05 October 2023
ESMA12-2121844265-3182

Report on the DLT Pilot Regime

Study on extraction of transaction data

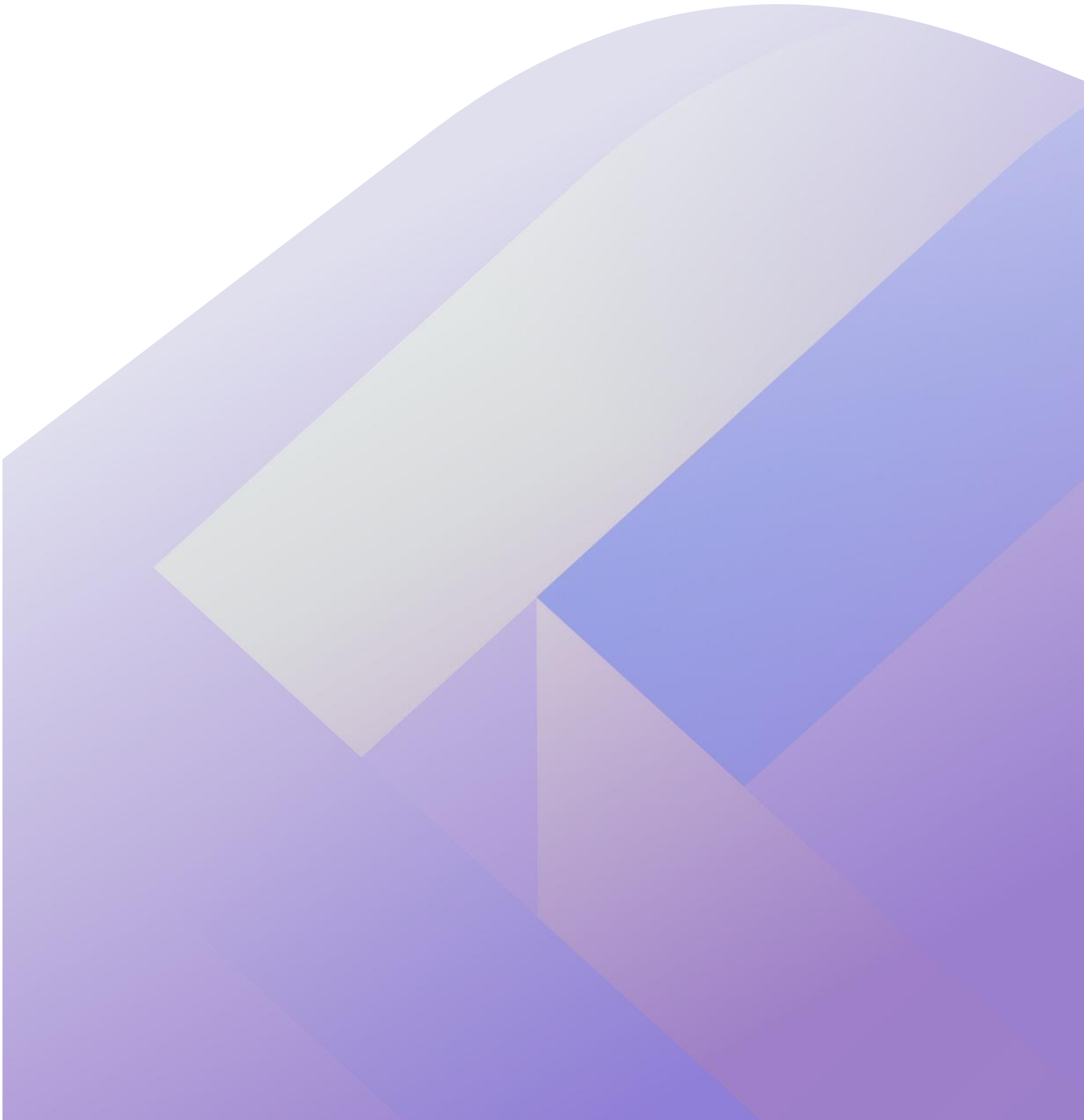


Table of Contents


ESMA
European Securities and Markets Authority

Acronyms used

API	Application Programming Interface
CA	Certificate Authority/Certification Authority
CCP	Central Counterparty
CPU	Central Processing Unit
CSD	Central Securities Depository
DBMS	Database Management System
DLT	Distributed Ledger Technology
DLTR	Regulation (EU) 2022/858 of the European Parliament and of the Council of 30 May 2022 on a pilot regime for market infrastructures based on distributed ledger technology, and amending Regulations (EU) No 600/2014 and (EU) No 909/2014 and Directive 2014/65/EU
DLT MTF	DLT Multilateral Trading Facility
DLT SS	DLT Settlement System
DLT TSS	DLT Trading and Settlement System
DTI	Digital Token Identifier
DTIF	Digital Token Identifier Foundation
EC	European Commission
EIP	Ethereum Improvement Proposal
EOA	Externally Owned Account
EOL	End of Life
ERC	Ethereum Request for Comment

ESMA	European Securities and Markets Authority
ETH	Ether
EU	European Union
EVM	Ethereum Virtual Machine
FX	Foreign Exchange
GB	Gigabyte
HLF	Hyperledger Fabric
IDE	Integrated Development Environment
IETF	Internet Engineering Task Force
ISIN	International Security Identification Number
ISO	International Organization for Standardization
KYC	Know Your Customer
MiFID II	Directive 2014/65/EU of the European Parliament and the Council of 15 May 2014 on markets in financial instruments and amending Directive 2002/92/EC and Directive 2011/61/EU
MiFIR	Regulation (EU) No 600/2014 of the European Parliament and of the Council on markets in financial instruments and amending Regulation (EU) No 648/2012
MSP	Membership Service Provider
MTF	Multilateral Trading Facility
NFT	Non-fungible Token
NCA	National Competent Authority
PII	Personal Identifiable Information
PoC	Proof-of-concept

PoS	Proof-of-stake
PoW	Proof-of-work
RTA	Registered Transfer Agent
RTS	Regulatory Technical Standard
RTS 22	Commission Delegated Regulation (EU) 2017/590 of 28 July 2016 supplementing Regulation (EU) No 600/2014 of the European Parliament and of the Council with regard to regulatory technical standards for the reporting of transactions to competent authorities
SDK	Software Development Kit
SEC	United States Securities and Exchange Commission
SHA	Secure Hash Algorithm
SS	Settlement System
TVTIC	Trading Venue Transaction Identification Code
UCITS	Undertakings for Collective Investments in Transferable Securities
URI	Uniform Resource Identifier
UTC	Coordinated Universal Time
XML	Extensible Markup Language

1 Executive Summary

Reasons for publication

The DLT Pilot Regime (DLTR) entered into force on 23 June 2022 and aims to foster innovation in the European Union's capital markets sector. It allows eligible firms to operate DLT market infrastructures to be used for trading and settlement purposes. A survey conducted during the ESMA workshop on the DLTR on 31 March 2022, identified three main DLTs (Corda, Ethereum, and Hyperledger Fabric) that might be used by DLT market infrastructures. The three DLTs are analysed in this study with respect to transaction reporting.

Transaction reporting plays a crucial role in current financial markets as it provides regulators with insights into market movements and trends as well as overall market stability. The objective of this study is to understand the implications of the use of DLT/blockchain in the context of transactions in financial instruments when an exemption to Article 26 of MiFIR is granted to a DLT market infrastructure. To do so, DLT developers, potential DLTR applicants from various European jurisdictions, and other stakeholders were interviewed to gain a better understanding of applicable data storage approaches and, more generally, transaction data produced by the DLT transactions. This practical knowledge was supplemented with theoretical knowledge gained by reading and analysing the DLTs' respective official documentation.

This document has been prepared for the European Securities and Markets Authority (ESMA) by PwC EU Services EESV (PwC). It reflects the views only of its authors, and the European Securities and Markets Authority is not liable for any consequence stemming from the reuse of this publication.

Contents

Sections 3.1, 3.2 and 3.3 take a closer look at approaches to extract data from the three DLTs analysed. In doing so, necessary background regarding the DLTs is outlined, considering certain specificities and appropriate use cases. Further, the sections dive into potential architectural designs for extraction of all relevant details of financial instrument transactions processed by DLT market infrastructures to provide regulators with all required details. The sections therefore explore the file-based approach, the API-based approach and the approach in where regulators obtain native access to each DLT and DLT network. Each of the approaches is compared based on its costs and related benefits within the cost-benefit analysis conducted in Section 3.4.

Additionally, recommendations about potential regulatory relevant information to be further included for market surveillance purposes as well as on-chain analysis is given under Section 3.5.

Lastly, Section 3.6 introduces possible scenarios in where on-chain analysis might be relevant and presents existing tools and software to conduct on-chain analysis.

The general conclusion of the study can be summarised as the file-based approach being the most cost-beneficial approach in combination with the complementation of the RTS 22 XML schema with relevant information, to provide regulators with all the necessary information on DLT financial instrument transactions.

2 Introduction to the DLT Pilot and MiFID II/MiFIR transaction reporting

1. On 23 June 2022, the Regulation (EU) 2022/858 on a pilot regime for market infrastructures based on distributed ledger technology (DLT) ("the DLT Pilot Regime", "DLTR") entered into force. As part of the Digital Finance Package of the European Commission (EC), it furthers innovation and competition in the capital markets sector as it allows eligible firms to operate DLT market infrastructures.

2. Three types of DLT market infrastructures exist as part of the DLT Pilot. These are DLT MTFs, DLT SSs, and DLT TSSs. A DLT MTF is defined as a multilateral trading facility (MTF) which only admits to trading DLT financial instruments. A DLT SS, on the other hand, is a settlement system (SS) only settling transactions in DLT financial instruments against payment or delivery. A DLT TSS combines the services performed by a DLT MTF and a DLT SS.[1] To operate a DLT market infrastructure, firms must apply with their National Competent Authorities (NCAs).

3. The focus of this study was on the trading side of DLT MTFs and DLT TSSs, which will be referred to as DLT market infrastructures for the purposes of this report. Among other conditions, they are subject to the requirements that apply to multilateral trading facilities (MTFs) under Regulation (EU) 2014/600 (MiFIR) and Directive (EU) 2014/65 (MiFID II). However, the NCA can exempt the DLT TSS from some of said requirements, if it complies with the conditions listed in Article 4 of the DLTR.[2]

4. Specifically, the DLT TSSs may permit natural and legal persons to deal on own account via their systems, granted they fulfil a variety of requirements. In that case, additional measures to protect such participants may be required. Additional measures ought to be proportionate to the participants' risk profiles. Moreover, DLT TSSs may be granted exemptions from transaction reporting requirements under Article 26 MiFIR. Should this exemption be granted, the DLT TSSs must nevertheless keep records of all transactions executed and further ensure that NCAs entitled to receive said data have direct and immediate access to it.

5. Initially set out for a duration of three years, the DLT Pilot will enable the trading of DLT financial instruments on DLT TSSs in the European Union, with the aim to stimulate innovation in the sector while guaranteeing investor protection. Annually, ESMA shall publish interim reports providing information on the functioning of the markets but also to provide clarifications on the Regulation's application. Following the initial three-year period, ESMA will present a report to the European Commission covering, among other things, the number of DLT market infrastructures, an overview of DLT financial instruments admitted

[1] https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX%3A32022R0858#d1e717-1-1
[2] https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX%3A32022R0858#d1e1048-1-1

to trading and recorded, as well as an overall assessment of the DLT Pilot's costs and benefits. Furthermore, a recommendation as to whether and how the regime will be continued ought to be made.[3]

6. Over the course of its initial three-year period, the DLT Pilot imposes certain restrictions, for instance, on the financial instruments it covers. More precisely, it encompasses shares, bonds, and UCITS, which are subject to further thresholds pertaining to, among other things, the issuer's market capitalisation. Operators of DLT market infrastructures shall activate their respective transition strategies should the aggregate market value of all DLT financial instruments admitted to trading or recorded on that infrastructure exceed EUR 9 billion.[4]

7. MiFID II/MiFIR and their legal framework aims to protect investors in financial markets, while providing market transparency and functioning as a harmonised set of financial regulation in the European Union (EU). Overall, 28 Regulatory Technical Standards (RTS) are in place ranging from organisational requirements to disclosure obligations and the reporting of transactions.[5] More precisely, Delegated Regulation (EU) 2017/590 (RTS 22) prescribe how transactions are ought to be reported in a consistent and standardised format to NCAs to enables said NCAs to be able to analyse the reported data effectively.

8. Article 2, Paragraphs 2(a) and 3(a) of the RTS 22 respectively define a transaction as the conclusion of an acquisition or disposal of a financial instrument.[6] Such RTS 22 transactions must be reported no later than the close of the following working day and entail complete and accurate details regarding the nature of the financial instruments acquired or disposed of. In total, the current RTS 22 transaction reporting logic consist of 65 fields inquiring about information regarding the buyer and seller of a financial instrument, its details and transmission, as well as specific transaction details including quantity, trading date time, and price.[7]

9. It might be considered, to extend the RTS 22 fields for use-case specific fields, derived from the DLT transaction flows, and specified in the Smart Contract between the participants. Therefore, consensus on the mandatory fields needs to be reached. The extension of fields within the RTS 22, specifically for the DLT Pilot Regime, is further described in the "Study on how transactions are registered in various blockchain solutions".

[3] https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX%3A32022R0858#d1e2493-1-1
[4] https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX%3A32022R0858#d1e966-1-1
[5] https://ec.europa.eu/finance/securities/docs/isd/mifid/its-rts-overview-table_en.pdf
[6] https://ec.europa.eu/finance/securities/docs/isd/mifid/rts/160728-rts-22_en.pdf
[7] https://ec.europa.eu/finance/securities/docs/isd/mifid/rts/160728-rts-22-annex_en.pdf

3 Methods of extracting data from selected Distributed Ledger Technologies

10. The purpose of this study to gain some understanding of the specificities of the DLT to enable ESMA to have informed discussions on the topic in the context of subsequent market consultations on a broader set of DLT. The three selected DLTs were considered as a good starting point for the analysis to ensure that the study could be delivered within the required timeframes.

11. Three DLTs, Corda, Ethereum, and Hyperledger Fabric, were selected to be analysed as part of this study. These DLTs were solely identified based on a survey launched during the ESMA workshop on the DLT Pilot Regime held on 31 March 2022. The participants in the workshop were individuals, or firms' representatives, who had responded to the ESMA Call for Evidence regarding the DLT Pilot Regime.[8] Neither ESMA nor PwC are endorsing any of these DLTs or the softwares used to perform the study.

12. This section describes three different approaches for extracting data from the DLT. The first approach is the "File-based approach" sharing data in files either via an sFTP server (similar to current market practice for other transaction reporting regimes) or via a shared drive made available by the regulators' software.

13. The data is organised in a specific format, such as text files or other formats. The common file-based format used for financial instruments transaction reporting is "Extensible Markup Language (XML)". The data inside XML file format is organised in tags, where data is placed between a start and an end tag. The structure of the XML document is defined by a schema that defines the rules for organising the data and the relationships between different elements.

14. The second approach is the "API-based approach" using an application programming interface (API) to facilitate communication between different software systems. An application programming interface is a set of rules and protocols that define how software components interact with each other, what types of requests can be made, and the data formats that are used.

15. An API is also considered an intermediate between different software programs, allowing them to communicate with each other. In this approach, one system (such as a web application) can access the functionality of another system (such as a database) by making requests to the other system's API. This allows the different systems to interact and share data with each other in a standardised way. APIs are commonly used in modern software development to enable integration between different systems and services.

[8] https://www.esma.europa.eu/press-news/consultations/call-evidence-dlt-pilot-regime

16. The third approach is the so called “native access approach”. Within this transaction reporting approach external stakeholders, such as regulators, are directly involved in each of the DLT networks in which financial instruments are traded. Involvement may be denoted in various ways, dependent on the nature of the DLT’s IT architecture, but always expressed through an active usage of the DLT’s functionalities.

3.1 Corda

3.1.1 Background

17. Corda is a permissioned, peer-to-peer DLT by technology provider R3 HoldCo LLC (“R3”).[9] It aims to optimise existing processes in regulated markets and has hence found widespread application in the financial markets sector. Due to its set-up as a permissioned DLT, information is shared between parties on a need-to-know basis. This means uninvolved third parties generally cannot access the stored information as there is no global broadcast of all Corda transactions. This allows for increased data and Corda transaction privacy with involved parties being able to flexibly structure Corda transactions and deciding which data they intend to make available to third parties.

18. Peers on Corda are best defined as network nodes owned and operated by specific parties that have unique identities. On the Corda DLT, nodes are typically operated by legal persons rather than natural persons. This is due to the costs and complexities associated with node set-up and the nodes’ continuous operation. Nodes validate Corda transactions, and each maintain their own copy of the ledger, ensuring the DLT’s security and integrity. Peers can communicate and enter Corda transactions with another, essentially making the peers the network’s participants. These activities are enabled in part by making use of X.509 certificates[10]. The X.509 certificate format was developed by the Internet Engineering Task Force (IETF) in 1988 and finds widespread application in networking and security protocols.[11]

19. To keep the Corda network secure, X.509v3 certificates are issued to the peers, i.e., the network participants, upon joining the network. This is done by a network component called certification authority (CA), which is typically operated by a trusted entity. The CA could be the initiator of the network or an organisation that possesses knowledge on how to technically operate network infrastructures. CAs occupy a crucial role in maintaining Corda’s security as they help ensure that network participants are correctly identified and thereby facilitate, for instance, the sending of Corda transactions. The certificates

[9] https://r3.com/products/corda/
[10] X.509 is a commonly used standard for public key infrastructure. X.509 certificates are used to bind an identity to a digitally signed public key. Among other things, they further contain information as to the certificate’s issuer and its validity period.
[11] https://sectigo.com/resource-library/what-is-x509-certificate

themselves then include, among other things, information on participants' public keys or their names.

20. States are further essential components of the Corda DLT. They are best defined as immutable objects stored on Corda's ledger and representing shared facts between network participants. A state's immutability ensures that it can neither be modified nor deleted once it is created. [12]

21. When such a fact changes, however, a new state is created by one of the network participants. To do so, the previous state serves as an input. Using a state as an input in a new Corda transaction is also called "consuming" a state. Once a state is consumed, it is marked as historic. Hence, an input state can be used to generate one or more new output states.

22. Current as well as historic states are stored in so-called vaults. Vaults are maintained by the network nodes. Every network node will thus have stored all applicable states, consumed and unconsumed, for the Corda transactions in which the node has been a participant.[13] Hence, Corda does not have a central ledger recording facts for all network nodes but rather individual nodes storing the data known to them.

23. Corda further utilises so-called smart contracts, which digitise and enforce agreements entered into between various network participants. Smart contracts can be designed and implemented in a variety of ways to reflect and support the use cases they are applied to. Typically, they are implemented to put certain constraints on how states will evolve over the course of their lifetime. For instance, it can be specified that the coupon payment on a bond must remain at 2% of its notional amount over the course of its lifetime.

24. A further example is the transfer of a DLT financial instrument in exchange for e-money tokens between Party A and Party B. In such a scenario, the smart contract checks the balances of the two contracting parties to ensure Party A possesses the DLT financial instrument and Party B possesses a sufficient number of e-money tokens to purchase it. This can be done by the smart contract querying the vault and assessing the respective balances of the parties or by checking the history of all Corda transactions the parties have been involved in. The correctness of this data is ensured, for instance, through the technical notarisation of Corda transactions the parties have been involved in. Upon the success of the check, the smart contract automatically executes the trade by transferring both the DLT financial instrument and the e-money tokens to their new owners, which are Party B and Party A respectively.

25. Another essential Corda component are the so-called flows. Corda provides a set of built-in flows that assist in automating tasks that commonly occur on the network. Flows can automate processes related to initiation, verification, or notarisation of a Corda transaction.

[12] https://docs.r3.com/en/platform/corda/4.8/enterprise/key-concepts-states.html
[13] https://docs.r3.com/en/images/vault-simple.png

More generally, flows enable inter-node communication. By facilitating communication between network participants, flows play a decisive role in coming to agreements regarding ledger updates.

26. In the financial sector, the Corda DLT has been used for a variety of activities, such as certain kinds of bond swaps or inter-custodian swaps between large financial institutions.[14]

27. This report takes into consideration Corda's Open Source Version 4.8.

28. The following sections explain components described above in more detail.

3.1.1.1 Network Map Service

29. The network map service is Corda's identity service and matches each node identity to an IP address. Nodes are identified by their IP addresses and use these to establish communication between each other. Generally, nodes are not operated by natural persons on Corda due to a variety of reasons. For instance, costs associated with node operation and the accompanying effort of their implementation and infrastructure set-up tend to be high making such an endeavour unrealistic. Theoretically nodes can be provided by other network participants such as DLT market infrastructures. Such nodes can be used by natural persons to access the network and engage with it.

30. Regarding individual transactions that must be conducted under boundaries of privacy principles, Corda offers the creation of confidential identities. This guiding principle distributes the certificate which is used to identify a node or a legal entity only on a need-to-know basis. The confidential identities are used in cases where attackers gain access to transactions and protect entities that are operating nodes within a Corda network from identification.

31. Furthermore, the network map service is Corda's mechanism to allow nodes to discover each other within the network. For said discovery purposes, the Network Map Service publishes a list of the peer nodes that exist as well as their metadata. Network participants can make use of this information to derive who else is involved within the network and the services they offer.

3.1.1.2 Nodes

32. A Corda node is best understood as a virtual machine running the software provided by Corda. Every node is represented by a unique identity and represents a legal entity. Due to cost of operation and various other efforts such as maintenance and IT security, it is uncommon and not intended for natural persons to operate Corda nodes.

