

Blockchain-oriented Inter-organizational Collaboration between Healthcare Providers to Handle the COVID-19 Process

Ilyass El Kassmi, Zahi Jarir

Computer Science Engineering Laboratory
Faculty of Sciences, Cadi Ayyad University
Marrakech, Morocco

Abstract—Collaborative business activities have aroused great interest from organizations because of the benefits they offer. However, sharing data, services, and resources and exposing them to external use can prevent organizations involved in collaboration from being engaged. Therefore, the need for advanced mechanisms to ensure trust between the different parties involved is paramount. In this context, blockchain and smart contracts are promising solutions for performing choreography processes. However, the seamless integration of these technologies as non-functional requirements in the design and implementation phases of inter-organizational collaborative activities is a challenging task, as reported in the literature. Consequently, we aim in the proposed approach to extend the modeling and implementation of the choreography lifecycle based on service-oriented processes. This is fulfilled by integrating blockchain transactions and smart contract calls, to allow collaboration and interoperability between different entities while guaranteeing trust and auditability. Moreover, to conduct this extension efficiently we use a BPMN choreography diagram combined with Finite State Automata to ensure a meticulous modeling which targets the processes' internal interactions. Hyperledger Fabric is used as a permissioned blockchain for proof-of-concept implementation. A use case of COVID-19 collaborative processes is used to experiment our approach, which aims to guarantee a fluid collaboration between healthcare providers and epidemiological entities at a national scale in Morocco.

Keywords—Blockchain; inter-organizational collaboration; choreography; permissioned blockchain; business process management; COVID-19

I. INTRODUCTION

Collaborative business activities have aroused great interest from organizations because of the benefits they offer. This success is due to the technological advances that the interconnected digital world has known. However, sharing data, services, and resources, and exposing them for external use can prevent organizations involved in the collaboration from being engaged. Therefore, the need for advanced mechanisms to ensure trust between the different parties involved is paramount. In this context, blockchain and smart contracts are promising solutions for performing choreography processes. They aim to create effective and secure infrastructures to ease coordination between interacting organizations, ensuring greater achievement

of common goals, and avoiding data redundancy caused by the multitude of exchanged messages in conventional schemes.

However, the seamless integration of these technologies as non-functional requirements (NFRs) [1] in the design and implementation phases of inter-organizational collaborative activities is a challenging task, as reported in the literature. This is the case for the inter-organizational healthcare domain, as an example, which is characterized by its complexity both in terms of the management of sensitive data and the heterogeneity of the activities involved in the collaboration process. Blockchain can be introduced to overcome the problem of lack of secure links connecting healthcare organizations including and not limited to public healthcare centers, private clinical centers, ministry regulation entities, specialized centers (dental, laboratory, imagery) and insurance organisms. Moreover, the privacy property characterizing the blockchain provides protection either for exchanged data through security policies or for healthcare professionals' identities as participants in the network.

Blockchain technology has gained attention over the last few years. It has extended its foundation to cover, in addition to the commonly known implementations of digital currencies, the integration of decentralized applications (DApps) [2] in a broad range of industries. This technology aims to overcome the reliance on a centralized authority to certify information's integrity and ownership, and provides a platform for decentralized business logic ensuring transparency and immutability of exchanged data between the distributed interacting parties. Advances in this technology have encouraged organizations to gain confidence in sharing information with other entities without requiring an intermediate trust third party [3]. In order to support the control and exchange of digital assets in the distributed ledger, for common blockchain infrastructures, smart contracts [4] are introduced for this purpose. These contracts are defined as executable programs that allow manipulation of data and functions within the blockchain. They aim to facilitate, verify, or negotiate a contract agreement under a set of conditions to which users agree. When these conditions are met, the agreement terms are automatically carried out.

In collaborative business processes [5][6], the integration of smart contracts aims to create an effective and secure infrastructure dedicated to facilitate collaboration between

interacting organizations and achieve common goals. Furthermore, the tamper-proof property and the exhaustivity provided by the logging characteristic of the blockchain allow the interacting organizations to track who, what, and when transactions are performed and shared to the ledger, according to the agreed security policies. This property guarantees auditability and thus, develops trust between participants without the need for an intermediate trust authority.

In previous contributions, we proposed approaches to tackle the modeling of both functional requirements (FRs) and NFRs [7] and their integration as part of composite services in order to produce an optimal composition, while guaranteeing a prior validation of user properties using model checking verification. Currently, we aim to enhance this approach by providing the ability to handle collaborative processes, ensuring that each process can be executed independently unlike one-time execution in the context of orchestrated service composition. The second important contribution of the proposed approach is the integration of blockchain transactions and smart contract execution as a potential implementation to support the collaboration of business processes based on a service-oriented architecture (SOA). The proposed solution is supported by a proof-of-concept applied to the healthcare domain to meet the emerging collaboration requirements raised by the coronavirus disease 2019 (COVID-19) pandemic.

Our current contribution aims to propose an approach that has the advantages of extending the modeling and implementation phases of the choreography lifecycle by integrating blockchain transactions and smart contract executions, to allow collaboration and interoperability between different entities while guaranteeing trust and auditability. As a case study, this approach was applied to COVID-19 collaborative processes integrating healthcare providers and epidemiological entities at a national scale in Morocco. The proposed solution has the following advantages:

- Proposing a modeling process based on two phases: first using the Business Process Modeling Notation (BPMN) choreography diagram to design the interactions between collaborative business processes, then using Finite State Automata (FSA) to design the internal interactions of the corresponding service compositions.
- Integrating the blockchain technology to tackle common challenges of collaborative business processes such as trust and auditing.
- Providing a support for transactions and smart contract executions, adapted to the context of collaborative business process as service compositions.
- Taking advantage of the security properties and policies granted by the integrated blockchain solution in order to enhance inter-organizational collaboration.
- Proposing a proof-of-concept based on a real-world problematic of inter-organizational collaboration among healthcare entities during the COVID-19 pandemic.

As stated, the proposed approach is based on SOA, consequently, it supports a service-based integration that can be adapted to different blockchain platforms. In this contribution,

we implemented a proof-of-concept based on the Hyperledger Fabric architecture [8]. This technology improves the security measures by integrating identity management, access control over assets and resources, and offering the ability to create and use special sub-networks restricted to users with granted permissions. Although we highlight some security improvements offered by the selected technology, our research scope does not focus on vulnerabilities and blockchain-related security threats. The focus is preliminarily oriented to the aspects of modeling and integrating the blockchain as a NFR, to tackle the challenges of inter-organizational business process collaboration.

The remainder of this paper is organized as follows. Section 2 presents a brief overview of the blockchain technology concepts. In Section 3 we expose some interesting contributions tackling business process collaboration challenges using blockchain and SOA. In Section 4, we present a novel approach for handling the integration of permissioned blockchain technology to manage decentralized business processes. Section 5 presents a COVID-19 related case study to demonstrate the behavior of this approach. Finally, in Section 6, we summarize the suggested approach and highlight future works and upcoming perspectives.

II. A BRIEF BLOCKCHAIN OVERVIEW

Blockchain is a decentralized ledger that records the provenance of digital assets, using a set of protocols and cryptography technologies to securely store data on the network [9]. It can be defined as a sequence of blocks that store all exchanged transactions in a peer-to-peer network [10]. Each block contains, in addition to the stored data, the hash of the previous block and the timestamp of its production. Thus, all the peers store a complete and exact copy of the database, ensuring immutability, decentralization, anonymity and auditability, which promotes trust between participants without the need for a central authority. These cited characteristics, among other remarkable features, have opened the opportunity for diverse research fields around the blockchain, and provided the foundations for a number of application domains, beyond cryptocurrency, such as non-fungible tokens (NFTs) [11], Internet-of-Things (IoT) implementations [12-14], decentralized finance platforms (DeFi) [15], blockchain gaming, and DApps [2].

According to the literature and recent advances in blockchain-oriented technologies, different architectures have been proposed that vary according to the implementation purpose. The most popular architecture is the public blockchain, also known as permissionless blockchain [16], it is a publicly accessible network that does not require special roles for connecting or sending transactions. Thus, any person can access the ledger and interact with it (i.e., read and write data). A public blockchain ensures full transparency of the ledger while maintaining user anonymity. Additionally, no particular participant has total control over the data, as block validation is achieved through consensus mechanisms implying all mining nodes. In contrast, private blockchains are restrictive blockchain architectures that operate in closed networks [17]. They are mostly adopted in enterprises, organizations, and applications that require handling sensitive data, with the aim of using this

technology for internal usage. Moreover, private blockchain is often considered a more centralized architecture than public blockchain because the maintainability of the network is commonly ensured by a single authority. Hyperledger Fabric [18], Sawtooth [19], and Corda [20] are examples of private blockchains. Finally, the permissioned blockchain constitutes a mixed architecture that combines both public and private blockchains. They support numerous customization options, such as identity management, allowing only certain activities to be performed on the network.

Each blockchain platform uses a specific consensus mechanism to achieve the necessary agreement for storing data on a distributed ledger. A consensus mechanism is a procedure through which all the peers of the blockchain network reach a common agreement about the present state of the distributed ledger [21]. A large variety of consensus protocols have been described in the survey [22], such as Proof-of-Work (PoW) [23], Proof-of-Stake (PoS) [24], Proof-of-Authority (PoA) [25], and practical Byzantine fault tolerance (PBFT) [26]. PoW is the most popular consensus mechanism allowing miners to compete to solve the puzzle in order to validate a block and append it to the ledger. Bitcoin [27] and Ethereum [28] are the most popular implementations of public blockchains that use PoW consensus.

Provided that the robustness of the consensus protocols is guaranteed, smart contracts are deployed on the distributed ledger to manipulate data and functions using an analogy similar to classes/objects in object-oriented programming (OOP). Smart contracts can then be composed of attributes and methods, allowing better code structuration and business-oriented modeling of assets and their relationships. This allows the design and development of DApps in a fine-grained manner [4]. They can be used to automate the execution of an agreement so that all participants can be certain of the outcome, without involving any third party. They are also used to automate a workflow or trigger an action when specific conditions are met.

To support business process management (BPM) using blockchain technology, Mendling et al. proposed an interesting survey describing the main challenges and opportunities for this emerging technology [29]. They concluded by presenting seven future research directions based on the conducted research. Among these research directions, two are linked to our proposed approach: (1) the investigation of methods for analyzing and engineering business processes based on blockchain technology, and (2) the definition of appropriate methods for blockchain evolution and adaptation. Another interesting systematic literature review was presented by Garcia-Garcia et al., which aimed to identify opportunities and gaps in the area of collaborative business processes using blockchain technology [30]. They provided different comparative tables to illustrate the contribution scope of each research study. Among the 34 research papers studied, most of the contributions are oriented to public blockchain architecture using Ethereum, while only 15% use Hyperledger blockchain solutions. On the other hand, only 37% integrated process modeling and 45% used or extended BPMN, against 3% using languages based on state machines. The authors concluded that there is a rapid and growing interest in scientific communities and software industries to explore opportunities provided by

blockchain technology to improve the management of collaborative business processes in a decentralized manner. Similarly, Xu et al. conducted a comparative analysis of consensus mechanisms [31], and presented an approach incorporating smart contracts and PBFT consensus to address challenges regarding time, prejudice, and trust of process executions in the context of collaborative business processes and IoT. They concluded by highlighting the advantages of the voting mechanism designed to decrease the delay caused by prominence consensus such as PoW.

III. RELATED WORK

We previously stated that our current approach consists of modeling collaborative processes, building appropriate service-oriented systems, and integrating the blockchain as a service to guarantee decentralization for desired interactions through transactions or smart contract executions. Therefore, in this section, we present some interesting approaches and implementations that address the modeling and integration of blockchain technology for collaborative business processes.

