1.1, case studies around axle bearing monitoring and the unattended inspection of overhead lines were analysed, with specific stakeholder mappings developed for the target systems and consideration of the wider regulatory context of the GB rail industry. These were then used to enable selection of an appropriate blockchain implementation model (private, permissioned) and implementation SDK (Hyperledger Fabric) for the core framework. Although the use cases were modified slightly by the time the team reached the proof of concept stage reported in Deliverable 3.1, primarily due to data access constraints, the first still focussed of axle bearing monitoring just using a different sensor solution, and the second had switched from the monitoring of the overhead lines to monitoring of the tracks. Critically, both of the use cases were still examples of the same "quadrants" of the rail condition monitoring space, with sensors on the infrastructure monitoring the health of the vehicle in use case 1, and sensors on the vehicle monitoring the fixed infrastructure in use case 2. The proof of concept platform has shown that the blockchain framework developed is capable of handing exchanges of the data from these use cases.

Objective 2: To develop an implementable blockchain framework for the attribution of data costs in systems crossing organisational boundaries

- Delivered: The B4CM team have produced and demonstrated a proof of concept blockchain framework that is capable of tracing such data exchanges. It should be noted that in the process of delivering the work it has become clear that the SDKs that are currently available for distributed ledger technologies still require a very specialised programming skillset that is not yet widely available. The transfer of the developed framework between systems / physical machines requires a complex configuration management task, and this raises questions over the capacity of industry engineers to adequately exploit distributed ledgers in their current form.

Objective 3: To evaluate mechanisms for the incorporation of the developed blockchain framework into the financial processes of the European rail sector

- Partially delivered: In developing the data access agreement and accounting models that underpin the B4CM framework, the team have reviewed and drawn on the existing industry state of the art for the deployment of templated contractual agreements for cross-industry condition monitoring projects (e.g. RSSB's project T1010). As a minimum, this activity brings the B4CM outcomes into line with the practices of other well-received initiatives in this area. The proof of concept makes use of a standard payment test harness to show that the underlying distributed ledger technology is compatible with / ready for APIs of the type the industry would employ in a full deployment of such a system. However, a gap remains in this area around formal understanding of the internal commercial structures of rail industry actors, and whether the assumptions of the project team around the ability, particularly in the case of arms-length government-funded stakeholders, to delegate authority for sign-off of small-value transactions to an automated system and whether this would limited the ability of its employees to access data / exchange data in a timely manner. This would certainly require further investigation before steps towards a more formal investigation/implementation of the technology could be delivered.

Objective 4: To develop a testbed, demonstrating the operation of the framework in the context of rail sector, enabling future developers to extend the tools produced based on a known working configuration

- Partially delivered: The B4CM proof of concept has been developed and successfully demonstrated through both the formal reporting channels of the project (Deliverable D3.1) and through short videos available on YouTube (links in Section 7.2 of this report). The framework is available via the project GitHub (see deliverable D1.1), however, as noted in the comments under Objective 1, it has become clear to the B4CM team over the life of the project that when using the currently available SDKs, transferring distributed ledger deployments between physical machines is an extremely complex task, and that complexity limits the extent of reusability of the framework in the way originally intended.

Objective 5: To disseminate the findings of the project and the lessons learned to influence best practice in innovation and technology uptake in a key and evolving field within the European rail sector

- Delivered: The B4CM team have produced journal (Deliverable 4.3) and conference (Deliverable 4.2) papers reporting on the progress of the work. Supporting these, the team have also produced project webpages (https://b4cm.co.uk) that will be maintained beyond the lifetime of the project, provided a twitter feed (@B4CM1), and presented to industry events including the Europe's Rail Innovation Day event in December 2022.

Objective 6: To support the development of a researcher in gaining a PhD and thus generating a skilled specialist valuable to the European rail sector

- Delivered: At the core of the B4CM project proposal was provision for a PhD studentship enabling the development of a new researcher in the domain. That student is now in the process of writing up her work with a view to submitting her thesis for examination in the spring of 2024.

