Book Chapter Proposal for Chapter 3

Blockchain Technology Use Cases in Healthcare

Peng Zhang, Douglas C. Schmidt, and Jules White
Vanderbilt University, Nashville, TN

Gunther Lenz
Varian Medical Systems, Palo Alto, CA

Abstract

Blockchain technology alleviates the reliance on a centralized authority to certify information integrity and ownership, as well as mediate transactions and exchange of digital assets, while enabling secure and pseudo-anonymous transactions along with agreements directly between interacting parties. It possesses key properties, such as immutability, decentralization, and transparency, that potentially address pressing issues in healthcare, such as incomplete records at point of care and difficult access to patients' own health information. An efficient and effective healthcare system requires interoperability, which allows software apps and technology platforms to communicate securely and seamlessly, exchange data, and use the exchanged data across health organizations and app vendors. Unfortunately, healthcare today suffers from siloed and fragmented data, delayed communications, and disparate workflow tools caused by the lack of interoperability. Blockchain offers the opportunity to enable access to longitudinal, complete, and tamper-aware medical records that are stored in fragmented systems in a secure and pseudo-anonymous fashion. This book chapter focuses on the applicability of Blockchain technology in healthcare by (1) identifying potential Blockchain use cases in healthcare, (2) providing a case study that implements Blockchain technology, and (3) evaluating design considerations when applying this technology in healthcare.

Keywords**: Blockchain **in Healthcare, Healthcare Interoperability, Smart Contracts.

3.1Introduction

Blockchain is a platform that alleviates the reliance on a single, centralized authority, yet still supports secure and "trustless" transactions directly between interacting entities [1]. It offers decentralization, immutability, and consensus via cryptography and game theory. This technology provides the foundations for a number of application domains, including cryptocurrency and *Decentralized Apps* (DApps) [2].

Smart contracts are enhancements to Blockchain technologies, as implemented in the Ethereum Blockchain [3], that provide code to directly control the exchanges or redistributions of digital assets (such as crypto-tokens or some pieces of data) between two or more parties according to certain rules or agreements previously established between involved participants. Smart contracts can store data objects and define operations on the data, enabling development of DApps to interact with Blockchains and provide seamless services to the application users. In the domain of healthcare, smart contracts can be applied to create secure and effective technical infrastructures to enhance care coordination and quality and thus improve the wellbeing of individuals and communities [4]. Ideally, software apps and technology platforms in an interoperable healthcare environment should be able to communicate securely, exchange data, and use the exchanged data across health organizations and app vendors [5]. These health systems should also ensure effective care delivery for individuals and communities by allowing care providers to collaborate within and beyond organizational boundaries, *e.g.*, by mediating secure access to electronic health records (EHRs).

Healthcare researchers and practitioners today, however, struggle with fragmented and siloed data, delayed communications, and disparate workflow tools. On the one hand, providers feel reluctant to exchange data due to (1) perceptions that patient health and identification information safe-keeping regulations prevent such sharing (even anonymized) and (2) potential liability and financial consequences associated with data sharing [6, 7]. On the other hand, vendor-specific and incompatible health systems create gaps in healthcare communications, making it hard to coordinate and provide patient-centric care [8].

A key problem in production healthcare systems today is the lack of secure links that can connect all independent health systems together to establish an end-to-end reachable network [9] while protecting healthcare professionals with some level of anonymity (privacy). Although data standards like HL7 [10] and FHIR [11] provide basic interoperability for data exchange between trusted systems, this level of interoperability is limited to the implemented standards and requires data mapping between systems in most if not all cases. Maintainability of these systems is also hard to achieve since an interface change on one system requires other parties in the trusted network to adapt the change as well.

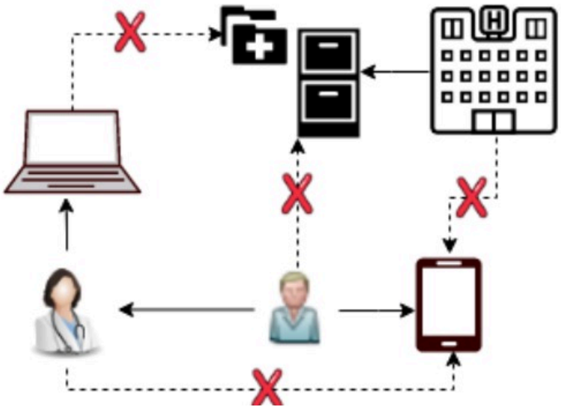


Figure 1 Production Healthcare Systems are Often Unable to Interoperate, and Access to Data Records is Disabled by Disparate and Siloed Systems[1]

For example, Figure 1depicts barriers to achieving healthcare system interoperability, including incompatible software (such as vendor-locked EHR systems) and the lack of access to data outside a healthcare environment (such as a firewall protected clinic database or a patient-collected mobile health data). A promising solution to these problems involves the application of Blockchain technology, which provides "trustless" transactions via decentralization with pseudo-anonymity.Complexities within the healthcare industry, however, yield additional challenges to employing Blockchain technology. This book chapter explores the fundamental properties of Blockchain technology that can assist in establishing these trusted links and other capabilities in several potential use cases and analyzes the key design considerations for creating production DApps in the healthcare domain.

The remainder of this book chapter is organized as follows: Section 3.2 identifies pressing issues in healthcare, focusing on interoperability and patient-centered care; Section

[1] © [2017] IEEE. Reprinted, with permission, from Peng Zhang, Douglas C. Schmidt, Jules White, Gunther Lenz, and Mike Walker, Metrics for Assessing Blockchain-based Healthcare Decentralized Apps, Proceedings of the IEEE Healthcom 2017, October 12-15, 2017, Dalian, China.

3.3 then describes seven specific healthcare scenarios, or use cases, where Blockchain technology can be leveraged to alleviate some of the major challenges associated; Section 3.4 explores four healthcare-specific challenges faced by Blockchain-based systems and their design implications; Section 3.5 presents a case study prototype we developed as an example that addresses the challenges; Section 3.6 uses the case study to highlight key design considerations for developing Blockchain-based DApps for healthcare; Section 3.7 summarizes key lessons learned from our experience; and Section 3.8 provides an outlook on future research directions of Blockchain technology in the healthcare space.

3.2Pressing Issues in Healthcare

This section explores pressing issues in healthcare today, focusing on interoperability challenges that limit data sharing and impede patient-centered care fostered by healthcare interoperability that would otherwise allow patients to access and control their own health information.

3.2.1 Interoperability

Healthcare interoperability describes the ability for heterogeneous information technology systems and software applications, such as the Electronic Health Record (EHR) system, to communicate, exchange data, and use the exchanged data [12]. Allowing information systems to work together within and across organizational boundaries is paramount to improve effective care delivery for individuals and communities [13]. For example, interoperability enables providers to securely and scalably share patient medical records with one another (given patient permissions to do so), regardless of provider location and trust relationships between them.

Secure and scalable data sharing is essential to provide effective collaborative treatment and care decisions for patients. Data sharing helps improve diagnostic accuracy [14] by gathering confirmations or recommendations from a group of medical experts, as well as preventing inadequacies [15] and errors in treatment plan and medication [16, 17]. Likewise, aggregated intelligence and insights [18-20] helps clinicians understand patient needs and in turn apply more effective treatments. For example, groups of physicians with different specialties in cancer care form tumor boards that meet regularly to discuss cancer cases, share knowledge, and determine effective cancer treatment and care plans for patients [21]. As another example, if a cancer patient under treatment is admitted to the emergency room (ER) in a different hospital, then it would be critical for the ER provider to access the patient's medical records to identify potential drug interactions; while the patient's primary cancer care provider should be notified that the patient is being treated in the ER.

Despite the importance of medical data sharing, today's healthcare systems frequently require patients to obtain and share their own medical records with other providers either via physical paper copies or electronic hard disk copies. This process of obtaining and sharing medical records is ineffective for the following reasons:

- **It is slow** since copies of medical data must be prepared, delivered, and picked up by patients. The law allows providers up to 30 days to supply medical records to patients, although some providers may only take 5-10 business days to prepare non-critical health records [22].
- **It is insecure** because data copies may be lost or stolen during their physical transmission by patients from one location to another.
- **It is incomplete** since as patient health history may be fragmented because their data is stored in disparate and siloed systems. There is no single source that stores all the medical records of an individual, so patients must therefore be responsible for keeping track of

when and where they received health services in order to request copies of their medical history.

- **It lacks context** since today's healthcare systems are provider-centric instead of patient-centric, thereby preventing patients from taking control of their own health records and having knowledge of what is done to their data or who has accessed their data [23].

The ineffective data sharing process in healthcare results in part from the lack of trust between providers and the lack of interoperability between health IT systems and applications today. Healthcare interoperability comprises of three levels: foundational, structural, and semantic, ordered lowest to highest fidelity [24], as summarized in Table 1 and described below.

Table 1: Summary of the Three Levels of Interoperability

Interoperability Level	Summary
Foundational	Data exchange is enabled; does not require data interpretation
Structural	Defines formats for the exchanged data
Semantic	Requires interpretation of the exchanged data

Foundational interoperability enables data exchanges between healthcare systems. It does not require providers receiving the data to be able to interpret the data. **Structural interoperability** additionally defines formats for exchanged clinical data and ensures that received data are preserved and can be interpretable at the data field level using predefined formats. **Semantic interoperability** demands the interpretability of exchanged data by not only syntax (structure) but also semantics (meaning) of the data.

These three levels help ensure that disparate health systems and applications deliver information with requisite data quality and safety. Foundational and structural interoperability are prerequisites for semantic interoperability, which is hardest to achieve but most desired to advance quality of care. This book chapter focuses on exploring Blockchain-based use cases that address foundational and structural interoperability from the technical infrastructure's perspective. Semantic interoperability requires clinical domain knowledge and medical policies that enforce the adoption of ontologies, *i.e.*, common data standards, to interpret myriad sources of health information, which is beyond the scope of this book chapter.

3.2.2 Patient-Centered Care

The healthcare industry is shifting from volume-based care (fee-for-service), in which providers are incentivized to provide more treatments because payment is proportional to the quantity of care, to value-based care that promotes patient-centered care with higher quality (hence "value"), in which patients are informed and involved in clinical decision making [25]. In patient-centered care model, patients are capable of incorporating Patient Reported Experience Measures (PREM) and Patient Recorded Outcome Measures (PROM) [26], such as symptoms or health status, collected from their wearable and mobile devices into their medical history. Patients should also be given easy access to their medical records with a comprehensive view of their entire health history, which could potentially reduce information fragmentation and inaccuracy caused by communication delays or coordination errors and in turn improve the care continuity and quality [27].

Ideally, all health systems (regardless of the type of care settings) would provide automatic notifications for patients to access their clinical data in (near) real-time, e.g., as records are entered into the system or when lab results are available. In addition, in patient-centric care it is necessary for patients to control when and to whom their health data is shared and to choose what pieces of information they would be willing to share. Healthcare systems today, however, do not provide the means for patients to modify or revoke a provider's access to their data.

As a result, once a provider has cared for a patient or has obtained access to patient data that data is permanently in possession of the provider. When a patient visits different providers many times throughout their lifetime, their health and other sensitive personal data is available at several sites. This diffusion increases the risk of data theft because it only takes one provider lacking sufficient and up-to-date security practices to put patient information vulnerable to attack (with the assumption that no malpractice or unethical usage of patient data is involved, of course). Alternately, a patient may wish to release their medical records to a new provider, which is not easily accomplished today as discussed in Section 3.2.1above.

Interoperability is also fundamental to support a patient-centric model that improves quality of care for individual patients. In practice, barriers exist in the healthcare technical infrastructure that impedes interoperability and thus patient-centered care, including (but not limited to):

- **Information security and privacy concerns**. Despite the need for data sharing, it increases the risk of sensitive data breaches without a highly secure infrastructure in place [7]. Providers could face severe financial and legal consequences [6] when data is compromised.
- **Lack of trust between providers**. Because of security regulations, care providers must be able to identify other providers and also trust their identities before any patient health-related communication occurs [28]. Trust relationships often exist between in-network providers and/or health organizations but they are particularly difficult to establish when the data receiving care office does not use the same health system with a shared provider directory, such as in a private practice or a hospital network.
- **Scalability concerns**. Medical data may contain large volumes of data like medical images, especially in cancer patients or patients with chronic conditions. These large-scale datasets are difficult to share electronically due to limitations in bandwidth or restrictive firewall settings, such as in rural areas [29].