[14] Bank consortium Fnality International and Luxembourg company HQLAx successfully completed the first proof of concept (PoC) of a cross-chain repo swap settlement between Enterprise Ethereum and Corda.

33. Furthermore, a Corda node has two components. A network interface and a Remote Procedure Call (RPC). The network interface is used to establish communication with other nodes in the network, while the RPC interface enables users to interact with a node. Hence, the RPC interface enables vault queries, which essentially are calls to the RPC interface.

3.1.1.3 Visibility

34. Visibility refers to the ability of a node to access and view the data that is stored on other nodes and the ability to view the data that is shared between two or more nodes. In Corda, visibility between nodes is typically restricted to protect the privacy and security of the data and to ensure that only authorised parties have access to it. Restricting access in Corda is applied to either a specific node or a group of nodes.

35. The visibility of data is implemented by following access rights granting mechanisms. Each node has access to a subset of facts. Implicitly, their own data as well as the data shared with others. By following this concept, within the Corda network, at no time, the entirety of the ledger is visible to a single node. Nevertheless, it is technically possible to assign a single node with the ability to view the entirety of the ledger.

36. Figure 1 shows an example of a potential implementation between five nodes within a Corda network. While **Node A** and **Node B** share the same information, **Node A has no access** to the information shared between **Node B and Node C**. **Node A, Node B as well as Node C have no access** to the information shared between **Node E and Node D**.

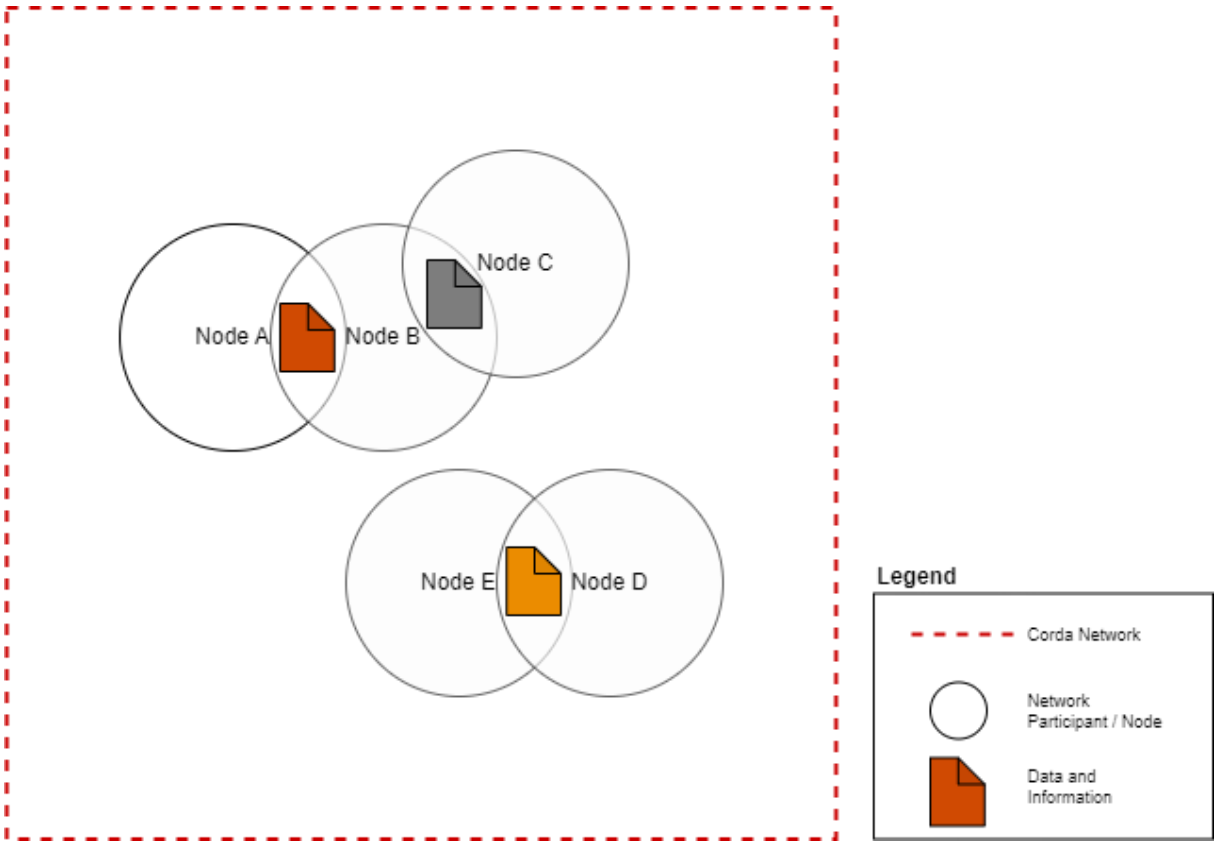


FIGURE 1: CORDA NETWORK VISIBILITY IMPLEMENTATION

3.1.1.4 Notary service

37. The notary on Corda is best understood as the network's consensus service, as it aims to prevent double spending and ensure the unique usage of input states. In the case where a participant has visibility on most transactions and/or is participating in a large amount of transaction flows, this participant can get corrupted. Therefore, the aim of the network's consensus service is also to minimise transaction tampering.

38. From an IT security perspective, there always remains the risk of tampering in any technology. Corda networks are flexible as they may consist of a single notary service or a multitude of services, so-called notary clusters. Notary clusters may have a positive impact on overall network performance and security, which will be explored further later in this chapter.

39. Notaries, or notary clusters, represent a network's final point. At this point, Corda transactions are either validated and signed as correct or rejected, deemed incorrect, and flagged. The notary's', or the notary clusters' signature is obtained, if a successful

verification of the proposed transaction's input states has occurred and it is confirmed that said input states have not already been consumed by way of a previous transaction.

40. Hence, notaries are essentially tasked with assessing a transaction's uniqueness. In a network architecture which implements a single notary, all transactions are validated by that single notary component. In productive networks, however, this is rarely the case and an implementation of several notaries within a notary cluster is recommended. As already described, a notary merely verifies the validity of the transaction by checking if a state has already been used and spent. It has no further insights regarding any transaction data. The below example in Figure 2 shows how a single notary may be implemented in a Corda network.

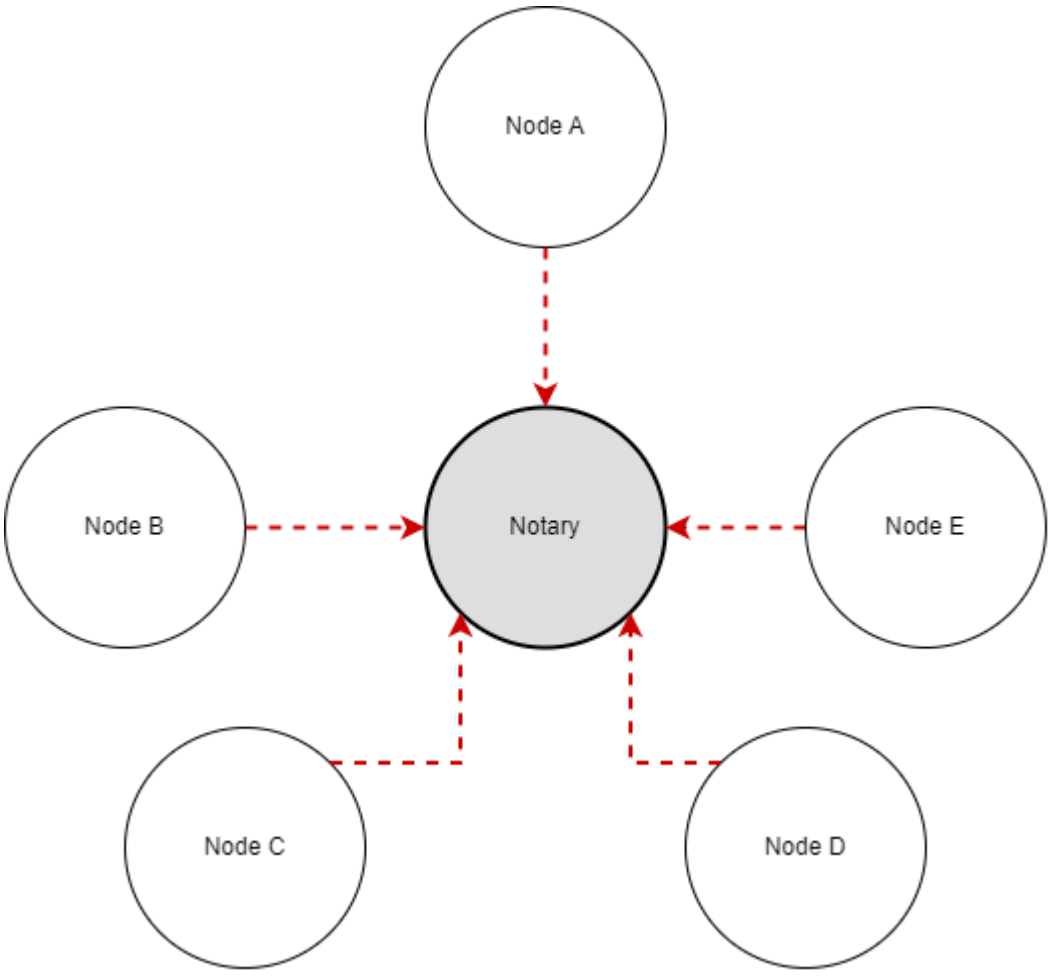


FIGURE 2: SINGLE NOTARY IMPLEMENTATION

41. Should a single notary find itself in a situation where many incoming transactions have to be processed and verified in a short amount of time, the notary might become a bottleneck to the smooth functioning of the network. This can potentially lead to transactions not being

processed as they should, hence affecting network efficiency. Implementing notary clusters remediate such operational issues due to their ability to achieve a higher transaction throughput resulting in an improved performance. This is shown in the below example.

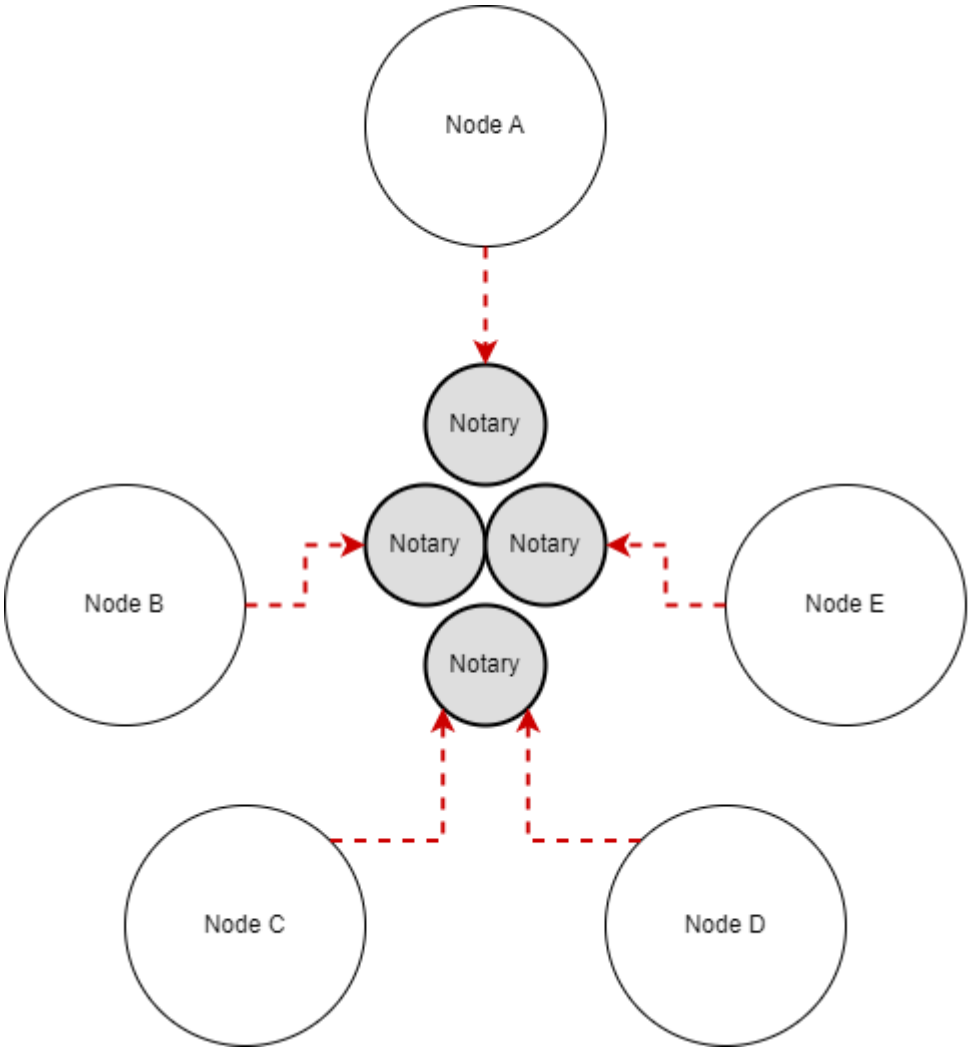


FIGURE 3: MULTIPLE NOTARY IMPLEMENTATION

42. On Corda, every state can have an appointed notary or notary cluster. This set-up ensures transaction privacy and security by only allowing these parties to validate a certain Corda transaction. Such a set-up becomes especially sensible in cases where notaries are tasked with validating transactions between nodes not sharing the same data. By separating validating notaries (able to validate transactions and observe transaction data) from non-validating notaries (which only validate the state and have no ability to observe transaction data), notary clusters ensure the privacy of transaction data during the validation process in the network.

3.1.1.5 Vault

43. The vault stores all data from the ledger that is relevant to a node. It is implemented as a table in the node's underlying SQL database to track transaction states. States that are stored inside of the vault are categorised in two different categories, unconsumed (or unspent) states and consumed (or spent) states.[15]

44. Unconsumed states represent states, which can be spent, changed or transferred to another participant.[16]

45. Consumed states represent immutable states, which cannot be changed. Consumed states are used for transaction reporting, auditing, and archiving.[17]

3.1.2 Approach methodology

46. As mentioned in 3.1.1.2, natural persons typically are not part of a Corda network due to the costs and complexities associated with node set-up and the continuous operation of nodes. For this chapter, therefore, natural persons were not considered. Rather the below approaches assume legal entities, such as financial institutions, to operate the nodes participating in the network.

47. Further, per Article 4(3) of the DLT Pilot Regime, DLT market infrastructures are required to keep records of transaction data at the disposal of the regulator. Hence, the below approaches - illustrate approaches in which this requirement is implemented with periodic reporting of the data subject to the record keeping obligation to the regulator via the DLT market infrastructure. Due to the implementation and operation effort, it is not common for private persons to operate nodes within a Corda DLT network and participants are likely to be institutions, such as investment firms, banks, and others. The participants which are part of a network depends on the intention of the business case which the network follows.

3.1.2.1 File-based approach

48. To extract said transaction data, the DLT market infrastructure must query the vault first and process the derived information in a specified file format. For instance, the file could be processed into XML format and adapt to the RTS 22 schema, or a new schema covering RTS 22 data and additional fields specific to the DLT (e.g DTI, transaction hash and wallet addresses), under the MiFID II/MiFIR transaction reporting regime. Such an approach would result in a single-sided transaction reporting and thus lead to a reduction in the overall effort required within the network.

[15] https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-vault.html
[16] https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-vault.html
[17] https://docs.r3.com/en/platform/corda/4.10/community/key-concepts-vault.html

49. As per regulation, all reported transactions from DLT market infrastructures must be registered and made available to regulators. Further, transactions that may be grouped by market participants and involve the DLT market infrastructure must be ensured to be part of the record keeping. Therefore, DLT market infrastructures must ensure necessary governance functionalities and smart contract implementations, so that all individual transactions can be identified by the regulators.

50. It is important to mention, that nothing prevents participants within a network from implementing their own transaction flows and settling a DLT financial instrument without integrating the DLT market infrastructure in the process. However, the implemented smart contract in the respective network cannot be tampered with. Such attempted transactions will be visible on the network and end up invalid.

51. Corda is a private DLT, meaning network participants must agree on a governance and certain conditions and functionalities that are implemented in the network through a smart contract. Should any tampering on the network occur, said tampering is traceable back to a specific node, and the participant operating it. Therefore, the incentive to tamper in a private DLT such as Corda, is negligible.

52. Since all transactions in financial instruments within a network are processed via a DLT market infrastructure, said DLT market infrastructure could be tasked with the obligation to gather the necessary transaction data and process it into the specified file format. This reduces the overall effort for network participants to process and report transaction data to a minimum. The below illustration (Figure 4) describes the exemplary process of the file-based approach, in which the DLT market infrastructure makes the data available to the regulator with a report of transaction data.

53. Following this approach, transactions occurring between participants are stored in the DLT market infrastructure's vault. To extract the accompanying transaction data, the DLT market infrastructure must query its own vault and parse the extracted data into a format requested by the regulator, such as the underlying XML structure of the RTS 22 schema and provide it to the regulator. More precisely, transactions between the participant operating Node A and the participant operating Node C, both for instance financial institutions, are processed via the DLT market infrastructure and the transaction data is stored in the vault of each participant involved in the pre-described transaction flow (Node

A to Node C via DLT market infrastructure.)

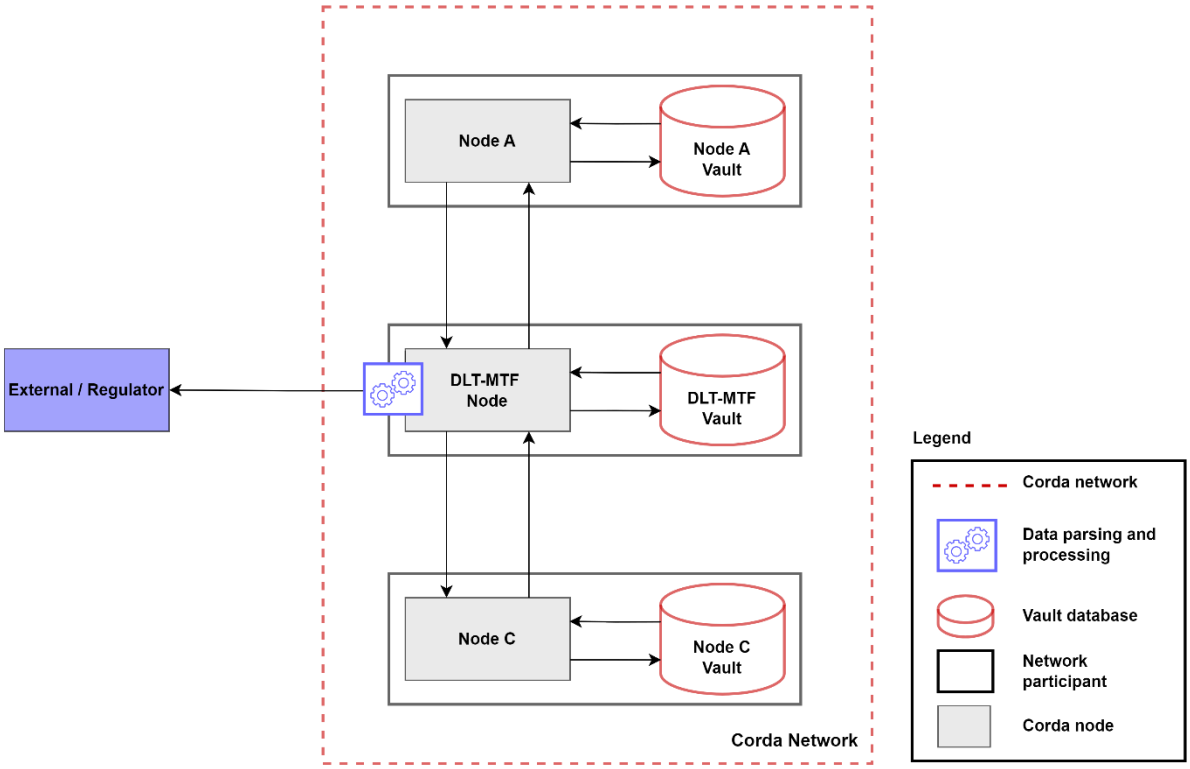


FIGURE 4: FILE-BASED TRANSACTION DATA EXTRACTION BY THE DLT MARKET INFRASTRUCTURE

a) Advantages

54. The file-based approach in which DLT market infrastructures fulfil the obligation to query their vaults and report the conducted transactions (Figure 4) is cost efficient, as there is no effort, to implement additional software and system components, such as an API on top of the extraction of the transaction data. Further, there is no running cost to maintain and operate the systems. Although, costs exist for processing the incoming XML data from DLT market infrastructures in order to be stored on the regulator's internal systems.

55. From an industry's perspective, already in place systems for file-based reporting can be used, which reduces the complexity of this approach.

56. Unlike other subsequent approaches, the file-based approach avoids the implementation of further interfaces and communication layers, such as APIs. Consequently, the major advantages for incumbent DLT market infrastructures that are already authorised under MiFIR to operate in traditional financial markets and regulators lie in the traditional mechanisms and systems already in place that can be used for the transformation of the transaction data.

57. In particular, the already supported standards such as the ISO 20022 XML schema, file encryption- and transmission mechanism within the same secured channels can be used. In addition, and mainly from a regulators perspective, the implemented components for file encryption and secure transmission stay loosely coupled from the DLT market infrastructure's implementation and don't result in dependencies towards systems and components provided by the DLT market infrastructure.