Modeling blockchain-oriented collaborative processes constitutes an open challenge because of the variety of blockchain platforms offering multiple possibilities and usages according to the desired blockchain architectures and consensus mechanisms. Most of the proposed approaches tackling the process choreography using blockchain technology are implementing permissionless blockchains, particularly Ethereum, which represents the first Bitcoin's alternative for public blockchains. Weber et. al proposed one of the first approaches to address the lack-of-trust problem in collaborative business processes based on the blockchain [X]. They aim to map the business processes, designed using BPMN choreography modeling, onto a peer-to-peer execution infrastructure that stores transactions in the blockchain. Smart contracts can then be used as a direct implementation of the mediator process control logic. Authors presented the built modules such as the translator allowing to parse and convert BPMN choreography diagrams files to Solidity scripts, and the Triggers that connect the participants' internal processes with the blockchain. The authors highlighted some of the limitation of their approach based on the presented permissionless platform, which consists on the lack of privacy once the organizations public keys are known, which constitutes opportunities for competitors to inspect and track the data on the ledger. Another approach presented by Garcia-Bañuelos et al. consists on presenting another aspect surrounding the use of blockchain in business process executions, which is the optimization [Y]. They introduced a two-phase modeling based on BPMN processes translated into Petri Net models, then reduced to be converted into Solidity smart contracts. The purpose consists on optimizing costs by reducing gas consumption for overall collaborative business processes. On the other hand, a model-driven methodology for choreography-based systems adapted to the Ethereum blockchain was proposed by Corradini et al. [34]. They implemented all process phases from modeling to execution using a framework that takes BPMN choreography models as input and provides the appropriate Solidity smart contracts [35] adapted to Ethereum. To provide more specifications and capabilities associated with blockchain technologies while dealing with choreography

processes, the authors in [36] assessed the capabilities and limitations of current choreography modeling approaches, and then proposed new language concepts adapted to BPMN choreography processes with their appropriate operational semantics. Di Ciccio et al. proposed another interesting study [37] that presented a comparison between two model-driven approaches, 'Lorikeet' [38] and 'Caterpillar' [39], to deal with the design and implementation of blockchain-based collaborative process execution. Both approaches are focused on BPMN-style process models and implement the Ethereum blockchain.

On the other hand, Pourheidari et al. proposed a study investigating the applicability of the execution of a real-world untrusted business process on a permissioned blockchain [40]. They highlighted the advantages of using the permissioned blockchain for collaborative business processes, and then proposed an order processing model based on the BPMN standard, considering the particularities of Hyperledger Fabric as a blockchain platform and Hyperledger Composer as a framework. This study focused on implementing blockchain to support an existing business process, taking advantage of the modeling capabilities offered by Hyperledger Fabric in addition to the use of identity management to handle permissions. According to the cited contribution, each process is executed independently and cannot be automated. In contrast, our proposed approach is based on a service-oriented architecture that automates the execution of interacting services through a service composition, by integrating blockchain transactions as services. Authors Autili et al. presented an approach to address the problem of trust in service choreography [41]. They proposed blockchain technologies to support decentralized and peer-to-peer collaboration in a trustworthy manner. A model using a BPMN2 choreography diagram is presented to illustrate the peer-to-peer communication between interacting parties. The proposed solution allows trust-based coordination between participants and the verification of the correctness of exchanged messages by providing the ability to validate transactions and manage permissions. The interactions with the blockchain are however weakly expressed during the modeling phase. In comparison, we use in our approach a second phase of modeling based on automata to define, in a fine-grained manner, each interaction with the blockchain to fulfill decentralization requirements.

The following contributions highlight approaches of tackling the interactions with blockchain technology based on SOAs. An interesting contribution [42] proposed a taxonomy based on an analysis of the Blockchain-as-a-Service (BaaS) market, according to service characteristics, the support of

different distributed ledger technology (DLT) protocols and consensus mechanisms, and related pricing models for service provisioning. This study highlights the predominance of large IT service providers in the BaaS market, offering a variety of services to match the requirements of corporate customers, who prefer permissioned distributed ledger solutions, and prioritize quality characteristics such as performance, ease of use, and availability. In the same context, Daming et al. proposed a systematic review of recent research studies covering BaaS models [43]. They aimed to categorize the applied scenarios, trends, evaluated quality of service (QoS) factors, new challenges, and open directions on BaaS models in IoT management. According to the conducted study, the BaaS models are mainly applied to the network layer to manage the IoT environment more than other layers. In addition, security and privacy are two important factors for evaluating existing BaaS models in cloud-edge IoT environments. In accordance with these results, Zheng et al. proposed a platform for BaaS called NutBaaS [44]. This platform is based on a four-layer model and provides blockchain services over cloud computing environments, such as network deployment and system monitoring, smart contract analysis, and testing. It also provides some key built-in features, such as identity-chain profile management and smart contracts' security-related vulnerability detection. The proposed solution is technically complementary and based on Hyperledger Fabric. Comparably, our proposed approach uses specific services to interact with the ledger through simple transactions or smart contract execution. However, prior to this integration, we provide a fine-grained modeling process to describe the overall interactions of these blockchain-oriented services, to meet the existing service composition scheme and accomplish the associated business process. Another contribution presents a unified blockchain as a service platform (uBaaS) [45] to support both the design and deployment of blockchain-based applications. This solution combines a deployment as a service, aiming to avoid lock-in to specific cloud platforms, with a design pattern-as-a-service that aims to apply design patterns for data management and smart contract design to address the common scalability and security issues.

Finally, Table I presents a summary of the common aspects of the cited contributions. The covered compared characteristics are as follows: (1) the modeling used, (2) the type of blockchain architecture, (3) the platform used, (4) the implemented consensus mechanism, (5) the adaptation ability to more complex use cases, (6) the approach evolution to cover other blockchain architectures and platforms, and (7) the security attributes and additional behavioral NFRs covered when applicable.

TABLE I. A COMPARATIVE TABLE DESCRIBING CHARACTERISTICS OF CITED CONTRIBUTIONS

	[32]	[33]	[34]	[40]	[41]	[43]	[45]	Our proposed approach
Modeling	BPMN2 Choreography Diagram	BPMN Process Petri Net	BPMN2 Choreography Diagram	BPMN	BPMN2 Choreography Diagram	N/A	BPMN	- BPMN2 Choreography Diagram - Finite State Automata
Architecture	Public Permissionless	Public Permissionless	Public Permissionless	Permissioned	Permissioned	Permissioned	Public Permissionless	Permissioned
Platform	Ethereum	Ethereum	Ethereum	Hyperledger	Hyperledger	Hyperledger	Ethereum	Hyperledger
Consensus	Proof-of-Work	Proof-of-Work	Proof-of-Work	N/A	N/A	Kafka	Proof-of-Work	RAFT
Adaptation	Yes	Yes	Yes	N/A	Yes	Yes	Yes	Yes
Evolution	Yes	No	N/A	N/A	N/A	Yes	N/A	Yes
Security & Behavioral NFRs	N/A	N/A	- Role Management	- Privacy - Access Control	- Transaction Validation - Permission Management	- Profile Management - Smart Contract vulnerability detection	- Permission Management	- Identity & Role Management - Permission Management - Privacy - Custom Security NFRs

We notice in the comparative table that most of the presented works used either BPMN process diagram or BPMN2 choreography diagram to model the case study workflow. However, the designed modeling lacks expressiveness regarding the interaction between the current system and the blockchain. In contrast, we introduce automata-based modeling to describe in a fine-grained manner each interaction between the existing services fulfilling the business requirements and the desired blockchain transactions. Additionally, most of the presented contributions are oriented into a unique blockchain platform, which constitutes a strength when considering the advanced features characterizing the platform, but also a limitation when dealing with the implementation of heterogeneous blockchain architectures that might be implied in the collaborative inter-organizational business processes. Our proposed approach tackles this challenge by adopting a service-oriented architecture in which blockchain interactions are considered as services executed to fulfill predefined behavioral NFRs. Another concern that most cited papers has pointed to is security. Privacy and identity management are key security attributes that are integrated differently, depending on the adopted blockchain platform. For instance, Hyperledger Fabric integrates a built-in permission management system that aims to manage participant roles in accordance to business requirements. However, in Ethereum-based implementations, the permission module can be developed and integrated separately to control access permissions through smart contracts. In our approach, we have a limitless choice of security attributes to integrate. We can either use the built-in security policies provided by the chosen blockchain platform, such as access control lists and channels, and on the other hand, our service-based approach also allows us to define any desired security process as a behavioral NFR during the modeling phase, which, consequently, calls appropriate concrete services and inject them into the generated service composition.

According to the parameters listed in Table I, our contribution covers the overall phases allowing the production

of a reliable business process collaboration with the integration of blockchain interactions. The proposed approach combines the benefits of two powerful technologies: SOA and blockchain, and provides an advanced modeling process that combines them with an exhaustive definition of their interactions.

IV. PROPOSED APPROACH

Inter-organizational collaboration implies interactions between heterogeneous components to achieve external processes controlled by foreign entities. This raises numerous challenges depending on (1) the number of interacting organizations, (2) the complexity of published processes, and (3) the agreed policies to secure access and manage permissions. To overcome these challenges, model-driven methods can be helpful to rigorously describe the overall interactions using modeling languages, propose a fine-grained overview abstracted from any technical implementation, and allow reliable interoperability with the ability to customize code execution according to the runtime environment.

In this study, we propose an approach for designing and implementing collaborative business processes based on SOA. Additionally, we aim to introduce the ability to integrate blockchain transactions as a way to guarantee a higher level of decentralization while ensuring reliable collaboration. The proposed approach is supported by a proof-of-concept proposed to fulfill the emerging need raised by the COVID-19 pandemic, which consists of ensuring a wide collaboration between healthcare organizations. The proposed solution will allow healthcare actors to interact and exchange key medical information in accordance with the Moroccan governmental information systems and policies.

In previous contributions, we focused our research on modeling and generating business-oriented service compositions capable of fulfilling both FRs and NFRs [46]. The proposed solution integrates a modeling module that allows the design of interactions between desired services with the ability

to inject behavioral and quality-oriented NFRs, and then generates the optimal appropriate service composition. In the current approach, we aim to extend the scope of handling orchestrated composite services to allow their interactions with external business processes and, therefore, promote inter-organizational collaboration. Thus, we describe a business process as a complex business workflow that can be achieved by executing a service composition to fulfill certain predefined functional and non-functional requirements. A comprehensive modeling of these complex business processes is performed using the FSA. BPMN choreography diagram modeling is used from a wider perspective to define the interactions between multiple inter-organizational business processes. Moreover, we enhance our approach by introducing the ability to inject specific services, allowing interactions with the blockchain through transactions or smart contract executions. These blockchain interactions are covered from the modeling to the execution phases.

A. Modeling Collaborative Business Processes based on Automata

We previously stated that, in our preceding contributions [7, 46], we proposed approaches tackling the service composition challenges and integrating a modeling module based on FSA, allowing a graphical description of the desired FRs and NFRs. The generated composition is an orchestration of concrete services that fulfill the requirements of abstract services designed in the modeling phase. The context of collaborative business processes is slightly different, as it consists of a choreography of different processes achieved on the need, discontinued over time, and not reliant on an orchestrator. For instance, a patient can perform the investigation operation on day d , and two days after he completes the PCR laboratory test on $d+2$. In the service composition context, the process is completely executed on the fly to fulfill all the desired requirements at once. In this contribution, we combine both concepts, allowing an initial modeling of the desired collaborative system using the BPMN choreography diagram. Subsequently, we design each process in a fine-grained manner using FSA modeling. Fig. 1 illustrates the proposed modeling approach with a graphical description of each phase.