Below we first present six potential use cases of Blockchains in the healthcare space and then focus on a concrete case study that demonstrates what design decisions can be made to leverage Blockchain-related technologies to address these interoperability challenges.

3.3 Blockchain Use Cases in Healthcare

This section explores seven Blockchain use cases focusing on various concerns in healthcare, as summarized in Table 2.

Table 2: A Summary of Seven Healthcare Use Cases that Blockchain Technology Can Address

Section	Use Case Summary
3.3.1	Prescription Tracking to Detect Opioid Overdose and Over-Prescription
3.3.2	Data Sharing to Incorporate Telemedicine with Traditional Care
3.3.3	Sharing Cancer Data with Providers Using Patient-Authorized Access
3.3.4	Cancer Registry Sharing to Aggregate Observations in Cancer Cases
3.3.5	Patient Digital Identity Management for Better Patient Record Matching
3.3.6	Personal Health Records for Accessing and Controlling Complete Health History
3.3.7	Health Insurance Claim Adjudication Automation to Surface Error and Fraud

3.3.1 Opioid Prescription Tracking

It is widely known that there is an opioid epidemic present in the United States [30]. While many efforts are being made to address this crisis (e.g., the Drug Supply Chain Safety Act (DSCSA) [31], the President's Commission on Combating Drug Addiction and the Opioid Crisis [32], and numerous prescription awareness campaigns [33, 34]), our current prescription tracking system still lacks the technology to do so effectively. Data hoarding, doctor shopping [35], provider ignorance, vulnerable and centralized data, and over-prescription riddle the current prescription opioid marketplace. The decentralization and auditability of Blockchain technology provides a promising approach to prescription monitoring that not only makes prescriptions safer, but also provides incentives for writing fewer prescriptions.

Healthcare providers today are incentivized to prescribe opioids to patients. For example, providers incur less face-time with patients, fewer costs associated with patient care, and thus greater profits from higher returns. Likewise, pharmacies are incentivized to produce and distribute opioids since the more they sell, the higher their bottom line and the greater their return to shareholders. Moreover, patients are incentivized to consume opioids. In the treatment of pain, physical therapy or post-surgery recovery can be frustrating and riddled with disappointment [36]. Opioids provide a short-term relief, at the cost of addicting a patient. This self-fulfilling cycle can thus benefit from a technological solution that realigns these incentives.

To offset these incentives contributing to the rise of the opioid epidemic, a Blockchain-based system can establish a trusted network of hospitals and pharmacies to store opioid-associated transactions (including prescriptions, fulfillment, etc.) in a secure and accountable manner. Such a distributed and shared permissioned Blockchain platform allows for loosely-coupled providers to access other data silos without explicit trust relationships between each other. Stakeholders within the system (hospitals, pharmacies, etc.) are likewise incentivized to onboard new members to the consortium because with each additional member, they can form a more complete dataset. Rules can be mutually predefined so the consortium can securely onboard new provider members to the system.

By distributing knowledge that an opioid transaction has occurred, rather than the entirety of the specific content of that transaction, this type of ecosystem can remedy a number of the problems in the current opioid system. More complete opioid prescription history can be available to detect overly prescribed opioid by providers and also patterns of doctor shopping in patients. Consequently, providers will be incentivized to meet the requirements to join the consortium of providers through potential access to data that will increase the quality of their care. Most importantly, by tracking the history of opioid prescriptions, patients will receive care more appropriate to their condition and thus be steered away from the dangers of opioids towards less addictive and thus longer lasting treatment actions.

3.3.2 Data Sharing between Telemedicine and Traditional Care

Traditionally, telemedicine offers widely accessible care to patients who are located in remote areas far away from local health facilities or in areas of with shortages of medical staff. Today, it has becoming increasingly popular among patients who wish to receive convenient medical care [37]. Connected patients can avoid wasting time waiting at a doctor's office and get immediate treatment for minor but urgent conditions on demand [38]. Due to the growing accessibility to smart mobile and telemedicine devices, many companies offer 24/7 continuous access to care, and many user-friendly apps have been created for patients to monitor, manage, and report their health using technology [39]. For example, Apple Health

[40] app allows patients to connect to equipment for measuring vitals and store these data on their iPhones . These records can then be reported to the provider as needed.

Telemedicine services are usually equipped with more advanced technologies and are much more far-reaching compared to traditional physical health services. Due to these on-demand services, it is common for providers from different regions or networks to treat patients, resulting in reduced care continuity. Health data collected during telemedicine care episodes may be inaccessible by patients' primary care providers, which creates an incomplete medical history and in turn risks the overall quality of care [41].

By removing the need for a third-party authority and empowering direct interactions between involved participants, Blockchain technology can potentially bridge the communication barrier between these providers. Blockchain technology alone, however, cannot address the complex data sharing challenge–it must be incorporated into existing disparate health systems and clinical data standards. Figure 2 shows a high-level conceptual infrastructure where a Blockchain (represented as the dashed ellipse) is connected to disparate health database systems (represented as cylinder database objects).

Each database system shown in Figure 2 opens up a new secure data channel (as represented by small circles on the ellipse border), similar to what is used to share data with other akin systems. A smart contract (the keyed file symbol) is then used to govern the data transactions between these systems based on mutual agreements and also create an immutable history of all the transaction records. In practice, a robust architecture will include many more design components than what is shown in Figure 2.

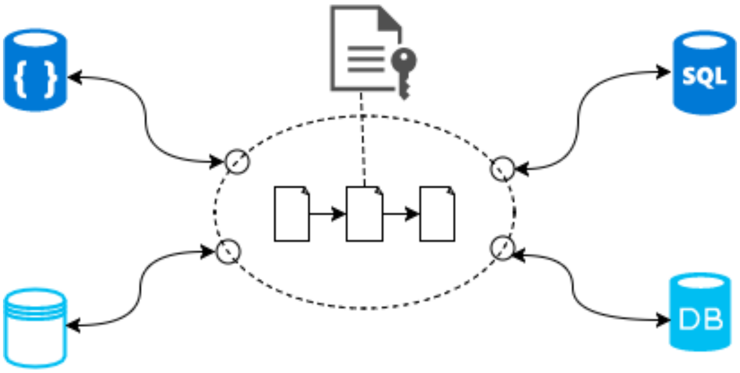


Figure 2 A High-Level Blockchain-Based Conceptual Infrastructure toConnect Disparate Health Database Systems and Record Data Exchange History

3.3.3 Patient-Controlled Cancer Data Sharing

Cancer diagnoses and treatment plans are rarely black-and-white, i.e., they involve many considerations due to the complexity of tumor cases and the number of available treatment options [42]. Getting fresh perspectives from different specialists can help narrow down the options and may shorten the time from (suspected) diagnosis to treatment for a cancer patient. In the U.S. today, most hospitals have included at least one tumor board, which is a multidisciplinary team of medical, surgical, radiation oncologists, and other specialists and care providers, to review and discuss individual cancer patient's condition and treatment options in depth [2]. Despite the increasing effort to encourage cancer care collaboration among oncologists, patients and families remain passive in the decision-making process. A large-scale enterprise hospital may involve specialists from a wider range of disciplines, whereas a smaller-scale care center may have limited resources to expand their tumor boards. The quality of life that is important to cancer patients may be neglected due to patient disengagement.

In reality, patients may wish to reach out to a new provider for a second opinion on their medical conditions and/or treatment plan. To share critical data today, patients have to obtain copies of their medical reports from their current provider, which may include their family history, visit history, prescriptions, current diagnoses and treatment options, and so on. All reports will then be delivered physically to the new provider. In this technologically advanced society, patients with critical conditions should be least involved in the manual data sharing process and have the critical data shared in a timely manner to prevent delays in treatment. A patient-controlled data sharing feature is missing from the existing health systems for cancer patients to promptly request a second opinion and also selectively share information.

Instead of creating a new trusted "middle man" that mediates the establishment of trust relationships between providers/hospitals, Blockchain technology offers the opportunity for trustless exchange and disintermediation that allow existing trust relationships to be aggregated and propagated across various organizations and providers. This approach is similar to the patient referral process, except the referrals are not limited to a single provider's network. Instead, they could be expanded across different regions, states, and even countries. A Blockchain-based system can also capture existing trust relationships between patients and providers, allowing patients to decide which provider(s) can share their data.

3.3.4 Cancer Registry Sharing

Data sharing is especially critical in cancer care where cases are usually complex and cures are rarely one-size-fits-all [43]. Being able to share cancer data helps ensure the integrity of results obtained from clinical trials by enabling individual confirmation and validation, but it can also agglomerate intelligence gathered to reduce unwarranted repetition in clinical trials [18-20]. It allows distributed clinical trials to achieve a significant cohort size and thus speeds up the discovery of more effective cancer treatments. In the U.S., only about 3% [44] of cancer patients are undergoing clinical trials today. As a result, most cancer patients receive treatments based on observations drawn from this small population of highly selective patients with different demographics, family medical history, secondary diagnoses, etc.

Population-based cancer registries (PBCR) are attempts to capture very rudimentary data from cancer incidences across geographic areas and for planning population-wide cancer control [45]. As with EHRs, cancer registries are often siloed and fragmented, which can similarly leverage Blockchain technology for expedited information exchange. In addition, with increased availability of richer data collected from many patients, artificial intelligence can be used to construct prognostic and predictive models for assisting care providers with decision support. A learning ecosystem can be designed using Blockchain technologies to also share predictive models and collaboratively improve accuracies of learned medical insights.

3.3.5 Patient Digital Identity

A fundamental component in health information exchange is patient identification [46] matching, which finds a patient in a healthcare database using a unique set of data. Systems like the Master Patient Index (MPI) and Enterprise Patient Master Index (EPMI) [47] have been created to manage patient identities within a healthcare organization or within a trusted network. Despite the increased development effort, accurately and consistently matching patient data remains hard. Patient identity mismatching has contributed to duplicated patient records and incomplete or incorrect medical data [48].

One study [49] estimated that 195,000 deaths occur each year due to medical errors, with 10 of 17 errors being identity or "wrong patient errors." There are also significant costs to healthcare organizations who maintain these duplicate records and correct mistakenly merged errors [50] and also patients who experience repeated tests or treatment delays. In addition, these errors also impact reimbursement as claims may be denied due to “out of date or incorrect information” [51], not to mention the security risks involved when patients disclose their personal information.

Without common standards for collecting patient identifying information, even the same patient’s identity can vary from one care facility to another. For instance, demographics data, such as name, date of birth, address, and Social Security Number (SSN) are often used to register a patient [46]. However, names may be stored in various formats, such as legal first and last name, nickname and last name, with or without middle initial, and patients may share identical or similar names; similarly, date of birth can be entered into the system in multiple ways; address can change as patients move to a new location; and patients may refuse to provide their SSN or do not have an SSN. Furthermore, patient information manually entered into the system may contain typos or errors, and the more data collected, the more opportunities for mistakes [51]. Although within each organization patient demographics data may be collapsed into a single unique ID, the ID generally does not translate across organizations.

Without a functional, unified identity management system, patient identification schemes employed at various care sites may continuously experience incompatibility and patient matching problems, unless a patient exclusively receives care within one organization. In fact, the very nature of Blockchain incorporates such a decentralized, unified identity system. Many existing Blockchains use cryptographically secured addresses to represent identities. Each address is mathematically linked with a unique key that is used to easily verify the ownership of an address or an identity yet does not reveal any personal information relating to the individual. The decentralized and auditable characteristics of Blockchain can help enforce standardized verifiable identities for patients via a universal patient index registry sharable across all healthcare facilitates within a nation and beyond [52]. In case of lost or stolen keys, new addresses are also trivial to generate and reassign to patients.