58. Furthermore, the file-based approach is similar to the currently implemented processes within the traditional world of transaction reporting. Therefore, no significant transformation activities, such as change management, are necessary. A specific aspect to the processes is the access to the latest status of transactions and access to the visibility of corrections and cancellations. By applying a file-based approach, no implementation of new processes is necessary since the current daily reporting process can be followed.

59. From both, the regulator's and DLT market infrastructure's perspective, the major advantage lies in the usage of a standard to interact during the overall transaction reporting process. Following the standard of file-based transaction reporting, in the form of the ISO 20022 standard and the RTS 22 XML schema, ensures an interoperable and stable transaction reporting process.

60. Compared to today's regular transaction reporting with T+1 or T+2 reporting, the file-based approach could, if necessary, provide a higher frequency of file generation. This can be done by firstly querying vault data and secondly transforming it into the requested schemas and files.

61. A further advantage, mainly from regulatory perspective, lies within governance implications of the file-based approach. Regulators keep the control of the standard XML schema, agree, and decide on updates, and control the timing of the implementation of the updates to give stakeholders time for implementation.

b) Disadvantages

62. Due to the architectural design of the Corda DLT, DLT market infrastructures can provide extracted data from the vault within reports in a near-time sequence. This possibility mainly exists by applying the API-based and native access approaches and is mentioned under this section. Although, it is not a disadvantage in a sense of weakening or threating the file-based approach, since near time reporting is not in line with current practice of file-based transaction reporting.

3.1.2.2 API-based approach

63. Extracting data as an external party by consuming an API, the external party must be part of the Corda network. Within this approach, the external party consumes the endpoints of the API, which offer transaction data produced in the network. To access the DLT market

infrastructure's API endpoints, DLT market infrastructures have to grant access to the specific API. Furthermore, regulators that might supervise several DLT market infrastructures within their jurisdiction need to be granted access to each of the DLT market infrastructure API layers.

64. Considering API-based approaches, a variety of possibilities exist to design the system architecture of the Corda network itself and the potential entry points for data extraction. Within this approach, the obligation of providing the API-layer resides with the DLT market infrastructure. As the DLT market infrastructure is part of the network and facilitates any transactions conducted within it, its vault will contain all transactions occurring in the network.

65. Hence, to minimise the efforts of network participants, the DLT market infrastructure could provide an API-layer above its vault whose endpoints would offer the relevant and necessary transaction data to be consumed by external participants, such as regulators. With the introduced API implementations in Corda 5 for external requests, external systems obtain the possibility to invoke REST[18]-endpoints and await the response, such as in a traditional HTTP[19] request/response model.

66. Consequently, parsing of data and therefore provision of a format that is requested and aligned with the regulator, also resides with the DLT market infrastructure. This may concern the transformation of the API response data, which commonly is derived in JSON format, into an XML data format which is the underlying of the RTS 22 schema.

[18] REST is the "Representational state transfer" and a software-architectural pattern for an interface between different separate software components. It is commonly used to establish a connection between a Client and a Server.
[19] HTTP is the „Hypertext Transfer Protocol". which is a communication protocol used for communication of hypermedia documents such as HTML between browsers and web server but is also used to establish communication between further software components apart from the already mentioned. HTTP is a communication protocol commonly used by the REST pattern.

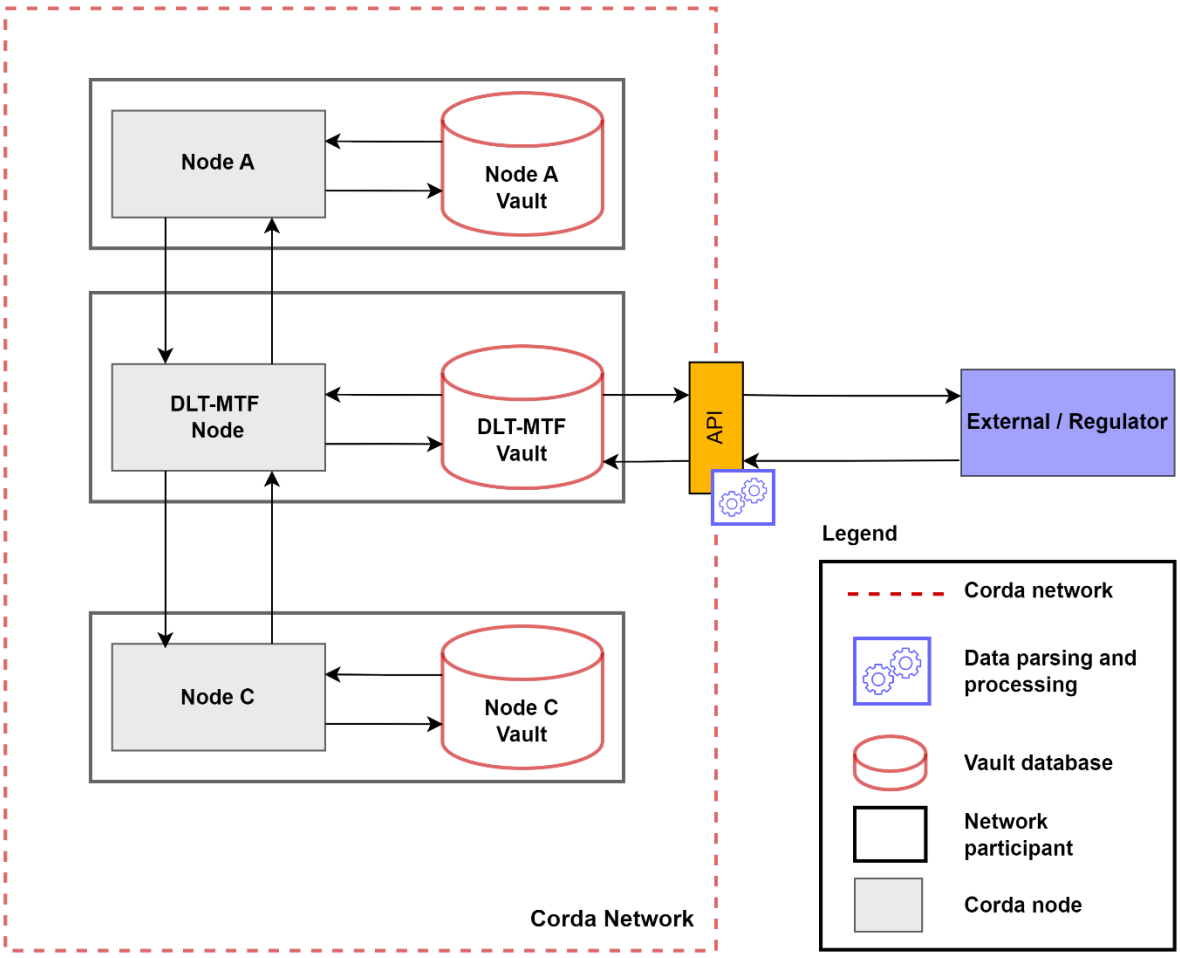


FIGURE 5: API-BASED TRANSACTION DATA EXTRACTION BY THE REGULATOR

a) Advantages

67. Firstly, with API-based approaches, real-time reporting and near-time reporting of transactions is possible. Real-time reporting defines the gathering and provision of data of transactions as they are executed. Near-time reporting, on the other hand, defines the gathering and provision of transaction data shortly after transactions are executed. Through near and real-time reporting, the reporting process is sped up and regulators can process the data sooner.

68. From both, a regulatory and DLT market infrastructure perspective, there can be an advantage in the future of minimising effort for data transaction and data transformation, **if an API standard** that enables all participants to maintain the same stable and interoperable transaction reporting process across all DLT networks and protocols, can be followed. The format provided by the API should be equivalent to the RTS 22 XML format,

such as encoded JSON objects of the equivalent message following the ISO 20022 standards, to maintain stable and interoperable procedures.

69. Since there is currently no existing standard that enables transaction reporting across several DLT networks and protocols, efforts reside for the DLT market infrastructure to implement the API functions, whereas regulators must implement the client[20]side of the API. For DLT market infrastructures it is an effort to ensure a stable underlying of the API functionality, which is discussed in Paragraph 77.

70. APIs are highly flexible system components and built to be accessed externally. Therefore, API interfaces, by default, provide high security standards, such as enhanced encryption. APIs are also more flexible in terms of their ability to adapt and configure to specific requirements and circumstances. Consequently, the overall IT-security valuation of the data extraction systems and components may be improved from regulator's perspective, compared to the file-based reporting systems.

71. As mentioned in Paragraph 63, APIs enable endpoints to be accessed externally and provide the requested data. By following the API-based approach, a simplified handling of the data is possible.

72. This approach could allow regulators to move away from the sequential cancellation/modification reporting process under MiFIR, which seems to be less relevant when moving to DLT. DLTs could potentially allow to simplify the process by removing cancellations from the picture and the DLT Pilot exemption from reporting could be used to explore it. For example, regulators could be enabled to access the latest state of the transaction, resulting in a removal of the need to have a correct sequence of NEW-CANC-NEW reports with the correct report record number attached.

b) Disadvantages

73. To benefit from the outlined advantages on security standards within API-based components, a deep understanding of the relevant security measures and the efforts associated with their implementation is fundamental.

74. As already touched in the respective advantage section of this chapter, standardisation of provided APIs is key to an interoperable and stable transaction reporting process. By using standardised API interfaces and components, a high level of technical interoperability between the components could be achieved in the future and would lead to an easier orchestration of the transaction data on the regulator's as well as on the DLT market infrastructure side. Although, there is currently no existing standard, advantages of the API-based extraction could only be achieved within one Corda network, whereas achieving

[20] Technical term

interoperability between different Corda networks, or possibly even different DLTs, in a standardised and interoperable way is not feasible in the short term.

75. From both, a regulator's and DLT-MFTs perspective, a disadvantage expresses within the cost of implementation. Regulators must make sure to receive data from APIs provided by DLT market infrastructures in a way that data can be further processed. Therefore, the implementation of a reliable client side is unavoidable, which leads to further efforts and costs for regulators. From a DLT market infrastructure's perspective, efforts stem from the implementation of API components and systems. API-based approaches must be implemented once and are then continuously improved by way of minor changes, which do not change the API's base implementation.

76. Although, the first-time implementation of API components and systems leads to more effort in a one-off manner on DLT market infrastructure side, DLT market infrastructures and regulators face further efforts due to the amount and complexity of accessing and implementing native protocol components, such as specific Corda APIs and Software Development Kits (SDKs) for the support, operation and maintenance of the exchanged data format and as well the response / request mechanisms.

77. For Corda the implementation and maintenance of the API becomes more complex since it must be adapted to the provided Corda APIs and SDKs. This leads to dependencies for API implementations across different networks. Hence, the larger the number of networks the regulator must connect to, the more significant are the efforts to do so. Besides the implementation, further the operation, maintenance, and performance monitoring of the API components brings with it significant additional efforts.

78. Above the implemented API-Layer, DLT market infrastructures need to come up with technical solutions to transform the transaction data derived from the vault into a format requested by the regulator. Exemplary is the data transformation of the data extracted by an underlying SQL[21] query from the vault, reachable from an API endpoint and finally transforming it from the provided JSON[22] formats into the data format requested by the regulator. At this point, it is worth to mention, that the previous effort of conversion is equally needed under the file-based approach.

79. DLT market infrastructures and regulators face major challenges regarding growth and creation of networks and the scaling of API-based implementations within a network. With the existence of several networks, the necessity for standardisation of API components and their monitoring arises. The more networks and the more DLT market infrastructures

[21] SQL is a standard language for accessing and manipulating databases (https://www.w3schools.com/sql/sql_intro.asp, 17.03.2023)
[22] JSON format is the JavaScript Object Notation and a format for storing and transporting data commonly between web pages and servers (https://www.w3schools.com/whatis/whatis_json.asp, 17.03.2023)

inside them, and various other networks, exist, the more effort is required for standardisation of orchestration.

80. One outline of the risen effort and complexity is shortly touched upon in Paragraph 63 and considered as a disadvantage of the API-based approach: Each DLT market infrastructure must grant access to the specific regulator which oversees the jurisdiction and expects the transaction report. This enhances to the prescribed complexity within orchestration, such as the necessity for regulators to store and manage credentials to access the API layers of the DLT market infrastructures. Furthermore, the amount of attack vectors as well as incident vectors is increased.

81. Additionally, there are further efforts from a regulator's perspective due to the implementation, operation, and maintenance of the systems deriving the transaction data from the API endpoints.

82. Regarding the dependencies and coupling of systems and components between the DLT market infrastructure and the regulator, a disadvantage arises within the API-based approach. Since the regulators' controls and systems are much more reliant on a fully functional API layer provided by the DLT market infrastructure, a failure might not only affect parts of the architectural design pattern, but lead to effects on several components, also affecting the regulators systems and components.

83. From an interoperability perspective, disadvantages are related to governance aspects of the API design and implementation. Within the file-based approach, full control over standards, schemas and implementation lies within the regulator, while within an API-based approach, the control over the implementation moves to the software providers of the respective underlying DLT protocols and networks, that might change over time and force DLT market infrastructures to continuously follow up the changes.

84. Changes in the API layer made by the DLT market infrastructure lead to a disadvantage for the regulator. It is impractical for regulators to follow up on the implementation of changes made independently by each DLT market infrastructure.

85. A further effect of the governance dimension additionally expresses within the access to the history and latest status of transactions as well as the access to oversee corrections and cancellations. DLT market infrastructures must implement similar procedures as within a file-based approach, to ensure that regulators receive latest transaction reports on a daily basis and can build a historical database as well as daily reports for data on corrections and cancellations. Regarding this effect, regulators are in first view dependent on the design and the correct implementation of the DLT market infrastructures.

86. Access to the history and latest status of transactions is especially important, since DLTs do not fully ensure to be tamper-proof. From governance perspective, in a network with a large number of participants and a small number of validating nodes, tampering may be hypothetically possible. Realistically, governance and DLT networks must be designed in

away, that tampering is not possible. This should be further ensured through compliance with the relevant regulatory frameworks such as Digital Operational Resilience Act (DORA)[23].

3.1.2.3 Native access to each DLT and DLT network: File-based transaction reporting

87. There are additional scenarios in which regulators may operate observer nodes and thereby become part of the Corda network. In such a case, network participants would still trade via the DLT market infrastructure which then could initiate and execute transaction flows transmitting the transaction data to the observer node. The transmitted data would then be stored in the observer node's vault and could be extracted from the Corda network in one of two ways.

88. The first scenario, illustrated in Figure 6 below, encompasses a file-based extraction solution. As the observer node is specified in the DLT market infrastructure's transaction flow, the transaction data will be stored in the observer node's vault. From there, the operator of the observer node, in this case the regulator, can extract the data from its vault by using the relevant provided Corda APIs and SDKs. In the next step, the extracted data is processed and made available via a file-based approach to make the data available outside of the Corda network.

[23] Regulation (EU) 2022/2554 of the European Parliament and of the Council of 14 December 2022 on digital operational resilience for the financial sector and amending Regulations (EC) No 1060/2009, (EU) No 648/2012, (EU) No 600/2014, (EU) No 909/2014 and (EU) 2016/1011

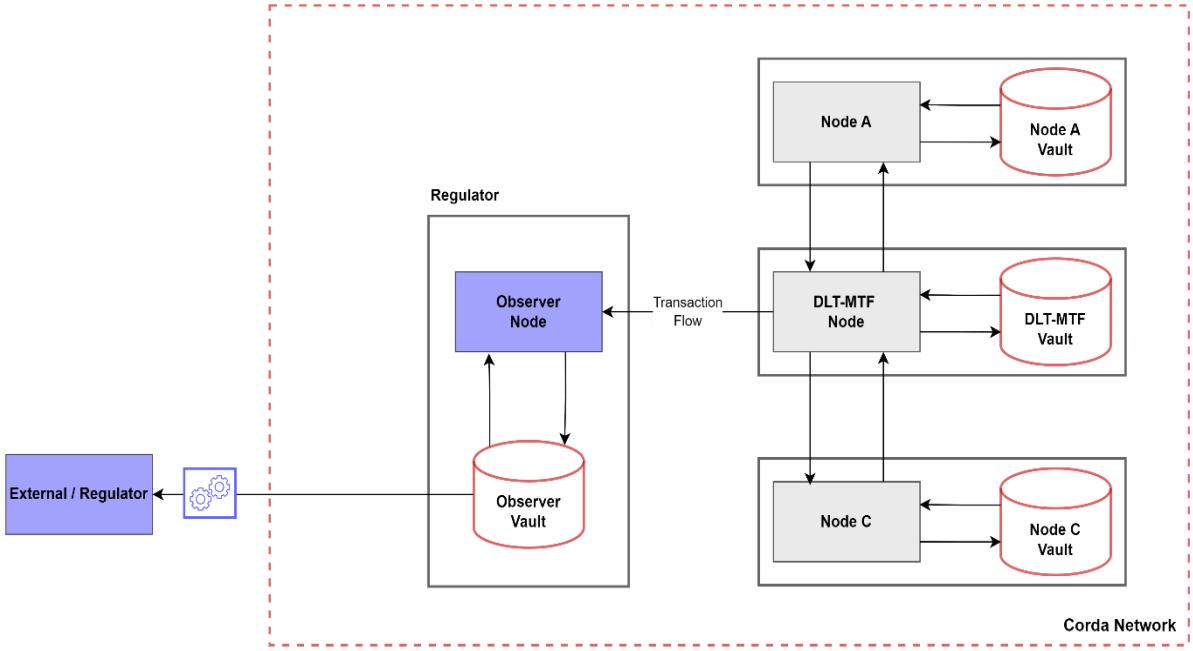


FIGURE 6: OBSERVER NODE EXTRACTING TRANSACTION DATA FILE-BASED

a) Advantages

89. Similar to API-based approaches, the setup displayed in Figure 6 allows for real-time reporting of transactions. This stems from the regulator's direct involvement in the network. Hence, communication does not need to flow through multiple other systems or components first, as the regulator's observer node is directly involved at the time of transaction execution.

90. By having regulators directly involved in the DLT network and provided with real-time access, the possibility for manipulation and tampering of transaction data is reduced. As previously described in Paragraph 85, governance and technical functionalities to prevent tampering of network participants must be ensured. The effect from real-time access is that regulators have a full view on the executed transactions and can identify manipulation in real-time. A further effect of the native access for regulators to each DLT and network the access rights for external participants and regulators result in a direct read access to all transactions conducted via the DLT market infrastructures platform.

91. The direct involvement of a regulator within the Corda network likely has additional positive effects on the real-world adoption and acceptance of DLTs as such, as it signifies a certain level of trust into the technology.

92. From a DLT market infrastructure's perspective, there is less effort associated with the extraction or processing of the transaction data, unless a full data set of transaction data

can be stored on the DLT. The regulator is directly involved in all transactions being executed and extracts the transaction data by itself. Within Corda and Hyperledger fabric, it may be suitable and technically possible to store the full set of transaction data including PII and other trade-relevant data directly on the DLT. In Ethereum as a public DLT protocol, it is not possible to store PII data on the DLT and common practice is to only store minimal information on the DLT.

b) Disadvantages

93. The technical setup of an observer node, as well as its operation and monitoring will require additional skills, resources, and efforts.

94. DLT market infrastructures using different DLTs, as well as different versions of said DLTs with various smart contract implementations can be limiting factors. Should the used versions change, it will be impossible for regulators to implement, maintain, and follow-up on the changes, proving to be another disadvantage. This effect can be observed within the regulator's security assurance in terms of confidentiality, integrity, and availability. By following the native access to each DLT network approach, security assurance depends on the native offering of each DLT and each present Corda network. Therefore, regulators must get familiar with the native offerings by the technology and adapt their assurance methodologies accordingly.

95. Data extraction and processing from the regulator's observer node's vault to, for instance, an XML schema and file format result in further efforts. Corda provides the possibility to implement transaction reporting based on ISO 20022 XML schemas. However, regarding the approach of regulators natively accessing the network via an observer node, it is dependent in which way transaction reporting is designed and implemented, which is decided by the network participants.

96. Should the transaction flow and therefore the reporting fail on the DLT market infrastructure side, additional efforts in accessing the executed transactions are required. For instance, regulators may have to request the relevant data directly from the participants involved in the unreported transactions, while another approach could be to implement back-up transaction flows. Either way, it is crucial that transaction flows between regulators and DLT market infrastructures are always fully functional and available. DLTs as such are regarded to be stable, however, they are still fairly new technologies. Hence, they are subject to continuous improvement and have not been established for long in production environments.

97. The above-described potential disadvantages which may arise for regulators additionally show the dependency allocation within the overall approach, in where regulators natively access several DLTs and their networks. Effectively, regulators are vastly dependent on the technology as well as on the network design and the decisions which are made by the

participants inside the Corda network. The existing tight coupling may be considered as a further disadvantage from a regulator's perspective.

3.1.2.4 Native access to each DLT and DLT network: API-based transaction reporting

98. The second scenario, illustrated in Figure 7 below, encompasses an API-based extraction solution. Again, the observer node is specified in the DLT market infrastructure's transaction flow resulting in the transaction data being stored in the observer node's vault. The regulator can extract the data from its vault via an API-layer being implemented at the fringes of the network allowing it to extract the transaction data from the Corda network.

