Thesis Final 9

by Petnathean Julled

Submission date: 13-Sep-2021 02:09PM (UTC+0700)
Submission ID: 1622337728
File name: Thesis_Final_9.docx (12.54M)
Word count: 23402
Character count: 138156

CROSS-ENTERPRISE DOCUMENT SHARING (XDS) IMPLEMENTATION BASED ON BLOCKCHAIN TECHNOLOGY

PETNATHEAN JULLED

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF SCIENCE
(CYBER SECURITY AND INFORMATION ASSURANCE)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2021

COPYRIGHT OF MAHIDOL UNIVERSITY

Thesis
entitled

CROSS-ENTERPRISE DOCUMENT SHARING (XDS) IMPLEMENTATION BASED ON BLOCKCHAIN TECHNOLOGY

.....
Mr. Petnathean Julled
Candidate

.....
Assadarat Khurat,
Dr.-Ing. (Computer Security)
Major advisor

.....
Pattanasak Mongkolwat,
Ph.D. (Computer Science)
Co-advisor

.....
Asst. Prof. Thitinan Tantidham,
Ph.D. (Computer Science)
Co-advisor

.....
Prof. Patcharee Lertrit,
M.D., Ph.D. (Biochemistry)
Dean
Faculty of Graduate Studies
Mahidol University

.....
Assoc. Prof. Vasaka Visoottiviseth,
Ph.D. (Computer Engineering)
Program Director
Master of Science Program in Cyber Security and Information Assurance (International Program)
Faculty of Information and Communication Technology
Mahidol University

Thesis
entitled

CROSS-ENTERPRISE DOCUMENT SHARING (XDS) IMPLEMENTATION BASED ON BLOCKCHAIN TECHNOLOGY

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science (Cyber Security and Information Assurance)
on
June 17, 2021

.....
Mr. Petnathean Julled
Candidate

.....
Dr. Chakan Pramkaew,
Ph.D. (Computer Science)
Chair

.....
Assadarat Khurat,
Dr.-Ing. (Computer Security)
Member

.....
Asst. Prof. Thitinan Tantidham,
Ph.D. (Computer Science)
Member

.....
Pattanasak Mongkolwat,
Ph.D. (Computer Science)
Member

.....
Prof. Patcharee Lertrit,
M.D., Ph.D. (Biochemistry)
Dean
Faculty of Graduate Studies
Mahidol University

.....
Pattanasak Mongkolwat,
Ph.D. (Computer Science)
Dean
Faculty of Information and
Communication Technology
Mahidol University

ACKNOWLEDGEMENTS

The success of this thesis would never be succeeded without the attentive support from Dr.Assadarat Khurat and Dr.Pattanasak Mongkolwat.

I would like to thank them for their kindness, dedication, and patience which always support me during the progression of this work. I also would like to thank all the persons who have advised whether it is about Blockchain technology or another context. Your advice is what greatly enhances the quality of this work.

Petnathean Julled

CROSS-ENTERPRISE DOCUMENT SHARING (XDS) IMPLEMENTATION BASED ON BLOCKCHAIN TECHNOLOGY

PETNATHEAN JULLED 5936474

M.Sc. (CYBER SECURITY AND INFORMATION ASSURANCE)

THESIS ADVISORY COMMITTEE: ASSADARAT KHURAT, Ph.D., PATTANASAK MONGKOLWAT, Ph.D., THITINAN TANTIDHAM, Ph.D.

ABSTRACT

On the increasing demand for a better quality of healthcare services, some topics involve healthcare information technology in terms of operational efficiency. Healthcare information sharing and interoperability between healthcare organizations are one of the major solutions to improve healthcare service quality. However, the healthcare industry poses many challenges that inhibit solutions to become reality. The Integrating Healthcare Enterprise (IHE) initiative to standardize healthcare information sharing methods to address health document sharing issues between different enterprises, Cross-Enterprise Document Sharing (XDS.b) Profile allows the adopted organizations to share health documents simultaneously using the central exchange.

Like other industries, cyber-security threats have threatened the healthcare information domain. These threats increase the difficulty in the development of health information sharing networks and causing damage to healthcare enterprises. These cyber-threats can cause damage to the industry in many aspects, especially those cyber-attack that targeting integrity and availability of data. These kinds of cyber-attacks can severely hinder the continuity of medical operations, potentially resulting in the cost of a patient's life. There are many solutions technology proposed to deal with these kinds of cyber-attacks. One of the technologies that are the trend to deal with cyber-threats threatening integrity and availability of data is Blockchain technology.

Several pieces of research have proposed the concept of using Blockchain technology to solve health information sharing issues but, there are still many limitations that prevented Blockchain technology from effectively integrated with data like health information. This is where the IHE XDS.b profile could be used with Blockchain technology to allow health document sharing through the decentralized networks while address cyber-security issues through unique characteristics of Blockchain technology. In this work, we propose the approach for integrating Blockchain technology with the IHE XDS.b profile which results in the new concept of health information exchange.

KEY WORDS: HEALTH INFORMATION / INTEROPERABILITY / INFORMATION SHARING / INFORMATION SECURITY / BLOCKCHAIN / SMART CONTRACT / IHE / XDS

298 pages

CONTENTS

	Page
ACKNOWKEDGEMENTS	**iii**
ABSTRACT	**iv**
LIST OF TABLES	**viii**
LIST OF FIGURES	**ix**
LIST OF ABBREVIATIONS	**xiv**
CHAPTER I INTRODUCTION	**1**
1.2 Objective	4
1.3 Problem Statement	4
1.4 Scope of Project	5
CHAPTER II LITERATURE REVIEW	**6**
2.1 Integrating the Healthcare Enterprise (IHE)	6
2.1.1 IHE Process	6
2.1.2 IHE Integration Profiles	7
2.1.3 IHE Information Technology Infrastructure Technical Framework	8
2.1.4 Cross-Enterprise Document Sharing Set-b (XDS.b) Profile	8
2.1.5 XDS Transaction Format Types	11
2.1.6 Transaction Object Type and Metadata Attributes	12
2.2 Blockchain Technology	13
2.2.1 Definition of Blockchain	13
2.2.2 Benefit of Blockchain	14
2.2.3 Blockchain Characteristics	14
2.2.4 Blockchain Types	15
2.2.5 Blockchain Components	16
2.3 Ethereum and Smart-Contract	20
2.3.1 Smart Contract	21
2.3.2 Quorum	21

Page

2.4 Related Work 22
2.4.1 A Blockchain-Based Approach to Health Information Exchange Networks 22
2.4.2 A Case Study for Blockchain in Healthcare: "MedRec" Prototype for Electronic Health Records and Medical Research Data 23
2.4.3 Blockchain-Based Data Preservation System for Medical Data 23
2.4.4 Blockchain-Based Electronic Healthcare Record System for Healthcare 4.0 Applications 24
CHAPTER III PROPOSED METHOD **25**
3.1 Use Case Scenario 25
3.2 Concept Design 26
3.3 Blockchain Design 27
3.4 Integrating Blockchain with XDS.b Profile 28
3.5 Design Functions 29
3.5.1 Document Register 29
3.5.2 Document Search 30
3.6 Process Flow 31
CHAPTER IV IMPLEMENTATION **33**
4.1 Blockchain setup 33
4.1.1 Machine Specifications 33
4.1.2 Go-Ethereum 33
4.1.3 Quorum Installation 35
4.1.4 Compile and Deploy Smartcontract Solidity Code 40
4.1.5 Deploy Smartcontract into Blockchain 45
4.1.6 Prepare NodeJS Coding Environment 46

Page

4.2 XDS Actors 47

4.2.1 XDS Document Repository Actor 47

4.2.2 XDS Document Consumer Actor 56

4.2.3 XDS Document Registry Actor 83

4.3 Implementation Result 123

4.4 Evaluation 133

4.4.1 Functionalities Test 134

4.4.2 Performance Test 145

CHAPTER V DISCUSSION AND CONCLUSION **153**

5.1 Discussion 153

5.2 Conclusion 154

REFERENCES **156**

APPENDIX **162**

BIOGRAPHY **270**

LIST OF TABLES

Table		Page
4-1	Modified metadata attributes in each sample transaction	133
4-2	Functionalities experiment result	143
A-1	SubmissionSet	162
A-2	Folder	163
A-3	DocumentEntry	164

LIST OF FIGURES

Figures		Page
2-1	IHE Process to create guideline for implementation of health information technology [20]	7
2-2	Cross-Enterprise Document Sharing - b Diagram [22]	11
2-3	Blockchain network formed from the participation of Blockchain nodes	18
3-1	XDS Profile within the scope of interest for this work	28
3-2	Integrating Blockchain into XDS.b Profile	29
3-3	The process Flow of XDS Document Registry Actor	32
4-1	Installation command-line for Go-Ethereum on Ubuntu [41]	34
4-2	Geth console accessed using "geth attach" command	35
4-3	Installing Quorum directly from its source	36
4-4	Cloning "7-Nodes" Quorum example from its repository available on Github	36
4-5	Initial configuration method for 7-Nodes example	37
4-6	Executing "istanbul-init.sh" with Linux Bash syntax	37
4-7	Content of "istanbul - genesis.json" file	38
4-8	IBFT 7-Nodes Blockchain activation script	38
4-9	The activation script activating all seven Blockchain nodes	39
4-10	All seven Blockchain nodes successfully activated	39
4-11	The content of "rebirth.sh" script	40
4-12	The content of "runmy7nodes.sh" script	40
4-13	ABI Code and Byte code generated can be copied and passed directly	41
4-14	ABI code (brown color) assigned into variable "abi"	43
4-15	Byte code (brown color) assigned into variable "bytecode"	44
4-16	The Web3js script for Smartcontract deploy	45
4-17	"npm install" command-line	46

Figures **Page**

4-18	Pseudocode represents general format of Register Document Set-b [ITI - 42]	48
4-19	XML Code snippet of Registry Document Set-b Response transaction sample	49
4-20	XML Code snippet of Registry Document Set-b [ITI-42] transaction sample	54
4-21	Javascript Code Snippet of XDS Document Repository Actor	55
4-22	Pseudocode represents general format of Registry Stored Query Request [ITI - 18]	57
4-23	Pseudocode represents general format of Query Response included “Object Reference” of search results	57
4-24	Pseudocode represents general format of Query Response included "Leaf Class" of search result	59
4-25	XML Code Snippet of RegistryStoredQueryRequest [ITI-18] Transaction Sample	60
4-26	XML Code Snippet of RegistryStoredQueryResponse Transaction Sample	65
4-27	The program prompt user to input query type	67
4-28	The program prompt user to input essential metadata attribute values	67
4-29	The program prompt user to input optional metadata attributes	67
4-30	The user chooses to start the query after input all known attributes	68
4-31	Javascript Code Snippet of XDS Document Consumer Actor	82
4-32	Pseudocode showing the process flow of the XDS Document Registry Actor	83
4-33	The pseudocode showing the process flow of XDS Document Registry Actor for Document Registering Function	84
4-34	Javascript Code Snippet of XDS Document Registry Actor Node Module import declaration and TCP Socket message receiver section	85