The first phase, surrounded by orange color, illustrates a BPMN choreography diagram, which defines the interactions between different organizations' processes to fulfill inter-organizational collaboration. As the choreography diagram lacks modeling expressiveness adapted to a more fine-grained level to address internal interactions, we complement it using the automata-based modeling. The automata formalization used was introduced in our previous contributions [7, 46]. It allows a service-oriented modeling of all business processes, and then offers the ability to integrate behavioral NFRs, when needed, to concisely express complex requirements. We used this modeling formalization to design (1) the desired FRs and (2) both measurable and behavioral NFRs. We take advantage of the automata expressiveness to implement complex security NFRs into the desired service composition. An example of implementing complex security attributes into automata modeling is the integration of the Shamir multi-cloud sharing

algorithm in a previous study [47]. In our current approach, we adopt this modeling method to address the challenges of designing blockchain interactions. Thus, a blockchain interaction is integrated as a service injected into an automatically orchestrated service composition. The Abstract Functional Automaton (AFA) is introduced to design the desired FRs of a specific business process, in order to allow its translation to a ready-to-use service composition that fulfills these requirements. These FRs are designed in the form of abstract services that constitute automata states, while transitions define the desired interactions between these abstract services. The formalization is inherited from the inputs/outputs/pre-conditions/results scheme (IOPR), allowing fine-grained modeling capabilities to describe complex case studies. From our perspective, the NFRs integrated into the service composition can either be measurable, such as quality-of-service attributes, or behavioral (e.g., security attributes). In this study, we aim to integrate blockchain transactions as behavioral NFRs. This integration can be introduced into AFAs to grant decentralization and immutability properties to any business process. The formalization used provides the ability to use behavioral scopes to graphically design the desired blockchain interactions. Behavioral scopes are graphical automata scopes that specify the subset of states associated with the integrated behavioral NFR. These are illustrated using dashed lines surrounding the target abstract service states. The arrow direction in the scope defines whether the concrete services fulfilling the behavioral NFRs should be implemented before or after scoped abstract services. In the illustrated example, we use a post-execution scope associated with the abstract service (AS_2), aiming to integrate a blockchain transaction as a service. The concrete representation of this integration is illustrated in the third phase, which defines the interactions between the concrete services and the integrated blockchain service.

B. The Proposed Blockchain based Service-Oriented Architecture

In our contribution, we aim to implement a service-oriented architecture that allows fluid interaction with the blockchain to ensure process collaboration using smart contracts. For the implementation, we opted for a permissioned blockchain because of the specificities of the introduced case study. It allows us to restrict interactions to only authorized participants due to the criticality of exchanged information in the context of healthcare business process collaboration. In this perception, despite the heterogeneity of the internal information systems used by collaborating organizations, we are able to provide on-the-fly invocation of blockchain transactions or smart contract execution to meet the requirements described in the initial modeling.

On a large scale, the diversity of providers' roles and the variety of functionally similar services published make business process collaboration a challenging task. The integration of API gateways allows the unification of service calls by using common URI suffixes for functionally similar services. This facilitates service lookup and consumption, especially when the number of organizations is important.

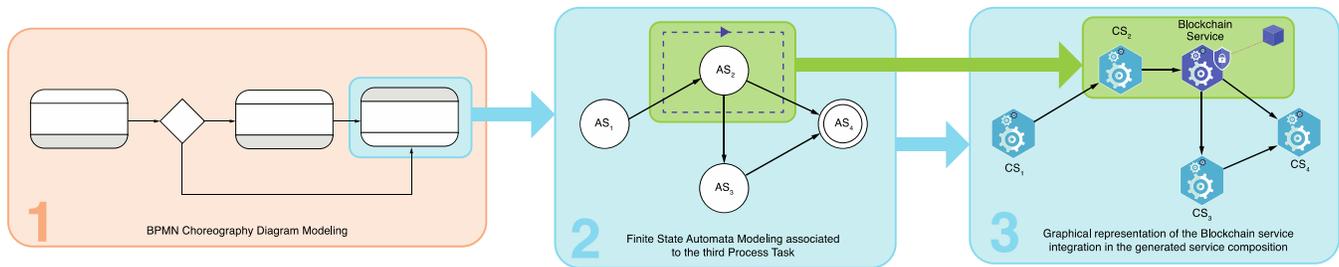


Fig. 1. Graphical Illustration of the Two Phases Modeling Process based on Choreography Diagram and Finite State Automata.

In our opinion, in addition to fulfillment of the primary requirements for a reliable business process collaboration, the privacy, identity management, and consensus mechanisms are crucial factors to consider when choosing a blockchain platform. In the proposed contribution, and to align with the context of the introduced case study, we integrate the Hyperledger blockchain as part of the proposed SOA, aspiring to improve collaboration between healthcare organizations, especially during the pandemic and post-pandemic periods. Hyperledger Fabric is a pillar platform for distributed ledger solutions that aims for democratization and standardization of blockchain for the business landscape, by allowing organizations to build custom blockchain apps to address their business requirements in a secure decentralized way [48]. It ensures a high level of confidentiality, resiliency, flexibility, and scalability. Hyperledger Fabric gained popularity through its plug-and-play approach, using the container technology to run its components to meet the desired business requirements and level of complexity. One of the essential components is the chaincode, which constitutes Hyperledger's native smart contract, defining the rules and operations for specific business processes. Additionally, Hyperledger comes with the ability to choose a consensus or even use a custom consensus implementation to define relations between members of the chain. Another significant improvement provided by Hyperledger Fabric is the support for channel creation [49]. This feature constitutes an additional security layer that allows a group of participants to create separate ledgers of transactions, especially for networks where some participants might be competitors and not want every transaction they make to be known to every participant. These previously cited features constitute the basis of our choice, which makes Hyperledger Fabric the most appropriate for the proposed blockchain proof-of-concept, especially when applied to a healthcare project.

The main components defining business logic in a blockchain solution are assets and transactions. The assets provide the ability to model business network logic in an object-oriented similar way. This feature allows designers to define business logic exhaustively using abstraction, inheritance, dependencies, and relationships. On the other hand, transactions constitute a structured and behavioral business-related description that allows the manipulation of assets through functions. These two components are explored to describe the business logic behind the implementation of collaborative processes.

The illustrated example presented in Fig. 2 describes a workflow showing the benefits of combining SOAs with

blockchain technology. The blockchain is integrated as a service to constitute a complementary ledger while maintaining the organization's internal information system functioning and updated continuously. The API gateway intercepts the user requests and executes the appropriate services. The first service endpoint called allows updating the internal database to keep the local information system up-to-date. The second endpoint executes the appropriate blockchain transaction. The ledger is then updated, and events are triggered when applicable. The end-user receives a response describing whether all the steps were successfully performed. On the other hand, Fig. 3 illustrates the interactions of the integrated blockchain service with the modeled service composition. We graphically describe a simplified workflow that represents the interactions with the adopted Hyperledger platform. It shows the usage of the provided software development kit (SDK) to automatically launch the endorsement process, and then to validate the transactions by the orderer once the consensus is reached. Hyperledger Fabric achieves consensus through its ordering service. This service establishes the total order of transactions submitted to the network. In previous versions, Hyperledger Fabric used Kafka to achieve consensus while guaranteeing crash fault tolerance (CFT). As Kafka is commonly used to manage messaging queues, it is known to not provide a Byzantine fault-tolerant (BFT) consensus. Thus, the system cannot be prevented from reaching an agreement in the case of malicious or faulty nodes. In the latest versions, Hyperledger Fabric relies on RAFT [50], which is a distributed consensus algorithm for managing replicated logs, following a "leader and follower" model, where a leader node is elected for each channel. Leaders' decisions are then replicated by followers. This allows different organizations to provide nodes that contribute to the formation of a distributed ordering service.

In the proposed approach, we provide a model based on a choreography diagram with FSA to design the desired business processes, including the ability to perform blockchain transactions. The automata-based modeling of each process is translated into a composite service handled by the orchestrator to fulfill the business process requirements. A blockchain transaction is wrapped into an atomic service that is injected into the generated composition according to the NFR scope defined in the abstract automaton. As presented in our previous study [7], the orchestrator performs a lookup into the NFR registry to select the matching service fulfilling the behavioral NFR defined in the scope. In the current contribution, the behavioral NFR consists of a smart contract execution, which constitutes a blockchain transaction in the Hyperledger context.

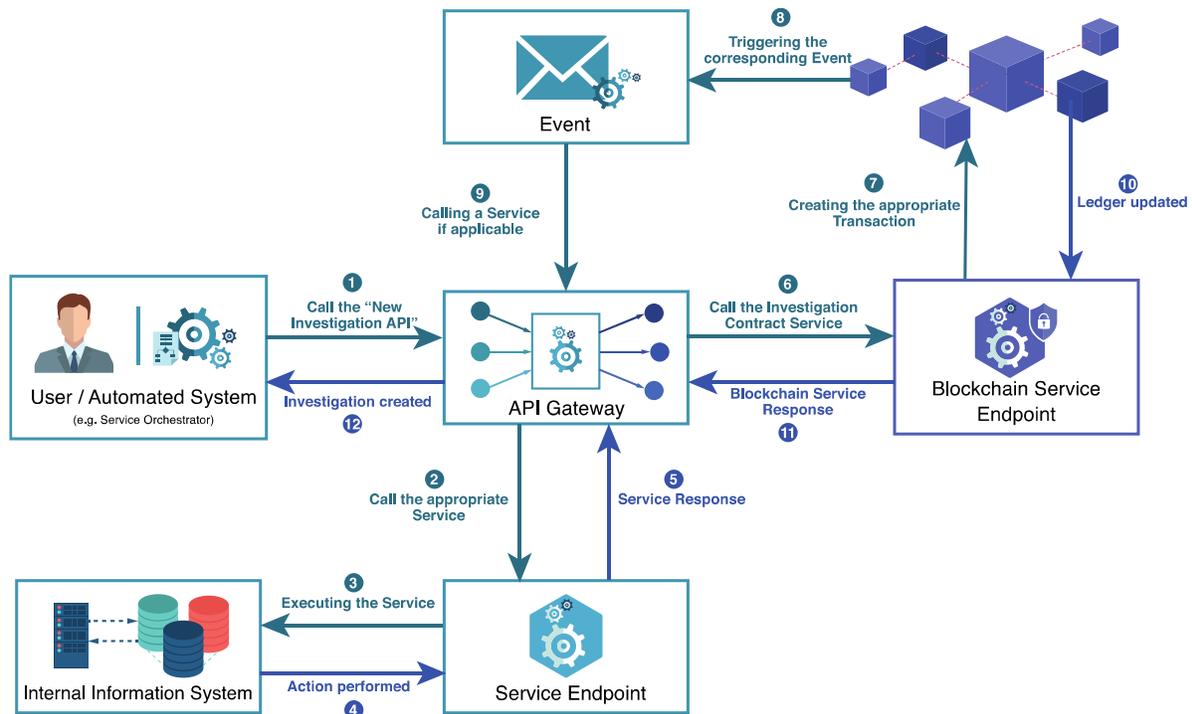


Fig. 2. The Workflow of a Blockchain as a Service Integration into the Service Composition.