3.3.6 Personal Health Records

Unlike the current standard practice of using provider-centric EHRs to maintain and manage patient data, personal health records (PHRs) are applications used by patients, the true data owners, to access and manage their health information. The ultimate goal for PHRs is to help patients securely and conveniently collect, track, and control their complete health records compiled from various sources, including provider visit data, immunization history, prescriptions records, physical activity data collected from Smartphone devices, and many more. PHRs enable patients to control how their health information is used and shared, verify the accuracy of their health records, and correct potential errors in the data [53]. Enterprises and technology companies, such as Apple and Microsoft, have begun exploring centralized solutions with their Apple Health [40] and Microsoft HealthVault [54] products. Centralized approaches do not resolve the data sharing problem at its core, however, and may therefore face similar hurdles as existing disparate EHR systems.

Blockchains, in contrast, allow distribution of control to individuals via decentralization enabled by consensus algorithms. By creating a widely accessible and secure data distribution service that connects to existing health systems, patients can easily aggregate their medical history without requesting a copy from every provider they have visited. Connections to personal smart devices are also possible as Blockchains remove the “distrust” between healthcare professionals and third party health tracking apps and services. Furthermore, permission-based data distribution can be set up with smart contracts to guarantee that patients (1) remain in control of their data access, (2) are aware of the origin of aggregated data sources, and (3) are informed when their data is accessed by providers. Data origin and access history are made transparent to the patients through immutable audit logs to always keep patients up-to-date of when and by whom their health information is retrieved.

3.3.7 Health Insurance Claim Adjudication

Health insurance is used to protect individual assets from the devastating costs of a major accident, medical emergency, or treating a chronic disease and to ensure care is provided when needed. It can cover the cost of doctor visits, medical, and surgical expenses, depending on the type of health insurance coverage [55]. Patients may a portion of out-of-pocket costs at the point of care, but the remaining costs are submitted as claims to the health insurance companies. Providers are then reimbursed through the “claim adjudication” process, whereby the insurer determines their financial responsibility for the payment and the payment amount (if applied)directly made to the provider [56].

The insurer may decide to pay the claim in full, deny the claim, or reduce the amount paid to the provider. The decision to reduce a payment to the provider is typically made when the insurance company has determined that the billed service is inappropriate or medically unnecessary for the diagnosis or procedure codes. Therefore, “it is important to ensure that all claims submitted for payment are coded accurately. As soon as an insurance company receives a medical claim, they begin a thorough review. Sometimes even small errors such as a misspelled patient name may cause a claim to be rejected” [57]. The majority of claims today are processed automatically without manual intervention, but as claims become more complex and are plagued by error and fraud [58], claim adjudication remains a challenging process.

Currently, twenty-two percent of claims get rejected either because they are not received by the insurer or they contain defects, such as incomplete or incorrect demographic data or lack of proof supporting the services billed [59].Fortunately, smart contracts present an opportunity for automating the adjudication process further by distributing and thus making claims transparent to the provider and insurer, exposing potential errors and frauds that can be corrected or investigated in a much timelier manner. Another benefit of creating these pre-established agreements via smart contracts is to ensure involved participants are up-to-date and properly notified as policies or rules change.

3.4 Healthcare Inoperability Challenges Faced by Blockchain-based Applications

While it is important to understand the fundamental properties that Blockchains possess, it is also crucial to analyze domain-specific challenges that this technology may face to ensure the practicality of Blockchain use-cases. This section examines four key interoperability challenges faced by Blockchain-based healthcare applications: system evolvability, data storage on Blockchain, healthcare information privacy, and system scalability, as summarized in Table 3.

Table 3: Summary of the Interoperability Challenges Faced by Blockchain Technologies

Challenge	Section(s)	Challenge Summary
Evolvability	3.4.1, 3.6.1	Blockchain is immutable but needs to support health system evolution
Storage	3.4.2, 3.6.2	Inefficient and costly on-chain storage should be minimized yet still provide data access
Privacy	3.4.3, 3.6.3	Data transparency of the Blockchain should be balanced with health privacy concerns
Scalability	3.4.4, 3.6.4	Relevant data should be filtered out from all events recorded on the Blockchain

3.4.1 Evolvability Challenge: Supporting System Evolution While Minimizing Integration Complexity

Many apps are created with the assumption that data is easy to change. This assumption is valid in most centralized systems where the data is managed by some trusted party, such as Amazon Web Services. In a Blockchain-based app with decentralized storage, however, data is difficult to modify *en masse* and its change history is recorded as an immutable log. A critical consideration when using Blockchain technology in health system design is thus to ensure that the data structures defined in a Blockchain (*i.e.* via smart contracts) are designed to facilitate evolution where needed to minimize change.

Although evolution must be supported, healthcare data must often be accessible from a variety of deployed systems that cannot easily be changed over time. Necessary data structures should therefore be designed in a way that are loosely coupled and minimize the usability impact of evolution on the clients that interact with data in the Blockchain. Section 3.6.1 shows how we design with the ABSTRACT FACTORY pattern to facilitate evolution while minimizing the impact on dependent clients, focusing on a concrete evolution challenge involving entity creation and management of healthcare participants.

3.4.2 Storage Challenge: Minimizing Data Storage Requirements on the Blockchain

Healthcare applications typically serve thousands or millions of participants, including providers, patients, billing agents, and so on. It may incur enormous overhead when large volumes of data are being stored in a Blockchain–particularly if data normalization and denormalization techniques are not carefully considered. Not only is it costly to store these data, but data access operations may also fail if/when the cost exceeds Blockchain-network defined data size limit. For instance, the Ethereum public Blockchain defines a "gas limit" that limits the capacity of data operations to prevent attacks manifest through infinite looping [3].

An important design consideration for Blockchain-based healthcare apps is thus to minimize data storage requirements on-chain yet provide sufficient flexibility to manage individuals' health concerns. Section 3.6.2 shows how to design smart contracts with the FLYWEIGHT pattern to ensure unique entity account creation that maximizes sharing of common intrinsic data across entities while still allowing extrinsic data to vary in individual entities.

3.4.3 Privacy Challenge: Balancing Data Sharing Capability with Privacy Concerns

The US Office of the National Coordination for Health Information Technology (ONC) has outlined a number of basic technical requirements for achieving interoperability [5]. These requirements include identifiability and authentication of all participants, ubiquitous and secure infrastructure to store and exchange data, authorization and access control of data sources, and the ability to handle data sources of various structures. Blockchains are emerging as promising and cost-effective means to meet some of these requirements due to their inherent design principles built upon secure cryptography and resilient peer-to-peer networks. Smart contract-enabled Blockchain-based DApps provide the healthcare domain with capabilities like digital asset sharing (such as health records sharing) and audit trails of transaction history (such as data access records, which are essential for improving healthcare interoperability.

Although storing patient health data in the Blockchain may provide substantial potential benefits to interoperability and immediate availability, there are also significant risks due to the transparency of the data because each Blockchain manager/miner maintains a complete copy of Blockchain data. In particular, even when encryption is applied, it is still possible that the current encryption techniques may be broken in the future or that vulnerabilities in the encryption implementations used may lead to private information potentially being decryptable and compromised in the future. In Section 3.6.3 we discuss how to a Blockchain-based app using the PROXY pattern can facilitate interoperability while keeping sensitive patient data from being directly encoded in the Blockchain.

3.4.4 Scalability Challenge: Tracking Relevant Health Changes Scalably Across Large Patient Populations

Communication gaps and information sharing challenges are serious impediments to healthcare innovation and the quality of patient care. Providers, hospitals, insurance companies, and even departments within the same health organizations experience disconnectedness caused by delays or the lack of information flow. Patients are commonly cared for at various sources, such as private clinics, regional urgent care centers, enterprise hospitals, and telemedicine practice. A provider may serve hundreds or more patients whose associated health activities must be tracked. Without any activity monitoring or filtering mechanism implemented, it would require tremendous computational effort for a provider to review a patient's health transactions on-demand. Section 3.6.4 shows how a Blockchain-based app design using the PUBLISHER-SUBSCRIBER pattern can be aid in scalably detecting and tracking relevant health changes.

3.5Case Study DApp Overview

This section presents the structure and functionality of a case study DApp for Smart Health (DASH)[2] we developed to explore the efficacy of applying Blockchain technology to the healthcare domain. This prototype was implemented on an Ethereum test Blockchain to emulate a minimal version of a personal EHR system. It provides a web-based portal for patients to self-report and access their medical records, as well as submit prescription requests. DASH also includes a staff portal for providers to review patient data and fulfill prescription requests based on permissions given by patients. Figure 3 shows the structure and workflow of DASH.

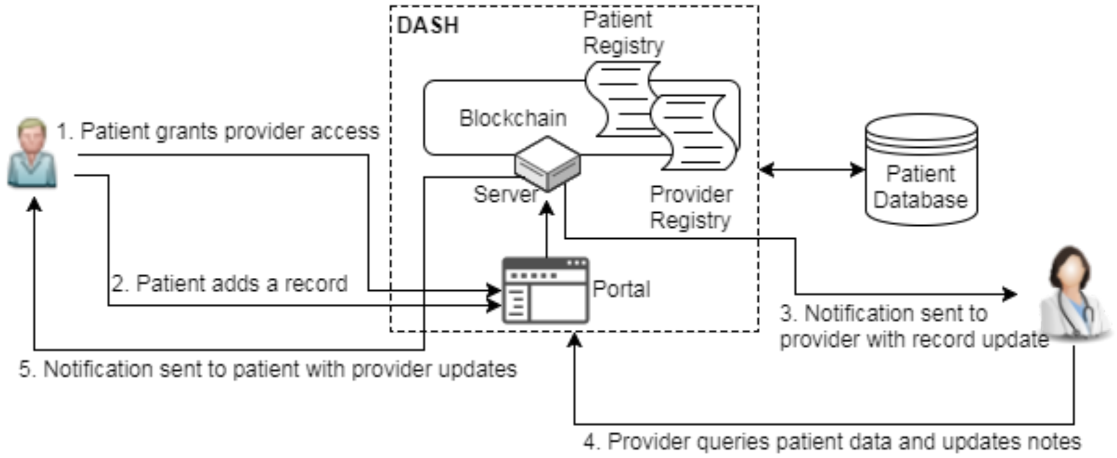

DASH
Patient Registry
Blockchain
Server
Provider Registry
Portal
Patient Database
1. Patient grants provider access
2. Patient adds a record
3. Notification sent to provider with record update
4. Provider queries patient data and updates notes
5. Notification sent to patient with provider updates

Figure 3 DASH Architecture Overview[3]

The core user features supported in DASH can be summarized as the follows:

[2] There is no relationship between our DASH app and the "Dash" cryptocurrency, even though they are both based on Blockchain technologies.

[3] Reprinted, with permission. Figure source: Peng Zhang, Jules White, Douglas C. Schmidt, and Gunther Lenz, Applying Software Patterns to Address Interoperability in Blockchain-based Healthcare Apps, the 24th Pattern Languages of Programming conference, October 22-25, 2017, Vancouver, Canada.

1. Patients can grant a provider permissions to access their health records or prescription requests via the DASH Portal
2. Patients can add a health record through a standardized, pre-formatted form through the DASH Portal
3. Health related activities (*i.e.*, prescription requests and health record additions) related to a patient are sent to providers with authorized access to the patient's data with secure notification messages
4. Provider with authorized access to a patient's records can query, make changes, and upload physician notes to the data, as well as fulfill the patient's prescription requests
5. Patients will be notified of any update to their health record performed by the provider

DASH uses a *Patient Registry* contract to store a mapping, or relationship that links unique patient identifiers to their associated *Patient Account* contract addresses (locations). Each *Patient Account* contract maintains a list of healthcare providers (via unique provider identifiers) who are granted read/write access to the patient's medical records. At its current state, DASH is limited to only provide data access services to two types of users: patients and providers[4]. Patient health records are stored off-chain in a secure database, implementing the FHIR data standards. The reason for storing patient data in a centralized database is to simulate a data silo, as it is in today's health systems, to later on exercise data integration with other siloed databases. Our database server creates a secure socket to exchange permission-based tokenized access to patient data using standard public key cryptography. Provider and patient users who are members of DASH are each equipped with two secure cryptographic key pair for (1) encrypting and decrypting data references for authorizing access to a patient dataset and (2) signing new health records and verifying signatures to prevent tampering to the data.