Figures **Page**

4-35	XDS Document Registry Actor This section checks if receiving message is ITI-42 or ITI-18 identified by its header	86
4-36	XDS Document Registry Actor Declaration of JSON variable to store all Metadata attributes by its position in the format	88
4-37	XDS Document Registry Actor Define variable of each Metadata attribute UUID label following IHE ITI Framework	89
4-38	XDS Document Registry Actor This section interprets and assort Metadata attribute value from ITI-42 to JSON	97
4-39	XDS Document Registry Actor This section passes JSON into Smart Contract as single string variable	100
4-40	Specified gas value spplying Ethereum Smartcontract execution	101
4-41	The pseudocode showing the process flow of Document Register Smartcontract	101
4-42	Solidity Code Snippet of Smart Contract (Highlight - green color) related to Document Registering Function	102
4-43	XDS Document Registry Actor Define variable of query request type UUID label following IHE ITI Framework	103
4-44	The process flow for the native-side Javasceipt program	103
4-45	XDS Document Registry Actor Identify query request type following received ITI-18 header and assort search keyword	107
4-46	XDS Document Registry Actor Check for the latest document ID published in Blockchain before beginning search operation	110

Figures **Page**

4-47	XDS Document Registry Actor Begin search operation by sequentially check each published contract one-by-one	113
4-48	XDS Document Registry Actor Check if value of Metadata attributes in each publish contract matched with search keyword before summarize search result.	118
4-49	XDS Document Registry Gather search result and response back to Document Consumer Actor	120
4-50	The pseudocode showing the process flow of Smartcontract function related to Document Search	121
4-51	Solidity Code Snippet of Smartcontract	122
4-52	All XDS Actors activated via its terminal (Top Left) XDS Document Registry Actor	124
4-53	XDS Document Registry Actor standby and wait for incoming XML Messages.	124
4-54	XDS Document Repository prompt for health document number to register	125
4-55	XDS Document Repository sent ITI-42 transaction to XDS Document Registry	125
4-56	XDS Document Registry received ITI-42 transaction and successfully registered the metadata set into the Blockchain while wait for other XML Messages	125
4-57	XDS Document Repository received response from XDS Document Registry	126
4-58	XDS Document consumer Actor prompt the user for input	127
4-59	XDS Document Consumer Actor prompt for essential search keyword values	127
4-60	XDS Document Consumer Actor prompt for optional search keyword values	127
4-61	XDS Document Consumer Actor sent ITI-18 transaction	128

Figures **Page**

4-62	XDS Document Registry received ITI-18 transaction then interpret the message	128
4-63	XDS Document Registry then begin search operation over Smartcontract	130
4-64	XDS Document Registry responding search result back to XDS Document Consumer	131
4-65	XDS Document Consumer received search result and display it to the user	132
4-66	Part of transaction samples content [59]	134
4-67	Ten of mockup transactions generated for the experiment	134
4-68	The XDS Document Repository initiate Document Register function	135
4-69	The XDS Document Registry received ITI-42 transaction	135
4-70	The XDS Document Registry Actor signal the Document Register success	136
4-71	The smartcontract return metadata attributes correctly when called	137
4-72	The XDS Document Consumer initiate Document Query	138
4-73	The XDS Document Registry Actor received the query	139
4-74	The XDS Document Registry Actor found the matching document and about to return the result to the XDS Document Consumer Actor	139
4-75	The XDS Document Consumer received the query result and display it to the user	140
4-76	Content of "permissioned-nodes.json" file define active nodes (node ids truncated for simpler explanation) [33]	142
4-77	Single node id represent single active node	142
4-78	"numNodes" variable in "istanbul-init.sh" file reassigned with new value	142
4-79	The 7-Node Example system log	144
4-80	Performance comparison of the system performing the Document Register function between when performed by only a single node and	147

Figures **Page**

when performed by all nodes at the same time
(without triggering the Document Query function at the time)

4-81 Processing time to complete Document Query compared between minimum keywords and maximum keywords performed by single node 148

4-82 Processing time to complete Document Query compared between minimum keywords and maximum keywords performed by all nodes 149

4-83 Processing time to complete Document Query compared between FindDocument and GetDocument performed by single node 150

4-84 Processing time to complete Document Query compared between FindDocument and GetDocument performed by all nodes 151

4-85 Processing time to complete Document Query compared between when performed by single node and when performed by all nodes 152

LIST OF ABBREVIATIONS

Abbreviation		Page
IHE	Integrating Healthcare Enterprise	1
XDS	Cross – Enterprise Document Sharing	1
XDS.b	Cross – Enterprise Document Sharing Set-b	1
IDE	Integrated Development Environment	44

CHAPTER I
INTRODUCTION

With the transition from the age of paperwork to digital records, the healthcare industry is now undergoing digital transformation. Efficiency and continuity are the main factors that driven the healthcare industry to change. Paperwork starts falling behind when a huge amount of data is produced by healthcare service operations from day to day. Health information undeniably has become an important component in developing efficient healthcare services [1–6]. On the increasing demand for the better quality of healthcare service, there is the topic that involves healthcare information technology in terms of operational efficiency. Healthcare information sharing and interoperability between healthcare organizations are one of the major solutions to improve healthcare service quality. Patient's health document data are scattered across different healthcare organizations, due to the foundation of healthcare informatics are separately developed by different organizations. Each healthcare organization has its own method to process and handle healthcare information. This makes it hard for one healthcare piece of information to interoperate with other. Lack of interoperability prevents many opportunities for healthcare service quality improvement. The patient may need to take extra repetitive care procedures when visiting a new hospital. Mistakes in communication between different physicians can cause misdiagnosis. So, there are many demands from the patient side that want their health journey to be connected and improve healthcare service quality.

To enable health information sharing from just one organization with another can cost much more than the benefit they can gain. Sharing health information with not fully-trusted party exposing vulnerabilities to the business model. The risks and benefits to the organization from sharing their patient information with others may not be worth it as compared to the risk. For example, health information sharing allows the physician to access a patient's health history in other hospitals give decent improvement to service quality, in turn, exposing the information access to cyber-criminal and provide a chance for business competitors to gain an advantage. This creates high friction for one

organization to share their information with others. It was even more difficult for an individual patient to integrate their healthcare with different providers. This makes the interoperability issue to be extremely difficult for every single organization to solve on its own. It revealed that these interoperation problems cause a huge decrease in inefficiency in healthcare operations and result in lower quality of healthcare service [7–14]. However, there is still no effective approach to tackle the problem. That means there still have an open issue on how to solve interoperability in the field of healthcare. [7,9–11]

That way many initiatives start to standardize healthcare information technology to allow healthcare organizations to be able to interoperate with each other. Integrating Healthcare Enterprise (IHE) is one of the well-known initiatives that provide specifications for using healthcare informatics standardization. IHE provides an implementation framework and guideline for developing a health informatics system. For health document sharing between different organizations, IHE provides a Cross-Enterprise Document Sharing (XDS.b) profile. The profile act as a guideline for the system developer to implement their system to meet the requirement where the system can share health document with other organizations. This profile will be the main focus of this work, to deal with the health information sharing problem.

In the current age of information digitalization, cybersecurity has become an important issue for many organizations and individuals. Anyone can become a target of cyber-attacks. The healthcare industry is one of the major targets that become a victim of cyber-attacks each year [15]. Followed by the digitalization of hospital operations and information systems, the amount of cyber-attack and variations rise as the technology developed. These incidents variant from breaches in personal health information to the larger size of attacks which can potentially halt hospital operations that cause damage in various kinds. It may cost the hospital more than a million, or even cost individuals' life because of the incident for the worst.

There are many kinds of incidents targeting the healthcare industry. In recent years, one of the major incidents found throughout the industry is a hospital data breach. Data breaches often appeared in the form that hospital data got compromised by hackers. The compromised data can be valuable in the criminal world as it can be used for various kinds of criminal activities like identity theft, blackmailing, or social engineering

because the data may include patients' personal information and their health condition. This kind of incident can potentially cost hospitals 'a trust' issues from their patients, as individuals' medical conditions and privacy are being exploited. Also, there is the case that not just gain unauthorized access to patient's private data but, take over the data or even wipe all important data out of existence. 'Ransomware' and 'Wipeware' are the main cause of these threats. Ransomware takes over ownership over data away from the hospital system and encrypts all the data which often takes an important role in hospital operation. At the same time, Wipeware will delete all the data from the victim machine. This mostly causes great disruption in hospital operation as consequence. Incidents that showed up in recent years seem to target healthcare organization more frequently, as the industry still have poor cybersecurity practices [16]. Many incidents [16–18] showed that social engineering launched on healthcare employees is on rising. The threat has the potential to seamlessly blend into hospital workflow and made it hard to be noticed. However, follow these incidents, many stakeholders in the healthcare domain start to implement cyber-security to their organization infrastructure.

At the foundation, each organization must start with educating their employees on cyber-security awareness to reduce the risk of cyber-incident that may cause by human error or human vulnerabilities. Next, define organization policy and management plan that help prepare against cyber-incident. When employees and management level of organization have prepared cyber-security, the organization will focus on cybersecurity of the technology layer. There is various kind of tools and technology that was invented to mitigate cyber-incidents. Some may have been made to prevent exploitation of existing technology while some may have been made to directly deal with known and upcoming threats.

One of many concepts invented to mitigate these threats is the decentralization of data. The concept of decentralization was made to mitigate most incidents and threats that involve single-point of failure vulnerability. In the case of the healthcare industry where the loss of patient data can cause many major damages to the affected organization and their patient, decentralization of data can help reduce damage caused by the case. There is more than one benefit that healthcare document data can gain from decentralization. Decentralization allows patient data that scattered across healthcare domains in different organizations to link to each other. As healthcare document data

can be scattered across the different organizations within the healthcare industry, it also increases the chance that its copies can survive cyber-incidents. Even in case, that document in one organization got compromised, there is a chance that copies of compromised data also exist in other organizations. The survived copies can make a substitute for the original that got compromised. However, this only possible if there is a point that lets every organization in the network known which document exists in which organization. This is where the concept of the IHE Cross-Enterprise Document Sharing Profile fits in. Combined with Blockchain technology that makes the Document Registry entry persist and immutable, this ensures that every organization in the network will always know the whereabouts of document they need within the network while the entry itself cannot be tempered or deleted by any actor with ill intention.