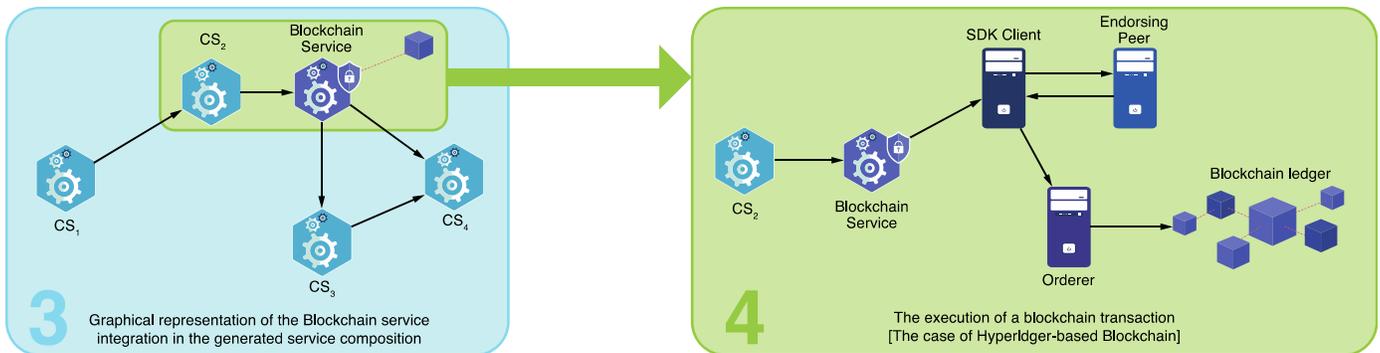


Fig. 3. The Blockchain Service Interactions with the Service Composition and the Ledger according on the Integrated Behavioral NFR Scope.

Finally, the advantage of using a service-oriented architecture is that it provides interoperability with the used blockchain platforms. The provided blockchain-as-a-service concept is not limited to Hyperledger technology, which is used in our proof-of-concept to meet the requirements of the considered case study. Subsequently, the approach is viable as long as the implemented blockchain platforms allow the automation of executed transactions using services, such as the Solidity-based smart contracts implemented by Ethereum blockchain.

V. THE PROPOSED IMPLEMENTATION

In this contribution, we aim to introduce a prototype to overcome the collaboration issues between healthcare entities during the hospitalization process for affected patients, with the ability to extend the current model to cover vaccination processes in an upcoming phase of the project. Our proposed proof-of-concept is based on previously highlighted technologies by combining the composition of services with the

blockchain. To fulfill this implementation, we opt for Hyperledger Composer, which is an open-source development toolset based on Hyperledger Fabric. Hyperledger Composer provides advanced abilities for modeling business networks and ensures the easy integration of blockchain applications with existing systems [51]. The main components of blockchain-related business network applications are the chaincode, assets, transactions, participants, events, and access-control rules. Chaincode is a program installed and instantiated onto a blockchain network to enable interactions with the ledger. It runs in a separate process from peers and is responsible for the initialization and management of the ledger state through transactions submitted by applications [52]. It provides also the ability of creating real-world decentralized applications, modeling business processes using assets and relationships, updating and exchanging information in the ledger through transactions, managing participants and their associated roles, and triggering specific business functions or services through events. The next subsections briefly define each of these

components with real-world examples inspired by the COVID-19 case study.

A. Case Study Overview

The world has experienced a remarkable health crisis, which has damaged the economy and negatively impacted healthcare systems. Most healthcare organizations are not logistically and structurally ready to deal with unpredictable large-scale healthcare-related challenges, which require urgent mobilization of resources to prevent additional losses. In the context of the Moroccan government, the healthcare ministry was very vigilant by publishing a new web-based platform allowing epidemiological centers to follow-up the cases of suspected and affected COVID-19 patients. In addition, they deployed a new mobile application allowing to notify users of any suspected physical interaction with affected patients using a geolocation tracking history. For both cited platforms, we noticed a lack of collaboration between healthcare providers, constituting the first and main destination for suspected and affected patients during the pandemic. These providers are required to maintain a continuous coordination with their respective regional epidemiological centers in order to process and investigate daily collected data, and then forward them to the national epidemiological organization within the healthcare ministry.

The collaboration between healthcare organizations in the cited context implies a good understanding of their interactions and their common processes. Briefly, the central epidemiological organization within the healthcare ministry collaborates closely with the regional epidemiological centers in each of the 12 regions of the kingdom. Similarly, each of these regional centers collaborates with all healthcare providers in their respective regions. Thus, in order to support the heterogeneity of methods and optimize interactions guaranteeing an exhaustive exchange of information, we introduce a complete process that allows the modeling and integration of collaborative business processes, with the extended ability to integrate the blockchain on demand. To present a realistic prototype, we aim to cover the overall healthcare processes and dependencies related to COVID-19, such as investigation, laboratory tests, and hospitalization, etc. Additionally, to allow an exhaustive collaboration and enhance inter-organizational interactions, we involve specialized healthcare entities, such as medical imagery centers and medical analysis laboratories. Each entity might contribute to one or multiple phases of the overall COVID-19 process, depending on the policies defined through our modeling process.

Among many interesting contributions integrating blockchain solutions to tackle collaborative challenges in the healthcare domain, especially during the COVID-19 pandemic, J. Xu et al. proposed a privacy-preserving scheme for fine-grained access control adapted to large-scale health data based on blockchain [53]. This platform, named Healthchain, is based on two blockchains to allow the users to upload IoT data and read doctors' diagnoses; on the other hand, it allows doctors to read users' uploaded data and upload appropriate diagnoses. The proposed blockchain-based platform prevents data from being tampered or denied; however, at any time, users can revoke doctors to preserve data privacy when needed. In contrast to the cited solution, and in the first phase of the

project, our contribution focuses on business-oriented inter-organizational collaboration, and thus does not integrate patient interactions with the ledger. We are currently exploring the combination of heterogeneous blockchain architectures in order to connect a permissionless ledger for patients' interactions with the permissioned architecture presented in our current approach. Another solution was proposed by Alsamhi et al., aiming to integrate blockchain and multi-robot collaboration to combat the COVID-19 pandemic [54]. The proposed conceptual framework can increase the intelligence, decentralization, and autonomous operations of connected multi-robot collaboration in a blockchain network. The proposed architecture integrates Ethereum as a permissionless blockchain platform. The proposed solution presents the scope of use of each consensus algorithm in addition to shard techniques to maintain connectivity between collaborating robots, avoid collisions, and thus improve real-time. In the same direction, the authors in [55] proposed a blockchain-based system using Ethereum smart contracts to generate statistical information based on reported data related to the number of new cases, deaths, and recovered cases obtained from trusted sources. They presented a comparative analysis of the security and cost incurred by the stakeholders to prove the feasibility of integrating the proposed solution to ensure data integrity, security, transparency, and data traceability among stakeholders. Finally, the authors in [56] conducted a scoping review to identify relevant studies by searching 11 bibliographic databases. They conducted backward and forward reference list checking of the included studies and relevant reviews. According to their study based on 10 use cases of blockchain to tackle COVID-19 challenges, the most prominent use cases were contact tracing and immunity passports. Public blockchain technology was the most commonly used type in the included studies. Out of these 10 studies that identified the platform used, nine studies used Ethereum to run the blockchain, and Solidity was the most prominent programming language used to develop smart contracts. Although Ethereum continues to be the most popular and used blockchain platform in various industries and research fields, it is still susceptible to privacy leakage as public keys are made transparent to members of the network. This constitutes a key factor impacting the choice of blockchain solution and migration to permissioned blockchains, especially for addressing business process collaboration.

In the proposed implementation, we aim to tackle the business process collaboration challenge by implementing comprehensive well-designed composite services, fulfilling all business process requirements, and providing the ability to perform transactions over the blockchain. As illustrated in Fig. 4, we allow healthcare organizations to collaborate through well-defined processes, described using exhaustive modeling. The validated automata models are translated into composite services, integrating blockchain interactions and further behavioral NFRs when necessary. The concrete business services in blue color describe the services executed to perform a specific business need, whereas the concrete blockchain services in purple specify the services executed to perform an action on the blockchain, such as obtaining data from the ledger, or executing transactions to update the ledger's world state.

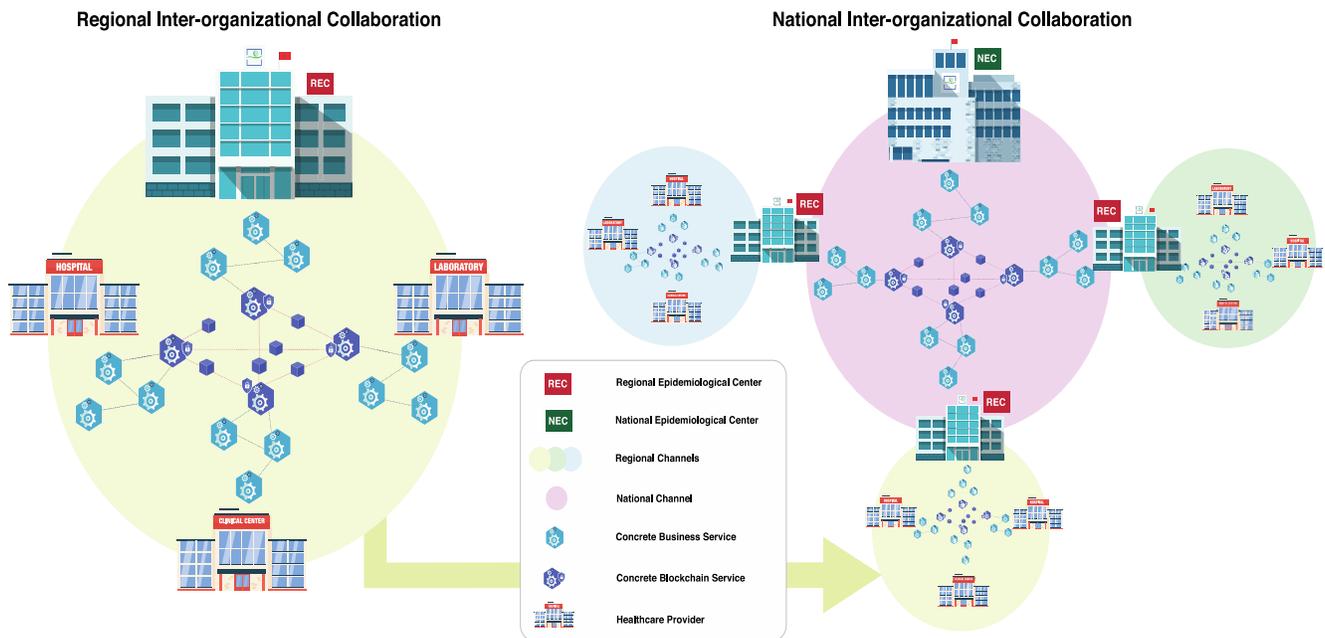


Fig. 4. The Proposed Implementation of Regional and National Inter-Organizational Collaborations based on the Blockchain.

To align the proposed architecture with the policies and hierarchical specificities of the healthcare case study, we integrate the channel feature, allowing the creation of sub-networks to ensure data privacy in accordance with each regional perimeter. This segmentation allows each regional epidemiological center (REC) to play a key role in federating and managing all organizations and participants falling under its regional administrative perimeter. Thus, coordinators of RECs will be in charge of managing the roles and permissions, and supervising the continuous conformance to healthcare ministry policies. Consequently, each REC coordinator interacts with two different ledgers: (1) the main ledger through the national channel, and (2) the regional ledger associated with the regional channel he manages. Examples of regional channels are depicted using blue, yellow, and green circles, whereas the pink circle defines the national channel.

B. COVID-19 Workflow Modeling

The initial phase in the proposed modeling process consists of defining the collaboration workflow through a choreography diagram. In Fig. 5, we present an illustrative model covering the main processes related to the COVID-19 workflow. The participant roles in this example continuously collaborate with RECs. These centers constitute, for each region in the kingdom, the main entities responsible for processing and analyzing collected data, before forwarding the daily COVID-19 key performance indicators (KPIs) to the national epidemiological center (NEC). Additionally, these RECs constitute the decentralized representatives of the healthcare ministry in the

region, forwarding all epidemiological-related decisions to healthcare providers, and supervising their execution. The illustrated modeling depicts all phases of a COVID-19 patient, taking into account all cases, depending on the PCR results for both diagnosis and control laboratory tests. The main phases in the choreographic scheme can be shared by different actors; for example, a hospitalization can either be processed by a healthcare public provider or private provider. Similarly, laboratory tests can be performed by either a healthcare provider or a private laboratory center. These processes are examples reproducing real scenarios, summarized and abstracted into one choreography diagram. This diagram is shared by the 12 regions, all replicating the same unified process with the same common roles. The organization's roles participating in the process example are briefly described below, and defined more in deep in the following subsections.