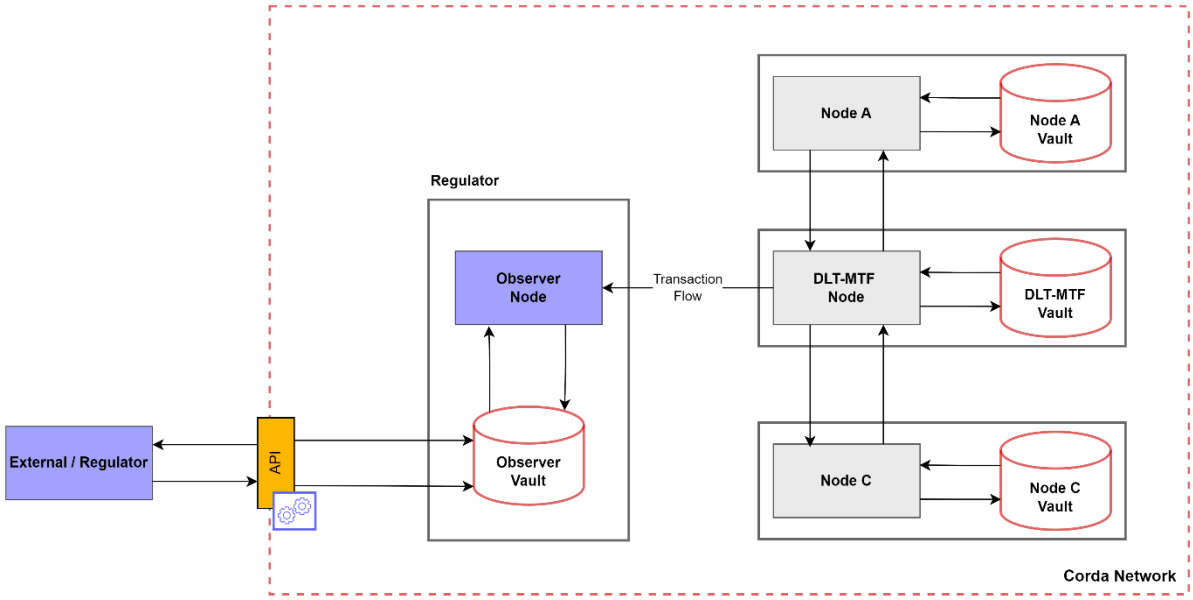


FIGURE 7: OBSERVER NODES EXTRACTING TRANSACTION DATA API-BASED

a) Advantages

99. The various advantages for this approach are the same as for the preceding approach displayed in Figure 6. Hence real-time reporting is possible, regulator involvement will likely lead to reduced manipulation and transaction data tampering, further additional positive effects due to the regulator's involvement are to be expected.

b) Disadvantages

100. The disadvantages are similar those associated with the approach illustrated in Figure 6. This means, regulators may be inexperienced in node setup as well as subsequent node

operation and monitoring. Further, again a fallback mechanism ought to be established in cases where a DLT market infrastructure is unable to report the relevant transaction data to the observer node.

101. A further disadvantage not present within the previous approach is that regarding security standards within API-based components, a deep understanding of the relevant security measures and the associated efforts is crucial. Lastly, from a regulator's perspective the efforts for the implementation, operation, and maintenance of the API-based components and systems are a further disadvantage.

102. According to the feedback provided by NCAs, it is currently not possible to apply the approach of native access to each blockchain. The main factor excluding this possibility is the disadvantage of a subsequent follow-up and monitoring of changes to the amounts of networks, protocols and further technical components relevant to the DLT architecture.

3.1.3 Conclusion on transaction data extraction within the Corda DLT

Criteria	File-based approach	API-based approach	Native access to each DLT network
Access rights for different participants and regulators	Not applicable – the DLT market infrastructure sends the reports to the regulator. Close to current transaction reporting approach e.g., under MiFIR Art. 26, EMIR Art. 9 or SFTR Art.4	Each regulator needs access rights granted on each DLT market infrastructure system to access the respective API.	Read access to the transactions processed via DLT market infrastructures is possible through the operation of observer nodes and in addition the involvement of external participants, such as regulators in the respective transaction flows of the smart contract.
Security assurance in terms of confidentiality,	Files can be exchanged in an encrypted way and	Depending on the DLT market infrastructure specific	Depending on the native technology offering of each

integrity, and availability	are transmitted on a secured channel.	implementation of the API.	Distributed Ledger Technology.
Timing and frequency of the updates, such as real-time or near-real-time	No real-time / near-real-time reporting, in-line with current transaction monitoring processes of T+1 transaction reporting. Reporting files may be made available at a pace which is to be agreed between the DLT market infrastructure and the regulator. Although the risk remains, that the frequency of the reporting is lower.	Near-real-time access is possible.	Real-time access is possible.
Technical standards supported	ISO 20022 standard and RTS 22 XML schema (or a new schema covering RTS 22 data and additional fields specific to the DLT, e.g., DTI).	There is currently no standard existing. Implementations based on ISO 20022 standards and RTS 22 XML schemas are also possible.	There is currently no standard existing.
Tight vs. loose coupling	Loosely coupled.	Tightly coupled to the DLT market infrastructure implementation.	Tightly coupled to the DLT implementation.
Governance implications	Regulators control the standard XML schema, agree on	Regulators have no control on changes of the DLT market	Regulators have no control on changes decided on the DLT

	updates, and control the timing of implementation of the updates to give stakeholders time for implementation.	infrastructure's implementations and need to follow changes on the client-side implementation.	protocol. Further, regulators have no control on changes of the DLT market infrastructure's implementations. DLT market infrastructures may have control on changes of the DLT network, in which they are part of, dependent on their level of involvement in steering of the network.
Access to the history and latest status of the transaction	Latest transaction reports may be received on a daily basis, although in sequences of CANC-NEW transactions, the latest version (NEW) is received on the following day. In addition, intraday reports of the same file are also possible.	Depending on the DLT market infrastructure specific implementation.	Depending on the DLT market infrastructure specific implementation.
Access to transaction corrections and cancellations	Regulators receive data on corrections and cancellations daily. The process is dependent on correct sequencing	Depending on the DLT market infrastructure specific implementation. Possibility to always have access to the	Depending on the DLT market infrastructure specific implementation. Possibility to always have

	of CANC-NEW reports. The corrected version of the report may arrive with some delay.	latest version of data can be explored.	access to the latest version of data can be explored.

TABLE 1: CONCLUSIONS ON DATA EXTRACTION IN THE CORDA DLT

3.2 Ethereum

3.2.1 Background

103. Ethereum was first described in a whitepaper[24] by its founder Vitalik Buterin in 2014 before being officially launched in 2015. It is an open source blockchain allowing for the building and operation of various use cases on it, including but not limited to decentralised applications, games, marketplaces, and financial instruments. Further, Ethereum carries its own native (crypto-)currency Ether (ETH). At the time of this report, Ether is only second to Bitcoin in terms of its market capitalisation[25] and will be further explored later in this section.

104. Smart contracts are of significant importance on the Ethereum network. Tokens, including those mentioned in Paragraph 118 and 119, and thereby DLT financial instruments, are based on specific smart contracts. Smart contracts contain certain functions, for instance *Transfer*, *Mint* and *Approve*. These functions emit certain events, which are stored within the transaction receipt, when triggered.[26]

105. The blockchain has a single inherent computer, known as the Ethereum Virtual Machine (EVM), embedded in it, a copy of which is kept by all Ethereum network participants, i.e., all nodes. The nodes all agree on the state of the EVM and possess the ability to request it to perform computations. Requests for computations are also called transaction requests.

106. Once such a request is broadcast, the computation is verified, validated, and executed by the other network participants. The computation's execution then changes the EVM's state, which is committed and sent throughout the network. The blockchain further stores the record of all past Ethereum transactions as well as the current state of the EVM.

107. Ethereum transactions can only be initiated by externally owned accounts (EOAs).[27] EOAs are one of the two types of Ethereum accounts, with the other one being contract accounts.[28] EOAs are owned and controlled by individual users, such as natural persons. They can receive, hold, and send Ether as well as other tokens by way of Ethereum transactions.[29] They can also interact with the second type of Ethereum accounts, which are contract accounts.

108. Contract accounts are used to deploy and execute smart contracts on Ethereum. Contract accounts are not controlled by any one entity but rather by the logic of the smart

[24] https://ethereum.org/669c9e2e2027310b6b3cdce6e1c52962/Ethereum_Whitepaper_-_Buterin_2014.pdf
[25] https://coinmarketcap.com/currencies/ethereum/
[26] https://goethereumbook.org/events/
[27] https://ethereum.org/en/developers/docs/accounts/#key-differences
[28] https://ethereum.org/en/developers/docs/accounts/#types-of-account
[29] https://ethereum.org/en/developers/docs/accounts/

contract code. In doing so, they enable a wide variety of use cases, including decentralised applications. An example of a decentralised application enabled by contract accounts and their smart contracts is the Uniswap protocol[30]. Uniswap is a decentralised application on which tokens can be traded without further intermediaries.[31]

109. Both kinds of Ethereum accounts have addresses, i.e., unique identifiers that represent the accounts on the Ethereum network. EOAs, however, unlike contract accounts, also have a private key. Private keys on Ethereum provide their respective owners with the ability to have control over access to funds as they prove ownership and control of an EOA on the network. They are also used to sign Ethereum transactions.

110. Ethereum transactions can be sent either to EOAs or contract accounts. In cases where the recipient to an Ethereum transaction is a contract account, the called upon smart contract takes the Ethereum transaction and its data as input to execute certain functions within that contract. For example, in the case of a decentralised exchange, trading processes are handled via smart contracts. If Party A wants to buy a certain token which is traded on a decentralised exchange, Party A can initiate an Ethereum transaction, specify the contract account, i.e., the smart contract as its recipient, the token he wants to buy and the amount at which to buy the token, and call a hypothetical *Buy* function implemented in the smart contract. The hypothetical "Buy" function will then calculate how many of the desired tokens Party A can buy given the amount he specified in the Ethereum transaction, deduct funds from Party A's account, and transfer the acquired tokens to his account.

111. As touched upon in Paragraph 105, Ethereum's network participants are better known as network nodes. Ethereum is designed in a way that the average personal computer is able to run a node.[32] However, it is recommended to use dedicated hardware to run nodes in order to minimise node downtime. Nodes can hence be run by anyone and ensure the network's decentralisation and security.

112. Nodes are instances of Ethereum client software connected to other computers running Ethereum software. Together, the multitude of nodes makes up the Ethereum network as such. In total, there are three different types of nodes, each of which consume data differently. While full nodes[33] store the entirety of blockchain data, assist in block validation and verification, light nodes [34] only download block headers carrying summarised information about block contents. Lastly, archive nodes build archives of historical states

[30] The Uniswap protocol is used to swap crypto-assets with liquidity providers which enable liquidity through keeping storing assets in respective Uniswap Liquidity Pools (LPs). The swap results in usage of the Uniswap protocol for trading and clearing as well as automatically settlement on the Ethereum (or other by Uniswap supported) blockchain.
[31] https://uniswap.org/
[32] https://ethereum.org/en/run-a-node/
[33] https://ethereum.org/en/developers/docs/nodes-and-clients/#full-node
[34] https://ethereum.org/en/developers/docs/nodes-and-clients/#light-node

and thus are an effective way of querying historical blockchain data, such as historical account balances.[35]

113. Two different types of clients, namely execution clients and consensus clients, exist on Ethereum.[36] The execution client executes new Ethereum transactions broadcasted. It further holds the network's most up-to-date version, i.e., the most up-to-date state of all Ethereum data. The consensus client, on the other hand, implements Ethereum's proof-of-stake consensus mechanism. Hence, it is crucial in that it facilitates network agreement regarding the data validated by the execution client. As part of "The Merge", these previously separated layers were connected and integrated into one network.

114. On September 15, 2022, Ethereum underwent The Merge, shifting the network's way of achieving consensus from proof-of-work (PoW) to proof-of-stake (PoS).[37] Network validators must now stake Ether into Ethereum smart contracts and are responsible for assessing that new blocks are valid and, at times, also creating and propagating new blocks through the network themselves. Validators' staked Ether acts as collateral which can be slashed from the staked balance in case the validator behaves in an undesired manner.[38] Further, "The Merge" was successful in decreasing Ethereum's energy consumption by 99.9%.[39]

115. Blocks on Ethereum store Ethereum transactions and reference previous blocks by making use of a hash. Ethereum makes use of Keccak-256 hashes, which, when compared to SHA-256 hashes, are stronger.[40] Should information within a block change, its accompanying hash would do so as well, as hashes are derived from block data. This is a useful mechanism for fraud prevention as changes to historical blocks would cause all subsequent blocks to become invalid.

116. Blocks and the Ethereum transactions contained within them are strictly ordered and must be agreed on by all Ethereum nodes. For a synchronised state of the network to be possible, blocks are propagated through the rest of the network, once assembled by a validator.[41] The other nodes then add the newly created block to the end of their blockchain before a new validator is randomly selected to assemble the next block. Other randomly selected validators will then vote upon the block's technical validity.

117. Ethereum transactions, are cryptographically signed instructions from accounts.[42] All Ethereum transactions, hence, update the overall state of the Ethereum network and are

---

[35] https://ethereum.org/en/developers/docs/nodes-and-clients/#archive-node
[36] https://ethereum.org/en/developers/docs/nodes-and-clients/#what-are-nodes-and-clients
[37] https://www.coindesk.com/tech/2022/09/15/the-ethereum-merge-is-done-did-it-work/
[38] https://blockdaemon.com/products/white-label-validator/ethereum-introduction/#staking
[39] https://cointelegraph.com/news/the-merge-brings-down-ethereum-s-network-power-consumption-by-over-99-9
[40] https://www.geeksforgeeks.org/difference-between-sha-256-and-keccak-256/
[41] Ethereum transactions are contained within blocks. Assembling the blocks refers to the process of selecting and including valid Ethereum transactions within a block. Once a block has been assembled, it will be appended to the previous block, thereby creating a chain.
[42] https://ethereum.org/en/developers/docs/transactions/

propagated throughout it. This means, all full and archive nodes have a copy of all transactions that have occurred on Ethereum. Light nodes, on the other hand, only store block headers, which contain information such as block numbers, timestamps, et cetera. For more detailed information regarding Ethereum transactions, they rely on full nodes. While there are various types of Ethereum transactions, their most rudimentary form refers to the sending of funds, e.g., Ether, from one account to another account.

118. As mentioned before, Ether is the network's native cryptocurrency. It has a wide variety of use cases in the broader DLT ecosystem and, on Ethereum itself, is used for such things as paying for "gas", i.e., transaction fees on the Ethereum network, as well as block proposal and validation. Ether is created in a "minting" process and distributed between the proposer and validators of a block.[43] Ether can also be "burned" and thereby permanently removed from circulation. Every Ethereum transaction leads to the burning of Ether, as a base gas fee whose amount is determined by transactional demand on the network gets destroyed along with the Ethereum transaction.[44] The process of minting Ether, however, does not lead to its burning.

119. Generally, there are a variety of token standards that find application on Ethereum. Some of the most popular token standards currently are ERC-20 tokens and ERC-721 tokens.[45] ERC-20 tokens are fungible, meaning an ERC-20 token will always be equal to another ERC-20 token. This makes them especially useful for usage as cryptocurrencies or security tokens. ERC-721 tokens, on the other hand, are the underlying interfaces for so-called non-fungible tokens (NFTs). ERC-721 tokens especially find application in the collectibles or ticketing industry.

120. A further popular token standard is the ERC-1155 standard, which allows for the creation of multi-token contracts. It derives its popularity from the ability to represent various types of digital assets on Ethereum, among those, such assets that are typically represented by ERC-20 and ERC-721 tokens. The ERC-3643 standard is a proposed token standard for regulated exchanges.[46] At the time of writing, however, the Ethereum Improvement Proposal (EIP) is stagnant. Hence, the proposal has seen no activity for at least the preceding six months, making its adoption uncertain.[47]

121. Ethereum as a network is constantly being developed by various parties, including but not limited to EIP authors and developers.[48] Although, there is no central party which is in charge of the network, changes to its core protocol are implemented regularly in order to maintain Ethereum's stability and security.

---

[43] Ether is minted, i.e., created as a reward for proposing blocks on the network. There is no alternative mechanism that can be applied to mint Ether.
[44] https://ethereum.org/en/developers/docs/intro-to-ether/
[45] https://ethereum.org/en/developers/docs/standards/tokens/
[46] https://eips.ethereum.org/EIPS/eip-3643
[47] https://eips.ethereum.org/
[48] https://ethereum.org/en/governance/

122. When it comes to DLTs, there is a difference between on-chain and off-chain governance. While on-chain governance typically involves stakeholder voting, for instance, by making use of a certain governance token, off-chain governance occurs through a process of social discussion. Ethereum itself makes use of off-chain governance, while it should be noted that some of the applications built on top of it utilise on-chain governance.[49]

123. Proposed changes to the network must undergo a process called Ethereum Improvement Proposal, as briefly touched upon in Paragraph 119. Several steps are involved in the decision of whether the proposal is approved or rejected. Should the EIP be approved, it will be scheduled as part of a network upgrade. As everyone on the network must upgrade simultaneously, multiple EIPs tend to be bundled together.[50]

124. Any application that interacts with the Ethereum blockchain is connected to an Ethereum node. In order to establish that connection, Ethereum clients provide a JSON-RPC specification. A Remote Procedure Call (RPC) is a mechanism, that allows an application to execute a message in another machine[51]. In Ethereum RPC is used to provide methods which applications can execute in order to read from and write to the Ethereum blockchain. The following code example describes a remote procedure call to the Ethereum blockchain which retrieves the current block number of the client[52]:

```
// Request
curl -X POST --data '{"jsonrpc":"2.0","method":"eth_blockNumber","params":[],"id":83}'
// Result
{
        "id":83,
        "jsonrpc": "2.0",
"result": "0x4b7" // Blocknumber: 1207
}
```

3.2.2 Approach Methodology

125. To derive potential approaches for extraction of transaction data from the Ethereum DLT, best practices, mostly based on the existing architecture and possibilities to extract data were taken into account. This information was supplemented with further learnings derived from potential applicants. Selected applicants to the DLTR were contacted to gain a better understanding of how Ethereum and EVM-based blockchains can facilitate transactions in DLT financial instruments and how the applicable transaction data is registered and stored.

126. EVM-based blockchains are blockchains that are compatible with the Ethereum Virtual Machine and thus allow for creation and execution of smart contracts on it. Notable

[49] https://ethereum.org/en/governance/#on-chain-vs-off-chain
[50] An overview of notable milestones of the Ethereum blockchain can be found here: https://ethereum.org/en/history/
[51] https://www.w3.org/History/1992/nfs_dxcern_mirror/rpc/doc/Introduction/WhatIs.html#:~:text=Remote%20Procedure%20Call%20is%20a,features%20in%20a%20transparent%20way. (20.02.2023)
[52] https://ethereum.org/en/developers/docs/apis/json-rpc/#eth_blocknumber (20.02.2023)

examples of EVM-based blockchains besides Ethereum itself are Polygon, Avalanche and the BNB chain.[53]

127. Regarding the outlined best practices and learnings from potential applicants, the content presented within this study does not claim to be fully exhaustive and further approaches on data extraction of transaction data from the Ethereum DLT may exist.

128. In any of the proposed approaches, a common foundational pattern is followed to obtain data directly from chain. On-chain data extraction requires the execution of a query, using RPC via an execution client, inspection, and investigation of the whole transaction. Several entry points to search for transactions exist.

129. For example, one obtains information via the block number, retrieves transaction hashes included in the block and then further investigates the specific transaction. Another one searches via the transaction hash to obtain the transaction receipt. A third possible entry point to extract transaction data from the Ethereum blockchain is to filter for any transactions and events of a specific smart contract. As a further approach, a filter on a specific account enables to extract all transactions specific to the observed account.

130. Overall, the most efficient and straightforward method is to query a known transaction hash in order to extract the relevant data. As described in the "Report on the DLT Pilot Regime - Study on transaction reporting based on RTS 22", the transaction receipt contains the relevant information on txhash, block, nonce, data etc. and the transaction logs[54]. The following graphic describes the abstracted and simplified interaction with the Ethereum blockchain via the remote procedure calls protocol called "JSON-RPC":

FIGURE 8: SIMPLIFIED JSON-RPC BASED INTERACTION WITH THE ETHEREUM BLOCKCHAIN

131. A foundational aspect of the applied methodology is used for any of the approaches presented: the Ethereum blockchain is a public blockchain protocol, which enables anyone to directly extract data from the chain. Consequently, there should be no incentive to tamper with the transaction data, since it is publicly available, and anyone (even outside the network) can check the correctness.

132. Another foundational dimension of the proposed methodology within this study is on the nature of tokenised assets and their underlying smart contracts. As mentioned above, due to the existence of ERC Standards and the required flexibility for blockchain-based

[53] https://medium.com/eligma-blog/what-are-evm-compatible-blockchains-64f91c97038e (18.04.2023)
[54] See chapter 3.2.3, Report on the DLT Pilot Regime - Study on how transactions are registered in various blockchain solutions

financial instruments and therefore tokenised assets, these would usually be implemented as a smart contract based on one of the above-mentioned standards on the Ethereum chain or with a self-created implementation.

3.2.2.1 File-based approach

133. To extract the data from the Ethereum Blockchain, or other respective EVM-based Blockchains, the DLT market infrastructure must create a full dataset containing data which is not stored on the blockchain, such as personal data of a retail customer, and the related transaction data directly from the Blockchain, commonly described as “on-chain” transaction data.

134. According to the interviewed potential applicants of the DLTR, a common architectural design is to have an internal data storage system at the DLT market infrastructure to store personally identifiable information (PII)[55]. In addition, market participants observed that the solely tokenised security could be separated from the relevant transaction data regarding the trading and settlement.