This work will introduce another way to allow health document sharing between healthcare organizations with increased protection against cyber-threats, by using a combination of Blockchain and IHE Cross-Enterprise Document Sharing (XDS.b) Profile.

1.2 Problem Statement

There is no reliable software platform that supports securely and confidentially sharing healthcare documents between healthcare systems and organizations. The platform must allow sharing of healthcare documents between different healthcare organizations while still maintaining the confidentiality of data and also help mitigate emerging cyber-threats in the healthcare domain that tend to tamper with integrity and availability of data. The platform must act as the health document exchange medium that has distributed, decentralized, persistent, confidential, and immutable availability characteristics.

1.3 Objective

1.3.1 Design and implement Document Registry Blockchain shall follow the requirement for document registry defined in the XDS.b integration profile from IHE.

1.3.2 Design and implement Blockchain smart contract shall provide the main function to Document Registry Blockchain as healthcare document registry which comprises of health document registering function and health document search

function.

1.3.3 Design and implement Blockchain smart contract shall have additional function to record healthcare document exchange between participate node.

1.3.4 Deploy and evaluate the functionality of Document Registry on Blockchain.

1.4 Scope of Project

1.4.1 Design and implementation of Document Registry Blockchain shall follow requirements defined in XDS.b integration profile from IHE.

1.4.2 Design and implementation of Blockchain smart contract within Document Registry Blockchain that gives the main function as healthcare document registry and additional function as healthcare document exchange history record.

CHAPTER II
LITERATURE REVIEWS

2.1 Integrating the Healthcare Enterprise (IHE)

Integrating Healthcare Enterprise initiative (IHE) is an initiative founded by RSNA and HIMSS that works closely with the healthcare industry to improve the integration, exchange, and sharing of information technology systems in the healthcare sector. IHE supports the use of standard practices like HL7 or DICOM to satisfy special demands or clinical needs in order to provide the best possible patient care. A system created in accordance with IHE will be able to interact more efficiently with one another, will be easier to set up, and will allow health care practitioners to use health data more efficiently. This helps enables accessibility to the information which enables usability whenever and wherever it is needed. The initiative is responsible for providing services for interoperability, specifications, and tools. They also collaborate with industries, health organizations, physicians, and users to design, test, and deploy solutions that satisfy important health information demands while adhering to defined protocols [19].

2.1.1 IHE Process

Figure 2-1 shows the IHE Process where the initiative gathers developers and users of healthcare information technology in a recurring four-step process. Technical and clinical experts develop critical usage scenarios for health information exchange in this phase, and then generate specifications for system communication to solve the use scenarios. IHE participants also select and optimize established standards during this step. After that, the industry implements the defined specifications which would be called “IHE Profile” into their healthcare information technology system. The initiative then tests implemented systems to ensure that the resulting implementation of IHE Profiles will provide benefit for the implementer and make their works compatible with others in the healthcare industry. The initiative committees follow the process as shown in Figure 2-1 to address interoperability in various healthcare service areas including Information Technology Infrastructure.

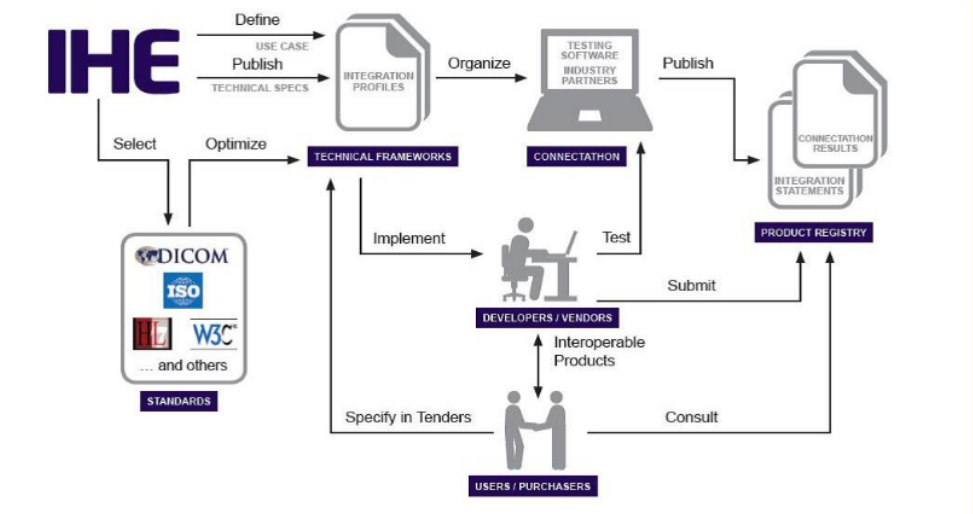


Figure 2-1 IHE Process to create guideline for implementation of health information technology [20]

2.1.2 IHE Integration Profiles

IHE Integration Profiles or IHE Profile are products of the IHE Process which provide a framework based on standards for sharing health information within the sites for healthcare services and across the networks. The framework address interoperability issues related to information access for the healthcare given to providers and patients in an area of clinical workflow, administration, information infrastructure, and security, etc. IHE Profile was purposely designed to offer a clear implementation path for IT developers to develop and implement IT systems for a healthcare organization that meets the needs and is compatible with the environment of the healthcare industry while also aiding them in dealing with various kinds of communication standards existing within the healthcare IT domain. These profiles arrange integrated functions through the coordinated execution of defined health information sharing standards (including HL7 W3C, DICOM, and safety standards). IHE Profiles also provide definitions for implementing standards to meet the clinical needs and demands. [21] By referencing the associated standards, each profile outlines the participants (actors), transactions, and availability of information needed for implementing to address the scenario. IHE Profiles that have undergone IHE Process, tested, and deployment in real-world settings and have reached the approved status, will be published in documents called "IHE

Technical Frameworks" (IHE TF). Each IHE clinical domain has one corresponding framework, with each framework may be comprised of multiple volumes. The Technical Frameworks provide detailed explanations for each IHE Profile specified by their interoperability issues and dependencies among the Integration Profile.

2.1.3 IHE Information Technology Infrastructure Technical Framework

The IHE IT Infrastructure Technical Framework (ITI TF) defines the implementation of existing standards to establish medical information sharing that supports the best patient care. After the public review phase, the framework is expanded every year, and it is maintained regularly by detection and corrections through errors. IHE ITI TF identifies a subset of the functional components of healthcare companies, called "IHE Participants", and uses "transactions" to illustrate their interactions based on a set of standards-based coordination. The body of transactions was described within the framework in detail. The IHE ITI TF is divided into four volumes. The first volume describes the concept detail of IHE ITI Integration Profiles. The second volume is divided into four sub-volumes; a, b, c, and x which describe the concept detail of all transactions present in the framework. The third volume provides a further explanation of the specifications of cross-transaction and content used in Document Sharing Profiles. The fourth volume provides additional national extensions related to the framework.

2.1.4 Cross-Enterprise Document Sharing Set-b (XDS.b) Profile

The configuration file XDS.b (Cross-enterprise document sharing set-b) makes it easier to register, distribute, and access patient EHR (Electronic Health Records) across the healthcare corporate network [22]. The profile aims at providing a description for handling the exchange of health documents among any healthcare organization, extending from doctors' hospital rooms to acute treatment in psychiatric wards, in compliance with standards. XDS is a generic term to reference all XDS profiles which are Cross-Enterprise Document Sharing Profiles [23]. In IHE ITI TF Vol.1 declared that the term XDS within the IHE ITI TF currently refers only to XDS.b. The main goal of the XDS.b profile is to allow XDS Affinity Domain members to share health documents via XDS Document Registry. That means, its process mainly about makes metadata of documents within the XDS Document Repository available on XDS Document Registry entry. This allows any XDS Document Consumer to visit XDS Document Registry and

seek the document they need before retrieve it from the XDS Document Repository that the document belongs to.

The process overview of Cross-Enterprise Document Sharing (XDS.b) profile is described in Figure 2-2. The figure also showed sequence of process along with involving XDS actors and XDS transaction format. At the beginning, each health document will be created from its sources along with its metadata attributes. These sources will be called 'XDS Document Source actor' which can be any machine involved in healthcare service, for example; CT scanner, laptop in each physician office, or central computer in medical lab. Next, these created documents along with its metadata will be sent to data storage which act as document repository. These repositories will be called 'XDS Document Repository actor' which usually be some kind of computer or server that was assigned to keep medical document available for use. According to XDS.b profile, XDS Document Source will send document metadata in the format of Provide and Register Document Set-b (ITI-41) format. In some case, XDS Document Source and XDS Document Repository may integrated together. This is called 'XDS Integrated Document Source Repository actor'. The XDS Integrated Document Source Repository functions the same way as XDS Document Source and XDS Document Repository will do but combined together.

The XDS Document Repository will index and make the document available for use after receiving the document and associated metadata. At the same time, XDS Document Repository registers metadata along with identifier and locator of the repository itself to local document registry. The message transaction in this process will follow format of Register Document Set-b (ITI-42) transaction. "XDS document registry participant" will be the name of the document registry. XDS Document Registry is software or machine that keep all document metadata and its corresponding repository from all connected repositories available for discovery. Commonly, XDS Document Registry should be database that keep document metadata from all connected repositories available for discovery through database query. However, there are no restriction from XDS.b profile for method to keep these data and how to discover each document metadata using specified document metadata attributes. There are just requirement that require XDS Document Registry to be able to accept value of specified

document attributes from XDS Document Consumer and return the matched document to the consumer.

In XDS.b profile, 'XDS Document Consumer actor' can be any kind of software or machine that allow user like healthcare employees to access health document or medical document they need. There are no restriction in XDS.b profile that specified XDS Document Consumer actor to be different software or machine from other actors. XDS Document Consumer actor will just require user to specify value of known document metadata attributes which will allow XDS Document Repository to search for matching document metadata in its database. After received document attributes value from its user, XDS Document Consumer actor will send the specified attributes to XDS Document Registry. This message transaction will follow format of Registry Stored Query (ITI-18). Then, XDS Document Registry process received attributes by search for matching document metadata and return full document metadata which it found to XDS Document Consumer. XDS Document Consumer actor show founded result to its user. The user pick the right document they need and issue to XDS Document Repository corresponding to the document for document retrieval via XDS Document Consumer actor. XDS Document Consumer will send document retrieval request transaction in the format of Retrieve Document Set-b (ITI-43). After XDS Document Repository received health document retrieval request from XDS Document Consumer, the repository will seek for the specified document and return the document to XDS Document Consumer. XDS Document Consumer actor will make the retrieved document available for user to use.