- The regional epidemiological center, which is the focal organizational authority in the region, is responsible for collecting, processing and analyzing epidemiological data from all healthcare providers within its region.
- The healthcare provider, which can be any healthcare entity among of the previously cited organizations. Its main role is to provide healthcare services to patients.
- The medical analysis laboratories, which are the medical laboratories authorized by the RECs to issue PCR tests to patients.

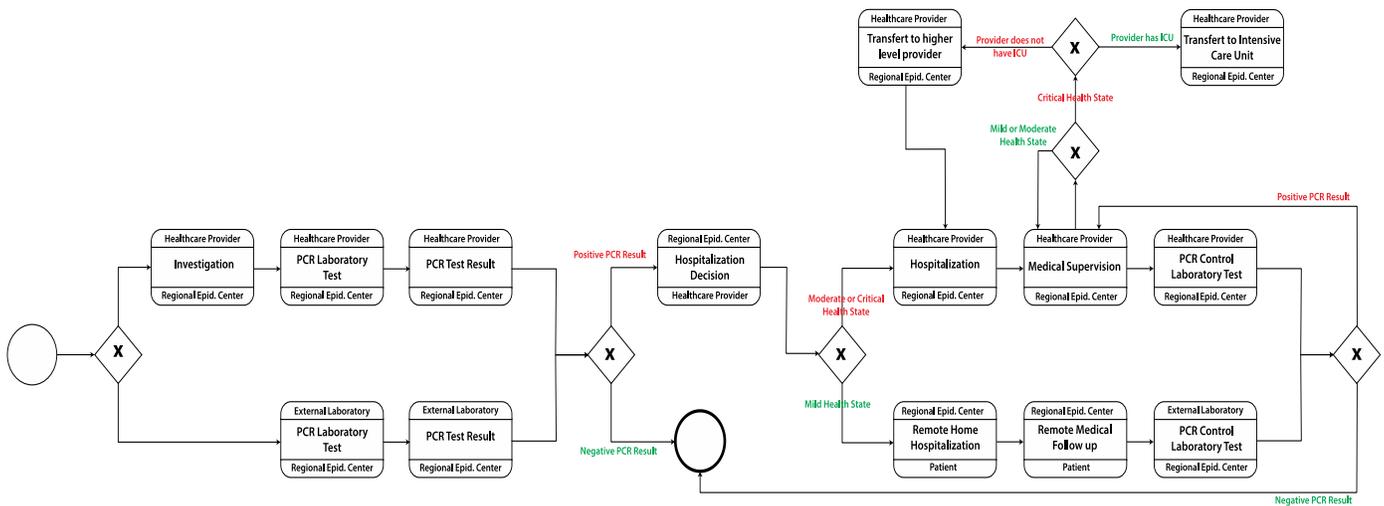


Fig. 5. A Proposed Choreography Diagram for the Adopted COVID-19 Business Processes.

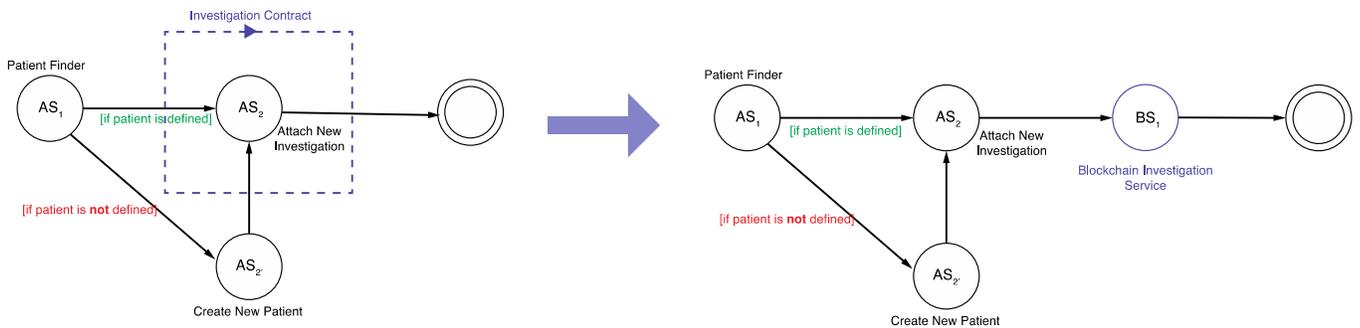


Fig. 6. A Proposed Modeling for the Investigation Process using the AFA.

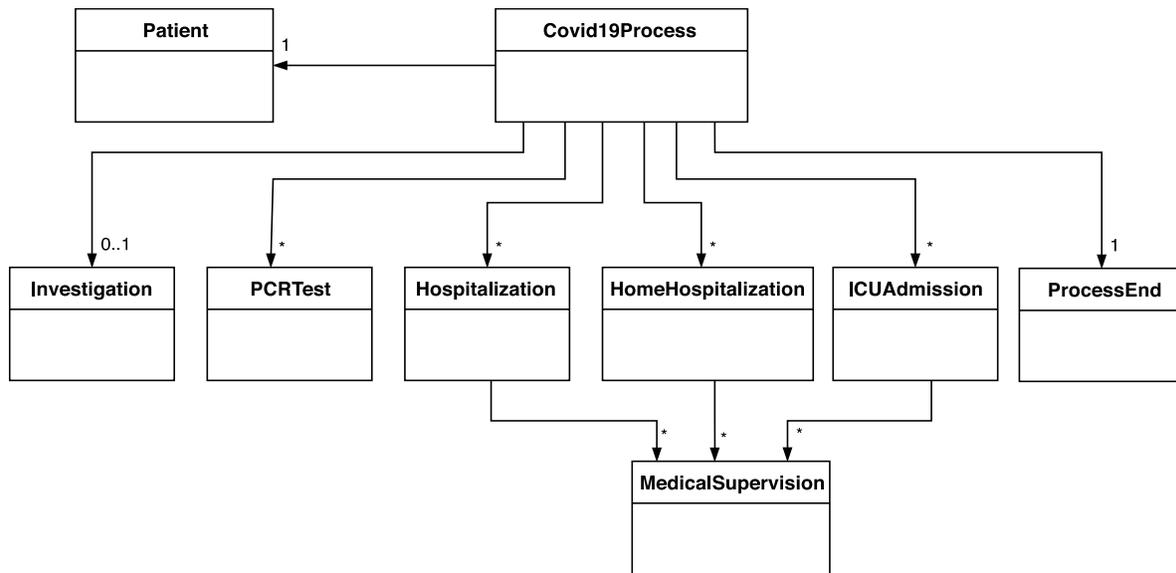
In the same modeling perspective, we illustrate in Fig. 6 a simple modeling example of interacting services that describes the investigation process. The service composition described in the automaton allows the creation of a new investigation for a patient. In this AFA modeling, we design the desired FRs using state interactions, in addition to the new scope defining the blockchain-oriented behavioral NFR. Thus, the concrete service composition integrates a service invocation that sends a transaction to the blockchain. Consequently, in addition to the usual persistence of the investigation data in the provider's internal information system, a new transaction will be executed in the blockchain, and propagated to local copies of all contributing participants.

The provided automata formalization uses descriptive names or regular expressions to index behavioral NFR [7]. In the context of blockchain, we integrate the possibility of performing the NFR-lookup operation taking as a parameter the label used to index the transaction or the smart contract to execute. In the provided example, a forward-direction arrow is used in the *InvestigationContract*, which means that the transaction service fulfilling the behavioral NFR should be executed after the scoped service. This integration implies a lookup operation performed in the behavioral registry, returning the concrete service metadata that matches the specified query.

C. The Asset Modeling

Recall that assets constitute the main resources in Hyperledger technology; they can range from tangible to intangible entities. Hyperledger Fabric provides the ability to modify assets using chaincode transactions. They are represented in Hyperledger Fabric as a collection of key-value pairs and can be represented in binary and/or JSON forms. In Composer, assets are defined in .cto files using a structure similar to the class structuration in object-oriented programming. In our case study, we used assets to model all phases of the COVID-19 healthcare processes.

Our case study modeling based on the presented choreography diagram can be defined in a fine-grained manner using appropriate assets. This allows us to design the desired information system exhaustively, and consequently, improve inter-organizational collaboration. Fig. 7 presents some asset definitions extracted from the modeling file *ma.gov.ehealthcare.assets.cto*. In the same figure we also provide the class diagram illustrating the relationships between the presented assets. In the current phase of implementation, the patient is integrated as an asset, which constitutes a subject and not an interacting participant.



```

namespace ma.gov.ehealthcare.assets
import ma.gov.ehealthcare.concepts.*
import ma.gov.ehealthcare.organizations.*
import ma.gov.ehealthcare.participants.*

asset Patient identified by govId {
  o String govId
  o String EHRId
  o PersonalInformation personalInformation
}

asset Investigation identified by investigationId {
  o String investigationId
  o DateTime investigationDate
  --> HealthcareOrganization healthcareOrganization
  --> Doctor investigator
}

asset PCRTest identified by PCRIId {
  o String PCRIId
  o DateTime sampleDate
  o DateTime resultDate optional
  o PCRResult PCRResult optional
  o PCRPhase PCRPhase
  --> HealthcareOrganization healthcareOrganization
}

asset Hospitalization identified by hospitalizationId {
  o String hospitalizationId
  o DateTime hospitalizationStartDate
  o DateTime hospitalizationEndDate
  o HealthState entryHealthState
  --> HealthcareProvider healthcareProvider
  --> Doctor authorizer
}

asset ICUAdmission identified by ICUAdmissionId {
  o String ICUAdmissionId
  o DateTime ICUAdmissionStartDate
  o DateTime ICUAdmissionEndDate
  o HealthState entryHealthState
  --> HealthcareProvider healthcareProvider
  --> Doctor authorizer
}

```

```

asset Covid19Process identified by covid19ProcessId {
  o String covid19ProcessId
  --> Patient patient
  --> Investigation investigation
  --> PCRTest[] laboratoryTests optional
  --> Hospitalization[] hospitalizations optional
  --> ICUAdmission[] ICUAdmissions optional
  --> HomeHospitalization[] homeHospitalizations optional
  --> HealthcareProviderTransfert[] healthcareProviderTransferts optional
  --> ProcessEnd processEnd
}

asset HealthcareProviderTransfert identified by healthcareProviderTransfertId {
  o String healthcareProviderTransfertId
  o Boolean acknowledgement
  o DateTime transfertDate
  o HealthState currentHealthState
  --> HealthcareOrganization healthcareRequester
  --> HealthcareOrganization healthcareReceiver
  --> Doctor transfertRequester
}

asset DailyPCRTestStatistics identified by dailyPCRTestStatisticsId {
  o String dailyPCRTestStatisticsId
  o DateTime endDate
  o Integer dailyPCRTestNumber
  --> Region region
}

asset HomeHospitalizationidentified ...
asset ImagingExam ...
asset MedicalSupervision ...
asset ProcessEnd ...
asset DailyPositiveCaseStatistics ...
asset DailyNegativeCaseStatistics ...
asset DailyRecoveryCaseStatistics ...
asset DailyDeathCaseStatistics ...
asset DailyICUAdmissionsStatistics ...

```

Fig. 7. A Proposed Example of Asset Definitions of COVID-19 Processes and their Appropriate Class Diagram Modeling.

The main assets constituting the COVID-19 healthcare processes are the following:

1) The investigation asset usually constitutes the first point of contact between the patient and the healthcare provider, allowing a doctor to investigate the patient's health state, symptoms and medical history for the recent period before his/her admission by the healthcare organization.