135. A common example for the separation is to have the tokenised security in form of an ERC-20 token on the Ethereum blockchain, with its native functions[56], while the order book with its trading and settlement mechanisms is implemented within the DLT market infrastructure’s internal system architecture. On the other hand, it was also observed that market participants implemented the tokenised asset in form of an ERC-20 token as well as the order book on the blockchain.

136. Either of the architectural patterns lead to different conditions in terms of file-based transaction reporting. In case of separation of data within an internal system and data on the blockchain, an additional effort to link further trade-relevant data to the blockchain data must be made before a full dataset for reporting can be created. A more detailed outline on how the linking works, is described in Figure 11.

137. In case of no separation, which means all trade-relevant data can be found on the blockchain the effort is limited to link a transaction to the respective personal data of the user. Usually, this is done by linking either personal data or an identifier of the personal data to the user’s wallet address and further to the specific transaction hash.

138. The following figure (Figure 9) describes the explanatory scenario of the extraction of the transaction data by using a file-based approach on a high-level.

---

[55] Personal data or personal identifiable information, shortly described as PII data, is information which can be linked to a living individual and identify the individual by data such as a name, address, IP addresses and further. (https://commission.europa.eu/law/law-topic/data-protection/reform/what-personal-data_en, 31.03.2023)

[56] For further information, see “Report on the DLT Pilot Regime - Study on how transactions are registered in various blockchain solutions”

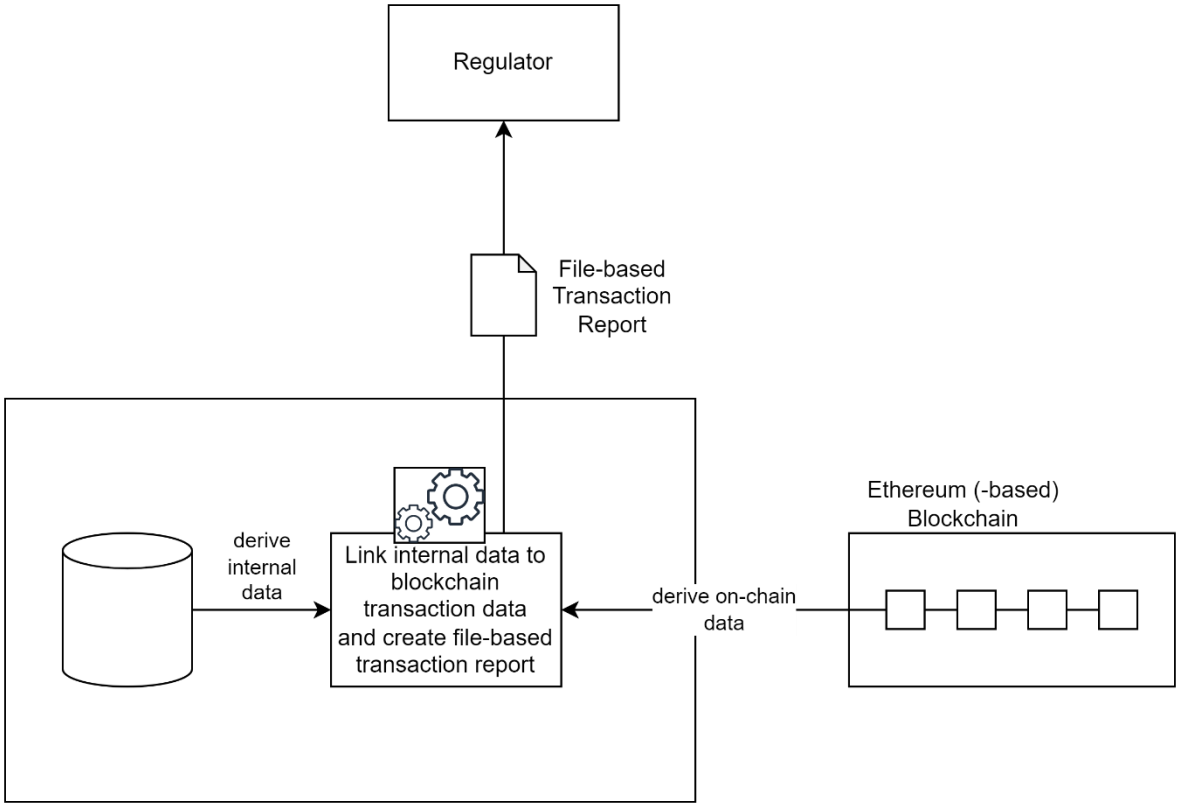


FIGURE 9: ETHEREUM DATA EXTRACTION: FILE-BASED TRANSACTION REPORT

a) Advantages

139. Within the file-based data extraction approach in the Ethereum technology, all advantages in Section 3.1.2.1 are applicable.

b) Disadvantages

140. In terms of the disadvantages of using a file-based data extraction approach in Ethereum, all disadvantages in Section 3.1.2.1 are applicable. Furthermore, data in Ethereum is extracted via JSON-RPC calls which are carried out via a client (see Paragraph 128). Major changes to either the client or the JSON-RPC implementation require software updates and are additional efforts on the DLT market infrastructure side. Changes to IT systems lead to interdependencies with other systems.

141. Although, Ethereum as a public Blockchain DLT represents specificities, which cannot be found in private and permissioned DLTs such as Corda and Hyperledger fabric, the previous described disadvantage of major changes within the software, comes into effect when discussing the respective specificities in Section 3.2.2.4.

3.2.2.2 API-based approach

142. In the API-based approach, a DLT market infrastructure provides an API with specific endpoints that can be accessed and called by the regulator to extract relevant data on the conducted transactions. According to the interviewed market participants and potential applicants for the DLTR using the Ethereum (or EVM-based) technologies, the API-based approach, as well was the file-based approach, was mentioned as a suitable method to process transaction reporting.

143. Within the API-based approach, the DLT market infrastructure provides an API to the regulator which exposes endpoints that allow to extract the transaction data. To provide any data, the DLT market infrastructure must access relevant data from the blockchain directly and match it with further trade-relevant data and personal data of the user that performed the trade. The process of accessing data directly from the Ethereum blockchain and linking it with further data to get a full dataset about the trade was already described in the previous chapter 3.2.2.1.

144. The difference within the API-based presentation of the data compared to the file-based reporting is not based in the extraction of data from the blockchain or different system components. The difference is solely based in the way of exposing the data to an external party, in this case the regulator. The following figure (Figure 10) describes the explanatory API-based data extraction for transaction data:

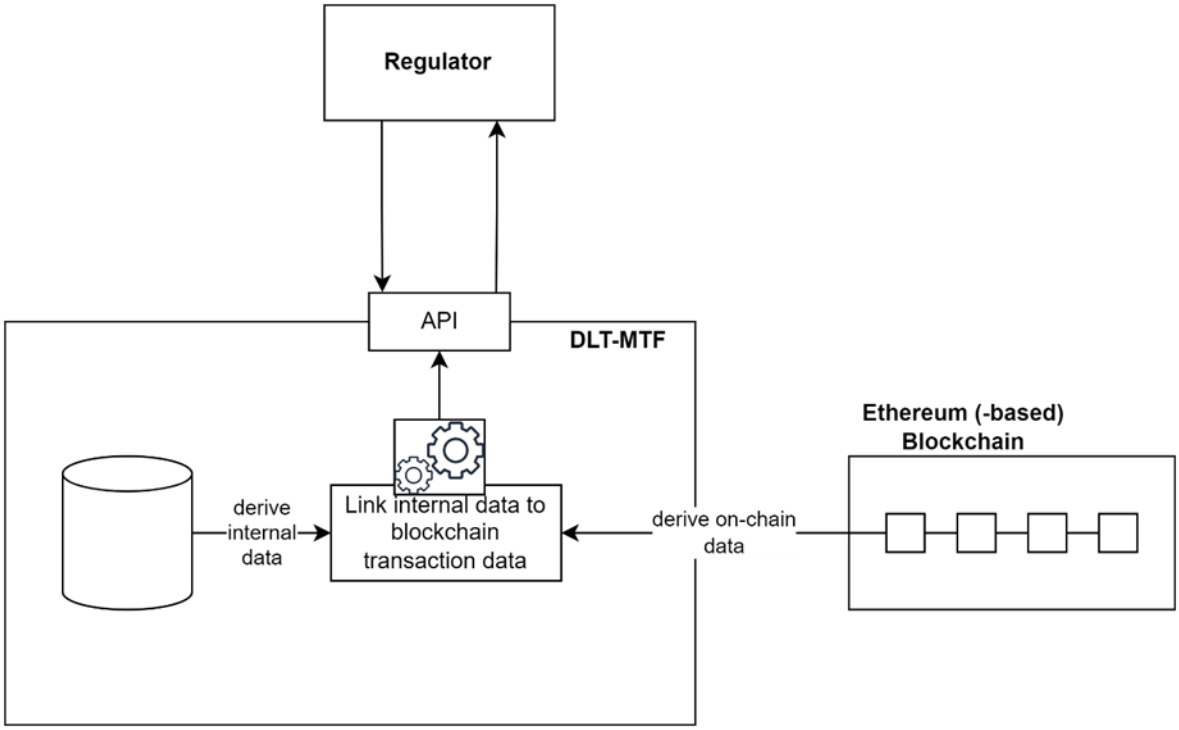


FIGURE 10: API-BASED TRANSACTION DATA EXTRACTION IN ETHEREUM

a) Advantages

145. Within the API-based data extraction approach in the Ethereum DLT, all advantages described in Section 3.1.2.2 apply as well.

b) Disadvantages

146. Within the API-based data extraction approach in the Ethereum technology, all disadvantages in Section 3.1.2.2 are applicable.

3.2.2.3 Native access to each DLT and DLT network

147. A third approach enables the regulator to access transaction data directly from the blockchain. The regulator will retrieve transaction data directly from the Ethereum blockchain and gain relevant data on the tokenised assets and its transactions, according to the basic fields that are always visible and additional fields, that are specified by the assets underlying smart contract implementation.

148. It may be impossible to obtain a full data set from the Ethereum DLT, due to the design as a public blockchain and storing personal data of private persons in a clear and readable

way may violate General Data Protection Regulation (GDPR), while ensuring and implementing GDPR compliance in a public Blockchain, such as Ethereum, results in huge effort and may be not feasible[57].

149. Extracting transaction data within Ethereum requires a linkage of transaction data on the DLT / blockchain to further trade relevant data as well as the associated personal data of the trading users. Usually, this information is kept as data internally stored within the DLT market infrastructure's infrastructure and systems. The linkage of data can be done in various ways.

150. A common method is a linkage of a trade through a transaction hash, the related wallet addresses of sender and receivers to the associated and internally kept personal data of the user, plus further internally kept data that could be relevant for the trade. The following figure (Figure 11) describes an exemplary approach for linking the data.

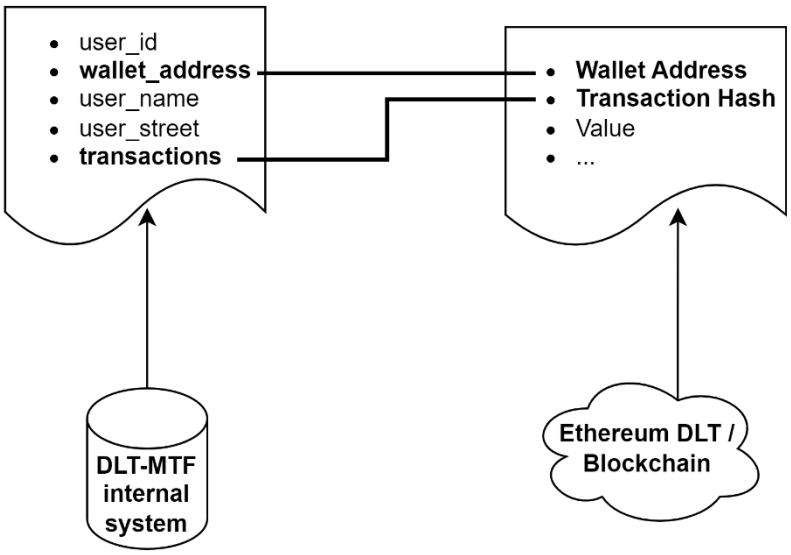


FIGURE 11: EXEMPLARY APPROACH FOR LINKING DLT MARKET INFRASTRUCTURE INTERNAL DATA TO PUBLIC DLT TRANSACTION DATA

151. Within the later scenario, the DLT market infrastructure provides the internal data in a file-based approach, following the status-quo procedures and standards, i.e., RTS 22 XML-Schema. Continuing from this step, the blockchain data must be converted and processed into a specific file format, such as XML. As mentioned in the example of the RPC call in Paragraph 128, the returned data from the comprehensive RPC call is in JSON format and therefore minimal effort is needed to transform the data into XML format.

[57] Refer for further insights on the tensions between public blockchains and GDPR requirements to: https://www.europarl.europa.eu/RegData/etudes/STUD/2019/634445/EPRS_STU(2019)634445_EN.pdf

152. The following figure (Figure 12) describes the abstracted procedure of data extraction, where the regulator extracts transaction data directly from the Ethereum blockchain and links it with the relevant internal reference data received from the DLT market infrastructure in file format. The respective link would have to be made by connecting the specific transaction hash and additional information, such as the wallet address, to the PII data and further relevant trade data received from the DLT market infrastructure's internal system.

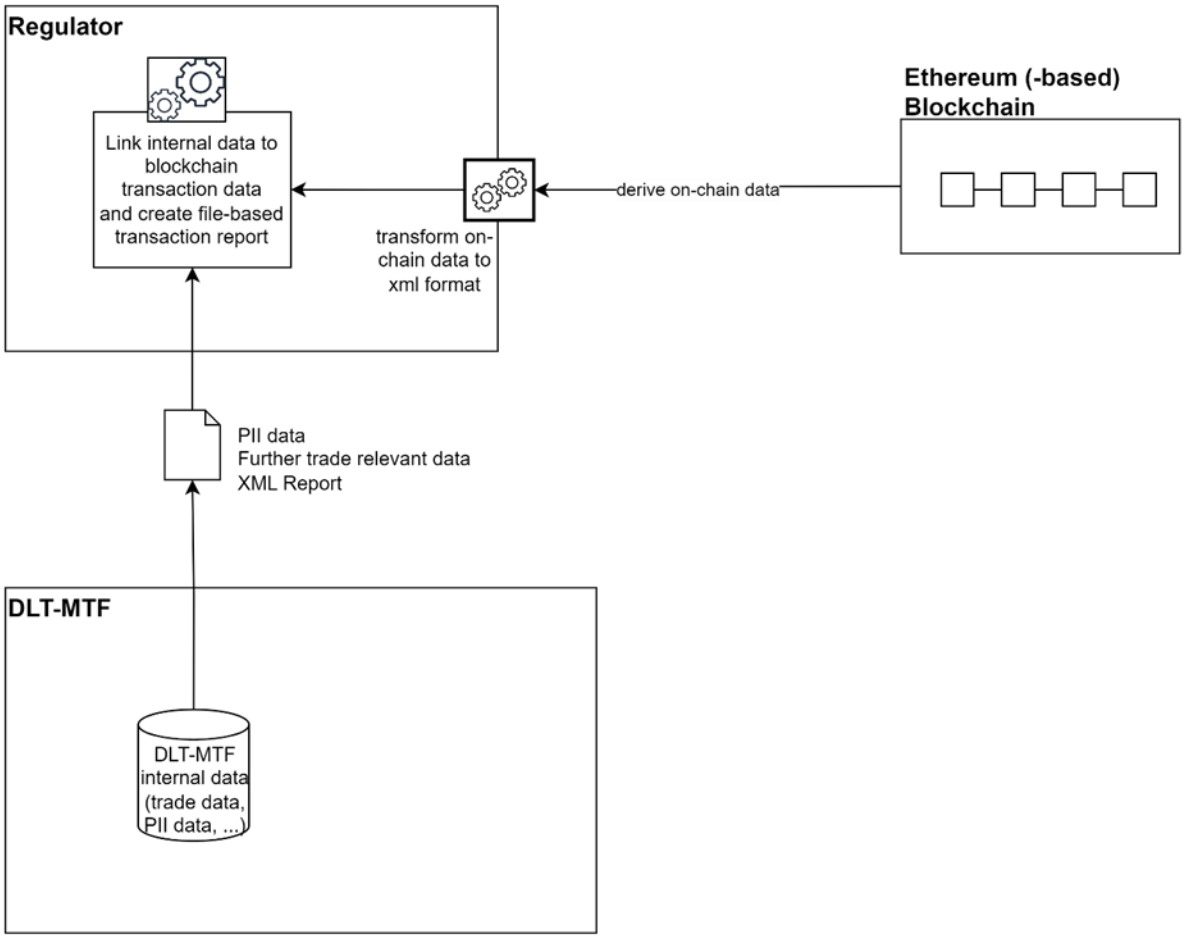


FIGURE 12: NATIVE ACCESS TRANSACTION DATA EXTRACTION BY USING A FILE-BASED APPROACH

a) Advantages

153. The native access approach applies all advantages described in Section 3.1.2.3.

b) Disadvantages

154. Regarding the disadvantages of this approach with native access all disadvantages already described in Section 3.1.2.3 apply.

155. Furthermore, within this data extraction approach, the reporting is dependent on the T+1 file report of the DLT market infrastructure until a final linkage between the file-based reported data and the extracted blockchain data is possible.

156. The second method within this native approach follows a quite similar procedure as mentioned in the beforehand paragraph. Instead of providing the data in file format, the DLT market infrastructure provides an API with respective endpoints that is called by the regulator to retrieve the further trade-relevant and personal data. The following figure (Figure 13) describes the procedure that the regulator can take to extract the transaction data directly from the Ethereum blockchain and linking it with further relevant data provided by the DLT market infrastructure via an API.

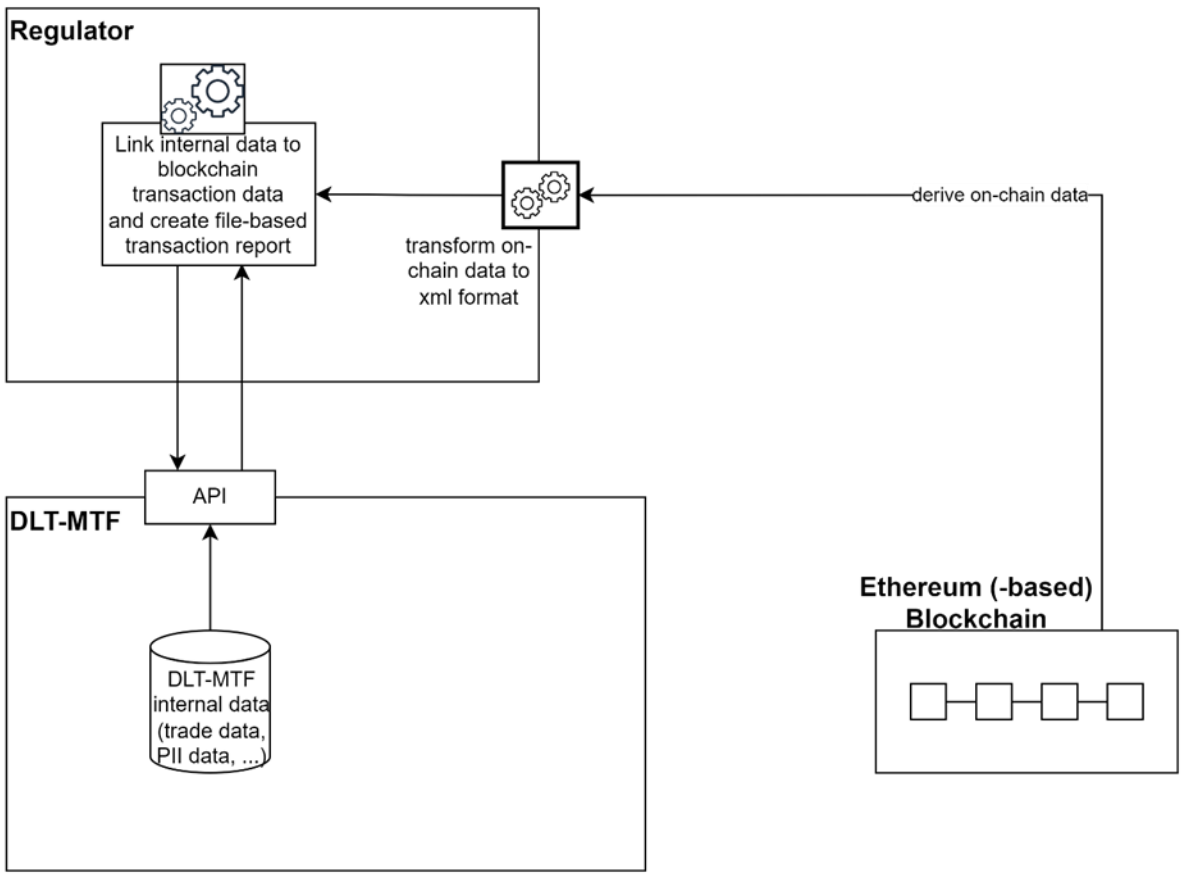


FIGURE 13: DIRECT ON-CHAIN DATA EXTRACTION BY REGULATOR METHOD TWO

a) Advantages

157. Similar advantages to the described ones of Corda's native access in Section 3.1.2.3 apply within the native access to extract data from the Ethereum blockchain. Due to the design of the Ethereum blockchain as a public blockchain protocol, especially the limitation to tampering and real time reporting are main advantages under this approach. Due to the necessity of separation of off-chain and on-chain data, real-time reporting is not achievable in practice. On-chain data can be derived in near-real-time, while off-chain data is handed in by DLT market infrastructures and therefore cannot be acquired at the same pace as on-chain data.

b) Disadvantages

158. Although there exist major advantages of limited tampering and less effort for DLT market infrastructures, indisputable disadvantages that drive the complexity for this approach must be evaluated. The already described disadvantages under Section 3.1.2.3 apply to this approach as well.