3.6 Design Considerations of Blockchain-Based Apps for Healthcare Use Cases

This section details the applications of familiar software patterns to address each healthcare interoperability challenge facing Blockchain apps. Namely, we focus on ABSTRACT FACTORY, FLYWEIGHT , PROXY, and PUBLISHER-SUBSCRIBER [60, 61] patterns and demonstrate how to incorporate these patterns in the DASH design. Figure 4 shows how an anatomy of DASH

[4] Naturally, there are other patterns relevant in this domain, which can be explored in future research.

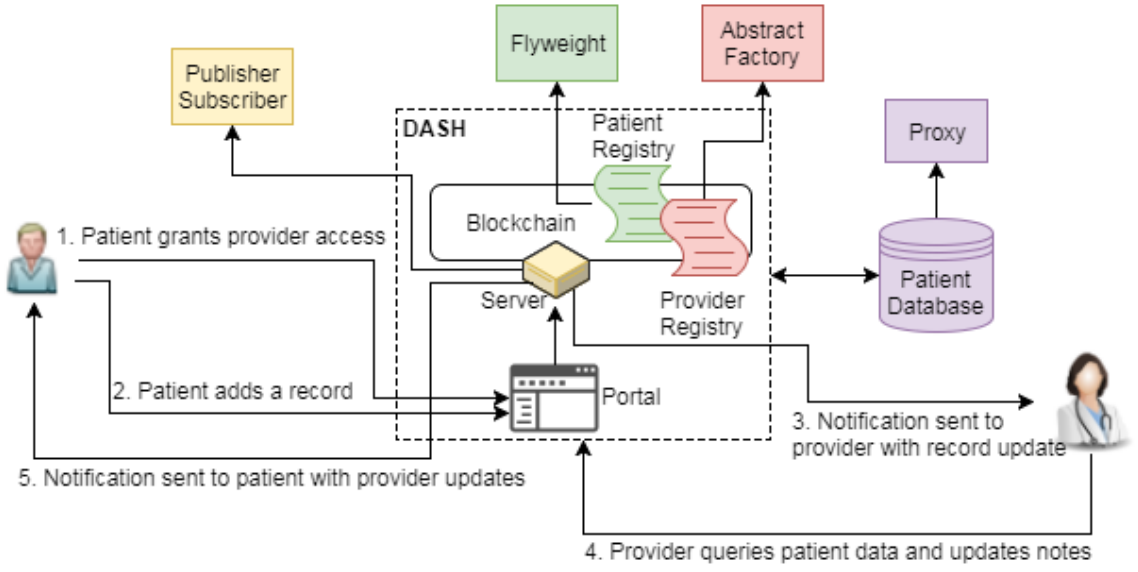

Publisher Subscriber
Flyweight
Abstract Factory
Proxy
DASH
Patient Registry
Blockchain
Server
Provider Registry
Portal
Patient Database
1. Patient grants provider access
2. Patient adds a record
3. Notification sent to provider with record update
4. Provider queries patient data and updates notes
5. Notification sent to patient with provider updates

Figure 4 Incorporating Software Patterns to DASH Design[5]

design with software pattern applications to address the following design challenges:

- ABSTRACT FACTORY assists with organization/individual account creation and management based on user types, especially in a structured or evolving organization
- FLYWEIGHT ensures unique account creation on the Blockchain and maximizes sharing of common, intrinsic data
- PROXY protects health information privacy while facilitating seamless interactions between separate components in the system to ensure appropriate levels of data accessibility
- PUBLISHER-SUBSCRIBER aids in scalably managing health change events and actively notifying healthcare participants when and only when relevant changes occur

Detailed applications of these patterns are discussed in depth in the following subsections.

3.6.1 Evolvability: Maintaining Evolvability While Minimizing Integration Complexity

Design problems faced by Blockchain-based apps. As part of the initial user on boarding process, account creation and management may occur on the Blockchain in order to define unique identifications for an organization or individual. Many organizations in the healthcare industry, such as enterprise hospitals and insurance companies, are hierarchical and evolving in nature. One design problem is thus to create an account structure that supports organization evolvability while minimizing integration complexity when new entities of different functions (e.g., a new division or department) are introduced. Specifically, the immutability property of Blockchain technology ensures that smart contract interface (including data member definitions and functions) cannot be modified. Each change to a smart contract must be deployed as a new contract object on the Blockchain and distributed among all the network nodes so it can be executed on demand.

To minimize interface changes over time, it is important to create a modular design. As a concrete example, suppose one smart contract defines a function that manages interactions between different departments in a highly-structured hospital. Without a modular design,

[5] Reprinted, with permission. Figure source: Peng Zhang, Jules White, Douglas C. Schmidt, and Gunther Lenz, Applying Software Patterns to Address Interoperability in Blockchain-based Healthcare Apps, the 24th Pattern Languages of Programming conference, October 22-25, 2017, Vancouver, Canada.

many decisions must be made to identify the appropriate departmental accounts involved, thereby creating a large number of branching statements. As new departments are introduced, the decision-making code will likely have to change more than once, making each previous version of smart contract obsolete. A desired model should minimize interface changes to a contract.

Solution → Apply the ABSTRACT FACTORY pattern to support design evolvability. Creating complex account structures in smart contracts for an evolving, hierarchical environment can be modularized by applying the ABSTRACT FACTORY pattern [60]. This pattern allows DApps like DASH to delegate the responsibility for providing account creation services to an abstract "factory" object (which is a contract instance itself). A concrete factory object can then inherit methods from the abstract factory and customize them to create accounts for a specific set of related or interacting sub-entities. For instance, with this pattern implemented in the account structure, when a new department is introduced, the app simply creates another concrete factory object for the new entity without affecting existing accounts in other smart contracts. Figure 5 shows the basic structure of this pattern in the context of DASH.

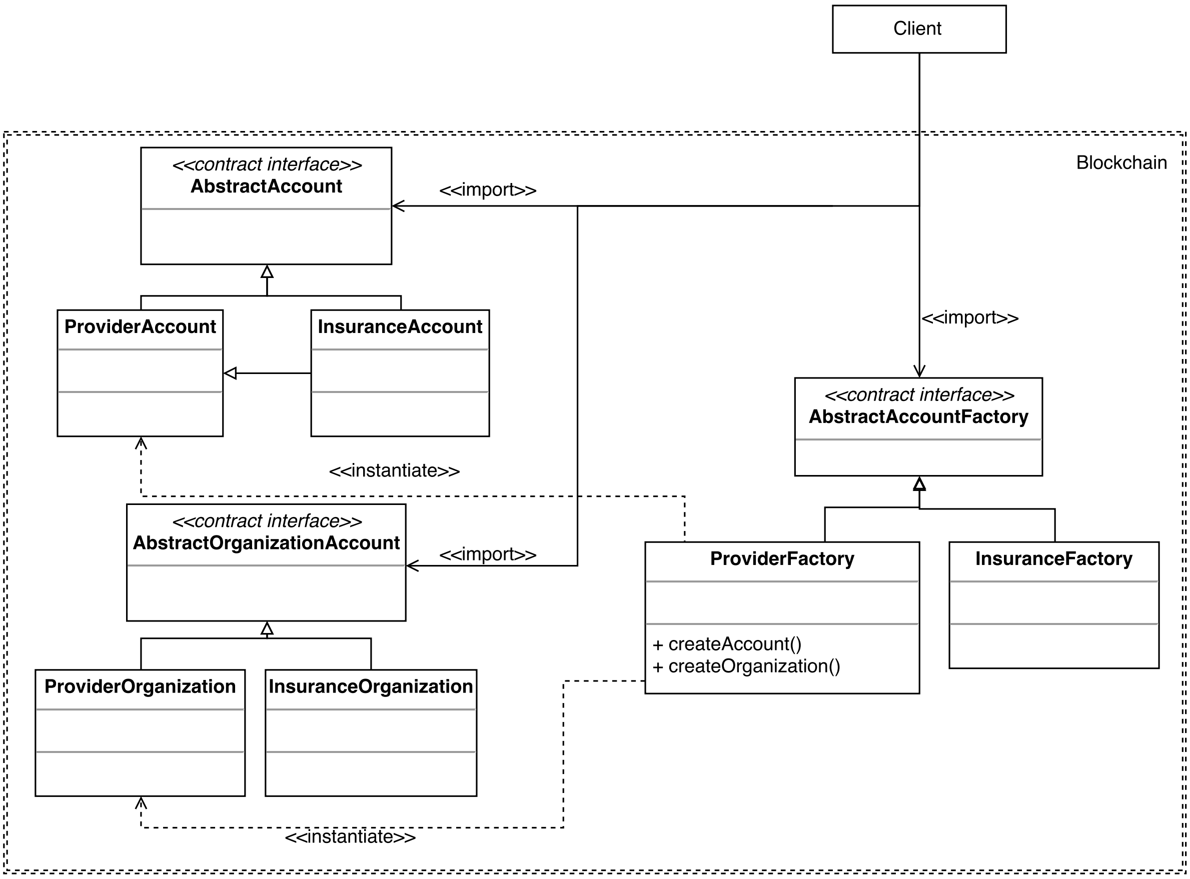


Figure 5 Organizational Account Creation Process with ABSTRACT FACTORY Applied in DASH Design[6]

DASH's Blockchain component uses an *AbstractAccountFactory*, the abstract factory contract object, to define some common logic (*i.e.*, *createAccount* and *createOrganization* methods) shared by all concrete factories (*e.g.*, *ProviderFactory*, *InsuranceFactory*, *etc.*). When *Client*, i.e. DASH user, requests a new entity account structure to be created, regardless of the entity type, DASH server creates a concrete entity factory inherited from *Abstrac-*

[6] Reprinted, with permission. Figure source: Peng Zhang, Jules White, Douglas C. Schmidt, and Gunther Lenz, Applying Software Patterns to Address Interoperability in Blockchain-based Healthcare Apps, the 24th Pattern Languages of Programming conference, October 22-25, 2017, Vancouver, Canada.

tAccountFactory to create one account type for storing individual users and another account type for storing information regarding the entity organization. This corresponds to the instantiation of a *Provider* account and a *ProviderOrganization* for the *Provider* entity in the example from Figure 5. The process of creating an account structure for an entity in this design is completely decoupled from already defined structures, thus leaving their corresponding account contract objects intact.

Consequences of applying ABSTRACT FACTORY. Without using a high-level abstraction, such as an abstract factory, creating constituent accounts for an entity would imply many if-else decisions made at runtime to determine the appropriate concrete account factory to execute. This tight coupling is cumbersome since *Client* (who interacts with the Blockchain component in the above example) has to be familiar with the implementation details in order to execute each concrete factory's methods properly. For example, to create an account structure for either *Provider*or*Insurance* entity in this case, *Client* must be exposed to both *ProviderFactory* and *InsuranceFactory* contracts and make decisions at runtime regarding which factory methods to call, as shown in Figure 6. The example only presents two concrete entity examples, but as the number of entities scales up or as the departments within an organization scale out, the *Client* will have to keep track of an overwhelming amount of detailed implementation.