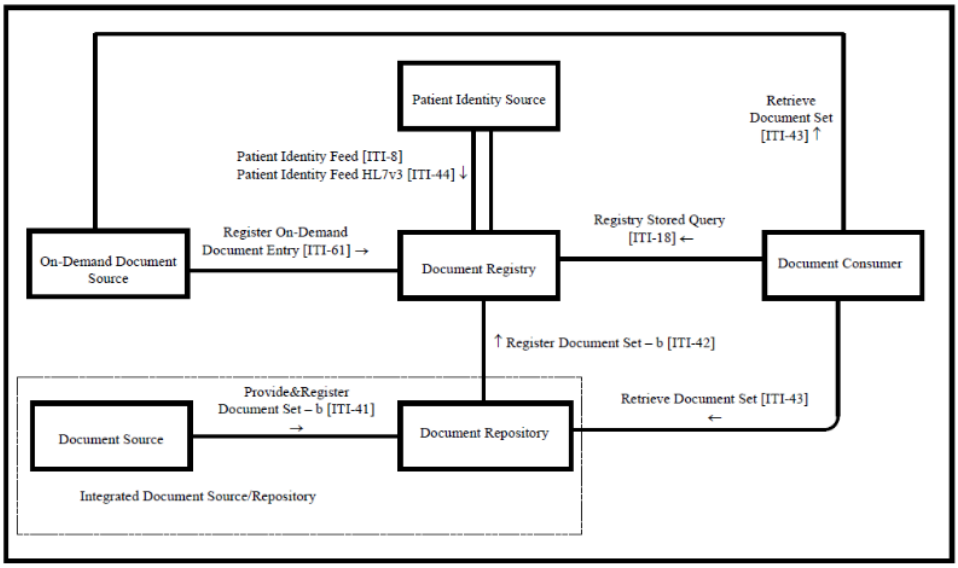


Figure 2-2 Cross-Enterprise Document Sharing - b Diagram [22]

2.1.5 XDS Transaction Format Types

In XDS.b profile, all messaging transaction will be in the form of XML format with schema depend on each types of transaction. Types of XDS transaction format vary upon involving actors and its purpose.

2.1.5.1 Provide and Register Document Set – b (ITI-41)

Provide and Register Doccument Set – b (ITI-41) transaction format defines XML schema for message that sends metadata of document from XDS Document Source actor to XDS Document Repository actor for store into document repository. This type of transaction mainly requires XDS Document Source to include all available metadata attributes of created document for other XDS actor. XDS Document Repository actor would need to acknowledge to XDS Document Source if it successfully received document and its metadata.

2.1.5.2 Register Document Set – b (ITI-42)

Registering a document set-b (ITI-42) defines an XML schema for a message that sends the metadata of the documents available in the repository from the XDS Document Repository participant to the XDS Document Registry participant to register the document to the Document Registry entry. The main goal of this form of transaction

is to send document metadata from the repository to the XDS Document Registry actor, along with repository properties. XDS Document Registry actor will need to respond back to XDS Document Repository actor when received the transaction and register it to document registry entry.

2.1.5.3 Registry Stored Query (ITI-18)

Register Stored Query (ITI-18) is general XML schema format that used by one actor to query for data from another actor in entire IHE IT Infrastructure Framework. The transaction will be used by the XDS Document Consumer actor to request document metadata from the XDS Document Registry actor in this operation. Any document metadata attributes known by XDS Document Consumer will be included in the transaction. XDS Document Registry will use specified metadata attributes to search for matching document metadata inside document registry entry. XDS Document Registry will need to respond to XDS Document Consumer actor that it received the request. XDS Document Registry also needs to return search result to XDS Document Consumer.

2.1.5.4 Retrieve Document Set (ITI-43)

Retrieve Document Set (ITI-43) define XML schema for XDS Document Consumer to request document retrieval from XDS Document Repository. Different to other transactions involved in XDS.b profile, Retrieve Document Set transaction only contain few essential attributes to allow retrieval of document from document repository. XDS Document Repository will need to acknowledge to XDS Document Consumer when received the transaction before return the requested document.

2.1.6 Transaction Object Type and Metadata Attributes

In each transaction, there are set of metadata attributes that represent the document. These metadata attributes are categorized to three sections. SubmissionSet (Table A-1) represent information associated with submission of document since it was created by the source. Folder (Table A-2) represent group that the document belongs to. DocumentEntry (Table A-3) represent the document itself. The table containing all metadata attributes can be further inspected in the Appendix Section A.

2.2 Blockchain Technology

2.2.1 Definition of Blockchain

The blockchain is a list of records, called "blocks", which are linked to each other in an encrypted manner and have security features [24]. Blockchain technology allows data to be stored and exchanged based on peer-to-peer levels. Blockchain data can be structurally shared, consulted, and secured relied on consensus-based algorithms (generally called "Consensus") [25]. It is a succession of blocks in the blockchain ledger that include a list of transaction records, similar to a standard public ledger. [26]. Participants in a Blockchain network have records of all blocks and transactions with these records stored locally on the computers that belong to each participant in the Blockchain network. Any kind of change made to the Blockchain requires consensus between the participants of the network. The idea of blockchain was later combined with other technologies and computing concepts to create modern digital currencies (called "cryptocurrencies"), which protect electronic cash through the encryption mechanism components of the Blockchain, replacing the central repository or authority.

In 2009, with the launch of a cryptocurrency network called "Bitcoin", Blockchain technology became widely known to the public. This was the first widely known Blockchain network, and its purpose was to serve as a new type of digital currency, followed by multifunctional Blockchain such as Ethereum has its own application platform. For cryptocurrency Blockchains like Bitcoin, the Blockchain is used to record the transmission of digital information representing electronic cash that occurs in a distributed system. These transactions act as digital evidence which allows Bitcoin users to transfer their rights over each unit of Bitcoin currencies to others, the distributed nature of the Blockchain ledger makes them available for public verification. Blockchain was designed to defy the concept of having a single centralized system as the host of the network which subsequently allows the concept of decentralization to take the place by having many members of the network equally maintain the replicated distributed ledger. Combining the concepts of “Block” and “Chain” of encrypted hashing, the Blockchain will be durable for any attempt to change the information recorded in the distributed ledger. With the contribution of the Blockchain developer community, the technology can now be used in various applications and is being studied for further use in many industries [27].

2.2.2 Benefit of Blockchain

Blockchain is tamper-proof. Blockchain ledgers are implemented in a distributed manner, giving them natural data distribution. Decentralization is also a feature of the technology, which reduces any need for a central body. The Blockchain allows the Blockchain user community to keep track of transactions in a shared ledger. Any transaction that has been released cannot be modified under the regular conditions of the blockchain network [27].

2.2.3 Blockchain Characteristics

Key characteristics of the Blockchain can be vary depend on its setup and environment of usage. According to many sources, key characteristics of the Blockchain may be summarized as followed:

2.2.3.1 Decentralization

Decentralization is the foundation of Blockchain technology as response to problem of centralized system. In centralized system, especially centralized database, there is a chance that the database got compromised by hacker. Other than rely on backup data, there are very few options to deal with the incident. This makes the compromised database become single point of failure which prevent follower system to operate. Decentralization of data was proposed to scatter the chance of single database from getting compromise. This makes decentralized database network have more resistant against incident threatening centralized data. Even hit by incident that aims to compromise the data. If at least half of decentralized network survived the incident, the data survives the attack.

2.2.3.2 Immutable

With utilization of cryptographically hashed chain combined with decentralized network, the Blockchain technology ensures that any data published on Blockchain cannot be deleted or modified. If there are any modification made to content of published data, it will cause change on the hash chain and detected the network. Any action that causes change to hash chain will be negate by majority of the network. This mean if anyone want to temper with published data on Blockchain, they will need to compromise the entire network at once. Any survived node has chance to notify the abnormal to the entire network.

2.2.3.3 Transparency

As the foundation of Blockchain is to have all participant nodes have the exact same copy of Blockchain ledger, it passively gives transparency to published data. It is impossible for anyone to secretly hide something inside Blockchain without let other participants in the Blockchain network know.

2.2.3.4 Distributed

Blockchain has distributed characteristic by design. All nodes will have the same Blockchain ledger. Any contents published to Blockchain ledger are passively distributed to all Blockchain node. With consensus algorithm, it requires that the publishing content either sent to all nodes before accepting to publish or being accepted then send to all node, to complete consensus. So, Blockchain ensures that any data published to the chain are distributed to all connected nodes.

2.2.3.5 Trusted

In public network where anyone can participate or in permissioned network where participants are not completely trust each other, trust is the main factor that define usability of decentralized network. Along with Blockchain technology, consensus solve the issue about trust by eliminate the chance of any single node participate in Blockchain to have absolute control over publishing data when certain condition is met. It can rely either on randomness or specially designed algorithm depend on each consensus method. When none of any single node can have absolute control over publishing data on the Blockchain, made it extremely difficult for someone to temper with target data. Many consensus methods ensure that it will much more expensive for anyone to attempt on tempering with publishing data when compared to benefit they can get. This passively establishes trust between all participant nodes.

2.2.4 Blockchain Types

When considering the scope of participants who can participate in a specific Blockchain network, Blockchain can be classified into three types of Blockchain networks.

2.2.4.1 Public Chain

Public Blockchain is the type that allows anyone to participate in the network either participating as client/user node or miner/validator node. This type of Blockchain mostly has no specific rule, policy, or agreement for participants to enter the network. The type is suited best with the network environment where its data is not required to be kept confidential from the public.

2.2.4.2 Private Chain

Private Blockchain is the type that allows only a limited number of members to participate. This type was invented to be more compatible with the environment that participant nodes are members of a specific organization or community where the Blockchain ledger may record confidential information limited only to participants.

2.2.4.3 Permissioned Chain

Permissioned Blockchain is the type that allows only selected members of a specific community or affinity domain to participate, and it is also known as consortium Blockchain. Permissioned Chain is different from Private Chain in terms of scalability. As the private chain was limited for pre-selected members, the permission chain may further extend its member to a larger group of members via policy or agreement accepted by original participants. At the same time, the permission chain will not allow anyone to participate in the network as the Blockchain ledger may contain confidential information limited to the accepted group of participants.