159. Additionally, regulators cannot match the data directly to a full dataset only from the blockchain data, since PII data and further relevant data for trades of the DLT financial instruments are not stored on the chain. Besides the effort for linking the data to a full dataset on regulator's reliability, this could lead to further disadvantages regarding data privacy and data security.

3.2.2.4 Specificities of public, permissionless DLTs, such as Ethereum

160. Besides the previously described advantages and disadvantages, the usage of the Ethereum DLT may cause further effects that are different from private and permissioned DLTs such as Corda and Hyperledger fabric, due to its public nature.

161. Within the Ethereum DLT anyone can publicly propose changes to the underlying protocol, which are reviewed by a group of developers and any other interested persons. The proposed changes, called Ethereum Improvement Proposals (EIPs), target changes that should be made to the underlying implementation of the Ethereum protocol.

162. Both, regulators and DLT market infrastructures, may be impacted especially when following the API-based and native access approaches, since the impact of a possible decision-making role within such update processes is rather minimal as an organisation. While Corda is designed to follow governance processes designed towards the interests of the organisations that are part of the respective network, Ethereum's governance focus is aligned on the decisions made by the majority of the existing public community.

163. As an effect from the previously described specificity, both DLT market infrastructures and regulators may be impacted in a rather negative way, if changes to the protocol layer

of the Ethereum DLT are made, which result in efforts to update the implementations in relevance to the transaction reporting process.

3.2.3 Conclusion on transaction data extraction methods in Ethereum

Criteria	File-based approach	API-based approach	Native access to each DLT network
Access rights for different participants and regulators	Not applicable – the DLT market infrastructure sends the reports to the regulator. Close to current transaction reporting approach e.g., under MiFIR Art. 26, EMIR Art. 9 or SFTR Art.4..	There is no difference in access rights for regulators, participants and other externals, since the data in Ethereum and other public DLTs is accessible by anyone.	In general, relevant transaction data is stored and publicly available to the regulator on-chain in Ethereum. However, off-chain data of particular importance to proper supervision (e.g., personal information or static instrument data) need to be submitted by the DLT market infrastructure in a T+1 file and need to be linked to the on-chain transaction data which nullifies advantages the native access has. There is no difference in access rights for regulators, participants and other externals, since the data in Ethereum and other public DLTs is accessible by anyone.

Security assurance in terms of confidentiality, integrity, and availability	Files can be exchanged in an encrypted way and are transmitted on a secured channel.	Depending on the DLT market infrastructure specific implementation of the API. Theoretically the same security levels as within the file-based approach are achievable.	Depending on the native technology offering of each Distributed Ledger Technology. Theoretically the same security levels as within the file-based and API-based approach are achievable.
Timing and frequency of the updates, such as real-time or near-real-time	No real-time / near-real-time reporting, in-line with current transaction monitoring processes of T+1 transaction reporting. Reporting files may be made available at a pace which is to be agreed between the DLT market infrastructure and the regulator. Although the risk remains, that the frequency of the reporting is lower.	Near-real-time access is possible.	Real-time access is possible.
Technical standards supported	ISO 20022 standard and RTS 22 XML schema (or a new schema covering RTS 22 data and additional fields specific to the DLT, e.g., DTI).	There is currently no standard existing. Implementations based on ISO 20022 standards and RTS 22 XML schemas are also possible.	There is currently no standard existing.

Tight vs. loose coupling	Loosely coupled.	Tightly coupled to the DLT market infrastructure implementation.	Tightly coupled to the DLT implementation.
Governance implications	Regulators control the standard XML schema, agree on updates, and control the timing of implementation of the updates to give stakeholders time for implementation.	Regulators and DLT market infrastructures are depended on implementations in the Ethereum public blockchain and need to follow changes.	Regulators have no control on changes decided on the DLT protocol. DLT market infrastructures have no control on changes of the DLT network, in which they are part of.
Access to the history and latest status of the transaction	Always the latest transaction reports on daily basis and can build a historical database.	Depending on the DLT market infrastructure specific implementation. Theoretically, the same level of access as within the file-based approach, to the history and latest status of the transaction is achievable.	Depending on the DLT market infrastructure specific implementation. Theoretically, the same level of access as within the file-based approach, to the history and latest status of the transaction is achievable.
Access to transaction corrections and cancellations	Regulators receive data on corrections and cancellations daily.	Depending on the DLT market infrastructure specific implementation. Theoretically, access to transaction corrections and cancellations can be	Depending on the DLT market infrastructure specific implementation. Theoretically, access to transaction corrections and cancellations can be

		achieved in near real-time.	achieved in real-time.

TABLE 2: CONCLUSION ON DATA EXTRACTION IN ETHEREUM

3.3 Hyperledger Fabric

3.3.1 Background

164. Hyperledger Fabric is a private and permissioned DLT platform, which is open source and was established in 2015 under the Linux Foundation. Its modular and configurable architecture allows for a variety of use cases, including but not limited to banking and financial services. Hyperledger Fabric also finds application in use cases pertaining to the representation of supply chains.

165. Hyperledger Fabric has a two-component ledger system comprised of the world state and the transaction log, i.e., the blockchain. While the world state can be considered the ledger's database as it describes its state at any given point in time, the blockchain keeps track of all Hyperledger transactions, which have resulted in the state of the ledger as it currently exists.[58] This means, the blockchain determines the world state. The relationship between the components is outlined in Figure 14 below.

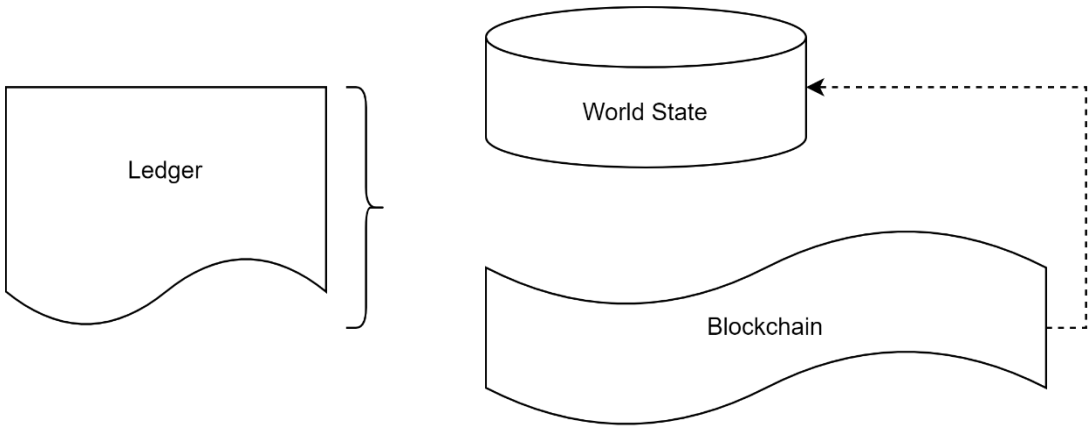


FIGURE 14: HYPERLEDGER FABRIC'S LEDGER COMPONENTS

166. The world state holds the current state of assets of a network. It can be queried to better understand the types of assets currently stored on it. In the context of the DLT Pilot Regime, the world state of a Hyperledger Fabric network could be queried to understand the types

[58] https://hyperledger-fabric.readthedocs.io/en/release-2.5/blockchain.html#what-is-hyperledger-fabric

of DLT financial instruments traded on it or to see, which network participant owns which DLT financial instruments. The world state is implemented as a database and different options exist depending on the type of data to be stored in it.

167. Popular options for this database are LevelDB and CouchDB.[59] If a market participant submits a Hyperledger Fabric transaction to the network, it must first be signed as defined within the endorsement policy before it updates the world state and is stored in the applicable database.[60] In order to retrieve ledger state information, the world state database can be easily queried.[61]

168. The blockchain is a sequence of interlinked blocks and each of its blocks contains a sequence of Hyperledger Fabric transactions. Overall, the blockchain contains historical records of how assets on the network change. In doing so, the blockchain determines the world state, as mentioned in Paragraph 165. Each of the block headers making up the blockchain itself contains a hash of the block's Hyperledger Fabric transactions and a hash of the prior block's header. This design links ensures network and data security.[62]

169. Members of the Hyperledger Fabric network must enrol through a trusted Membership Service Provider (MSP).[63] Hyperledger Fabric's default MSP uses X.509v3 certificates as identities.[64] These identities are crucial as they determine which accesses and permissions network participants have. X.509v3 certificates are then issued and managed by a Certificate Authority (CA).[65] The issued X.509v3 certificates contain the CA's digital signature and link the network participant to the network participant's public key. Hence, the certificate is used to identify network participants whenever they interact with the network.

170. A Hyperledger Fabric network can also consist of multiple CAs, depending on network configuration. Multiple CAs may be helpful insofar as that they can lead to improved efficiency due to a distribution of the overall workload. Typically, the CA is operated and maintained by a trusted third party. This could be a government agency, a regulator, or as part of the DLT Pilot Regime, a DLT market infrastructure.

171. A further element of Hyperledger Fabric are policies. Policies are used to define decision making on the network. In doing so, they outline the rights a network participant has.[66] For instance, policies prescribe the accesses network participants have or the amount of network participants that must be in agreement regarding updates to a channel.

---

[59] https://hyperledger-fabric.readthedocs.io/fa/latest/ledger.html
[60] https://hyperledger-fabric.readthedocs.io/en/release-2.5/ledger/ledger.html#world-state
[61] https://hyperledger-fabric.readthedocs.io/en/release-2.5/ledger/ledger.html#world-state-database-options
[62] https://hyperledger-fabric.readthedocs.io/en/release-2.5/ledger/ledger.html#blockchain
[63] https://hyperledger-fabric.readthedocs.io/en/release-2.5/blockchain.html
[64] https://hyperledger-fabric.readthedocs.io/en/release-2.5/identity/identity.html
[65] https://hyperledger-fabric.readthedocs.io/en/release-2.5/identity/identity.html#certificate-authorities
[66] https://hyperledger-fabric.readthedocs.io/en/release-2.5/policies/policies.html#what-is-a-policy

172. Channels are useful in order to engage in private and confidential Hyperledger Fabric transactions.[67] All Hyperledger Fabric transactions occur on channels and each channel contains its own Hyperledger Fabric transaction ledger.[68] Typically, various organisations come together to form a channel on Hyperledger Fabric.[69]

173. Smart contracts on Hyperledger Fabric are called chaincodes. Chaincodes are used to handle the business logic previously defined by the network participants. [70] Hence, chaincodes, among other things, are involved in the execution of Hyperledger Fabric transactions. They, for instance, assess whether a certain Hyperledger Fabric transaction is valid.

174. Chaincodes further always possess a so-called endorsement policy. Endorsement policies define which network participants within the respective channel must run the chaincode and endorse the execution results in order for Hyperledger Fabric transactions to be considered valid. In order to validate the Hyperledger Fabric transaction, the validating peers ensure that it contains a sufficient number of endorsements from the expected sources, while also checking the validity of these sources. [71]

175. Hyperledger Fabric provides Software Development Kits (SDKs), in order to enable clients, i.e., end users to interact with the underlying chaincodes and blockchain protocol. The SDKs are provided in three programming languages, which are Java, Golang, and Node.js. They enable clients to execute functions to initiate Hyperledger Fabric transactions, retrieve historical Hyperledger Fabric transaction data, or query information from the underlying ledger. This can be done to retrieve block data of a specific numbered block.

176. Another crucial part of Hyperledger Fabric networks are peers or peer nodes. Peers are in charge of managing ledgers and chaincodes as well as transaction proposals and endorsements.[72] There are various types of nodes in Hyperledger Fabric that differ in their responsibilities.

177. So-called endorsers, i.e., endorsement nodes are responsible for verifying and approving Hyperledger Fabric transactions. To do so, they, among other things, run the associated chaincode. The number of peers which must endorse a specific Hyperledger Fabric transaction depends on the applicable endorsement policy.[73]

178. So-called orderers, i.e., ordering nodes, are responsible for collecting relevant information regarding Hyperledger Fabric transactions from the endorser nodes. They take

[67] https://hyperledger-fabric.readthedocs.io/en/latest/channels.html
[68] https://docs.aws.amazon.com/managed-blockchain/latest/hyperledger-fabric-dev/hyperledger-work-with-channels.html
[69] https://hyperledger-fabric.readthedocs.io/en/release-2.4/network/network.html#what-is-a-blockchain-network
[70] https://hyperledger-fabric.readthedocs.io/en/release-1.3/chaincode.html
[71] https://hyperledger-fabric.readthedocs.io/en/latest/endorsement-policies.html
[72] https://hyperledger-fabric.readthedocs.io/en/release-2.5/peers/peers.html
[73] https://hyperledger-fabric.readthedocs.io/en/release-2.0/Fabric-FAQ.html

care of arranging submitted Hyperledger Fabric transactions into a well-defined sequence and package them into the blocks making up the blockchain.[74] Similar to CAs, multiple orderers can be part of a Hyperledger Fabric network depending on network specifications and requirements. Orderers do not see the contents, i.e., the data contained within the Hyperledger Fabric transactions, as they merely order rather than open them.

179. Furthermore, in cases where one wants to bypass the orderers altogether, this can be achieved multiple ways. For instance, Hyperledger Fabric's private data feature can be utilised.[75] Data, which is stored in private collections, is shared only between network participants that are permitted to view said data.[76] This is used to keep certain Hyperledger Fabric transactions and their associated data secret from other network or even channel participants.

180. Consensus on Hyperledger Fabric is reached by following three distinct steps, endorsement, ordering, and validation. The endorsement step is driven by the respective endorsement policy upon which a Hyperledger Fabric transaction is endorsed by other network participants. Secondly, within ordering, the endorsed Hyperledger Fabric transactions are ordered as agreed upon and in the way in which the eventually will be committed to the ledger. Lastly, validation refers to taking a block containing ordered Hyperledger Fabric transactions and validating their correctness, by, among other things, checking the applicable endorsement policy and assessing whether double spending has occurred.[77]

3.3.2 Approach methodology

181. To explore transaction data extraction approaches within the Hyperledger Fabric DLT (HLF), best practice IT architecture designs of enterprise scale DLT's were used as the main source. Furthermore, existing examples from the market were investigated and considered. As a result, it must be noted that the existing architectural examples as well as the methods for extraction of transaction data do not represent exhaustiveness and further methods and architectural designs might be possible and exist.

182. To understand how data extraction from HLF can be made, it is necessary to discover potential entry points before the description of different extraction approaches. The entry points to query for transactions in HLF are client applications which use SDKs, provided within the HLF software stack. Querying for information on the ledger can be done in

[74] https://hyperledger-fabric.readthedocs.io/en/release-2.2/orderer/ordering_service.html#:~:text=Ordering%20service%20nodes%20receive%20transactions,and%20package%20them%20into%20blocks https://hyperledger-fabric.readthedocs.io/en/release-2.2/orderer/ordering_service.html#:~:text=Ordering%20service%20nodes%20receive%20transactions,and%20package%20them%20into%20blocks.
[75] https://hyperledger-fabric.readthedocs.io/en/release-2.0/Fabric-FAQ.html
[76] https://stackoverflow.com/questions/74313642/hyperledger-fabric-transaction-payload-visible-in-ordering-node
[77] https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf

various ways. A common method is querying a transaction by its transaction identifier (Transaction ID), through a *GetTransactionByID* method.

183. Further, it is quite common that data is stored on the ledger in key value pairs, where the key could be a unique identifier and the value an attribute. An exemplary and high-level implementation of a security transfer to a new owner could implement the financial instrument as:

Key	Value
“stock_1”	{"isin": "...", "owner":"new_owner"}

TABLE 3: KEY-VALUE REPRESENTATION OF A HIGH-LEVEL SECURITY TRANSFER TO A NEW OWNER

184. Observing the above example, another entry point to query for transaction data is the key field. By querying the key field, the respective values can be derived and used for further processing.

3.3.2.1 File-based approach

185. Very similar to the previously described approach on file-based data extraction under the Corda chapter, HLF the DLT market infrastructure queries its state database, in HLF the CouchDB, to extract information on transactions, transforms the data and creates an XML file report in alignment to the RTS 22 reporting schema, which is sent to the regulator.

186. To query for transaction data, HLF offers the flexibility to define query functions that can also perform more complex queries on the respective data. An exemplary query could fetch data by querying for an identifier, but also using further parameters, such as financial asset owners and other necessary fields. The possible queries and their complexity must be defined inside the chaincode.

187. The DLT market infrastructure will use the provided SDKs by HLF to perform queries for extracting the transaction data. A client must be established on top of the DLT market infrastructure’s node infrastructure executing the query functions described in the channel-wide implemented chaincode. The derived results from a query response will then be transformed into the requested formats, such as XML, to generate the RTS 22 aligned report. The following figure (Figure 15) shows the exemplary extraction of file-based transaction reporting in HLF:

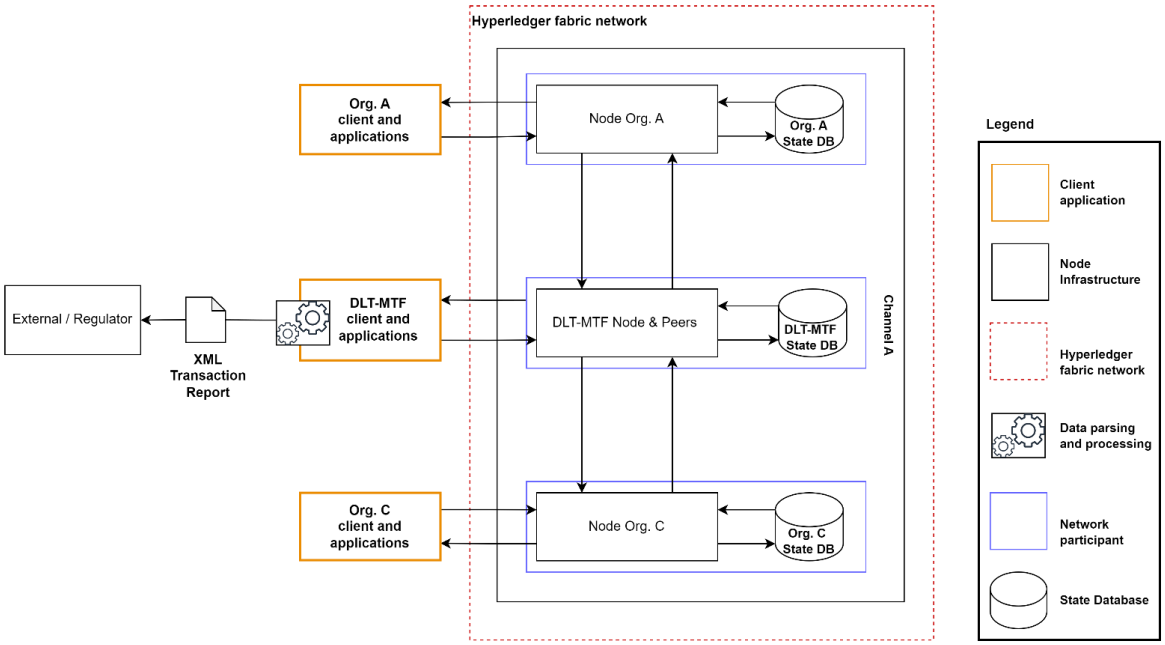


FIGURE 15: FILE-BASED DATA EXTRACTION APPROACH IN THE HYPERLEDGER FABRIC DLT

a) Advantages

188. Within the file-based approach in HLF, all previously described advantages of Paragraph a) in Section 3.1.2.1 are applicable.

b) Disadvantages

189. The disadvantages within HLF are very similar to the disadvantages that are previously described in Section 3.1.2.1.

3.3.2.2 API-based approach

190. The API-based approach to extract transaction data in HLF is very similar to the previously described approach of extracting transaction in Corda in chapter 3.1.2.2. DLT market infrastructures query their maintained state database, extract the data by using the relevant and necessary SDKs and methodologies provided within the HLF code-base and extract the data.

191. Further, the extracted data will be processed, transformed, and incorporated into an API-layer that provides respective endpoints, which can be called by regulators in order to fulfil necessary operations of transaction data extraction according to the RTS 22 schema.

192. The following figure (Figure 16) describes an exemplary and high-level architecture description, in which DLT market infrastructures provide an API to external stakeholders, such as regulators, in order to get access to extracted transaction data of financial instruments trades within a HLF DLT network.