2) The PCR test asset gathers all PCR-related information and constitutes the key element in defining the consequent phases according to the test results. In other words, in addition to other factors, a positive result may imply a patient hospitalization, whereas a negative result implies a non-affection for diagnosis tests or a patient recovery for control tests.

3) The hospitalization asset describes the process of taking care of a patient in a healthcare organization during a period of time according to medical instructions related to his/her health state.

4) The home hospitalization asset defines the newly introduced process of taking care of a patient remotely in his/her home through continuous supervision based on telecommunication and regularly filled reports.

5) The ICU admission asset defines the process of transferring a patient internally to the intensive care unit (ICU). This transfer depends on the health status of the patient and the availability of the ICU room.

6) The medical supervision asset describes the operation of performing regular medical checks in order to follow up and update the patient's health state.

On the other hand, in order to keep track of all participating organizations in each phase of the workflow, we adopt a model that considers collaborating organizations as assets. This allows us to identify the healthcare providers responsible for any performed action, thereby improving supervision and auditability. In each region, these healthcare organizations communicate with the REC, which constitutes their higher focal authority. In their turn, RECs are continuously reporting to the NEC. The collaborating organizations defined using assets in the studied scenario are the following:

1) The healthcare organization asset is the parent of the overall organizational assets, containing common attributes such as the identifier, name, address, and region.

2) The healthcare provider asset inherits from the healthcare organization asset and gathers all key information regarding the provider's capacities in terms of hospitalization, laboratory tests, and imaging examinations.

3) The public healthcare center asset inherits from the healthcare provider and unites both university healthcare centers and regional healthcare centers.

4) The private clinical center asset inherits from the healthcare provider asset, and defines the external private healthcare entities collaborating with the public healthcare organizations during the pandemic.

5) The private imaging center asset inherits from the healthcare organization and defines the key capacity

information for the main imaging examinations related to COVID19 affections.

6) The private laboratory asset inherits from the healthcare organization and defines the laboratory daily PCR exam capacity supported during the pandemic.

7) The regional epidemiological center asset inherits from the healthcare organization and defines the administrative entity responsible for collecting, processing, analyzing and reporting all epidemiological-related information through the built collaborative system.

8) The national epidemiological center asset inherits from the healthcare organization and defines the head entity responsible for collecting, processing, analyzing and reporting all epidemiological-related information gathered from RECs through the built collaborative system.

D. The Participants

Participants are the members of a business network. They constitute the main actors of the blockchain, managing assets and submitting transactions. For each participant, one or more identities can be assigned, allowing him/her, depending on the attributed rules, to perform actions and interact with the ledger. In the first phase of this project, we omit adding the patient as a participant, as in this initial phase of the project, we consider only inter-organizational collaborative processes, which excludes patient interactions. In other words, all patient data are updated by coordinators of healthcare organizations that they interact with, such as doctors, healthcare provider coordinators, etc.

We introduce different role definitions, which all inherit from the healthcare participant role. They are listed below:

1) The healthcare provider coordinator role defines the main actor representing his/her *healthcare organization*. It allows to manage internal data and also to update the ledger. We link this role to *the healthcare provider's asset* to easily identify the organization of each coordinator.

2) The regional epidemiological center coordinator plays a key role in our case study. On the one hand, this participant is responsible for managing and supervising all blockchain information related to his/her attributed region. On the other hand, it allows interaction with the NEC through the main national channel.

3) The national epidemiological center coordinator role allows users of the healthcare ministry to investigate and follow the daily summaries received through the national channel. It has higher administrative privileges allowing to manage participants, organizations, and entities on a national scale.

4) The laboratory coordinator role allows interaction with the ledger as a coordinator of a private laboratory, in order to update the patients' COVID-19 processes by integrating information related to PCR tests and their respective results.

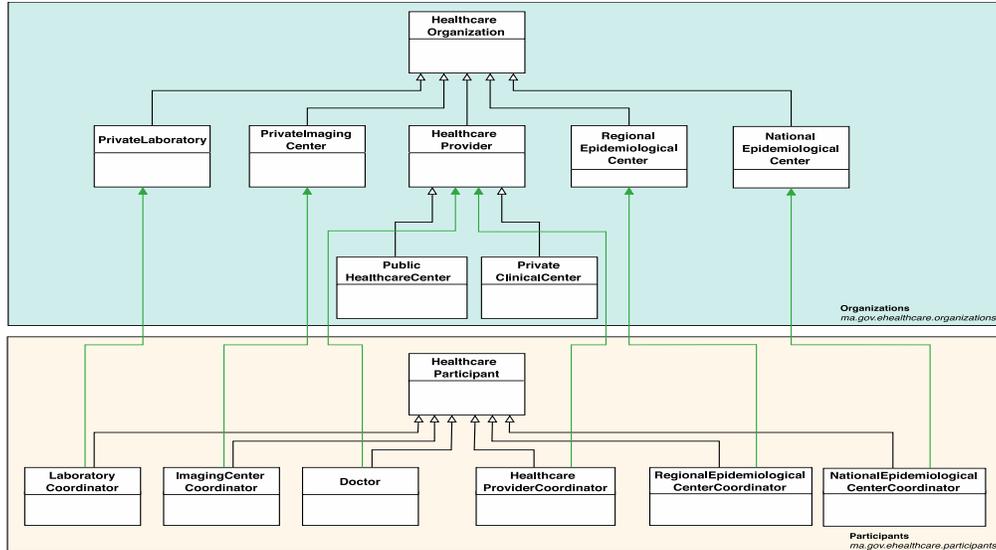
5) The imaging center coordinator defines the role assigned to the participants responsible for coordinating the private imaging centers. Their main contribution consists of

integrating all affected patients' imaging data in the blockchain, such as chest X-ray, CT scan, or MRI results.

In Fig. 8, we present some examples of collaborating organizations and their associated participants supported by the appropriate class diagram.

E. The Transactions

Transactions are the mechanisms by which participants interact with assets. They define the core business logic of the process. Once executed, the state of the associated assets changes. Consequently, the world state can be recreated by replaying all transactions with respect to their accurate order, which constitutes one of the strengths of blockchain technology.



```
namespace ma.gov.ehealthcare.organizations
import ma.gov.ehealthcare.concepts.*
import ma.gov.ehealthcare.assets.*
```

```
asset HealthcareOrganization identified by
healthcareOrganizationId{
  o String healthcareOrganizationId
  o String healthcareOrganizationName
  o Address healthcareOrganizationAddress
  --> Region region optional
}

asset HealthcareProvider extends HealthcareOrganization{
  o Integer capacity
  o Integer dailyPCRSampleCapacity optional
  o Integer dailyChestXRayCapacity optional
  o Integer dailyCTScanCapacity optional
  o Integer dailyMRICapacity optional
}

asset PublicHealthcareCenter extends HealthcareProvider{
  o HealthcareCenterType type
}

asset PrivateImagingCenter extends HealthCareOrganization{
  o Integer dailyChestXRayCapacity
  o Integer dailyCTScanCapacity
  o Integer dailyMRICapacity
}

asset PrivateLaboratory extends HealthCareOrganization{
  o Integer dailyPCRSampleCapacity
}

asset PrivateClinicalCenter extends HealthcareProvider ...
asset PrivateLaboratory extends HealthcareOrganization ...
asset RegionalEpidemiologicalCenter extends
HealthcareOrganization ...
asset NationalEpidemiologicalCenter extends
HealthcareOrganization ...
```

```
namespace ma.gov.ehealthcare.participants
import ma.gov.ehealthcare.concepts.*
import ma.gov.ehealthcare.organizations.*
```

```
abstract participant HealthcareParticipant identified by
participantId {
  o String participantId
  o PersonalInformation personalInformation
}

participant HealthcareProviderCoordinator extends
HealthcareParticipant {
  --> HealthcareProvider healthcareProvider
}

participant RegionalEpidemiologicalCenterCoordinator extends
HealthcareParticipant {
  --> RegionalEpidemiologicalCenter
regionalEpidemiologicalCenter
}

participant NationalEpidemiologicalCenterCoordinator extends
HealthcareParticipant {
  --> NationalEpidemiologicalCenter
nationalEpidemiologicalCenter
}

participant LaboratoryCoordinator extends
HealthcareParticipant {
  --> PrivateLaboratory laboratory
}

participant ImagingCenterCoordinator extends
HealthcareParticipant {
  --> PrivateLaboratory laboratory
}

participant Doctor extends HealthcareParticipant {
  o Boolean isSpecialist
  o Speciality speciality optional
  --> HealthcareProvider healthcareProvider
}
```

Fig. 8. A Proposed Example of Organizations and Participants Assets and their Appropriate Class Diagram Modeling.

We use transactions to populate blockchain assets using organizational and healthcare data. First, we use our proposed automata-based approach to illustrate the modeling of each business process to produce a valid abstract service composition. Then, we integrate the appropriate blockchain transactions using behavioral scopes associated to the abstract services. In Fig. 9, we present a model of the patient investigation phase, which consists of collecting patient data, such as health state, symptoms, and comorbidities, etc. According to patient collected data, the investigating doctor estimates if there is a risk that patient might be affected, and consequently redirects him to the appropriate service for a PCR test or for an imaging examination. On the other hand, the *healthcare provider coordinator* performs a lookup to verify whether the patient already has an EHR, and creates a new patient profile otherwise. The second step consists of creating a specific *COVID-19 process* blockchain asset to assign to the patient if he/she interacts for the first time with the *healthcare organization*. All these scenarios are modeled as part of the investigation process, where states identify the abstract services described as follows: The states outlined with a black color define the abstract services denoted AS. Each of these abstract services will be associated with a pool of concrete services fulfilling the desired FRs defined in the AFA. The states outlined in blue are abstract services fulfilling behavioral blockchain NFRs of requesting data from the ledger on a read-only basis. Thus, their associated concrete services will interact with the blockchain, but will not impact the world state of the ledger. Finally, the states outlined in green constitute the abstract services defining blockchain transactions, and whose concrete services have the ability to update the world state of the ledger.

We provide below a brief description of the provided Automata modeling example:

- The concrete service CS1 associated with the first abstract service AS1 (Patient Finder) allows to return, from the internal information system, the unique electronic health record (EHR) for a given patient, using a set of personal information as input parameters.
- The concrete services CS2/CS2' associated with the abstract services AS2 and AS2' respectively, are conditioned by the result of execution of the first concrete service CS1 during the runtime. Thus, if CS1 returns a valid patient, the concrete service CS2 will be executed to fulfill the query "Covid Process Finder". In this case, CS2 will only interact with the blockchain on a read-only basis to obtain the requested data. However, in the second case, the concrete service CS2' will be executed to fulfill the query "New Covid Process and Patient Assignment". In this case, CS2' will call a blockchain transaction to create a Covid Process asset with a nested patient asset filled with the patient information.
- The concrete service CS3 fulfills the requirements of the AS3 associated with the query "Investigation Assignment". This service will perform a blockchain transaction to create a new investigation asset and assign it to the previously returned Covid Process asset.

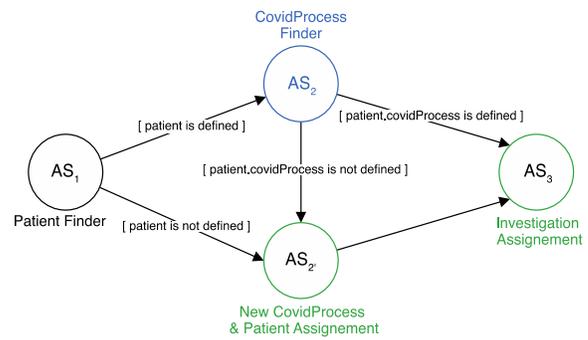


Fig. 9. The Proposed Automata Modeling of the Investigation Process Including Blockchain-Oriented Abstract Services.