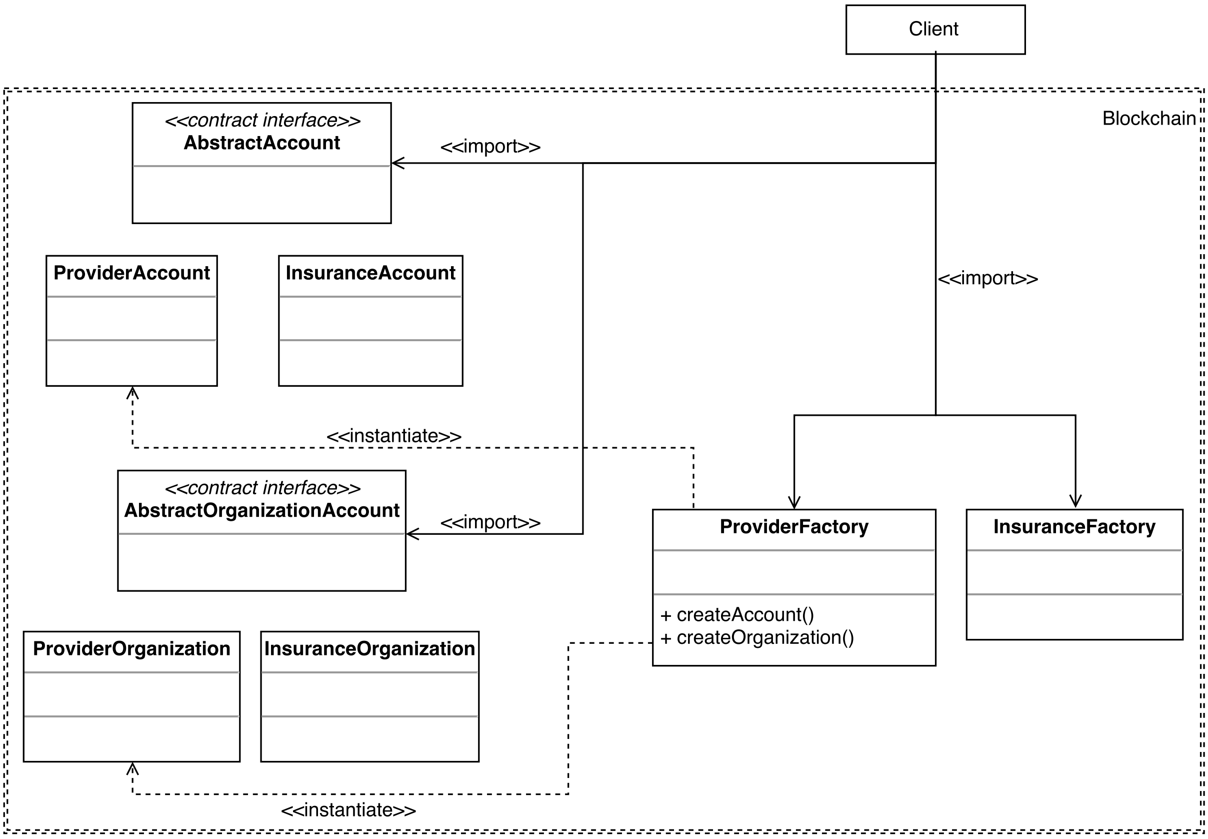


Figure 6 Organizational Account Creation Process without Applying the ABSTRACT FACTORY Pattern[7]

Entity creation with ABSTRACT FACTORY introduces a loose coupling between the client (*e.g.*, DApp server) and specific smart contract implementations (*i.e.*, the Blockchain component). In particular, newly defined entity interactions can inherit from the abstract con-

[7] Reprinted, with permission. Figure source: Peng Zhang, Jules White, Douglas C. Schmidt, and Gunther Lenz, Applying Software Patterns to Address Interoperability in Blockchain-based Healthcare Apps, the 24th Pattern Languages of Programming conference, October 22-25, 2017, Vancouver, Canada.

tract to preserve common properties and have entity-specific customizations. Because of the loose coupling, existing contracts remain unmodified, minimizing contract deprecation. The downside of using this design in a Blockchain, however, would be the extra storage (and therefore extra cost if used in a public Blockchain with a mining incentive) overhead incurred by an added layer of indirection for the abstract factory and its instantiation.

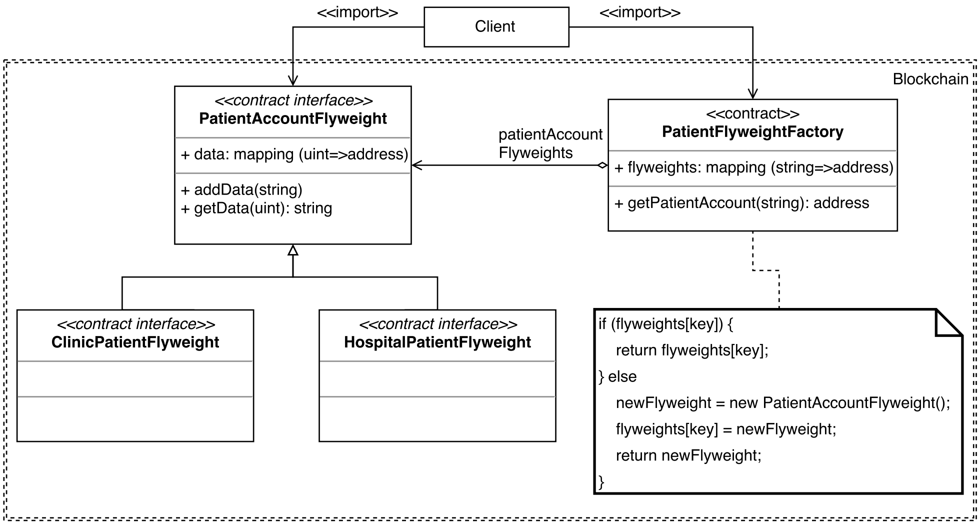
3.6.2 Storage: Storage Challenge: Minimizing Data Storage Requirements

Design problem faced by Blockchain-based apps. It may be inevitable that a Blockchain-based healthcare app requires some data to be maintained on-chain. However, a good design should minimize the storage requirements by maximizing data sharing on the Blockchain to avoid storage and cost overhead. The price for Blockchain to have transparency and immutability is that all data and transaction records maintained in the Blockchain are replicated and distributed to every node in the network. To avoid unnecessary data storage, such as duplicated data or unattended data, it is therefore important to create a design that maximizes shared data on-chain.

If a Blockchain is used as a database to store patient billing data, then in a large-scale healthcare scenario, millions of records will be replicated on all Blockchain miners. Moreover, billing data could include detailed patient insurance information, such as their ID number, insurance contact information, coverage details, and other aspects that the provider needs to bill for services. Capturing all this information for every patient generates excessive amounts of data in the Blockchain.

For example, suppose it is necessary to store some insurance and billing information (encrypted) in the Blockchain. In reality, most patients are covered by one of a relatively small subset of insurers (in comparison to the total number of patients, *e.g.*, each insurance policy may cover 10,000s or 100,000s of patients). A substantial amount of non-varying information, such as details on what procedures are covered by an insurance policy, is common across patients that can therefore be reused and shared. To bill for a service, however, this common intrinsic information must be combined with extrinsic, varying information (such as the patient's ID number and billing address) that is specific to each patient. A good design should maximize sharing of such common data to reduce on-chain storage and, meanwhile, is capable of providing access to complete data objects on demand.

Solution → Apply the FLYWEIGHT pattern to minimize data storage in the Blockchain. Combining the FLYWEIGHT pattern [60] with a factory object can help minimize data storage in the Blockchain. In particular, the factory can establish a registry model that stores shared data between a set of entities in a common contract, *i.e.*, the registry, while externalizing varying data to be stored in entity-specific contracts. The registry can also maintain references (*i.e.*, addresses) to entity-specific contracts and return a combined extrinsic and intrinsic (common) data set upon request. Figure 7 shows the flyweight registry model applied to DASH.



```
if (flyweights[key]) {
    return flyweights[key];
} else
    newFlyweight = new PatientAccountFlyweight();
    flyweights[key] = newFlyweight;
    return newFlyweight;
}
```

Figure 7 Applying the FLYWEIGHT Registry Model to Create Unique Patient Accounts in DASH Design[8]

In the example above, DASH implements a registry model to manage patient information. Specifically, *PatientFlyweightFactory* is a patient registry that creates a *PatientAccountFlyweight* contract for each patient and links the flyweight reference address to the patient via a unique identifier to avoid account duplication. It stores some data common to different groups of patients (such as insurance coverage as described previously), preventing an exorbitant amount of memory usage from saving repeated data in all patient accounts. *PatientAccountFlyweight* can have concrete implementations, e.g. *ClinicPatientFlyweight* and *HospitalPatientFlyweight* based on the patient's primary care type, to store varying, patient-specific contact and billing information based. *PatientFlyweightFactory* only create a new account if the specified patient identifier does not yet exist in the registry; otherwise, the registry retrieves the address associated with the existing *PatientAccountFlyweight* contract. To retrieve the complete insurance and billing information of a particular patient, *Client* (DASH server) only needs to invoke a function call from *PatientFlyweightFactory* with the patient identifier to obtain the *PatientAccountFlyweight address*. The function *getData()* of *PatientAccountFlyweight* is then responsible for returning the combined intrinsic and extrinsic data object back to the *Client*.

Consequences of applying the FLYWEIGHT pattern. The flyweight registry model provides better management to the large object pool (*e.g.*, patient accounts in the example above). It minimizes redundancy in similar objects by maximizing data and data operation sharing. Particularly in the above example, if some common insurance policy details were stored in each patient's contract directly, any change to a policy detail would be costly since it would require rewriting a huge number of impacted contracts. Data sharing with flyweight registry can help minimize the cost of changes to the common intrinsic state in Blockchain-based apps. One downside of applying the Flyweight pattern, however, is the added layer of complexity. For example, creating the flyweight contract is another transaction to verify and include in the Blockchain before it can be executed. Although, this extra step can be outweighed by the efficiency in entity/data management that the registry model provides.

3.6.3 Privacy: Balancing Data Accessibility with Privacy Concerns

[8] Reprinted, with permission. Figure source: Peng Zhang, Jules White, Douglas C. Schmidt, and Gunther Lenz, Applying Software Patterns to Address Interoperability in Blockchain-based Healthcare Apps, the 24th Pattern Languages of Programming conference, October 22-25, 2017, Vancouver, Canada.

Design problems faced by Blockchain-based apps. If a Blockchain-based healthcare app must expose sensitive data or common Meta data (such as patient identifying information or insurance information in the example in Section 3.5.2) on the Blockchain, it must be designed to maximize health data privacy on-chain while facilitating health information exchange. In particular, a fundamental aspect of a Blockchain is that data and all data change history stored in the Blockchain are public, immutable, and verifiable. For financial transactions focused on proving that the transfer of an asset indeed occurred, these properties are critical. When the goal is to store data in the Blockchain, however, it is important to understand how these properties will impact the use case.

For example, storing patient health records in the Blockchain can be problematic since it requires that data be public and immutable. Although data can be encrypted before being stored, should all patient data be publicly distributed to all Blockchain nodes? If a consortium Blockchain were deployed amongst large U.S. hospital organizations with more than 10 hospitals, we would be storing these patient data roughly 40 times [62]. Even if encryption is used, the encryption technique may be broken in the future or bugs in the implementation of the encryption algorithms or protocols used may make the data decryptable in the future. Immutability, however, prevents owners of the data from removing data or its change history from the Blockchain when a security flaw is found.

Many other scenarios, ranging from discovery of medical mistakes in the data to changing standards may necessitate the need to change the data over time. In scenarios where the data may be changed, the public and immutable nature of the Blockchain creates a fundamental tension that must be resolved. On the one hand, healthcare providers would want incorruptible data so it can be trusted and never compromised. At the same time however, providers may need the data to be changeable so that they can account for possible errors. A practical Blockchain-based health app should protect patient privacy and also ensure data integrity.

Solution → Apply the PROXY pattern to enable secure and private data services. Combining the PROXY pattern [60] with a secure data retrieving service, such as an oracle [63], can enable secure and private data exchange services. The oracle network is a third-party service that allows a smart contract to query or retrieve data sources outside the Blockchain address space and ensures that retrieved data is genuine and uncompromised. To reduce computation overhead on-chain, a proxy can be created as a lightweight representation or placeholder for the real data until its retrieval is required. Figure 8 shows the application of a proxy in DASH.