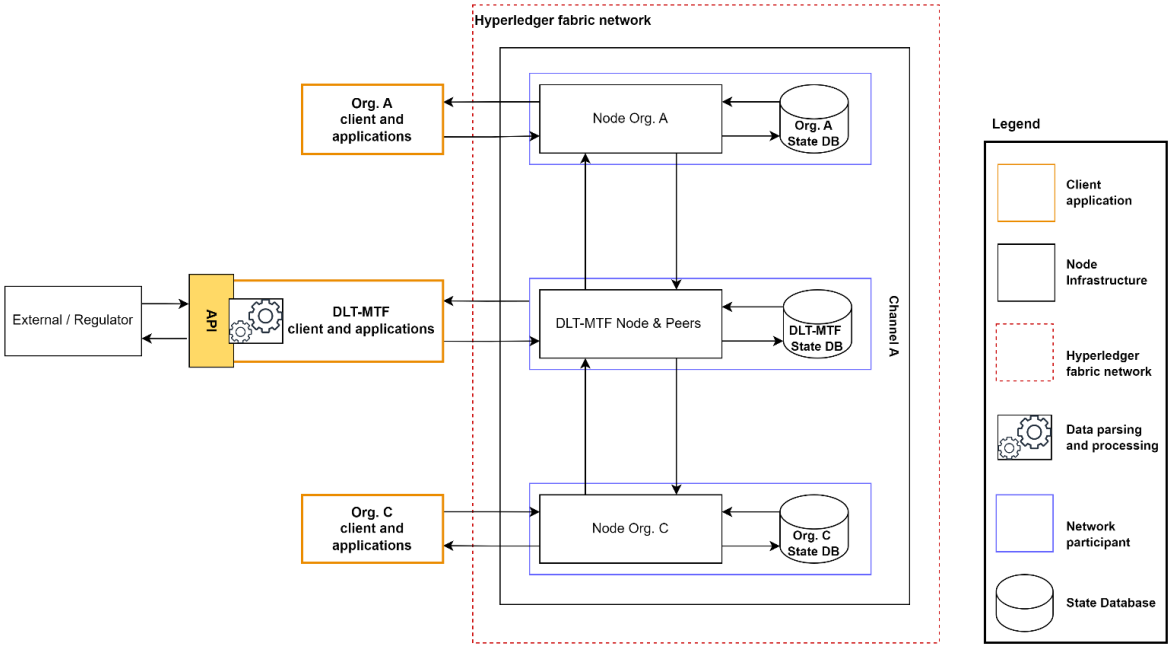


FIGURE 16: API-BASED TRANSACTION DATA EXTRACTION APPROACH IN THE HYPERLEDGER FABRIC DLT

a) Advantages

193. All advantages that were previously described in the Corda chapter under the Section 3.1.2.2 apply to the advantages within the HLF DLT as well.

b) Disadvantages

194. All disadvantages that were previously described in the Corda chapter under the Section 3.1.2.2 apply to the advantages within the HLF DLT as well.

3.3.2.3 Native access to each DLT and DLT network

195. Within the native access approach, in which external stakeholders, such as regulators, natively access each DLT network by themselves to extract transaction data, regulators could maintain a node within a HLF network and have access to each of the channels in which financial instruments are traded.

196. With this approach, regulator's nodes and state databases will be synced to the latest state of the ledger and therefore hold the most relevant information on financial transactions within the channel and network. Therefore, they can query and extract the transaction data from their implemented state database. Afterwards the data must be transformed into the necessary format and provided to the internal regulator's internal reporting infrastructure.

197. Data processing after data extraction from the state database can be done in several approaches, such as in the creation of a XML file in compliance with the RTS 22 reporting schema or via provision of an API that offers the data through implemented and callable endpoints.

198. The following figure (Figure 17) describes the process of transaction data extraction from a HLF network by natively accessing each network and extracting the data in form of an XML file:

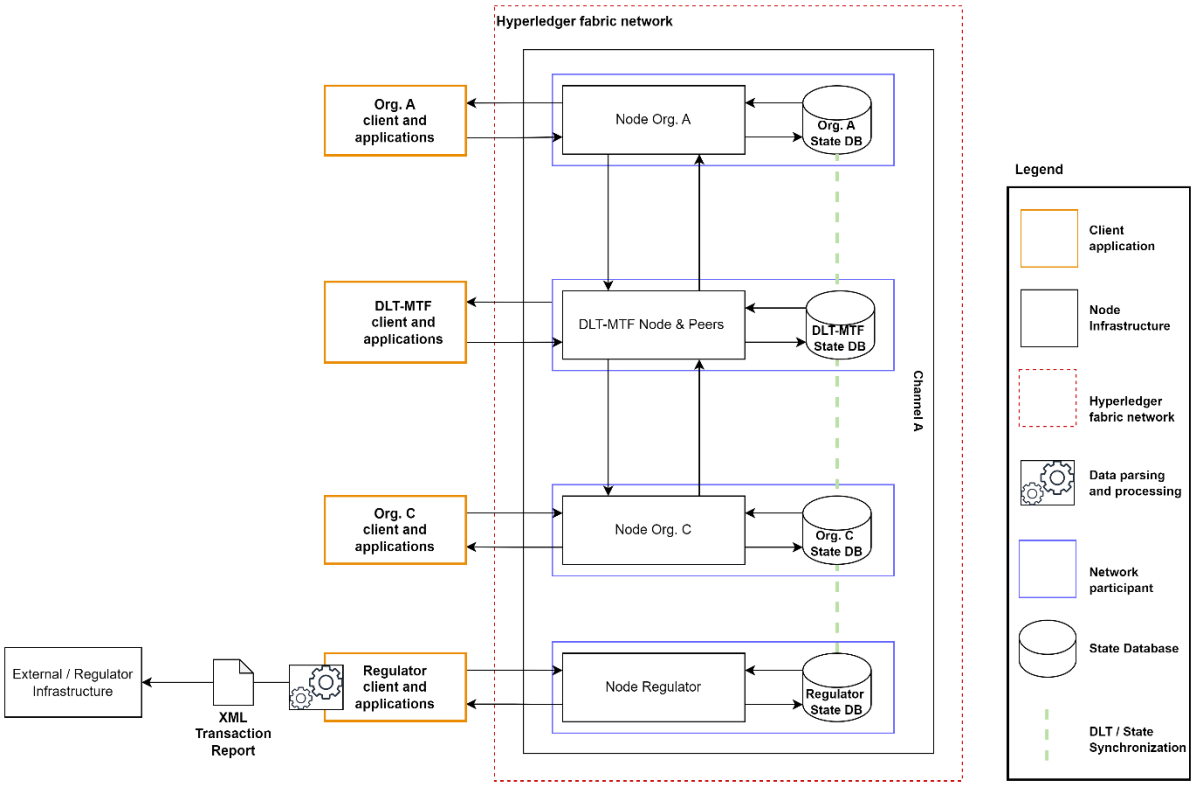


FIGURE 17: NATIVE ACCESS TO EACH BLOCKCHAIN NETWORK IN THE HYPERLEDGER FABRIC DLT AND FILE-BASED TRANSACTION DATA EXTRACTION

199. The following figure (Figure 18) describes the process of transaction data extraction from a HLF network by natively accessing each network and extracting the data in through provided API endpoints.

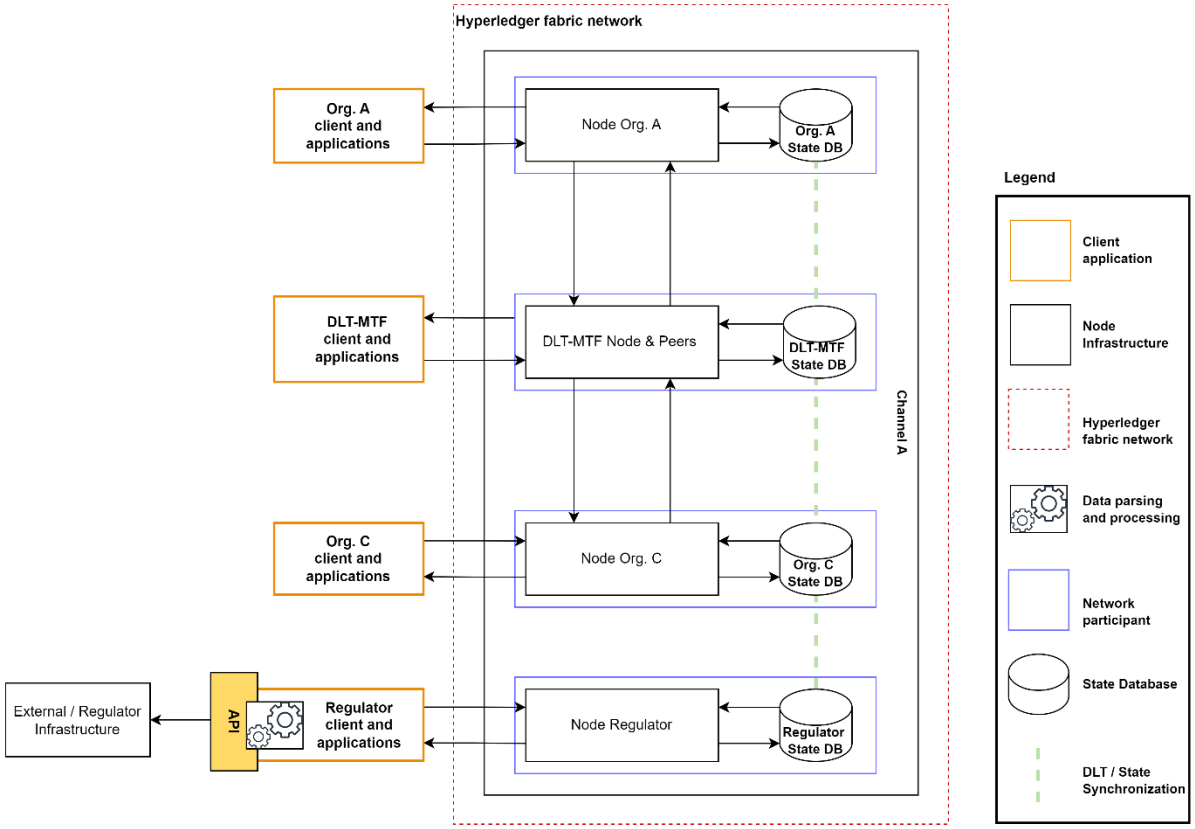


FIGURE 18: NATIVE ACCESS TO EACH BLOCKCHAIN NETWORK IN THE HYPERLEDGER FABRIC DLT AND API-BASED TRANSACTION DATA EXTRACTION

a) Advantages

200. All disadvantages that were previously described in the Corda chapter under the Section 3.1.2.3 apply to the advantages within the HLF DLT as well.

b) Disadvantages

201. The disadvantages already described under the Corda Section in 3.1.2.3 apply for HLF as well.

202. The design of participants being part of channels expresses an impact for DLT market infrastructures and as well for regulators. At first, the impact expresses to an additional many-to-many onboarding process. DLT market infrastructures and regulators must be part of each active channel within the network, which follows the respective use cases of DLT financial instrument trading. Therefore, they maintain within their state databases

several tables of transaction data for each channel. The latter results in a more complex data extraction due to the access to different tables within the databases.

203. The effect herein is that all tables must be investigated, and the data must be consolidated in order to generate the respective transaction reports. Since this is a general specificity to HLF and comes to effect within all of the discussed approaches, a detailed discussion is done in Section 3.3.2.4.

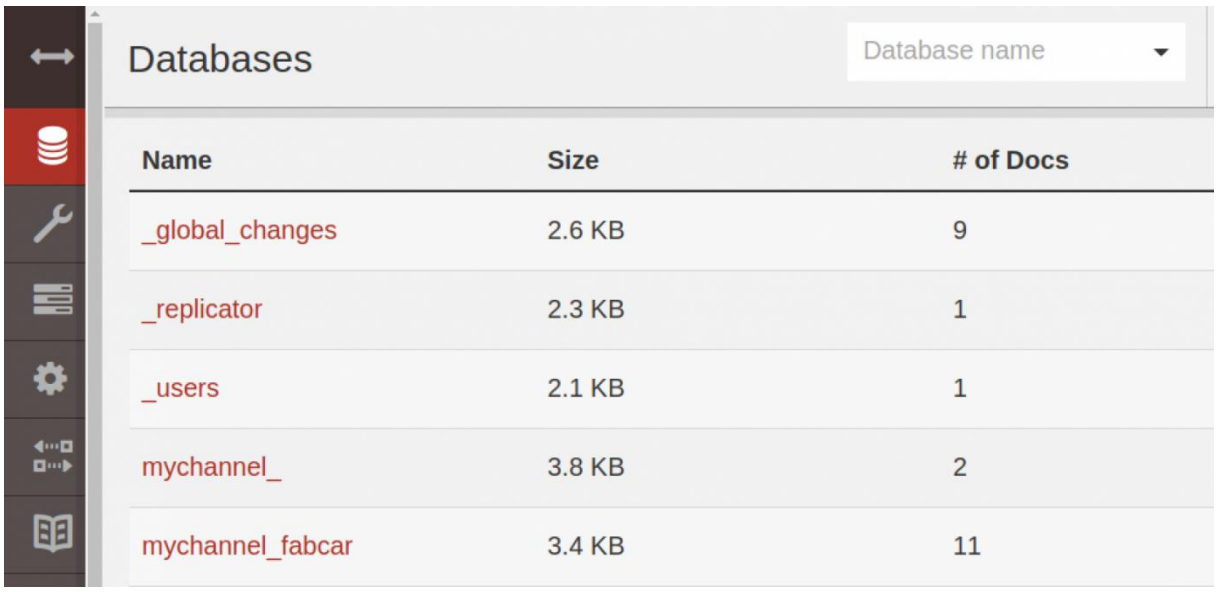
3.3.2.4 Specificities to HLF blockchain networks

204. HLF's concept of implementing an IT architecture which ensures privacy and separation through channels impacts the view on use cases of trading of DLT financial instruments and therefore it is important to highlight these circumstances.

205. In any of the beforehand described approaches, DLT market infrastructures must be onboarded on any existing channel within a network, where chaincode processing trades in DLT financial instruments is executed, in order to get the full dataset of all trades within the overall network. Additionally, and regarding the approach in which regulators have access to each DLT network, regulators have to be onboarded to each channel in the network as well.

206. Apart from the governance related and technical effort of onboarding to each channel, the state of the ledger regarding the respective channel is held within each node's state database. A separate database for each channel exists. If a DLT market infrastructure or a regulator is member of several channels, the state database infrastructure will maintain several separate databases for each channel.

207. The following figure (Figure 19) is an example, in where an organisation is part of the channels named "mychannel" and "mychannel_fabcar". In effect of the separation, a data query must be made to each separate database and the consolidation of the queried data to extract a full transaction report dataset must be processed in addition.



[78]

FIGURE 19: STATE DATABASE EXAMPLE OF AN ORGANISATION PARTICIPATING IN SEVERAL SEPARATE CHANNELS IN THE HYPERLEDGER FABRIC DLT

3.3.3 Conclusion on transaction data extraction in the HLF DLT

Criteria	File-based approach	API-based approach	Native access to each DLT network
Access rights for different participants and regulators	Not applicable – the DLT market infrastructure sends the reports to the regulator. Close to current transaction reporting approach e.g., under MiFIR Art. 26, EMIR Art. 9 or SFTR Art.4	Each regulator needs access rights granted on each DLT market infrastructure system to access the respective API.	Read access to the transactions processed via DLT market infrastructures is possible through the operation of a node by the regulator and therefore direct involvement in the respective transaction flows of

[78] https://blockgeeks.com/guides/hyperledger-fabric-tutorial-2/ (visited 13.04.2023)

			the smart contract is intended.
Security assurance in terms of confidentiality, integrity, and availability	Files can be exchanged in an encrypted way and are transmitted on a secured channel.	Depending on the DLT market infrastructure specific implementation of the API.	Depending on the native technology offering of each Distributed Ledger Technology.
Timing and frequency of the updates, such as real-time or near-real-time	No real-time / near-real-time reporting, in-line with current transaction monitoring processes of daily transaction reporting.	Near-real-time access is possible.	Real-time access is possible.
Technical standards supported	ISO 20022 standard and RTS 22 XML schema (or a new schema covering RTS 22 data and additional fields specific to the DLT, e.g., DTI).	There is currently no standard existing. Implementations based on ISO 20022 standards and RTS 22 XML schemas are also possible.	There is currently no standard existing.
Tight vs. loose coupling	Loosely coupled.	Tightly coupled to the DLT market infrastructure implementation.	Tightly coupled to the DLT implementation.
Governance implications	Regulators control the standard XML schema, agree on updates, and control the timing of implementation of the updates to give	Regulators have no control on changes of the DLT market infrastructures implementations and need to follow changes on the	Regulators have no control on changes decided on the DLT protocol. Further, regulators have no control on changes of the DLT market

	stakeholders time for implementation.	client-side implementation.	infrastructure's implementations. DLT market infrastructures may have control on changes of the DLT network, in which they are part of, dependent on their level of involvement in steering of the network.
Access to the history and latest status of the transaction	Always the latest transaction reports on daily basis and can build a historical database.	Depending on the DLT market infrastructure specific implementation.	Depending on the DLT market infrastructure specific implementation.
Access to transaction corrections and cancellations	Regulators receive data on corrections and cancellations daily.	Depending on the DLT market infrastructure specific implementation.	Depending on the DLT market infrastructure specific implementation.

TABLE 4: CONCLUSION ON DATA EXTRACTION IN THE HLF DLT

3.4 Cost-benefit analysis

3.4.1 Stakeholder description

3.4.1.1 DLT market infrastructure

208. Under the DLT Pilot Regime, DLT market infrastructures must keep track of all transactions and all their relevant details. Therefore, DLT market infrastructures must ensure the availability of all transaction details which are required by regulators for supervisory purposes. In order to achieve this, DLT market infrastructures must implement the required infrastructure software components to ensure the compliance of the transaction reporting data, irrespective of following a file-based or API-based approach, with the RTS 22 schema (or a new schema covering RTS 22 data and additional fields

specific to the DLT, e.g., DTI, wallet address, smart contract address, transaction hash, and other) under the MiFID II/MiFIR transaction reporting regime.

3.4.1.2 Regulators

209. Regulators authorise trading venues and DLT market infrastructure applicants to operate under the DLT Pilot Regime and supervise their trading activities. Further, regulators maintain IT systems, which enable processing of transaction information and, in particular, detection of market abuse. Therefore, regulators require the full RTS 22 transaction information and may need additional data to capture DLT specific transaction information.

3.4.1.3 Market participants

210. Market participants refer to the individuals, firms and entities that engage in buying, selling, and trading of financial instruments and with regards of this study, on a distributed ledger technology. Market participants are impacted by the liquidity, efficiency, and stability of financial markets and therefore, they are subject to regulatory oversight and compliance requirements. In terms of DLT-based transaction reporting processes, market participants might not be directly involved or obtain an active role.

3.4.2 Cost-benefit analysis of the file-based extraction of transaction data

3.4.2.1 Description of the file-based approach

211. The file-based transaction reporting approach follows the process, in which DLT market infrastructures create a transaction reporting file in compliance to the RTS 22 schema or a new schema, covering RTS 22 data and additional fields specific to distributed ledger technologies (e.g., DTI), under the MiFID II/MiFIR transaction reporting regime. The respective transaction reporting file is created in XML file format and transmitted to the regulator. The transmission is conducted in an encrypted and secured manner via channels that implement a file transfer protocol to enable transmission.

3.4.2.2 Cost-benefits of the file-based approach

Infrastructure to extract and transform data from the DLT into RTS 22 XML Schema	
Cost to DLT market infrastructure:	DLT market infrastructures may incur some one-off and on-going infrastructure costs for extracting on-chain and off-

One-off costs On-going costs	chain information, and processing the transaction data, into the RTS 22 XML file schema. Corda and HLF may allow to store both on-chain and off-chain information in a single database, while Ethereum requires to merge off-chain information with on-chain information. DLT market infrastructures may incur some one-off and on-going infrastructure costs for extracting and processing the transaction data into the RTS 22 XML file schema or a similar enriched version. The magnitude of costs additionally depends on the design of the DLT market infrastructure system and may increase or decrease with cost driving factors, such as the amount and complexity of implemented infrastructure components or the use of external providers.
Cost to regulator: On-going costs One-off costs	Regulators may incur on-going costs in order to maintain and operate secure infrastructure for file exchange and transmission channels. Further, regulators will face limited one-off costs to adapt the current RTS 22 schema in terms of enrichment with blockchain specific information, such as the DTI.
Cost to market participants:	No costs identified.
Benefits:	Besides the costs for transforming data from DLT networks and / or internal IT systems, no further costs occur for DLT market infrastructures and market participants to maintain capabilities and infrastructure for transaction reporting. The file-based approach offers flexibility and ease of adaptability to changing DLT characteristics, resulting in minimal changes and cost implications.

	A standardised and interoperable transaction reporting process results in benefits for all participants through improved efficiency, transparency, and ease of use. There are no additional costs associated with the development and maintenance of an API for the DLT market infrastructure. The interface for conversion and handling of transaction data relies on proven standards (XML) which features automated format validation at no additional costs (XML schema).
Governance	
Cost to regulator On-going costs	Regulators are in control of the standard XML schema, agree on updates and therefore control the timing of the implementation of updates to give stakeholders enough time for implementation. The need for all involved regulators to agree on changes and considering the impact on the industry ensures stability of the interface definition. Regulators need to follow up discussions on possible change requests at the level of the ESMA Standing Committees / Task Forces, and therefore dedicate resources to the relevant ESMA groups to involve in the ongoing governance, which may incur on-going costs.
Cost to DLT market infrastructure One-off costs On-going costs	DLT market infrastructures follow the agreed standards that are under control of the regulators and process adoptions in case of updates, agreed by the regulator. The adoption towards the regulatory framework may incur one-off costs for DLT market infrastructures, whereas updates do not occur on a frequent basis. Exemplary, the XML schema for transaction reporting in its current version did not change since 2019[79]. The magnitude of the related one-off costs is dependent on the effort needed to adopt

[79] https://www.esma.europa.eu/data-reporting/mifir-reporting#mifir-transaction-reporting

	the update. Cost-driving factors may be for example the size and the complexity of the respective update. Ongoing costs are associated with the control framework and IT service continuity for the transmission and exchange of XML files via SFTP.
Cost to market participants:	No costs identified.
Benefits	Following the EU regulator's governance process and maintaining of a standardised XML schema for transaction reporting leads to benefits for all participants, regulators, DLT market infrastructures and market participants. This results in an interoperable and stable transaction reporting interface.
Security assurance	
Cost to regulator: On-going costs	Regulators incur on-going costs for maintaining secured channels to enable an encrypted file transmission and exchange with DLT market infrastructures a manner that the availability, integrity and confidentiality of the data is ensured"
Costs to DLT market infrastructure: One-off costs On-going costs	DLT market infrastructures must ensure the security of the client side, in order to maintain an encrypted and secure file transmission. The implementation as well as the maintenance will incur one-off and on-going costs.
Costs to market participants:	No costs identified.
Benefits	No additional costs occur for DLT market infrastructures and market participants to maintain secure transmission and exchange of transaction reporting data. Therefore, all stakeholders are continuously able to follow an already standardised, secure and interoperable transaction reporting process.