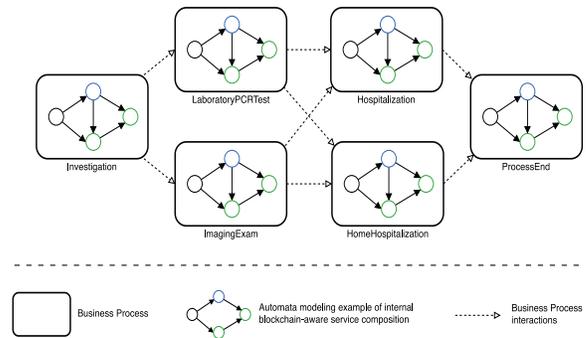


Fig. 10. Simplified Representation of the Process Choreography and Associated Internal Service-Oriented Automata.

Similarly, we extend the same modeling to design the remaining processes. In Fig. 10, we describe a simplified example representing how collaborating business processes can be seen from a wider and closer perspective, combining both FSA modeling and choreography diagrams.

Thus, in practice once a business process is required, the system executes the appropriate service composition, which includes regular business-oriented services and blockchain-oriented services injected as behavioral NFRs in the designed AFA.

Finally, in the context of our proposed case study, we present in Table II a comparative table describing the implementation and collaboration aspects of some of the main COVID-19 operations. It specifies how these operations are implemented, and whether (1) they allow an automatic collaboration with external organizations or (2) are limited to the internal information systems and are shared manually with external organizations through messaging or data transfers. Most public healthcare providers are using local web applications, handling common operations regarding hospitalizations, medical consultations and exams. This application is exhaustive for handling medical data, however, does not cover the specificities of some important COVID-19 operations, such as investigation which comes with a long questionnaire storing meticulous information about patient's health state, laboratory and imaging exams data which are handled using complementary applications connected to automates, etc. Accordingly, the healthcare ministry deployed urgently a new COVID19-oriented web application to gather investigation information and daily statistics. These applications

are not connected to each other, and do not provide web services or APIs to allow their interoperability. Thus, an automatic collaboration is only available using the web application that can be accessed by multiple participants and roles to handle investigation, patient health state, recovery/death and statistic data. For other daily operations not covered by any of implemented applications, manual registration is used then to ensure follow-ups and data transfers to higher organizations. The proposed contribution based on service-oriented collaboration and using decentralized ledgers to parallelly store and update information overcomes all these collaboration limitations, and provides accessible operations through blockchain transactions, with respect to the implemented access-control policies and regional channel configuration.

F. The Events

Events in Hyperledger Fabric, as for their common use in software engineering, provide a way to interact with external

components or systems. In Hyperledger Composer, they are defined in the same way as assets or participants, and can be emitted by transactions to indicate to external systems that something of importance has happened to the ledger. Applications can then subscribe to events through the composer-client API [57]. In our context, events are mainly associated with the blockchain services updating the ledger, outlined in green in the described automata model. In the present case study, events allow the notification of the higher authority of updates regarding the overall healthcare status related to COVID-19. For instance, transactions launched by healthcare organizations trigger events to update statistics and notify the REC. Similarly, at a higher level, transactions launched by the RECs trigger events notifying the NEC of updates regarding the epidemiological status in their respective regions.

TABLE II. A RECAPITULATIVE TABLE CLASSIFYING COLLABORATION LIMITATIONS FOR SOME OF COVID-19 PROCESS ACTIONS

Action	Organization	Initial Collaborative Environment					Current Implementation	
		Local Application	Complementary Documents/Apps	COVID-19 Web Application	On demand Collaboration (Messaging, Data Transfers)	Automatic Collaboration	Decentralized Service-oriented Solution	Automatic Collaboration
COVID19 Investigation	Healthcare Provider	X	✓	✓	X	✓	✓	✓
Laboratory PCR Data	Healthcare Provider	Partial	✓	X	✓	X	✓	✓
	Private Laboratory	✓	X	X	✓	X	✓	✓
Laboratory PCR Results	Healthcare Provider	X	✓	✓	X	✓	✓	✓
	Private Laboratory	✓	X	On approval	X	On approval	✓	✓
Hospitalization	Healthcare Provider	✓	X	X	✓	X	✓	✓
Medical Supervision History	Healthcare Provider	X	✓	X	✓	X	✓	✓
Imaging Exam	Private Imaging Center	✓	X	✓	X	✓	✓	✓
Health Status History	Healthcare Provider	X	✓	X	✓	X	✓	✓
Home Hospitalization	REC	X	✓	X	✓	X	✓	✓
ICU Admission	Healthcare Provider	✓	X	X	✓	X	✓	✓
External Transfer	Healthcare Provider	Partial	✓	X	✓	X	✓	✓
Process End (Recovery)	Healthcare Provider	✓	X	✓	X	✓	✓	✓
Process End (Death)	Healthcare Provider	Partial	✓	✓	X	✓	✓	✓
Data Anonymization Process		X		X			✓	
Access Control – Application Level		✓		✓			✓	
Access Control – Module Level		✓		✓			✓	
Access Control – Action Level		Partial		-			✓	
Access Control – Resource/Data Level		X		-			✓	
Channel Restrictions		X		-			✓	

To make our proposed implementation more exhaustive, we implement some statistic-oriented assets allowing the COVID-19 statistics to be updated daily for each region. These assets are automatically updated when appropriate events are triggered. In our case study, we use events to notify REC coordinators about updates regarding COVID-19 daily status in their respective regions. Another concrete example consists of using events to notify a target healthcare provider about a transfer request for a patient in a critical health state, where the requesting participant receives an acknowledgement as soon as the target provider validates the transfer.

Finally, in a service-oriented approach ensuring a choreography between different organizations, events constitute an essential pillar allowing an efficient collaboration, and facilitating the implementation of real-time notification systems to increase reactivity towards critical scenarios. In Fig. 11, we provide a definition of the investigation transaction associated with an event emission to subscribed components.

G. Access Control and Security Implementation

Hyperledger proposes a high-end permissioned system that allows to manage and control participants' permissions for each action. This constitutes the backbone and main purpose behind using an identity-based permissioned blockchain. Hyperledger provides an access control language based on declarative permissions over each element of the domain model. Access control rules are defined in specific files and are described using two different methods. The first definition type, which is the simplest, allows to control the access to a namespace, an asset, or an asset's property for a participant type or participant instance. The second type is a more advanced conditional rules system, based on JavaScript Boolean expressions. These expressions are evaluated in runtime to allow or deny access to a resource for a participant. Participants can have their access to transactions restricted based on their role's permissions defined in the access control files. In our perception, collaborating organizations and their attributed participants are dynamically managed. Identities and business network cards can be managed under administrative privileges through the Hyperledger API. The roles of coordinators for both regional and national epidemiological centers are given the ability to manage participants, identities and network cards, for their regional and national organizations, respectively.

In the present case study, we notice that none of the essential assets constituting the main COVID-19 process contain or point to the patient's assets. The aim behind omitting this relationship is to enhance confidentiality and ensure data anonymization, which constitutes an essential feature implemented in our contribution, as highlighted in Table II. Accordingly, participants with restricted permissions can only access and handle the data of assets associated to their scope. For example, they can handle COVID-19 hospitalization information, PCR results information or statistics information without accessing the corresponding patients' data (e.g., personal information). This modeling-based separation of assets, combined with the fine-grained layers of access-control management provided by the Hyperledger Fabric blockchain, constitute the backbone of security attributes implemented in this contribution, in order to provide a meticulous management of roles and permissions for

overall participants, according to their administrative attributions.

On the other hand, and in order to meet the ambitions and efforts of the Moroccan government to deploy the advanced regional decentralization, we aimed in the proposed case study to use an exclusive channel for each individual region to promote data privacy. From our perspective, these channels allow to concentrate the efforts of regional epidemiological centers coordinators to securely manage the permissions of collaborating organizations falling under their administrative perimeters. Similarly, a national channel is implemented, allowing communication between all RECs with the ministry entities (e.g., National Epidemiological Center, Ministry Secretary). The aim of this channel is to ensure continuous data forwarding and reporting based on the daily statistics generated for each region.

```
/**
 * Create Investigation Transaction
 * @param
 * {ma.gov.ehealthcare.assets.CreateInvestigation}
 investigationData
 * @transaction
 */

function createInvestigation(investigationData) {

    // 1. Get the asset registry
    return
    getAssetRegistry('ma.gov.ehealthcare.assets.Investigation')
        .then(function(investigationRegistry) {

            // 2. Get resource factory
            var factory = getFactory();
            var NS = 'ma.gov.ehealthcare.assets';

            // 3. Create the Resource instance
            var investigationId = generateTxId();

            var investigation = factory.newResource(
                NS, 'Investigation', investigationId);

            // 4. Set the data and relationships
            investigation.investigationDate = new Date(
                investigationData.investigationDate);
            var doctorRelationship = factory.newRelationship(
                'ma.gov.ehealthcare.participants', 'Doctor',
                investigationData.investigator.participantId);

            investigation.investigator = doctorRelationship;

            var healthcareOrganizationRelationship =
                factory.newRelationship(
                    'ma.gov.ehealthcare.organizations',
                    'HealthcareOrganization',
                    investigationData.healthcareOrganization.healthcar
                    eOrganizationId);

            // 5. Emit the event InvestigationCreated
            var event = factory.newEvent(NS,
                'InvestigationCreated');
            event.investigationId = investigationId;
            emit(event);

            return
            investigationRegistry.addAll([investigation]);
        });
}
```

Fig. 11. A Definition Example of the Investigation Transaction and the Associated Emitted Event.

VI. CONCLUSION AND FUTURE WORK

In this study, we aimed to provide a complete solution for the design and implementation of inter-organizational collaborative processes based on a two-phase modeling using BPMN's choreography diagram and FSA. We provided the ability to integrate blockchain transactions as services to promote decentralization, immutability, integrity, and trustless interactions between collaborating organizations. To illustrate our approach, we presented a proof-of-concept implementation adapted to the healthcare domain, and handling COVID19-related collaborative processes. Although we opted for Hyperledger Fabric as a permissioned blockchain platform to meet the collaboration requirements of the studied scenario, the introduced blockchain as a service concept can be adapted to any blockchain platform as long as it supports the automation of transaction calls. Our proposed proof-of-concept modeling is based on real business processes aiming to promote collaboration between healthcare organizations in Morocco.

Using an architecture based on a specific blockchain topology will always rely on its limitations in term of performance, scalability, security, and other key aspects characterizing the chosen blockchain platforms. On the other hand, the misuse of permissioned blockchains can cause colossal damage, especially in the context of inter-organizational collaboration, as the reliability of the ledger relies on the integrity of its members. Thus, identity and role management are key features implemented to ensure healthy collaborations. In this contribution, the proposed service-oriented approach does not rely on any specific blockchain topology and can integrate both permissioned and permissionless blockchain platforms, since the execution of a transaction or a smart contract is wrapped into a concrete service.

In future work, we will focus on two principal aspects: (1) Covering intra-organizational collaboration by encouraging progressive migration of internal information systems toward the blockchain, requiring the deployment of new blockchain transactions as concrete services and an exhaustive definition of interacting assets using the proposed modeling process. (2) Integrating end-users (e.g., patients) as participants in the collaborative ledger by experimenting the integration of a hybrid topology, pairing permissionless and permissioned blockchains. To achieve this objective, numerous challenges arise and are associated, for example, to the correlation of performance with the scalability, shared identity management, and the implementation of unified security requirements while merging the heterogeneous platforms.