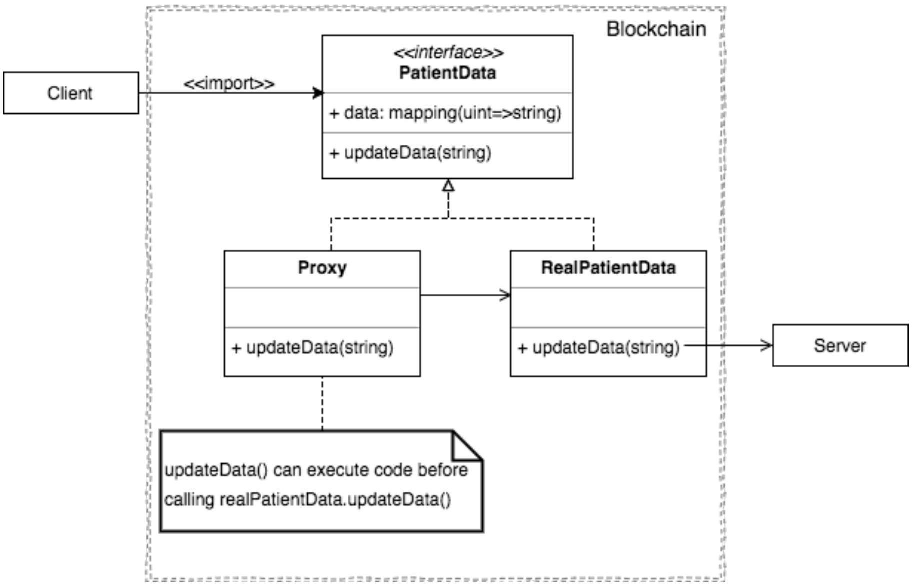


Figure 8 Structure of a PROXY Application in DASH[9]

DASH uses a *Proxy* contract to expose some simple metadata about a patient and later refer to the actual heavyweight implementation *RealPatientData* on demand to retrieve the complete data object via an Oracle. Each query request and modification operation are logged in an audit trail that is transparent to the entire Blockchain network for verification against data corruption or unauthorized data access. In the case of a proxified contract (heavyweight implementation) being updated with a new storage option (e.g., replacing an Oracle with some other data service), the interface to the proxy contract can remain unchanged, encapsulating the low-level implementation variations in the proxified contract.

Consequences of applying the PROXY pattern. A proxy object can perform lightweight housekeeping or auditing tasks by storing some commonly used metadata in its internal states without having to perform expensive operations (such as retrieving health data via an Oracle service). It typically follows the same interface as the real object and can execute the original heavyweight function implementations as needed. It can also hide information about the real object as needed to protect patient data privacy. However, PROXY may cause disparate behavior when the real object is accessed directly by some *Client* while the proxy surrogate is accessed by others. Nonetheless, proper usage of a proxy with an Oracle service can provide a private channel for protected information exchange.

3.6.4 Scalability: Tracking Relevant Events Scalably Across Large Traffic

Design problem faced by Blockchain-based apps. A practical Blockchain-based health system may need to manage and track relevant health events across large patient populations. Therefore, it should be designed to filter out useful health-related information from all communication traffic (*i.e.*, transaction records) occurring on the Blockchain. For example, the Ethereum public Blockchain maintains a transparent record of all contract creation and operation execution history along with regular cryptocurrency transactions. The availability of information makes Blockchain a potentially autonomous approach to improve care coordination across different participants and teams (*e.g.*, physicians, pharmacists, insurance agents,

[9] Reprinted, with permission. Figure source: Peng Zhang, Jules White, Douglas C. Schmidt, and Gunther Lenz, Applying Software Patterns to Address Interoperability in Blockchain-based Healthcare Apps, the 24th Pattern Languages of Programming conference, October 22-25, 2017, Vancouver, Canada.

etc.) who would normally communicate through various channels that are manual and time-consuming, such as through telephoning or faxing [64].

However, records on the Blockchain are continually growing. Without a meticulously crafted design, capturing any specific health-related topic from all occurred events on-demand would imply exhaustive transaction receipt lookups and topic filtering, which requires non-trivial computations and may result in delayed responses. A good design should support relevant health information relays to facilitate coordinated care as the events occur. For instance, care should be seamless from the point when a patient self-reports illness (*e.g.*, through a familiar web or mobile interface) to the point when the patient receives the necessary prescriptions created by their primary care provider to treat the reported symptoms. Moreover, clinical reports and follow-up procedure should be relayed to and from the associated care centers (*e.g.*, care provider office and pharmacy) in a timely manner.

Solution → Apply the PUBLISHER-SUBSCRIBER pattern to facilitate scalable information filtering. Incorporating a notification service using the PUBLISHER-SUBSCRIBER pattern [61] can facilitate scalable information filtering. In this design, health activities are only broadcast to providers who subscribe to events relating to their patients. It alleviates the tedious filtering process of determining which care provider should be notified about what patient activities as large volumes of transactions take place. This design also fosters an interoperable environment, which allows providers across various organizations or regions to participate. To avoid computation overhead on the Blockchain, the actual processing of patient activities data can be performed off-chain by the interfacing DApp server. Specifically, when the publisher sends an update, its subscribers only need to do a simple update to an internal state variable that records the publisher's contract address, which the DApp server actively monitors for changes. When a change occurs, the responsibility for the computation-heavy content filtering task (e.g., retrieving the change activity from the publisher using the address) is delegated to the DApp server from the Blockchain. The DApp server is context-aware at this point because each subscriber has an associated contract address accessible by the server. The server can then filter the content based on subscribed topics and update the contract states of appropriate subscribers as needed. Figure 9 shows the PUBLISHER-SUBSCRIBER pattern applied in our DASH design for the notification service.

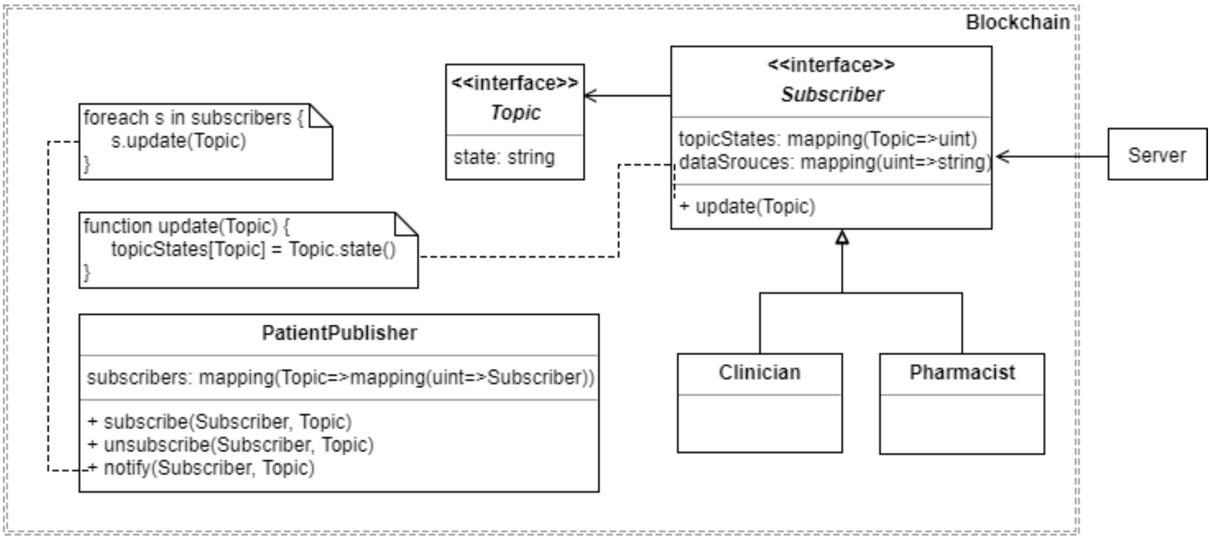

Blockchain
<<interface>>
Topic
state: string
<<interface>>
Subscriber
topicStates: mapping(Topic=>uint)
dataSrouces: mapping(uint=>string)
+ update(Topic)
Server
foreach s in subscribers {
s.update(Topic)
}
function update(Topic) {
topicStates[Topic] = Topic.state()
}
PatientPublisher
subscribers: mapping(Topic=>mapping(uint=>Subscriber))
+ subscribe(Subscriber, Topic)
+ unsubscribe(Subscriber, Topic)
+ notify(Subscriber, Topic)
Clinician
Pharmacist

Figure 9 Using the PUBLISHER-SUBSCRIBER Pattern to Provide Notification Service in DASH[10]

In DASH, events associated with patient-reported sickness symptoms (a concrete *Topic*) are subscribed to by the patient's primary care provider and pharmacist (*i.e.*, *Clinician* and *Pharmacist* types of *Subscribers*). When this event occurs, the *PatientPublisher* contract notifies the *Subscribers* by updating the *state* variable value in the concrete *Subscriber* contracts. DASH server, which is actively listening for changes in the *Subscriber* states, finds the updated status and in turn queries the latest patient health changes to pass onto the proper *Subscriber* objects.

Consequences of applying the PUBLISHER-SUBSCRIBER pattern. Applying a notification service in a healthcare DApp design is useful when a state change in one contract must be reflected in others without keeping the contracts tightly coupled. Adding or removing subscribers and topics is trivial as it only requires minimal changes in the state variables and not the interface or implementation. It also makes topic/content filtering more manageable when subscription relations are clearly defined in each participant's contract state. In addition, this design enables communication across participants from various organizations, as required of an interoperable system.

However, unlike a traditionally centralized notification service, there are Blockchain-specific limitations, which include:

(1) Potential delays in updates received by subscribers due to the extra step of validation required by the Blockchain infrastructure and

(2) High storage requirements associated with defining very fine-grained topic subscriptions on-chain.

Concretely, in a Blockchain the order of which transactions to be executed and verified is determined by the miners based on some pre-defined rules. In Blockchains with financial mining incentives for instance, the priority of a transaction may be based on transaction fees paid by its sender and how long it has been in the transaction pool. Transactions with the highest priority will be executed first and added to the Blockchain sooner, which could cause delays in the notification service for transactions with lower priority. If subscribers want to

[10] Reprinted, with permission. Figure source: Peng Zhang, Jules White, Douglas C. Schmidt, and Gunther Lenz, Applying Software Patterns to Address Interoperability in Blockchain-based Healthcare Apps, the 24th Pattern Languages of Programming conference, October 22-25, 2017, Vancouver, Canada.

receive fine-grained message topics, the amount of computation for filtering out the messages sent out by publishers may also cause overhead. This overhead may result in either failure to publish messages due to network limitation or restriction enforced on the computation. One mitigation strategy may be to have broader topics with fewer filter requirements on-chain and to handle more detailed message filtering off the Blockchain.

3.7 Summary of Key Design Lessons Learned

Blockchain and programmable smart contracts provide a platform for creating decentralized systems and applications that may serve a wide range of use cases in the healthcare industry, such as facilitating data sharing, managing patient digital identity, enabling personal health records, tracking prescriptions, expediting claims adjudication, and many more. Properly leveraging Blockchain technologies, given the complexity of the health care domain, however, requires that key domain-specific concerns be addressed in the ecosystem design. These concerns include but are not limited to application system evolvability, storage requirements minimization, patient data privacy protection, and application scalability across large user populations. This chapter described these concerns and recommended approaches to mitigate these challenges through an example using our Blockchain-based DApp for Smart Health (DASH). Specifically, we detailed the applications of four familiar software patterns, namely, ABSTRACT FACTORY, FLYWEIGHT, PROXY, and PUBLISHER-SUBSCRIBER.

Based on our experience developing the DASH case study presented in this book chapter, we learned the following lessons:

- The public, immutable, and verifiable properties of the Blockchain enable a more interoperable environment that is not easily achieved using traditional approaches that mostly rely on a centralized server or data storage.
- Each time a smart contract is modified, a new contract object is created on the Blockchain. Important design decisions must therefore be made in advance to avoid the cost and storage overhead introduced by changes in contract interface.
- To best leverage these properties of Blockchain in the healthcare context, concerns regarding system evolvability, storage costs, sensitive information privacy, and application scalability must be taken into account.
- Combining time-proven design practices with the unique properties of the Blockchain helps guide the design for health systems that are more modular, easier to integrate and maintain, and less susceptible to change.

3.8 Research directions in Healthcare-Focused Blockchain Applications

There are many research directions in applying Blockchain technology to the healthcare industry due to the complexity of this domain and the need for more robust and effective information technology systems. An interoperable architecture would undoubtedly play a significant role throughout many healthcare use cases that face similar data sharing and communication challenges. From the more technical aspect, much research is needed to pinpoint the most practical design process in creating an interoperable ecosystem using the Blockchain technology while balancing critical security and confidentiality concerns in healthcare. Whether to create a decentralized application leveraging an existing Blockchain, such as Ethereum [3], IBM's Hyperledger Fabric [65], or J.P.Morgan's Quorum [66], or to design a healthcare domain- or use case-specific Blockchain remains an open question.