	Furthermore, regulators can maintain the secure, interoperable and stable process of transaction reporting, without the need for major adaption of infrastructure and enhancement of further capabilities. Regulators ensure the establishment of availability, integrity, and confidentiality through a secure FTP infrastructure.

TABLE 5: COST-BENEFIT ANALYSIS ON THE FILE-BASED APPROACH

3.4.3 Cost-benefit analysis of the API-based extraction of transaction data

3.4.3.1 Description of the API-based approach

212. In the API-based approach, DLT market infrastructures provide APIs with specific endpoints that can be accessed and called by the regulator to extract relevant data on the conducted transactions. To provide any data, DLT market infrastructures must access relevant data from their infrastructure, such as state database, and process it further for provision in an API layer.

213. Regulators conduct calls to the exposed endpoints on the API layer through a client and must process the called data into their internal systems. Regulators need to get access to the same level of information as through the file-based transaction reporting approach. Therefore, APIs must allow regulators to extract all transaction details (enriched RTS 22, e.g., with DTI) of all participants on the DLT market infrastructure, during a certain day or any other flexible period of time, but at least on a daily basis. DLT market infrastructures must ensure the proper performance of the API's functions which are required to be used by regulators.

3.4.3.2 Cost-benefits of the API-based approach

Infrastructure to extract and transform data from the DLT into an API providing access to RTS 22 – like data	
Cost to DLT market infrastructure: One-off costs On-going costs	DLT market infrastructures may face API integration as well as implementation costs, which are one-time expenses related to the development, testing, and integration of the API and which may also include customisation and configuration. An example for implementation efforts may be the development of an API to support the required data

	for the transaction reporting, as well as the linking of previously on- and off-chain separated data sets. Furthermore, the process of mapping and/or transforming off-chain and on-chain data to meet regulatory reporting requirements of the regulator, may result in one-off costs. These costs are likely to be similar to the process of extracting and processing transaction data within the file-based approach. On-going costs for DLT market infrastructures may be associated with data transmission, error handling, and regular monitoring. The operation and maintenance of hardware as well as software required to support the API, may result in additional on-going costs for the DLT market infrastructures. DLT market infrastructures must ensure that they are compliant with the regulatory requirements established by authorities, which may lead to on-going expenses
Cost to regulator: One-off costs On-going costs	The hardware, software, and network resources required to establish connectivity to the API, will incur one-off costs. The magnitude for these is dependent on the number of APIs provided by DLT market infrastructures to which regulators must connect. Training and expertise are required for personnel to utilise the API and extract the respective transaction data. This will result in additional on-going costs. Both, one-off and on-going costs are multiplied when each DLT market infrastructure uses its own API, and no standard API can be offered. As previously described among the different technologies, there are currently no existing standard for APIs and smart contracts. As technology continues to evolve rapidly, it is essential to stay informed and up to date on the latest advancements in the respective DLT. Therefore, it is crucial to prioritise the

	ongoing education and training of individuals involved in transaction reporting. This includes investing in resources such as workshops, training sessions, and seminars to keep them informed and updated on the latest developments in DLT.
Cost to market participants:	No costs identified.
Benefits:	Usage of APIs may reduce the effort which is required for data transformation and therefore could result in further benefits, such as cost savings. Although, the reduction of effort is only reached, in case the API is able to natively convey the expected RTS 22 data in addition with potential future amendments (e.g., DTI).
Governance	
Cost to regulator One-off costs On-going costs	Regulators will have to define or to be involved in the definition of the APIs, at least with respect to the associated requirements and specifications. Maintaining the control framework even in the event of updates on the DLT market infrastructure side leads to staff costs and additional expenses. Similar to the infrastructure section of this approach's cost-benefit analysis, it is essential to stay informed and up to date on the latest advancements in the respective DLT, which may lead to further on-going costs towards education of staff.
Cost to DLT market infrastructure On-going costs	DLT market infrastructures will have to be involved in the definition of the API(s), as they will have to implement the API(s) in compliance with the regulatory requirements. Providing resources for defining the APIs may cause governance related costs for DLT market infrastructures. The identified costs for DLT market infrastructures are the same as the on-going costs expected for DLT market

	infrastructures in the governance section of the file-based approach. By following the to be established API requirements and specifications, necessary to comply with the guidelines outlined in the regulator's control framework, DLT market infrastructures can demonstrate their commitment to compliance, mitigate risks, and maintain the integrity of financial transactions.
Cost to market participants:	No costs identified.
Benefits	There are no specific benefits for both, regulators and DLT market infrastructures, identified in relation to governance within the API-based approach.
Security assurance	
Cost to regulator: One-off costs On-going costs	Regulators must ensure the security at the software client side of the API (such as penetration testing and other), which may incur one-off costs during the first-time implementation, as well as on-going costs to operate and maintain latest security standards and needs.
Costs to DLT market infrastructure:	DLT market infrastructures must ensure the security in accordance with the integrity, availability, and confidentiality of the extracted transaction data throughout the implemented API layer. This may result in one-off costs for the first-time implementation, as well as on-going costs to operate and maintain the latest security standards and needs. The above-mentioned costs also express within the implementation and maintenance of the API's server-side, which is accessed by regulators API clients.
Costs to market participants:	No costs identified.

Benefits	There are no specific benefits for both, regulators and DLT market infrastructures, identified in relation to security within the API-based approach.

TABLE 6: COST-BENEFIT ANALYSIS ON THE API-BASED APPROACH

3.4.4 Cost-benefit analysis of the native access approach to each DLT network

3.4.4.1 Description of the native access to each DLT network approach

214. The native access transaction reporting approach is the process, in which external stakeholders, such as regulators, are directly involved in each of the DLT networks in which financial instruments are traded. Involvement may be denoted in various ways, dependent on the nature of the DLT's IT architecture.

215. Within Corda and HLF, to extract transaction data, involvement of regulators could be denoted as operation and maintenance of a node that is part of the network and included within the transaction flow. Involvement regarding Ethereum could be denoted as operation and maintenance of node infrastructure, or using a third-party provider, to access the data from the public DLT.

216. Within the native access approach within the Corda and HLF DLT, DLT market infrastructures deploy, maintain, and update the necessary smart contracts for trading financial instruments and within these smart contracts, the required functionality for querying transaction data and involvement in transaction flows. Regulators need to either access the querying functionalities or must be included in the transaction flows, to be able to extract transaction data from the DLT.

217. Within the Ethereum DLT, DLT market infrastructures must provide PII data of their customers as well as further trade-relevant data which is not stored on the blockchain to the regulator in form of a file or API and results in the burden, that DLT market infrastructures as well as regulators cannot rely solely on the native access in any case.

3.4.4.2 Cost-benefits of the native access approach

Infrastructure to extract and transform data from the DLT through natively accessing each DLT and DLT network	
Cost to DLT market infrastructure: One-off costs	The setup of necessary infrastructure for the DLT market infrastructure to operate a node in the DLT network incurs one-off costs, as well as on-going costs for the operation and maintenance of the infrastructure.

On-going costs	By providing regulators with access to the Smart Contracts/Chaincode functionality, to enable them to query transaction data or involving them into transaction flows, DLT market infrastructures may incur one-off costs. DLT market infrastructures may incur on-going costs associated with providing maintenance and updates of the Smart Contract/Chaincode functionality to regulators to ensure the extraction process remains effective. Regarding the Ethereum DLT, DLT market infrastructures will incur one-off costs as well as on-going costs for maintenance, to ensure secure and stable transmission of PII and further trade-relevant data to regulators. In addition, if relevant data is partially stored on the blockchain, an API or file-based approach must be implemented on top of the provided native access to cover the transmission of the relevant data – which will lead to further costs.
Cost to regulator: One-off costs On-going costs	The setup of necessary infrastructure to enable the regulator to operate a node in the DLT market infrastructures network incurs one-off costs. The magnitude of these is dependent on the underlying DLT and its necessary components to implement. Integration costs involve implementing interfaces, protocols, and technical configurations to enable communication and data exchange for the extracted transaction data, which may result in additional one-off expenses. On-going maintenance and operational costs are associated with operating a node and its relevant system components. These costs depend on the complexity of the necessary implementation and the number of system components of the underlying DLT to maintain and operate. An Ethereum node may require less monitoring effort, compared to the Corda or HLF IT architecture, in which participants might implement several components, dependent on the targeted flexibility, such as implementing several Certification Authorities and Ordering Nodes in HLF. Further, high on-going costs may occur to adapt to

	changes made over time by each Distributed Ledger Technology involved, as well as for the adaption to each DLT market infrastructure specific implementation. For data storage outside the network, databases/warehouses need to be implemented as well as the required software for file export needs, resulting in additional one-off costs for implementation and on-going costs for their operation and maintenance. Training and expertise of staff are required in each involved distributed leger technology and further in each specific implementation by the DLT market infrastructure, for proper operation and maintenance of the infrastructure, resulting in additional on-going costs.
Cost to market participants:	No costs identified.
Governance	
Benefits:	Expected benefits for regulators could be expressed through improved regulatory oversight and increased transparency in long-term, due to the real-time monitoring of transactions to detect potential market abuse for ensuring the integrity and stability of DLT-based financial markets. However, this is only possible at the expense of very high cost for adapting to each DLT technology, each DLT market infrastructure specific implementation, and follow-up associated IT development lifecycle over time. DLT market infrastructures may face less effort for the overall transaction reporting process since the regulator extracts the transaction data by itself. Effort remains for transmitting any data stored off-chain, such as personal information in the case of Ethereum, which translates into the need to support and API-based or file-based approach on top of the provided native access.
Cost to regulator	Same as in the governance section of the API-based approach, on-going costs may incur for regulators in terms

One-off costs On-going costs	of staffing and education to stay up to date with latest technological advancements of the respective DLTs. Furthermore, if regulators have the possibility to be involved in decision processes of private and permissioned DLT consortia, they may face high on-going costs related to staffing and partial steering. Although, it is expected that involvement for regulators may not be possible.
Cost to DLT market infrastructure	No additional costs in relation to governance identified.
Cost to market participants:	No costs identified.
Benefits	Regulators may be involved in steering processes of private and permissioned DLT consortia, which could lead to some level of cost control for changes and updates. While DLT market infrastructures have no responsibility regarding the extraction and transmission of on-chain transaction data, benefits arise for both, DLT market infrastructures and regulators, due to the involvement of regulators within the steering processes. This could lead to reduced effort regarding coordination of changes and updates for all involved stakeholders. Although, these benefits are not in effect whenever trade-relevant data is partially stored off-chain.
Security assurance	
Cost to regulator: One-off costs On-going costs	One-off and on-going costs may incur for regulators to ensure the security of the regulator's node. An implementation to establish well known security measures for operating nodes within a DLT network could result in one-off costs, while keeping up with latest vulnerabilities and latest security updates may incur on-going costs for regulators.
Costs to DLT market infrastructure:	DLT market infrastructures must ensure the integrity, availability, and confidentiality of transaction data throughout the overall trading process. Within the Corda

One-off costs	DLT, regulators are involved in the transaction flow of the smart contracts, and in the HLF DLT, regulators are involved in the transaction flow through participation in the respective channels. The implementations regarding a secure transaction flow and the ability to securely querying of vaults and state databases may incur one-off costs for implementation from a DLT market infrastructure's perspective. In the Ethereum DLT, DLT market infrastructures must ensure that the PII- and further trade-relevant data is transmitted securely either file- or API-based.
Costs to market participants:	No costs identified.
Benefits	Within the approach in where regulators natively access the DLT network, a technical separation between PII / further off-chain data and on-chain transaction data could be accomplished at the expense of the added complexity due to the need to link both datasets.

TABLE 7: COST-BENEFIT ANALYSIS ON THE NATIVE ACCESS TO EACH DLT AND DLT NETWORK

3.4.4.3 Conclusion on the cost-benefits analysis

218. The cost-benefit analysis compares the three presented approaches of transaction reporting under investigation of one-off costs, on-going costs and their related benefits. All of the approaches target at the provision of all the relevant data in terms of financial transactions to regulators, to enable supervisory.

219. As a summary, the file-based approach is seen as the most stable and cost-effective for both, regulators and DLT market infrastructures due to its standardised and secure nature, as well as its proven stability.

220. Further, the API-based approach, while technically feasible, would necessitate additional implementation costs and complexity, for example in the linking of off-chain to on-chain data in public DLTs, without clear benefits.

221. Native access to each DLT and each DLT network, despite its potential for providing in-depth insights, is deemed to be the most expensive and complex due to the need for off-chain data storage, interoperability issues, and the difficulty in keeping up with changes in DLT technologies for both, DLT market infrastructures as well as for regulators.

222. The findings of this study primarily focus on evaluating the costs associated with various data access solutions in the context of the EU DLT Pilot Regime. Based on this cost-centered analysis, the final recommendation is to implement, as a first step, the transaction reporting exemption within the DLT Pilot using the file-based approach, supplemented by enriching RTS 22 files with relevant to allow optional DLT exploration for regulators.

223. This additional measure is seen as a complement to the existing file-based system. However, the inherent intent behind this exemption should not be overlooked. The exemption is meant to foster the exploration and eventual adoption of innovative strategies for data access. This approach acknowledges the fact that while cost is a significant factor, the ultimate goal is to find a more effective and efficient method for accessing data.

3.5 Recommendations regarding relevant regulatory information to be included

3.5.1 Additionally relevant fields for market surveillance purposes

224. Additional relevant fields for market surveillance purpose could be *Transaction Hash, From* and *To*[80].

3.5.2 Additionally relevant fields to perform on-chain analysis

225. On-chain analysis typically involves collecting data from the DLT network, structuring, and cleaning the data to use statistical and analytical methods to draw conclusions. On-chain analysis within private DLTs such as Corda and Hyperledger fabric is limited to the access rights an external party is granted.

226. With Ethereum as a public DLT protocol, there remains the possibility to extract further information that might be useful for on-chain analysis. The following table (Table 8) describes potential additionally relevant fields to perform on-chain analysis in the Ethereum DLT.

Field	Potential value for on-chain analysis
Wallet Addresses	Every Ethereum transaction is associated with a wallet address. Reporting on the wallet addresses involved in the trading of financial instruments can help identify potential market manipulators or insider traders.

[80] See Report on the DLT Pilot Regime - Study on how financial instrument transactions are registered in various Distributed Ledger Technologies (Chapter 3.2.4.3)

Smart Contract Addresses	Ethereum smart contracts are self-executing programs that can automate financial transactions, such as the issuance of financial instruments like equity, bonds, or derivatives. Reporting on the smart contract addresses involved in these transactions can help regulators better understand the issuance and trading of these instruments.
Timestamps	Ethereum blocks are timestamped, providing a chronological record of transactions. Including timestamps in on-chain analytics can help regulators detect suspicious patterns of trading activity or market manipulation.
Gas Fees	Ethereum transactions require the payment of gas fees, which are paid in ether (the cryptocurrency native to Ethereum). Reporting on gas fees can help identify high-frequency traders and market makers[81], as well as provide insight into the liquidity of various financial instruments.
Quantity/ Current Total Supply	These fields identify the current floating amount of the asset that was traded. Common token standards for financial instruments, such as ERC-20 and ERC-721, allow minting and burning mechanisms which may modify the current floating supply of a token. Linked to the current market price, this information can be used to monitor market liquidity and identify potential price manipulation.
Token ID	Ethereum allows for the creation and trading of custom tokens associated with an ID (e.g., ERC-721 NFTs), which can represent financial instruments or other assets. Reporting on the token IDs involved in trades can help regulators track the trading of specific financial instruments and their associated tokens.

TABLE 8: ADDITIONAL RELEVANT FIELDS IN ETHEREUM TO PERFORM ON-CHAIN ANALYSIS

[81] High-frequency traders and market makers may process batched transactions in one transaction, resulting in high amounts of gas fees. Although, it is important to note, that high gas fees could also be associated with complex smart contract interactions.

3.6 Recommendations regarding on-chain analysis scenarios and tools

3.6.1 Relevant scenarios of on-chain analysis to complement transaction data monitoring

227. On-chain analysis can be a powerful tool for regulators to complement transaction data monitoring when overseeing the use of DLT financial instruments. It provides valuable insights into the activity on the blockchain by utilizing various data sources including transaction history recorded on the blockchain that contains information about the sender, recipient, amount transferred, and timestamp. Along with smart contract activity investigation to gain insights into executed transactions and token transfers, regulators can identify patterns and anomalies on the flow of funds, transaction patterns, and address interactions that may indicate market manipulation, insider trading, or other fraudulent activities. For instance, regulators could identify suspicious trading patterns by tracking vast amounts of transfers between wallets, monitor token movements to identify potential pump-and-dump schemes, and investigate abnormalities in smart contract execution to uncover potential fraud.

228. One way on-chain analysis can assist regulators is by tracking token transfers to known bad actors through monitoring transaction records from and to blacklisted wallets or addresses associated with criminal activities. Additionally, on-chain analysis can help regulators trace the source of funds used to purchase tokens by examining transaction histories and correlated address balances on the blockchain. By leveraging this information, regulators can identify a suspicious address which received notable amount of funds from multiple wallets along with their timestamp information, leading to the discovery of possible illegitimate activities such as money laundering or terrorist financing activities.

229. Another critical aspect is analysing smart contract activity to ensure compliance with regulations and identify potential vulnerabilities or flaws in the code by monitoring the interactions between contracts, smart contract code, transaction data, and the contract's state changes. For instance, scrutinizing the changes in the state of a smart contract generated for executing a lending protocol and the associated flow of funds during its execution help determine unanticipated behaviors caused by potential security loopholes or improper input validation as well as irregularities such as insufficient collateral requirements or faulty interest rate calculations. Additionally, required identity verification measures, e.g., procedures for requesting user identification info in smart contract code (such as identification claims in the ERC-3643 standard) can be analyzed to prevent large transfers being made from unverified wallets and to verify compliance with the regulatory requirements on AML and KYC. Therefore, on-chain analysis can help regulators prevent potential issues from arising and protect investors.

230. Moreover, regulators can use on-chain analysis to monitor token price manipulation by identifying suspicious trading patterns or large transactions from transaction history as well as token transfers and address interactions that may artificially inflate or deflate token prices. By detecting coordinated market manipulation through examining trading patterns across addresses, smart contract interactions, and the timing, regulators can promptly protect investors and prevent potential issues from arising.

231. Finally, on-chain analysis can help regulators identify market trends and patterns by identifying increased market activity or swift spikes in trading based on transaction volume and frequency information. Additionally, token transfers between addresses on blockchain to identify frequent token fluctuations can clarify how tokens are being used and indicate if markets may emerge. This information can help regulators make informed decisions about regulatory oversight and intervention.

232. In conclusion, on-chain analysis is an essential tool for regulators to ensure compliance with regulations and laws related to the use of DLT financial instruments. By complementing transaction data monitoring with on-chain analysis, regulators can better protect consumers and ensure market integrity.

3.6.2 State-of-the-art tools and their capabilities

233. Regulators can utilise state-of-the-art tools for on-chain analysis, however, the level of access that regulators have to these tools depends on the specific permissions granted to them. Especially in Corda and HLF, the tools may need to be built by consortia or network participants and then provided to the regulators, since no publicly available state-of-the-art tools for on-chain analysis (and especially for transactions in DLT financial instruments) exist.

234. In Ethereum and other public DLTs, some state-of-the-art-tools exist to perform on-chain analytics. The following table (Table 9) states common tools, while the list does not claim to be exhaustive and many other tools may exist.

Tool Name	Capabilities
Chainalysis Products Reactor KYT Storyline Business Data	Provides blockchain analysis and compliance software to government agencies and businesses. Enables tracking of cryptocurrency transactions for anti-money laundering (AML) and know-your-customer (KYC) purposes, as well as flow-of-funds analysis, to identify suspicious activity and provide risk assessment reports.

Dune Analytics	Offers a platform for accessing, analysing, and visualising Ethereum data based on SQL data models. Enables users to publicly as well as privately create and share data queries and dashboards and integrating them into internal solutions via an API. Dashboards are commonly utilised to provide insights for projects in decentralised finance (DeFi).
Flipside	Similar to Dune Analytics
Nansen Products Portfolio Research Query	Nansen provides products for users to track their token portfolios, offers research insights based on on-chain data and additionally allows businesses to programmatically access on-chain data via SQL models, to create dashboards.
Messari	Provides market intelligence and data for the cryptocurrency industry. Offers research reports, data feeds, and an API for accessing crypto data. Further, Messari offers asset tracking of cryptocurrencies and tokens.

TABLE 9: STATE OF THE ART TOOLS TO PERFORM ON-CHAIN ANALYTICS