REFERENCES

- [1] L. Chung and J. C. S. do Prado Leite, "On Non-Functional Requirements in Software Engineering," in *Conceptual Modeling: Foundations and Applications*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 363–379.
- [2] W. Cai, Z. Wang, J. B. Ernst, Z. Hong, C. Feng, and V. C. M. Leung, "Decentralized Applications: The Blockchain-Empowered Software System," in *IEEE Access*, vol. 6, pp. 53019–53033, 2018, DOI: 10.1109/ACCESS.2018.2870644.
- [3] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology: Beyond bitcoin", *Applied Innovation*, vol. 2, pp. 6–10, 2016.
- [4] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *2016 IEEE Symposium on Security and Privacy (SP)*, 2016.
- [5] B. Niehaves and R. Plattfaut, "Collaborative business process management: status quo and quo vadis," in *Business Process Management Journal*, vol. 17, pp. 384–402, 2011, DOI: <https://doi.org/10.1108/14637151111136342>.
- [6] Q. Chen and M. Hsu, "Inter-enterprise collaborative business process management," in *Proceedings of 17th International Conference on Data Engineering*, 2002.
- [7] I. E. Kassmi, R. Belkeziz, and Z. Jarir, "Deep Attention on Measurable and Behavioral-driven Complete Service Composition Design Process," in *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 3, 2021, DOI: 10.14569/IJACSA.2021.0120377.
- [8] C. Cachin, "Architecture of the Hyperledger blockchain fabric," in *Workshop on Distributed Cryptocurrencies and Consensus Ledgers (DCCL 2016)*, Chicago, Jul. 2016.
- [9] M. Swan, "Blockchain: Blueprint for a New Economy," Newton, MA, USA: O'Reilly Media, 2015.
- [10] M. Pilkington, "Blockchain Technology: Principles and Applications", in *Research Handbook on Digital Transformations*, 2016.
- [11] A. Muddasar and B. Sikha, "Introduction to NFTs: The Future of Digital Collectibles" *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 10, 2021. DOI: <http://dx.doi.org/10.14569/IJACSA.2021.0121007>.
- [12] A. Reyna, C. Martín, J. Chen, E. Soler, M. Díaz, "On blockchain and its integration with IoT. Challenges and opportunities," *Future Generation Computer Systems*, Vol. 88, pp. 173–190, 2018, DOI: <https://doi.org/10.1016/j.future.2018.05.046>.
- [13] A. Jawad, A. Toqeer, M. Shahrulniza and A. Zahrani, "Towards Secure IoT Communication with Smart Contracts in a Blockchain Infrastructure" *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 9, no. 10, 2018. DOI: <http://dx.doi.org/10.14569/IJACSA.2018.091070>.
- [14] S. Al-Rakhami M. and Al-Mashari M., "Blockchain and Internet of Things for Business Process Management: Theory, Challenges, and Key Success Factors" *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11, no. 10, 2020. DOI: <http://dx.doi.org/10.14569/IJACSA.2020.0111069>.
- [15] Y. Chen and C. Bellavitis, "Blockchain disruption and decentralized finance: The rise of decentralized business models," *Journal of Business Venturing Insights*, vol. 13, 2020, DOI: <https://doi.org/10.1016/j.jbvi.2019.e00151>.
- [16] S. Solat, P. Calvez, and F. Nait-Abdesselam, "Permissioned vs. Permissionless blockchain: How and why there is only one right choice", in *Journal of Software*, pp. 95–106, 2021.
- [17] S. Pahlajani, A. Kshirsagar, and V. Pachghare, "Survey on private blockchain consensus algorithms", in *2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT)*, 2019.
- [18] E. Androulaki et al., "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proceedings of the Thirteenth EuroSys Conference*, 2018.
- [19] K. Olson, M. Bowman, J. Mitchell, S. Amundson, D. Middleton, and C. Montgomery, "Sawtooth: An introduction", in *Hyperledger: Blockchain Technologies for Businesses*, 2018.
- [20] M. Valenta, P. Sandner, "Comparison of Ethereum Hyperledger Fabric and Corda," in *FSBC Working Paper June 2017*, Frankfurt School Blockchain Center, 2017.
- [21] S. Bano et al., "SoK: Consensus in the age of blockchains," in *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, 2019.
- [22] W. Wang et al., "A survey on consensus mechanisms and mining strategy management in blockchain networks," *IEEE Access*, vol. 7, pp. 22328–22370, 2019, doi: 10.1109/ACCESS.2019.2896108.

- [23] J. A. Garay, A. Kiayias, and G. Panagiotakos, "Proofs of Work for Blockchain Protocols," IACR Cryptology ePrint Archive, Report 2017/775, Aug. 2017.
- [24] F. Saleh, "Blockchain without Waste: Proof-of-Stake," *The Review of Financial Studies*, vol. 34, pp. 1156–1190, 2021, DOI: <https://doi.org/10.1093/rfs/hhaa075>.
- [25] S. De Angelis, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri and V. Sassone, "PBFT vs proof-of-authority: applying the CAP theorem to permissioned blockchain," *Italian Conference on Cyber Security*, 2018.
- [26] M. Castro and B. Liskov, "Practical byzantine fault tolerance and proactive recovery," *ACM Transactions on Computer Systems*, vol. 20, no. 4, pp. 398–461, 2002.
- [27] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system", Self-published Paper, May 2008. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>.
- [28] V. Buterin, "Ethereum: A next-generation smart contract and decentralized application platform", Ethereum Foundation, Technical Report, 2014. [Online]. Available: <https://github.com/ethereum/wiki/wiki/White-Paper>.
- [29] J. Mendling et al., "Blockchains for business process management - challenges and opportunities," *ACM Transactions on Management Information Systems*, vol. 9, no. 1, pp. 1–16, 2018.
- [30] J. A. García-García, N. Sánchez-Gómez, D. Lizcano, M. J. Escalona and T. Wojdyński, "Using Blockchain to Improve Collaborative Business Process Management: Systematic Literature Review," in *IEEE Access*, vol. 8, pp. 142312–142336, 2020, doi: 10.1109/ACCESS.2020.3013911.
- [31] L. D. Xu and W. Viriyasitavat, "Application of Blockchain in Collaborative Internet-of-Things Services," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1295–1305, 2019.
- [32] I. Weber, X. Xu, R. Riveret, G. Governatori, A. Ponomarev, J. Mendling, "Untrusted business process monitoring and execution using blockchain". In *International Conference on Business Process Management*, pp. 329–347, 2016, Springer.
- [33] L. García-Bañuelos, A. Ponomarev, M. Dumas, I. Weber, "Optimized execution of business processes on blockchain". In *International conference on business process management*, pp. 130–146, 2017, Springer.
- [34] F. Corradini, A. Marcelletti, A. Morichetta, A. Polini, B. Re, and F. Tiezzi, "Engineering trustable choreography-based systems using blockchain," in *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, 2020.
- [35] C. Dannen, "Solidity Programming," in *Introducing Ethereum and Solidity*, Berkeley, CA: Apress, 2017, pp. 69–88.
- [36] J. Ladleif, M. Weske, and I. Weber, "Modeling and enforcing blockchain-based choreographies," in *Lecture Notes in Computer Science*, Cham: Springer International Publishing, 2019, pp. 69–85.
- [37] C. Di Ciccio et al., "Blockchain support for collaborative business processes," *Informatik Spektrum*, vol. 42, no. 3, pp. 182–190, 2019.
- [38] A. B. Tran, Q. Lu and I. Weber, "Lorikeet: A model-driven engineering tool for blockchain-based business process execution and asset management", *CEUR Workshop Proceedings*, 2018.
- [39] O. López-Pintado, L. García-Bañuelos, M. Dumas, I. Weber, and A. Ponomarev, "Caterpillar: A business process execution engine on the Ethereum blockchain," *Software: Practice and Experience*, vol. 49, no. 7, 2019.
- [40] V. Pourheidari, S. Rouhani, and R. Deters, "A Case Study of Execution of Untrusted Business Process on Permissioned Blockchain," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018.
- [41] M. Autili, F. Gallo, P. Inverardi, C. Pompilio, and M. Tivoli, "Introducing Trust in Service-oriented Distributed Systems through Blockchain," in *2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, 2019.
- [42] A. Kernahan, U. Bernskov and R. Beck (2021). "Blockchain out of the Box – Where is the Blockchain in Blockchain-as-a-Service?", *54th Hawaii International Conference on System Sciences*, 2021, DOI: 10.24251/HICSS.2021.520.
- [43] L. Daming, D. Lianbing, C. Zhiming and S. Alireza, "Blockchain as a service models in the Internet of Things management: Systematic review," *Transactions on Emerging Telecommunications Technologies*, 2020, DOI: <https://doi.org/10.1002/ett.4139>.
- [44] W. Zheng, Z. Zheng, X. Chen, K. Dai, P. Li and R. Chen, "NutBaaS: A Blockchain-as-a-Service Platform," in *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2941905.
- [45] Q. Lu, X. Xu, Y. Liu, I. Weber, L. Zhu and W. Zhang, "uBaaS: A unified blockchain as a service platform," *Future Generation Computer Systems*, vol. 101, pp. 564–575, 2019, DOI: <https://doi.org/10.1016/j.future.2019.05.051>.
- [46] I. El Kassmi and Z. Jarir, "Measurable and Behavioral Non-Functional Requirements in Web Service Composition," in *Handbook of Research on Contemporary Perspectives on Web-Based Systems*, pp. 340–361, 2018, IGI Global, DOI: <http://doi:10.4018/978-1-5225-5384-7.ch015>.
- [47] I. El Kassmi and Z. Jarir, "Towards security and privacy in dynamic web service composition," *2015 Third World Conference on Complex Systems*, 2015, doi: 10.1109/ICoCS.2015.7483260.
- [48] Hyperledger Fabric: Main Documentation [Online]. Available: <https://hyperledger-fabric.readthedocs.io/en/latest/whatis.html> (Visited on 02/02/2021).
- [49] Hyperledger Fabric: Channel Capabilities [Online]. Available: https://github.com/hyperledger/fabric/blob/release-2.2/docs/source/capabilities_concept.md (Visited on 02/21/2021).
- [50] W. Fu, X. Wei and S. Tong, "An Improved Blockchain Consensus Algorithm Based on Raft," *Arabic Journal for Science and Engineering*, 2021, DOI: <https://doi.org/10.1007/s13369-021-05427-8>.
- [51] E. Kinory, S. S. Smith, and K. S. Church, "Exploring the playground: Blockchain prototype use cases with Hyperledger Composer," *Journal of Emerging Technologies in Accounting*, vol. 17, no. 1, pp. 77–88, 2020.
- [52] Hyperledger Fabric: Smart Contracts and Chaincode [Online]. Available: <https://github.com/hyperledger/fabric/blob/release-2.2/docs/source/smartcontract/smartcontract.md> (Visited on 02/23/2021).
- [53] J. Xu et al., "Healthchain: A Blockchain-Based Privacy Preserving Scheme for Large-Scale Health Data," in *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8770–8781, Oct. 2019, DOI: 10.1109/JIOT.2019.2923525.
- [54] S. H. Alsamhi and B. Lee, "Blockchain-empowered multi-robot collaboration to fight COVID-19 and future pandemics," *IEEE Access*, vol. 9, pp. 44173–44197, 2021.
- [55] D. Marbough et al., "Blockchain for COVID-19: Review, opportunities, and a trusted tracking system," *Arabian Journal for Science and Engineering*, vol. 45, no. 12, pp. 1–17, 2020.
- [56] A. A. Abd-alrazaq et al., "Blockchain technologies to mitigate COVID-19 challenges: A scoping review," *Computer Methods and Programs in Biomedicine Update*, vol. 1, p. 100001, 2021.
- [57] Hyperledger Fabric: Peer channel-based event services [Online]. Available: https://github.com/hyperledger/fabric/blob/release-2.2/docs/source/peer_event_services.rst (Visited on 03/01/2021).