9 Conclusions and Recommendations

Blockchains and other variants of distributed ledger technologies are a disruptive technology that have the potential to provide significantly improved traceability of the exchange of assets within industrial contexts. By providing an immutable, auditable record of transactions that is visible to all parties within an exchange, it will be possible to attribute costs and distribute revenues across stakeholders more fairly. This in turn should, intuitively, make it easier for businesses to invest in technology that has benefits to the railway as a system, but not necessarily demonstrating a significant benefit to their organisation as an individual actor within it.

The overarching aim of the B4CM project was to develop and deliver a blockchain-based testbed for the attribution of RCM data costs across organisational boundaries, and to demonstrate the operation of that framework and in the context of the European Rail Industry. The framework developed was to be made available to developers as the basis for future work in the area.

In this document, the closing report for the project, the B4CM team have summarised the work that has taken place, and commented on the degree to which the project has delivered against its specific objectives. Despite the challenges the industry (and indeed the rest of the world) has experienced in recent years due to the financial crisis, the Covid-19 pandemic, and Brexit, many of those outcomes can be regarded as a success. The team have demonstrated that it is possible to develop a blockchain based system to audit condition monitoring data transfers, and have shown how, through a combination of off-chain storage on on-chain hashes, that data can be accessed and verified by the data consumer without incurring the huge computational overhead that would be associated with directly storing the exchanged data on-chain. Awareness of the technology has also been raised within the industry, and the academic and technical publications authored by the team have begun to be reported in the context of other studies.

Despite this, there are areas where the expected impact at the time of preparation of the bid have not yet been realised. In the process of delivering the work it has become clear that the SDKs that are currently available for distributed ledger technologies still require a very specialised programming skillset that is not yet widely available; as such, the B4CM team believe that priority (in terms of industrial exploration of these technologies) should be given to those areas of the business for which the technology is a more traditional fit, e.g. ticketing, enabling an easier developmental journey and more targeted allocation of the limited human resource available. Other studies performed by members of the same team, e.g. the EU-funded STUB project[4], have shown that the route to implementation in these sections of the industry have greater potential to provide the “silver bullet” that will prove the business benefits and stimulate widespread adoption of distributed ledger technologies within rail. The technological limitations have impacted on delivery, this is particularly true in terms of the reusability of the framework, for which the deployment of a new instance is a significant undertaking if you’re not already very familiar with the underlying technology.

In conclusion, the B4CM team recommend that:

- While this work has shown that distributed ledger technologies do offer significant potential value to the rail sector, early adoption should focus on use cases that are more traditionally associated with the transfer of tokens, such as rail ticketing, in order to prove the business case and build staff capability within the industry;
- Once a sufficiently experienced staff base has been established within rail, initial industrial deployments of the technology around audit of data shadow transactions being made in other data exchange platforms, such as the GB Rail Data Marketplace, providing firm evidence of the applicability of the technology in a way that does not impact on mission critical activities. Only once this has been demonstrated, and performance at representative scale proven, will it be appropriate to move forward with live operational systems.

10 References

[1] Rail Safety and Standards Board, 'Detailed overview of selected RCM areas - Monitoring of axle journal bearings (T857 Report)', *https://www.sparkrail.org/Lists/Records/DispForm.aspx?ID=9916*. 2010.

[2] Rail Safety and Standards Board, 'Detailed overview of selected RCM areas - Monitoring of overhead line integrity (T857 Report)', *https://www.sparkrail.org/Lists/Records/DispForm.aspx?ID=9919*. 2010.

[3] S. Newcombe and G. Tucker, 'Enabling greater use of cross-industry remote condition monitoring', in *International Conference on Railway Engineering (ICRE 2016)*, Institution of Engineering and Technology, 2016, pp. 11 (8 .)-11 (8 .). doi: 10.1049/cp.2016.0520.

[4] J. D. Preece, C. Morris, and J. M. Easton, 'Towards STUB 2.0: Using Graph-Based World States in Hyperledger Besu to Facilitate Distributed Transport Ticketing', in *2022 IEEE International Conference on Big Data (Big Data)*, IEEE, Dec. 2022, pp. 3838–3844. doi: 10.1109/BigData55660.2022.10020309.