2.2.5 Blockchain Components

2.2.5.1 Transaction and 'Block'

Each of individual information represent change or cause of actions in information system are stored within Blockchain as "Transaction". Several transactions being publish to Blockchain within the same time interval are put in the same "Block". To form each single block, miner or validator needs to hash transaction together. The resulting hash value represents integrity of each block. If there are any change apply to transaction in the block, it will cause hash value of the block to change. Format of block vary depends on each Blockchain platform and its use case. Some platforms may publish in a form of plaintext just to act as the source of truth for every participating node to look without constraint. Some platforms may bound transaction or block to unique address to extend variation in accessibility. Some platforms may encrypt block to

maintain confidentiality of data. Transaction and Block are the key component which determine purpose and application of Blockchain.

2.2.5.2 Cryptographically hashed 'Chain'

Other than the concept of "Block", The Blockchain concept also introduced the concept of "Chain. As integrity of each Block represent by its hash value, integrity of entire Blockchain represent by all hash value of all Block within "Chain". The foundation of "Chain" concept is by chaining hash value of all blocks together. This can be done by include hash value of block formed in previous time interval into the current block to generate its hash value. Any changes made to any one single block will alter hash value of the entire chain that come after. This makes it harder to alter data that published within Blockchain. It requires anyone who want to alter the data to apply change to all blocks that come after the target block until the current one to make the change valid. Combined with decentralization characteristic of Blockchain network, this makes data exist in Blockchain nearly impossible to alter.

2.2.5.3 Distributed network of participate 'Node'

Any machine that participates in the Blockchain network is called "Node". Nodes represent the population of each Blockchain network. Each node keeps the same copy of data in Blockchain which is the Blockchain ledger and always tends to in sync with each other via Blockchain-specific protocol depend on each platform. In the case of Ethereum, the platform utilize devp2p protocol which allows each Ethereum node to connect with others at a peer-to-peer level [28]. If there are any differences in data between nodes, the version of data is being held by a minority of participating nodes. They will be clarified as false before rejected by the entire network and replaced by the right version accepted by the network. In each Blockchain network, some nodes may participate as "miners" or be elected as "validators" of the network at each different time interval. Miner/Validator nodes have a duty to perform the task which maintaining the consensus of the network. The Blockchain can be alive only if they are at least one participating node to maintain the Blockchain, while the strength of its characteristics depends on the number of participating nodes. In most cases, more participating nodes mean stronger Blockchain.

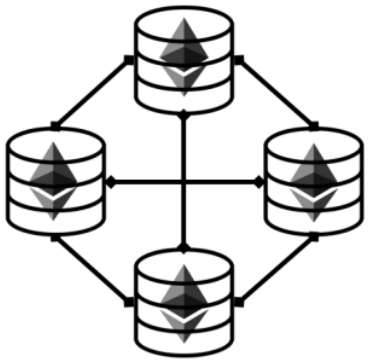


Figure 2-3 Blockchain network formed from the participation of Blockchain nodes

2.2.5.4 Consensus

Each Blockchain network must have its own consensus within the network. Specifically, it is not just the consensus between the member but the built-in consensus mechanism within the communication protocol connects each member in the network. The consensus mechanism is critical for maintaining the integrity of the Blockchain ledger because it ensures that everyone in the network has the same replica of the Blockchain ledger so no one has complete control over data added to the Blockchain ledger or the ability to choose which version of the Blockchain ledger the network should maintain. There are many variations of consensus invented since the beginning of Blockchain technology. Each has its own method of investing resources to achieve complete consensus. The existing consensus concept at the time can be divided into three major types based on the participation of Blockchain members who declared to act as maintainers of the Blockchain ledger.

2.2.5.4.1 Competition-based consensus

The type was the first to introduced along with the beginning of Bitcoin Blockchain which called "Proof of Work" (PoW). Competition-based consensus, such as PoW, forces Blockchain maintainer nodes (also known as "miners") to invest computational resources in competing with other miners to solve a specific mathematical puzzle that can only be solved with computational power. The first miner who can solve the puzzle will have the right to add the adding Block to the Blockchain ledger and receive a reward declared by the network which would compensate the

investment. In the case of PoW, a miner must generate a valid "nonce" number at random, which, when hashed with the hash value of the added Block, yields a hash value beginning with the digit "0" as required by the network (such as 0x0000000abcd). This kind of puzzle ensures that the chance where malicious actors want to attempt malicious activity on the specific transactions on certain Blocks is at least possible. Combined with the increase in the number of miners entering the competition given that chance becomes nearly impossible to achieve. This consecutively ensures the integrity of the Blockchain ledger and gives transparency to Blockchain. The scenario of competition-based consensus suited best to Public Blockchain where anyone can participate in Blockchain. The more miner entering the competition means the more reliability and transparency for the Blockchain. Additionally, the reward-based nature of the concept can even further synergize the Blockchain to have more miners participate in the network. However, due to the competition will have major of computational power invested in achieving consensus, that means the environment where computational power is limited and precious to its member will not be compatible with this type of consensus.

2.2.5.4.2 Randomness-based consensus

The type was originally introduced as an alternative to a competition-based consensus like PoW and to address the problem where major computational power will be wasted in achieving consensus. The concept proposes utilizing randomness to aid in the selection of the Blockchain maintainer at a certain time. Widely known consensus mechanisms that can be categorized to this type are "Proof of Stake" (PoS) and "Proof of Authority" (PoA) which using pseudo-random algorithm combined with additional factor to determine for the node that has the right to add Block to the Blockchain ledger at a certain time (which would be called "validator). Both PoS and PoA require validator candidate to place "the bet" on "the stake" which would mostly be cryptocurrency circulating in the network. The person who has placed a larger bet on a stake will have a better probability of being chosen as the time validator. However, there is still a chance that the one with a lower bet can be selected as the validator instead. The one that has been selected as the validator will gain all the bet placed on the stake. That means each candidate needs to take an equal risk to gain and loss their available bet. This consecutively results as distributed right amongst the network similar sense to what achieved in competition-based consensus.

As this type of consensus act as an alternative to competition-based consensus, that means the environment best suited with this type of consensus is where its member cannot effort to lose computational power in competition-based consensus altogether.

2.2.5.4.3 Majority-based consensus

This type of consensus was also introduced as another alternative to a competition-based consensus like PoW. The original concept of this consensus type was originated from the "Practical-Byzantine False Tolerance" (PBFT) method. The method was invented for the traditional logic systems to determine for decision the system would take in the assumption that the majority of its members are on the "good side" and will take responsibility to help the system achieve the best decision. Implement to Blockchain, the consensus mechanism requires all participate node to act in a similar fashion to the validator. The network will only accept the adding Block to the Blockchain ledger when the adding version is the similar version in a majority of the network. This means it requires the adding Block to be the version that 2/3 of all member nodes propose to add to the Blockchain ledger. This eliminates the chance where malicious actor which assumed to be the minority of the network to attempt malicious activity on the adding Block. However, due to the majority-based nature of the type, it is only compatible with Blockchain types with only known members including Private and Permissioned Blockchain.

2.3 Ethereum and Smart-Contract

Ethereum is one of the well-known open-source Blockchain platforms. The platform initially invented by the developer named Vitalik Buterin and further develop and maintain by the Ethereum community [29]. The main approach of Ethereum Blockchain is about the use of Blockchain technology for applications other than cryptocurrency. The platform was the first to propose the concept of a ‘smart contract’ that enables programming over Blockchain technology.

2.3.1 Smart Contract

Ethereum was the first to propose the concept of a smart contract [29, 30]. Now the word 'smart contract' become common word to describe feature that allows developer to design the content that publishes to Blockchain and its computational behavior. In Ethereum, smart contract code is written with Solidity programming language. Smart contracts define what behavior the contract will do when open/view by user. Smart contracts rely on Ethereum Virtual-Machine (EVM) which allow host machine of Ethereum client to be able to execute smart contract Solidity code. EVM was designed to allow portability of Ethereum platform and always packed with Ethereum client. Now there are many interface tools developed by Ethereum community that allow Ethereum client to work with major programming languages. This further extend usage of smart contract to infinite possibilities.

Solidity is Javascript-like programming language that is specifically designed to use with Ethereum smart contract [30]. The main purpose of the programming language is to act as the middle between human-understandable language and computer language. It reduces difficulty for developer to design behavior of their smart contract on Ethereum Blockchain. The language is update and maintain by Ethereum community.

2.3.2 Quorum [31, 32]

Quorum (or later renamed as GoQuorum) is an Ethereum-based distributed ledger protocol forked of go-ethereum enabled for transaction/contract privacy and a wider range of majority-based consensus mechanisms compatibility. The platform was initially developed by "JPMorgan" and further developed and maintained by the Quorum community. Quorum enables the usage of PBFT by inventing a PBFT-inspired consensus algorithm called Istanbul BFT which was adapted to be compatible with the Ethereum Blockchain environment. The platform also offers the "7-Nodes Example" [33] for developers to invent and test their Blockchain concept which is useful for the development with the limited computational resources available.

2.4 Related Work

There are many research proposing about decentralize healthcare information with Blockchain technology. The goal of decentralization and implementation of each work have many variants. These are several works that proposed interesting idea and concept about implement healthcare informatics system based on Blockchain technology.

2.4.1 A Blockchain-Based Approach to Health Information Exchange Networks [34]

The study advocated that Blockchain be used as a single center for health data exchange. The main goal of this Blockchain concept is to connect all bread and crumb of patient health information together by allow participate node to discover health information data they seek and its location within Blockchain ledger, increase interoperability in health information exchange. Their main contribution is the concept that suggests the use of FHIR health information exchange standard combine with Blockchain technology. Each transaction on Blockchain will contain FHIR locator of actual data along with its index which make each transaction available for search. Due to the limit of health information that it requires certain amount of confidentiality, this makes it not really compatible with platform open to public like Blockchain. Store actual data somewhere else outside Blockchain and put its locator into Blockchain for use. With known secure index, this Blockchain helps connect patient information that is scattered across healthcare industry together. The work also gave suggestion about how health information Blockchain should look like and what it should have by common. There also other major contributions that use secure index for searching on encrypted data and 'Proof of Interoperability'. This work suggests that if health information is kept within Blockchain in encrypted form, it should also contain secure index which will allow data search even the data is encrypted. This should reduce the difficulty of implementing health information with Blockchain. And other major concept proposed in this work is 'Proof of Interoperability'. Based on Proof of Work consensus, the work suggest that computational resource should not be wasted unnecessarily. Instead of putting computational resource to competition for consensus, it should be used to verify interoperability of participate health data instead. However, they did not propose about how the consensus should work in detail. This work gave a good example of how

Blockchain can have potential to solve issue that common in healthcare industry like interoperability. Additionally, they also proposed many concepts that can be a good foundation for using Blockchain technology with health information.