Additional research on secure and efficient software practice for applying the Blockchain technology in healthcare is also needed to educate software engineers and domain experts on the potential and also limitations of this new technology. Likewise, validation and testing approaches to gauge the efficacy of Blockchain-based health care architectures compared to existing systems are also important (e.g., via performance metrics related to time and cost of computations or assessment metrics related to its feasibility). In some cases, a new Blockchain network may be more suitable than the existing Blockchains; therefore, another direction may be investigating extensions of an existing Blockchain or creating a healthcare Blockchain that exclusively provides health-related services.

3.9 Conclusion

In this chapter, we described two pressing issues in healthcare, focusing on the need to (1) create an interoperable system to facilitate clinical communications and data exchange and (2) enable patient-centric care to provide patients with access and control of their complete medical history. We then identified seven concrete healthcare scenarios that share similar technical pain points, which can be alleviated with blockchain technology. However, the complexities associated with healthcare involvement and regulations create additional challenges inevitably facing blockchain-based systems, such as system evolvability, information privacy and communication scalability. Targeting a subset of these healthcare-specific challenges, we presented four design recommendations demonstrated with a case study prototype that we have previously developed. From our experience, we have seen the great potential of blockchain technology in creating secure and effective healthcare ecosystems with its inherent unique properties. In addition, we have also observed the importance of integrating domain-specific concerns and needs into blockchain-based designs. Overall, blockchain has a wide range of possibilities in healthcare, which invites many research opportunities in this space.

References

1 Nakamoto, S.: ‘Bitcoin: A peer-to-peer electronic cash system’, 2008

2 Johnston, D., Yilmaz, S.O., Kandah, J., Bentenitis, N., Hashemi, F., Gross, R., Wilkinson, S., and Mason, S.: ‘The general theory of decentralized applications, dapps’, GitHub, June, 2014, 9

3 Buterin, V.: ‘Ethereum white paper’, GitHub repository, 2013

4 Zhang, P., White, J., Schmidt, D.C., and Lenz, G.: ‘Applying software patterns to address interoperability in Blockchain-based healthcare apps’, arXiv preprint arXiv:1706.03700, 2017

5 DeSalvo, K.B.: ‘RE: Connecting Health and Care for the Nation: A Shared Nationwide Interoperability Roadmap’, 2015

6 Olson, S., and Downey, A.: ‘Sharing clinical research data: workshop summary. 2013’, in Editor (Ed.)^(Eds.): ‘Book Sharing clinical research data: workshop summary. 2013’ (Washington, DC: National Academies Press, 2016, edn.), pp.

7 The biggest healthcare breaches of 2017. (2017, December 06). Retrieved March 01, 2018, from http://www.healthcareitnews.com/slideshow/biggest-healthcare-breaches-2017-so-far?page=18 Adler-Milstein, J., and Bates, D.W.: ‘Paperless healthcare: Progress and challenges of an IT-enabled healthcare system’, Business Horizons, 2010, 53, (2), pp. 119-130

9 Zhang, P., Walker, M., White, J., Schmidt, D.C., and Lenz, G.: ‘Metrics for assessing Blockchain-based healthcare decentralized apps’, Proceedings of 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom), October 12-15, 2017, Dalian, China

10 Dolin, R.H., Alschuler, L., Beebe, C., Biron, P.V., Boyer, S.L., Essin, D., Kimber, E., Lincoln, T., and Mattison, J.E.: ‘The HL7 clinical document architecture’, Journal of the American Medical Informatics Association, 2001, 8, (6), pp. 552-569

11 Bender, D., and Sartipi, K.: ‘HL7 FHIR: An Agile and RESTful approach to healthcare information exchange’, in Editor (Ed.)^(Eds.): ‘Book HL7 FHIR: An Agile and RESTful approach to healthcare information exchange’ (IEEE, 2013, edn.), pp. 326-331

12 Reid, P.P., Compton, W.D., Grossman, J.H., and Fanjiang, G.: 'Building a better delivery system: a new engineering/health care partnership' (National Academies Press Washington, DC, 2005. 2005)

13 Middleton, B., Bloomrosen, M., Dente, M.A., Hashmat, B., Koppel, R., Overhage, J.M., Payne, T.H., Rosenbloom, S.T., Weaver, C., and Zhang, J.: 'Enhancing patient safety and quality of care by improving the usability of electronic health record systems: recommendations from AMIA', Journal of the American Medical Informatics Association, 2013, 20, (e1), pp. e2-e8

14 Castaneda, C., Nalley, K., Mannion, C., Bhattacharyya, P., Blake, P., Pecora, A., Goy, A., and Suh, K.S.: 'Clinical decision support systems for improving diagnostic accuracy and achieving precision medicine', Journal of clinical bioinformatics, 2015, 5, (1), pp. 4

15 Singh, H., Giardina, T.D., Meyer, A.N., Forjuoh, S.N., Reis, M.D., and Thomas, E.J.: 'Types and origins of diagnostic errors in primary care settings', JAMA internal medicine, 2013, 173, (6), pp. 418-425

16 Schiff, G.D., Hasan, O., Kim, S., Abrams, R., Cosby, K., Lambert, B.L., Elstein, A.S., Hasler, S., Kabongo, M.L., and Krosnjar, N.: 'Diagnostic error in medicine: analysis of 583 physician-reported errors', Archives of internal medicine, 2009, 169, (20), pp. 1881-1887

17 Kaushal, R., Shojania, K.G., and Bates, D.W.: 'Effects of computerized physician order entry and clinical decision support systems on medication safety: a systematic review', Archives of internal medicine, 2003, 163, (12), pp. 1409-1416

18 Warren, E.: 'Strengthening research through data sharing', New England Journal of Medicine, 2016, 375, (5), pp. 401-403

19 Taichman, D.B., Backus, J., Baethge, C., Bauchner, H., De Leeuw, P.W., Drazen, J.M., Fletcher, J., Frizelle, F.A., Groves, T., and Haileamlak, A.: 'Sharing clinical trial data—a proposal from the International Committee of Medical Journal Editors', in Editor (Ed.)^(Eds.): 'Book Sharing clinical trial data—a proposal from the International Committee of Medical Journal Editors' (Mass Medical Soc, 2016, edn.), pp.

20 Geifman, N., Bollyky, J., Bhattacharya, S., and Butte, A.J.: 'Opening clinical trial data: are the voluntary data-sharing portals enough?', BMC medicine, 2015, 13, (1), pp. 280

21 Gross, G.E.: 'The role of the tumor board in a community hospital', CA: a cancer journal for clinicians, 1987, 37, (2), pp. 88-92

22 Nourie, C. E. (Ed.). (2015, February). Your Medical Records. Retrieved March 01, 2018, from http://m.kidshealth.org/en/teens/medical-records.html

23 Schoenberg, R.: 'Bridged patient/provider centric method and system', in Editor (Ed.)^(Eds.): 'Book Bridged patient/provider centric method and system' (Google Patents, 2013, edn.), pp.

24 Lumpkin, J., Cohn, S.P., and Blair, J.S.: 'Uniform data standards for patient medical record information', National Committee on Vital and Health Statistics, 2003, 53

25 Heath, S. (2016, June 23). How is Interoperability Supporting Patient-Centered Care? Retrieved March 01, 2018, from https://ehrintelligence.com/news/how-is-interoperability-supporting-patient-centered-care

26 Black, N., Varaganum, M., and Hutchings, A.: 'Relationship between patient reported experience (PREMs) and patient reported outcomes (PROMs) in elective surgery', BMJ Qual Saf, 2014, pp. bmjqs-2013-002707

27 Ash, J.S., Berg, M., and Coiera, E.: 'Some unintended consequences of information technology in health care: the nature of patient care information system-related errors', Journal of the American Medical Informatics Association, 2004, 11, (2), pp. 104-112

28 Hripcsak, G., Bloomrosen, M., FlatelyBrennan, P., Chute, C.G., Cimino, J., Detmer, D.E., Edmunds, M., Embi, P.J., Goldstein, M.M., and Hammond, W.E.: 'Health data use, stewardship, and governance: ongoing gaps and challenges: a report from AMIA's 2012 Health Policy Meeting', Journal of the American Medical Informatics Association, 2014, 21, (2), pp. 204-211

29 LaRose, R., Strover, S., Gregg, J.L., and Straubhaar, J.: 'The impact of rural broadband development: Lessons from a natural field experiment', Government Information Quarterly, 2011, 28, (1), pp. 91-100

30 Laxmaiah Manchikanti, M., Standiford Helm, I., MA, J.W.J., PhD, V.P., MSc, J.S.G., and DO, P.: 'Opioid epidemic in the United States', Pain physician, 2012, 15, pp. 2150-1149

31 Bernstein, I.: 'Drug Supply Chain Security Act', in Editor (Ed.)^(Eds.): 'Book Drug Supply Chain Security Act' (2017, edn.), pp.

32 Christie, G.C., Baker, C.G.C., Cooper, G.R., Kennedy, C.P.J., Madras, B., and Bondi, F.A.G.P.: 'THE PRESIDENT'S COMMISSION ON COMBATING DRUG ADDICTION AND THE OPIOID CRISIS'

33 CDC Rx Awareness Campaign Overview. (2016, December). Retrieved March 1, 2018, from https://www.cdc.gov/rxawareness/pdf/RxAwareness-Campaign-Overview-a.pdf

34 Arlotta, C. (2015, July 31). Opioid Overdose and Abuse Awareness Campaigns Continue To Grow. Retrieved March 01, 2018, from https://www.forbes.com/forbes/welcome/?toURL=https%3A%2F%2Fwww.forbes.com%2Fsites%2Fcjarlotta%2F2015%2F07%2F31%2Fopioid-overdose-abuse-awareness-campaigns-continue-to-grow%2F&refURL=https%3A%2F%2Fwww.google.com%2F&referrer=https%3A%2F%2Fwww.google

35 James, J.: 'Dealing with drug-seeking behaviour', Australian prescriber, 2016, 39, (3), pp. 96

36 Nelson, L.S., Juurlink, D.N., and Perrone, J.: 'Addressing the opioid epidemic', Jama, 2015, 314, (14), pp. 1453-1454

37 10 Pros and Cons of Telemedicine | eVisit® Telehealth Solutions. (n.d.). Retrieved March 01, 2018, from https://evisit.com/10-pros-and-cons-of-telemedicine/

38 Sood, S., Mbarika, V., Jugoo, S., Dookhy, R., Doarn, C.R., Prakash, N., and Merrell, R.C.: 'What is telemedicine? A collection of 104 peer-reviewed perspectives and theoretical underpinnings', Telemedicine and e-Health, 2007, 13, (5), pp. 573-590

39 What is telemedicine? - Definition from WhatIs.com. (n.d.). Retrieved March 01, 2018, from http://searchhealthit.techtarget.com/definition/telemedicine

40 IOS - Health. (n.d.). Retrieved March 01, 2018, from https://www.apple.com/ios/health/

41 Zhang, P., White, J., Schmidt, D.C., Lenz, G., and Rosenbloom, S.T.: 'FHIRChain: Applying Blockchain to Securely and Scalably Share Clinical Data'

42 Multidisciplinary Cancer Care: The Benefits of a Tumor Board. (n.d.). Retrieved March 01, 2018, from https://www.maacenter.org/blog/multidisciplinary-cancer-care-the-benefits-of-a-tumor-board

43 Wanner, M. (n.d.). Why is cancer so difficult to cure? Retrieved March 01, 2018, from https://www.jax.org/news-and-insights/2015/december/why-no-cure-for-cancer

44 Cancer patients face the ultimate choice, with no room for error. (2017, January 10). Retrieved March 01, 2018, from https://www.statnews.com/2016/10/06/immunotherapy-cancer-clinical-trials/

45 Parkin, D.M.: 'The evolution of the population-based cancer registry', Nature Reviews Cancer, 2006, 6, (8), pp. 603

46 Just, B.H., Marc, D., Munns, M., and Sandefer, R.: 'Why patient matching is a challenge: research on Master Patient Index (MPI) data discrepancies in key identifying fields', Perspectives in health information management, 2016, 13, (Spring)

47 Feied, C., and Iskandar, F.: 'Master patient index', in Editor (Ed.)^(Eds.): 'Book Master patient index' (Google Patents, 2007, edn.), pp.