2.4.2 A Case Study for Blockchain in Healthcare: "MedRec" Prototype for Electronic Health Records and Medical Research Data [35]

Main goal of MedRec is to provide Blockchain that acts as a middle for health information exchange while allow Blockchain participants to gain benefit from participation. They chose Ethereum as Blockchain platform for the system. Ethereum provide smart contract and address based access for the work. This work assumes that miner/validator nodes are health institution that have demand for large amount of health information data to use in their research. Miner/validator node will be rewarded with anonymized health data which can be used in research involve health data analysis. Additionally, MedRec proposed about allowing patient to have consent about usage on their data. Give more control over individual health data. The work also adopted cryptographic key scheme proposed by Zyskin et al. [36], to ensure that only authorized party can access patient health information published on Blockchain. Additional to these main contributions, they also gave suggestions about factor that should keep continuity of Blockchain and how Blockchain element provided by platform like Ethereum can be useful. One of interesting concept is about using Ethereum address as patient identifier. Due to all identities exist on Ethereum Blockchain are assigned with unique address, these unique addresses can reduce complexity in patient identifier management if designed properly. MedRec gave a good example of concept that needed to maintain continuity of Blockchain network by allowing participant to gain benefit from participation in some way. At the same time, MedRec is another good example that using Blockchain technology to aid health information exchange issue. And the last, MedRec have shown flexibility of smart contract and how it can be useful when implement with healthcare information.

2.4.3 Blockchain-Based Data Preservation System for Medical Data [37]

This work used Blockchain to keep data that need to have confidentiality preserved. Regardless of what kind of data, this Blockchain allows user to design what data they want to keep in Blockchain. The chosen data will be encrypted before publishing into Blockchain. The goal of this Blockchain concept is to preserve medical

data inside Blockchain away from any tempering attempt while keep it secret and always available for its owner. Instead of let data available to public, this work has demonstrated how Blockchain technology can be used in different approach like keeping medical data available to only authorized entity.

2.4.4 Blockchain-Based Electronic Healthcare Record System for Healthcare 4.0 Applications [38]

The work has gathered research proposing the Blockchain concept from 2016 to 2019 that would benefit the healthcare industry by enhancing the capabilities of electronic health records. The work has well explained the overall concept of implementing Blockchain technology to electronic health records developed over the years. They also proposed another approach of implementing Blockchain technology for electronic health records by using IBM Hyperledger fabric as a medium for health information exchange in a similar fashion with MedRec which prioritize efficiency in handling huge number of transactions in the meantime. The contribution in the work inspires and encourages the idea of enhancing the existing EHR system with Blockchain technology as it provides the characteristics of distribution and decentralization.

CHAPTER III
PROPOSED METHOD

This chapter explains the proposed method. The first section introduces the use case scenario of the proposing concept. The second section explains the main concept design of this work. The third section explains how to construct Blockchain infrastructure that is compatible with the IHE XDS.b Profile's implementation. The fourth section explains how Blockchain technology is integrated into the IHE XDS.b Profile. The fifth section further explain the detail regarding the main function of our implementation. The last section shows the process flow of the XDS Document Registry Actor as an interface between the XDS system and the Blockchain ledger.

3.1 Use Case Scenario

Imagine the example case, Mr. Peter Parker is a normal person. One day he got bit by a strange spider. Mr. Parker then visits Hospital A located in Bangkok for diagnosis. A doctor at Hospital A diagnosed that there is no unusual effect on Mr. Parker and heal the bite wound for him. The event and diagnosis result were recorded within Hospital A. Later, during his travel to Chiang Mai, Mr. Parker found an unusual effect from the bite. He then pays a visit to Hospital B for diagnosis. Without the health information sharing standard implemented, A doctor at Hospital B will have difficulty accessing the last diagnosis result for Mr. Parker, which is unknown to them that it can be located in Hospital A. The situation becomes even worse when the health information system of Hospital B was hit by ransomware and went down. That means a doctor at Hospital B cannot know that what happened to Mr. Parker. On the other hand, with IHE XDS Blockchain implemented, even Hospital B hit by ransomware, a doctor at Hospital B can use an alternate machine to access the XDS Blockchain ledger and discover that there is the latest diagnosis result for Mr. Parker available at Hospital A. A doctor then gains access to the diagnosis result and able to identify that Mr. Parker was afflicted by unknown genetic mutated symptom caused by the spider bite. Allow the doctor to prevent a further harmful consequence.

3.2 Concept Design

As introduced in Chapter I, the unique nature of the healthcare environment that emphasizes confidentiality of data gave limits to implementing Blockchain technology into the industry. Patient data cannot be put directly into Blockchain as it will become persistent following Blockchain characteristics while increasing difficulty in ensuring data confidentiality when its replica is distributed all over the network [26, 27, 39]. So instead of risk confidentiality of healthcare data by publishing it directly into the Blockchain network, we propose using IHE XDS.b Document Registry Actor to act as a health document exchange medium for the Blockchain network. The profile is best compatible with Blockchain technology as decentralization will secure the availability of the health information exchange by eliminating the need for the organization that will act as the central hub for the exchange avoiding a single point of failure problem. At the same time, there is no longer a need to publish health documents directly into the distributed network, reduce the risk against the confidentiality of the data. Additionally, in this work, we further extend the usability of the profile by allowing the organization that has shared health documents from its source to also act as an additional data backup for the original by providing additional access points (URLs) for the document. That mean, even the source of health document become unavailable due to unpredictable circumstance like a cyber-incident (i.e., Ransomware threat), the network will still have a chance to access the compromised document via an alternative source available from a shared peer. This extends the benefit of the health document sharing network and encourages the growth of the health document sharing community even further indefinitely.

Following Figure 2-2, the XDS Document Registry actor who acts as a hub for health document exchange would normally host a database that allows XDS Document Consumer Actor to query for information of health Document they seek. The existing solution for the database is the utilization of SQL or non-SQL centralized database depend on each XDS Affinity network. In adaptation for this work, we propose replacing these centralized databases with a Blockchain ledger which innately distribute the registry amongst the network while also benefit from Blockchain characteristics. This consecutively transforms the XDS Document Registry Actor host by each XDS Affinity domain member into a Blockchain node. Each node will now serve as a

decentralized XDS Document Registry Actor who will joint cooperatively keep, operate, and maintain the Blockchain ledger which now contains the entire health document registry entry for the network.

3.3 Blockchain Design

Considering the environment of the XDS Affinity domain network, the network is comprised of members who are hospital or health institution that entered the network intending to share their health document and access health document shared from other. Each member entering the network are expected to have been selected by the network and have an agreement with the network to voluntarily share health document from their XDS Document Repository Actor to the network while not exposing information shared within the network to the outside. Following Blockchain Types mentioned in 2.3.2, the network is not suitable with the Public Chain as they are not accepting anyone into the network without pre-selection and a proper agreement. At the same time, they are also not suitable with the Private Chain as the XDS Affinity domain network was designed for scalability and not fixed to any specific organization. That means the XDS Affinity domain network is best suitable with the Permissioned Chain type as it was designed for scalability welcoming more members to join the network over time under the condition that joining members are accepted by the network and have a proper data sharing agreement.

Meanwhile, following the Blockchain consensus mentioned in 2.3.3.4, the XDS Affinity domain members were not pre-determined to invest a high amount of computational resources for entering the network and were not expected to gain direct profit from participation in maintaining the network. That means the consensus mechanism with computational resources inefficiencies like competition-based and profit-dependence mechanism like randomness-based are not the choice of consensus for the network. Then the remaining majority-based consensus will be the most suitable choice available. The start point of majority-based consensus will be PBFT as the basis of the type. With the nature of the XDS Affinity domain network which pre-determined the joining member, this should prevent the chance for the bad actor to control the majority and secure Blockchain characteristics for the network consecutively.

3.4 Integrating Blockchain with XDS.b Profile

For the implementation, following Figure 2-2, we assume that the ITI-61 transaction is unessential for the current usage of the work. The patient identification was assumed to be already standardized amongst all members of the network, eliminates the need for the ITI-44 transaction from consideration. This left the XDS Document Registry Actor Blockchain to only interact with the remaining XDS Document Repository Actor via ITI-42 transaction and XDS Document Consumer Actor via ITI-18 transaction (as shown in Figure 3-1).

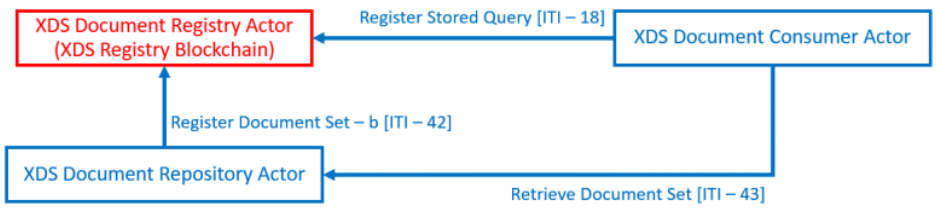


Figure 3-1 XDS Profile within the scope of interest for this work

Following Section 2.1.4, all Blockchain nodes will receive ITI-42 transactions from their local XDS Document Repository Actor as normal XDS Document Registry Actor would do, before transitioning the transaction into Blockchain Smartcontract and publish into the Blockchain ledger. Likewise, the health document query via ITI-18 transactions from local XDS Document Consumer will be interpreted and interact with Smartcontract consecutively. Note that the XDS Document Consumer Actor will still be required to directly issue ITI-43 transaction to the XDS Document Repository Actor hosting the health document to retrieve it. The Smartcontract will act as a medium for each node to perform the task to add data, read data, and search for data within the Blockchain ledger (as shown in Figure 3-2) allow the Blockchain technology to effectively integrated into the XDS system.

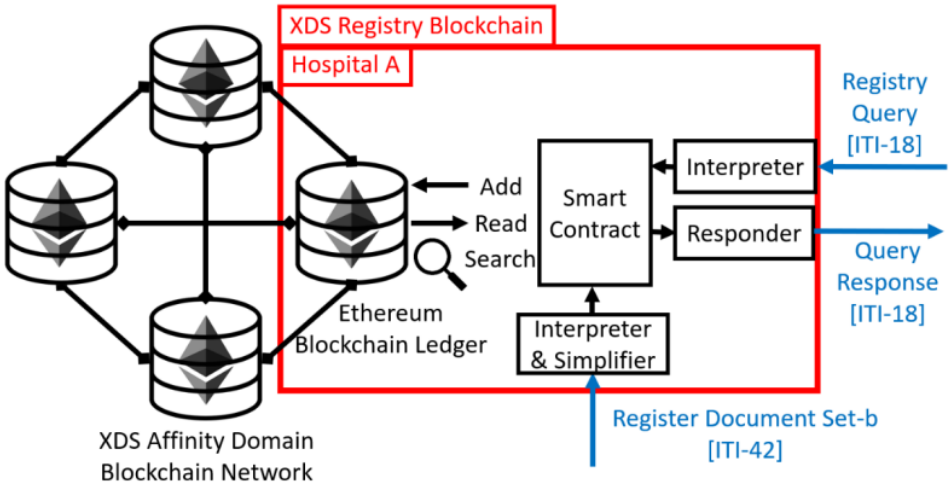


Figure 3-2 Integrating Blockchain into XDS.b Profile

3.5 Design Functions

3.5.1 Document Register

This is where XDS Document Registry Actor registers health document metadata set within ITI-42 transaction received from XDS Document Repository into the Blockchain ledger. This function enables sharing of health documents to the XDS Affinity Domain Blockchain network as well as allowing the shared document to be registered as an alternative source. There are 2 Smartcontract functions related.

The first function is the document registering function, the function act as part of the XDS Document Registry Actor to store the value of health document metadata interprets from ITI-42 transaction into the Blockchain ledger. This function act in a fashion similar to a programming variable where certain values were assigned to a specific variable for usage within the program, a whole set of metadata value is assigned into single Smartcontract function transaction. The function also automatically assigns each set of metadata with an identification number to be used for common understanding amongst the Smartcontract to differentiate each set of metadata belong to each health document. The identification number also essential for search operation which would be further explained later.

The second function is the checker function which will check for the last identification number assigned to the published set of metadata. This function allows

Smartcontract to keep track of the identification number previously used and prevent duplication. Each time a new set of metadata entering the Blockchain ledger, the identification number which would be assigned to the metadata set will be additively increased by 1 from the previous.

3.5.2 Document Search

The search operation allows members of XDS Affinity Domain Blockchain to discover health documents existing in the network by searching for registered metadata set belong to the document within the Blockchain ledger and gain access to actual documents using access information provided in the metadata. There are 2 Smartcontract functions related.

The first function is the read function where Smartcontract allows the XDS Document Registry Actor to read the value of metadata stored within the Blockchain ledger. This function only needs identification number input to return metadata value to the XDS Document Registry Actor program.

During the search operation, the XDS Document Registry Actor will be the one to handle the search keyword. The Actor performs sequential searches on each set of registered metadata using the assigned identification number for iteration until the matching result was found or reached the end of the iteration. The Actor then triggers the return query result function.

The second Smartcontract function is the return query result function where the Smartcontract returns the whole set of metadata specified as the search result to the XDS Document Registry Actor. The version of the return value omitted by this function is different from the read function in terms of compatibility with the ITI-18 transaction format. The XDS Document Registry Actor then sorts the result into an ITI-18 transaction and return it to the XDS Document Consumer Actor.

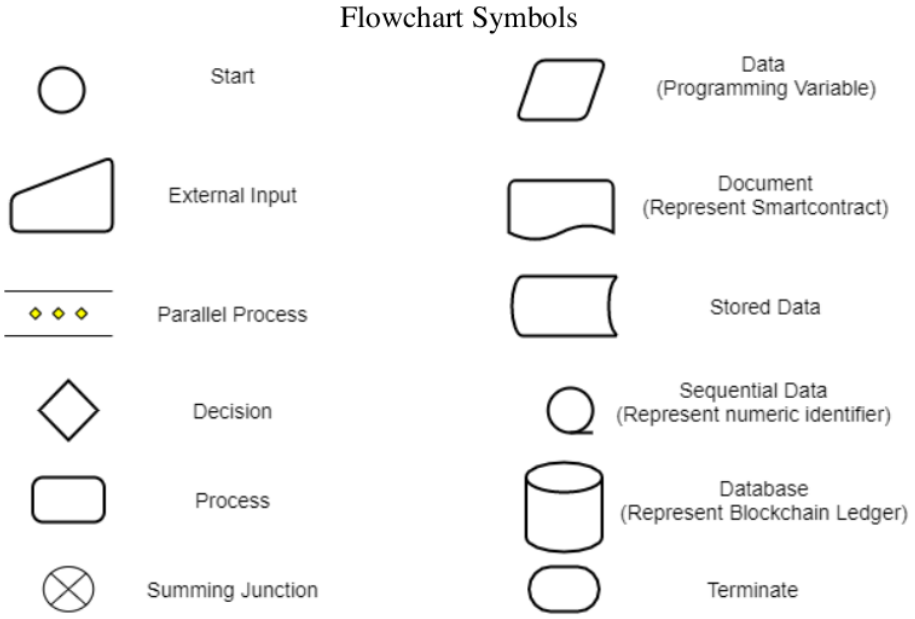
Following the normal process of IHE XDS.b Profile in Section 2.1.4, the XDS Document Consumer will then use health document access information provided within the search result metadata to gain access to the XDS Document Repository hosting the document. Then negotiate for document exchange using the ITI-43 transactions outside the Blockchain network.

As mentioned in Section 3.2, after receiving the document shared from its original source, the shared peer will also want to register the document into the XDS

Document Registry to let the network know that now they can act as an alternative source of the document for the network.

3.6 Process Flow

Figure 3-3 showing the process flow of the XDS Document Registry Actor which acts as a medium between the Blockchain ledger and XDS System.



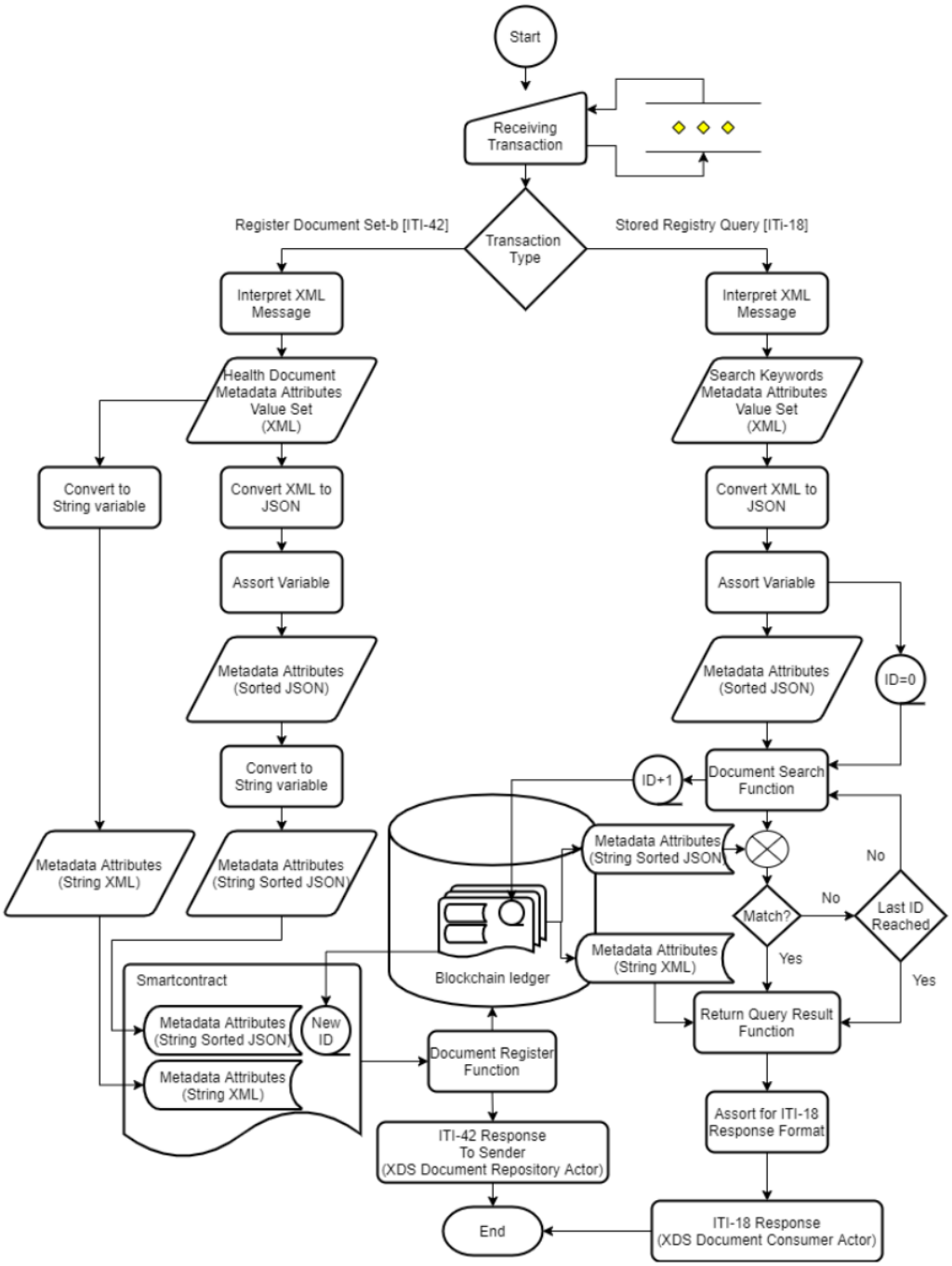


Figure 3-3 The process Flow of XDS Document Registry Actor

CHAPTER IV
IMPLEMENTATION

This chapter emphasizes the implementation step for this work. Begin with the setup method to establish a small Blockchain network for development which will describe the installation method and system specification. The second section then explains the implementation of each XDS Actor including its native programming code and corresponding Smartcontract detail. After that, the third section will explain the setup for the performance evaluation experiment and following with the experiment result in the last section.

4.1 Blockchain setup

4.1.1 Machine Specifications

For the implementation, the main machine was the best machine available for the implementation and compatible workflow which will be used for the coding process, which is a laptop Alienware 17 R5, Intel Core i9 with 32 GB installed RAM, running on Windows 10 operating system version Home Single Language (20H2) 64-bit, x64-based processor. The code then deploys on the test machine. As most Ethereum-related software is initially designed for Linux OS, so Ubuntu was selected OS for the test machine. The test machine is where all Blockchain-related environment was deployed. A test machine is a virtual machine established within the main machine using Oracle VM VirtualBox. The virtual machine running on Linux Ubuntu (64-bit) version 18.10 with 8 GB RAM and 100 GB storage dynamically shared from the host main machine.