48 Cross Organizational Patient Identity Management: Challenges and Opportunities. (2016). Retrieved March 1, 2018, from http://sequoiaproject.org/wp-content/uploads/2017/02/2017-02-22-HIMSS-2017-Patient-Matching-Challenges-and-Opportunities-v001.pdf

49 Fernández-Alemán, J.L., Señor, I.C., Lozoya, P.Á.O., and Toval, A.: 'Security and privacy in electronic health records: A systematic literature review', Journal of biomedical informatics, 2013, 46, (3), pp. 541-562

50 Patient Matching Errors Risk Safety Issues, Raise Health Care Costs. (2017, June 29). Retrieved March 1, 2018, from http://www.pewtrusts.org/en/multimedia/data-visualizations/2017/patient-matching-errors-risk-safety-issues-raise-health-care-costs

51 Patient matching peril: Why unique patient identifiers are a unique problem for hospitals. (n.d.). Retrieved March 01, 2018, from https://www.beckershospitalreview.com/healthcare-information-technology/patient-matching-peril-why-unique-patient-identifiers-are-a-unique-problem-for-hospitals.html

52 Krawiec, R., Housman, D., White, M., Filipova, M., Quarre, F., Barr, D., Nesbitt, A., Fedosova, K., Killmeyer, J., and Israel, A.: ' Blockchain: Opportunities for health care', in Editor (Ed.)^(Eds.): 'Book Blockchain: Opportunities for health care' (2016, edn.), pp. 1-16

53 Tang, P.C., Ash, J.S., Bates, D.W., Overhage, J.M., and Sands, D.Z.: 'Personal health records: definitions, benefits, and strategies for overcoming barriers to adoption', Journal of the American Medical Informatics Association, 2006, 13, (2), pp. 121-126

54 Microsoft HealthVault. (n.d.). Retrieved March 01, 2018, from https://www.healthvault.com/en-us/

55 What Is Health Insurance. (2016, January 5). Retrieved March 1, 2018, from https://www.medicalnewstoday.com/info/health-insurance

56 Peterson, B.E., Kwant Jr, J.W., Cecil, V.C., and Provost, W.A.: 'Electronic creation, submission, adjudication, and payment of health insurance claims', in Editor (Ed.)^(Eds.): 'Book Electronic creation, submission, adjudication, and payment of health insurance claims' (Google Patents, 2002, edn.), pp.

57 Claims Processing: What is Claims Adjudication? (n.d.). Retrieved March 01, 2018, from http://www.apexedi.com/what-is-claims-adjudication/

58 Blockchain's potential use cases for healthcare: hype or reality? (2017, September 28). Retrieved March 01, 2018, from http://www.healthcareitnews.com/news/blockchains-potential-use-cases-healthcare-hype-or-reality

59 5 Key Challenges in Medical Billing Industry. (2017, August 01). Retrieved March 01, 2018, from http://www.invensis.net/blog/infographics/5-key-challenges-in-medical-billing-industry/

60 Gamma, E.: 'Design patterns: elements of reusable object-oriented software' (Pearson Education India, 1995. 1995)

61 Buschmann, F., Henney, K., and Schimdt, D.: 'Pattern-oriented Software Architecture: on patterns and pattern language' (John wiley & sons, 2007. 2007)

62 Top 30 Largest Hospital Systems in America. (2017, December 05). Retrieved March 01, 2018, from http://www.compassphs.com/blog/healthcare-trends/healthcare-fast-facts-top-30-largest-hospital-systems-in-america/

63 Swan, M.: ' Blockchain thinking: The brain as a dac (decentralized autonomous organization)', in Editor (Ed.)^(Eds.): 'Book Blockchain thinking: The brain as a dac (decentralized autonomous organization)' (2015, edn.), pp. 27-29

64 EHRs and healthcare interoperability: The challenges, complexities, opportunities and reality. (2015, November 30). Retrieved March 01, 2018, from http://www.healthcareitnews.com/blog/ehrs-healthcare-interoperability-challenges-complexities-opportunities-reality

65 Hyperledger Fabric. (n.d.). Retrieved March 01, 2018, from
http://www.hyperledger.org/projects/fabric

66 Quorum. (n.d.). Retrieved March 01, 2018, from
https://www.jpmorgan.com/global/Quorum

Key Terminology &Definitions[11]

Abstract Factory Pattern–Abstract Factory encapsulates a group of individual factories that have a common theme without specifying their concrete classes. It separates the details of implementation of a set of objects from their general usage and relies on object composition, as object creation is implemented in methods exposed in the factory interface.

Cryptocurrency– A cryptocurrency is a digital or virtual currency that uses cryptography for security, making it difficult to counterfeit. It is not issued by any central authority and is immune to government interference or manipulation.

Decentralized App (DApp)– A DApp is an autonomously operated open-source application that cryptographically stores its data and records of operation in a public, decentralized Blockchain (via a smart contract for instance) to avoid central points of failure. It uses a native or an existing form of cryptographic tokens for monetizing the Dapp. The tokens must be necessary for the use of the app.

Digital Asset – A digital asset is anything existing in a binary format that comes with (some) rights to use. It could be a native asset lacking physical substance that can be owned or controlled to produce value, such as digital music, images, movies, electronic funds, software, etc. It could also be a digital representation of some traditional paper-based asset, such as certificates and titles of property, gold, autos, stock, currency, etc.

Electronic Health Records (EHRs)– EHRs are a digital version of a patient's paper medical records and chart that make information available instantly and securely to authorized healthcare practitioners. They contain the medical and treatment histories of patients and can also store information beyond standard clinical data collected in a provider's office, such as diagnoses, medications, treatment plans, allergies, and lab results.

Factory – A factory is a function or method that creates an object of a varying prototype or class from some method call. It creates abstraction or encapsulation so that program code is not tied to specific classes or objects, allowing the class hierarchy or prototypes to be changed or refactored without modifying code that uses them.

Flyweight Pattern–A flyweight object minimizes memory usage by sharing as much data as possible with other similar objects. It is particularly when a simple repeated representation would use an unacceptable amount of memory. Common parts of the object state can be shared internally, while varying parts of the data are stored externally in entity-specific objects. When all data regarding a specific object are requested, both external and internal data can be retrieved.

Interoperability –Interoperability allows two or more systems to exchange information and use the exchanged information. The three levels of health information technology interoperability ordered from lowest to highest fidelity are: 1) Foundational interoperability that enables data exchanges between healthcare systems without requiring the ability for the receiving party to interpret the data, 2) Structural interoperability that defines the formats of exchanged clinical data and ensures that received data are preserved and can be interpretable at the data field using the predefined formats, and 3) Semantic interoperability that allows for interpretation of data exchanged by not only syntax (structure) but also semantics (meaning) of the data.

[11] Key technology/technical terms used in the book chapter will be explained wherever they appear or at the "Key Terminology & Definitions" section. Apart from regular References, additional References are included in the "References for Advance/Further reading" for the benefit of advanced readers.

Master Patient Index (MPI) – Master Patient Index is a single registration system of all patients across various departments within a hospital. Similarly, an Enterprise Master Patient Index (EMPI) is a database that consolidates patient identities from multiple healthcare organizations. The goal of MPI or EMPI is to provide uniquely identifying patient information so that it may be queried and used to match existing records.

ProxyPattern–A proxy is a wrapper object that is used by the client to access the real serving object behind the scenes. It implements the same interface as the real object and can execute the original heavyweight function implementations as needed. A proxy can provide extra functionality that is typically lightweight housekeeping or auditing tasks, such as checking preconditions or caching when operations on the real object are resource intensive.

Publisher-Subscriber Pattern– This is a messaging pattern where senders of messages, publishers, do not directly send to specific receivers, called subscribers. Instead, publishers categorize published messages into topics without knowledge of which subscribers. Subscribers express interest in one or more topics and only receive messages that are of interest, without knowledge of which publishers.

Smart Contract – Smart contracts are enhancements built atop some Blockchain technologies (such as Ethereum). They are code that directly controls the exchanges or redistributions of digital assets between two or more parties according to certain rules or agreements established between the involved parties. They enable development of DApps to interact with Blockchain and support on-chain storage.

Software Pattern– A software pattern is a general repeatable solution to a commonly occurring problem in software design. It is not a finished design that can be transformed directly into code. Instead, it provides a description or template for how to solve a problem that can be used in many different situations. Software patterns allow developers to communicate using well-known, well understood names for software interactions. Common design patterns can be improved over time, making them more robust than ad-hoc designs.

Authors Bios

Miss Peng Zhang: *Peng Zhang* is a Computer Science PhD student at Vanderbilt University, Nashville, TN, USA. Her research interests include model-driven design for engineering and healthcare IT systems, intelligent model constructions using machine and deep learning, decentralized algorithms and protocols for facilitating and securing clinical communications, and application and enhancement of Blockchain technologies for moving towards patient-centered care.

Affiliation/Address:
E-mail: peng.zhang@vanderbilt.edu
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, TN, USA

Dr. Douglas C. Schmidt: Dr. Douglas C. Schmidt is Cornelius Vanderbilt Professor of Computer Science, the Associate Chair of the Electrical Engineering and Computer Science department, and a Senior Researcher at the Institute for Software Integrated Systems, all at Vanderbilt University. He is also a Visiting Scientist at the Software Engineering Institute (SEI) at Carnegie Mellon University. As an internationally renowned and widely cited (h-index of 80) researcher, his work focuses on patterns, optimization techniques, and empirical analyses of object-oriented and component-based frameworks and model-driven engineering

tools that facilitate the development of distributed real-time and embedded (DRE) middleware frameworks and mobile cloud computing applications on parallel platforms running over wireless/wired networks and embedded system interconnects. He has published over 10 books and 600+ papers (including 100+ journal papers) in top IEEE, ACM, IFIP, and USENIX technical journals, conferences, and books that cover a range of topics. He has mentored and graduated over 40 Ph.D. and Masters students working on these research topics and has presented 550+ keynote addresses, invited talks, and tutorials on mobile cloud computing with Android, reusable patterns, concurrent object-oriented network programming, distributed system middleware at scores of technical conferences.

Affiliation/Address:
E-mail: d.schmidt @vanderbilt.edu
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, TN, USA

Dr. Jules White: Dr. Jules White is an Assistant Professor in the Department of Electrical Engineering and Computer Science at Vanderbilt University. He was previously a faculty member in Electrical and Computer Engineering and won the Outstanding New Assistant Professor Award both at Virginia Tech. His research has published over 120 papers and won 5 Best Paper and Best Student Paper Awards. Dr. White's research focuses on securing, optimizing, and leveraging data from mobile cyber-physical systems. His mobile cyber-physical systems research spans focus on: (1) mobile security and data collection, (2) high-precision mobile augmented reality, (3) mobile device and supporting cloud infrastructure power and configuration optimization, and (4) applications of mobile cyber-physical systems in multidisciplinary domains, including energy-optimized cloud computing, smart grid systems, healthcare/manufacturing security, next-generation construction technologies, and citizen science. His research has been licensed and transitioned to industry, where it won an Innovation Award at CES 2013, attended by over 150,000 people, was a finalist for the Technical Achievement at Award at SXSW Interactive, and was a top 3 for mobile in the Accelerator Awards at SXSW 2013. His research is conducted through the Mobile Application computinG, optimizatoN, and secUrity Methods (MAGNUM) Group at Vanderbilt, which he directs.

Affiliation/Address:
E-mail: jules.white@vanderbilt.edu
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, TN, USA

Mr. Gunther Lenz: Mr. Gunther Lenz is an influential Software Solution Architect/Executive, MBA, published author, invited speaker, and strategic technology visionary with 16+ years of achievements to drive successfully drive software innovation projects. Established technology, process, and organizational transformation agent. Recognized expert in software architecture, big data, and cloud computing. Diverse experience in enterprise, high-growth, and incubation environments within a variety of industries. Published two books and numerous novel articles in the area of software engineering and software architecture. Selected Program Committee Member at prestigious international conferences. Awarded Microsoft Most Valuable Professional for Software Architecture (180 worldwide) 2005-2008.

Affiliation/Address:
E-mail: gunther.lenz @varian.com
OCS Technology Office
Varian Medical Systems
Palo Alto, CA, USA

INDEX