4.1.2 Go-Ethereum

Go-Ethereum or "Geth" client is the open sources software engine requiring to operate Ethereum Blockchain within each node. The client allows the user to issue commands to the node like initiate the Blockchain, start-stop mining/validating process for the Blockchain, and activate devp2p protocol to sync Blockchain data with other nodes. The client is available at [40]. Geth can be installed as standalone or included in the installation package of the Ethereum platform variant or other kinds of service

interacting with Ethereum Blockchain (i.e., Ethereum Wallet Client). Geth's interface was initially designed to operate on Linux OS and later extend compatibility to other OS via Linux console simulated platform or work under graphic user interface of another client. The Geth client has no specific system requirement as it only is a set of Golang scripts that has no restriction to any system but, as it is mostly integrated within another client, so it requires the machine to compatible with the main client for installation. In the implementation, the test machine has Geth globally installed by installing Quorum, so the client can be launched from anywhere regardless of environment path.

Geth can be installed as standalone for ready the machine for working with Ethereum Blockchain using Linux installation from source repository command i.e., “apt-get install ethereum” in Ubuntu. The installation instruction can be located at [41]. Figure 4-1 showing installation method for Ubuntu. All available command lines for Geth can be located at [42]. The most used command is "geth attach <IPC path or Link>" which required for accessing Geth console of each active node, as shown in Figure 4-2. The console is where the user can input the command line to directly control the behavior of each Ethereum Node.

```
sudo add-apt-repository -y ppa:ethereum/ethereum
#Enable launchpad repository for Ethereum

sudo apt-get update
sudo apt-get install ethereum
#Install Ethereum using apt-get
```

Figure 4-1 Installation command-line for Go-Ethereum on Ubuntu [41]

```
[redacted]-VirtualBox:~/Documents/7NodesTry/quorum-examples/examples/7nodes/qdata
/dd1$ geth attach ipc:geth.ipc
Welcome to the Geth JavaScript console!

instance: Geth/v1.8.18-stable-a603e745(quorum-v2.4.0)/linux-amd64/go1.10.4
coinbase: 0xd8dba507e85f116b1f7e231ca8525fc9008a6966
at block: 34875 (Thu, 10 Jun 2021 13:14:32 +07)
 datadir: /home/pcep/Documents/7NodesTry/quorum-examples/examples/7nodes/qdata/d
d1
 modules: admin:1.0 debug:1.0 eth:1.0 istanbul:1.0 miner:1.0 net:1.0 personal:1.
0 rpc:1.0 txpool:1.0 web3:1.0

>
```

Figure 4-2 Geth console accessed using "geth attach" command

4.1.3 Quorum Installation

As mentioned in Section 2.3.4.2, Quorum is an Ethereum-forked that allows the Blockchain to adopt a consensus mechanism other than PoW and PoS which default to Ethereum Blockchain. Quorum's source code and installation package can be accessed at [31]. The installation method for the latest stable release is located at [43]. The platform was designed to specifically operate with a Linux-based interface and can be compatible with non-Linux OS with the aid of 3rd party software as a medium. Other than that, the platform has no specific system requirement for installation. However, from the test during the implementation, it is recommended that the machine running Quorum should have more than 6 GB of available RAM. Otherwise, there will be a performance issue that occurred during the run. In this implementation, as the test machine running on Ubuntu 18.10, there is no other 3rd party software required to operate Quorum. The Geth client was included in the Quorum installation package, which means a developer can immediately start their Quorum Blockchain development right after the installation.

Figure 4-3 showing the installation method for Quorum from its source which was used to install the latest stable Quorum released available at the time for the implementation. Some parts or functions used in the implementation may differ from the current released due to the version difference.

```
git clone https://github.com/Consensys/quorum.git
#Clone Quorum from its source repository

cd quorum
make all
#Make the cloned file as an installation

make test
#Check if the installation was successful
```

Figure 4-3 Installing Quorum directly from its source

Other than the Blockchain platform, Quorum also offers the "7-Nodes Example" for developers to locally deploy in their machine to test the functionality and performance of Smartcontract during the development. The source code can be cloned directly from its repository available in Github [33] as shown in Figure 4-4. The "7-Nodes" will simulate seven Blockchain nodes in the host machine in a similar fashion to a virtual machine using the required library called "Tessera" [44] and "Constellation" [45] included in the package as a running engine. The source code can be initiated, activated, and simultaneously controlled using the control script provided within the example. Each node can be accessed using the Geth client. For the implementation, the control script provided within the example was further modified to be compatible with our usage. The Blockchain initiation script was modified to be able to re-initiate the entire Blockchain by deleting the existing chain and replace with the empty one. This allows reset of published Blockchain during the development. The transaction publishing script was modified to run other specific code developed for XDS Document Registry Actor and its Smartcontract, allow testing and running of XDS Document registry-related code on the 7-Nodes. It must be noted that transaction-related scripts only operate on specified single nodes amongst the seven, not the entire set of the seven nodes. That means the activity of each node is independent.

```
git clone https://github.com/jpmorganchase/quorum-examples.git
#Clone the 7-Nodes Example from its source repository
```

Figure 4-4 Cloning "7-Nodes" Quorum example from its repository available on Github

All initial configuration instructions are available at [46]. Primarily, it is required to configure the genesis Block for the Blockchain ledger and issue the initiation command using the script provided in the example as shown in Figure 4-5. Each script is specific for each consensus mechanism.

```
cd path/to/7nodes
#Navigate to the folder path where the cloned repository locate

./istanbul-init.sh
#Initiate the genesis block with IBFT as its consensus

./istanbul-start.sh
#Activate all seven nodes to start the Blockchain based on the genesis block
```

Figure 4-5 Initial configuration method for 7-Nodes example

In this implementation, we only use the script "istanbul-init.sh" (execute using Linux Bash syntax as shown in Figure 4-6) to initiate the genesis Block for the Blockchain ledger as we going to use IBFT as its consensus mechanism. The script will generate "istanbul-genesis.json" file as a configuration script for the genesis block which its content should be configured as shown in Figure 4-7. Then the activation of the IBFT Blockchain can be done using "istanbul-start.sh" script as shown in Figure 4-8 which will start the activation of all seven nodes and bring the 7-Nodes Blockchain network to become alive as shown in Figure 4-9 and Figure 4-10. It must be noted that this activation process may take several minutes.

```
./istanbul-init.sh
```

Figure 4-6 Executing "istanbul-init.sh" with Linux Bash syntax

```
{
    "alloc": {
      "0xed9d02e382b34818e88b88a309c7fe71e65f419d": {
        "balance": "1000000000000000000000000000"
      },
      "0xca843569e3427144cead5e4d5999a3d0ccf92b8e": {
        "balance": "1000000000000000000000000000"
      },
      "0x0fbdc686b912d7722dc86510934589e0aaf3b55a": {
        "balance": "1000000000000000000000000000"
      },
      "0x9186eb3d20cbd1f5f992a950d808c4495153abd5": {
        "balance": "1000000000000000000000000000"
      },
      "0x0638e1574728b6d862dd5d3a3e0942c3be47d996": {
        "balance": "1000000000000000000000000000"
      }
    },
    "coinbase": "0x0000000000000000000000000000000000000000",
    "config": {
      "homesteadBlock": 0,
      "byzantiumBlock": 0,
      "constantinopleBlock":0,
      "chainId": 10,
      "eip150Block": 0,
      "eip150Hash": "0x0000000000000000000000000000000000000000000000000000000000000000",
      "eip155Block": 0,
      "eip158Block": 0,
      "isQuorum": true,
      "maxCodeSize" : 35,
      "istanbul": {
        "epoch": 30000,
        "policy": 0,
        "ceil2Nby3Block": 0
      }
    },
    "extraData":
"0x0000000000000000000000000000000000000000000000000000000000000000f897f893946571d97f340c8495b661a823f2c2145ca47d63c2948157d
    "gasLimit": "0xE0000000",
    "difficulty": "0x1",
    "mixHash": "0x63746963616c2062797a616e74696e65206661756c7420746f6c6572616e6365",
    "nonce": "0x0",
    "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
    "timestamp": "0x00"
  }
```


Figure 4-7 Content of "istanbul - genesis.json" file [33]

```
./istanbul-start.sh tessera --tesseraOptions "--tesseraJar /home/[redacted]/
Documents/7NodesTry/tessera/tessera-dist/tessera-app/target/tessera-
app-0.11-SNAPSHOT-jdk11_app.jar"
```

Figure 4-8 IBFT 7-Nodes Blockchain activation script

```
  Found geth: "Quorum Version: 2.4.0"
[*] Starting Tessera nodes
Tessera version (extracted from manifest file): 0.11-SNAPSHOT
Config type -09-
[*] Starting 7 Tessera node(s)
java   -Xms128M -Xmx128M -jar /home/pcep/Documents/7NodesTry/tessera/tessera-di
st/tessera-app/target/tessera-app-0.11-SNAPSHOT-jdk11_app.jar -configfile qdata
/c1/tessera-config-09-1.json >> qdata/logs/tessera1.log 2>&1 &
java   -Xms128M -Xmx128M -jar /home/pcep/Documents/7NodesTry/tessera/tessera-di
st/tessera-app/target/tessera-app-0.11-SNAPSHOT-jdk11_app.jar -configfile qdata
/c2/tessera-config-09-2.json >> qdata/logs/tessera2.log 2>&1 &
java   -Xms128M -Xmx128M -jar /home/pcep/Documents/7NodesTry/tessera/tessera-di
st/tessera-app/target/tessera-app-0.11-SNAPSHOT-jdk11_app.jar -configfile qdata
/c3/tessera-config-09-3.json >> qdata/logs/tessera3.log 2>&1 &
java   -Xms128M -Xmx128M -jar /home/pcep/Documents/7NodesTry/tessera/tessera-di
st/tessera-app/target/tessera-app-0.11-SNAPSHOT-jdk11_app.jar -configfile qdata
/c4/tessera-config-09-4.json >> qdata/logs/tessera4.log 2>&1 &
java   -Xms128M -Xmx128M -jar /home/pcep/Documents/7NodesTry/tessera/tessera-di
st/tessera-app/target/tessera-app-0.11-SNAPSHOT-jdk11_app.jar -configfile qdata
/c5/tessera-config-09-5.json >> qdata/logs/tessera5.log 2>&1 &
java   -Xms128M -Xmx128M -jar /home/pcep/Documents/7NodesTry/tessera/tessera-di
st/tessera-app/target/tessera-app-0.11-SNAPSHOT-jdk11_app.jar -configfile qdata
/c6/tessera-config-09-6.json >> qdata/logs/tessera6.log 2>&1 &
java   -Xms128M -Xmx128M -jar /home/pcep/Documents/7NodesTry/tessera/tessera-di
st/tessera-app/target/tessera-app-0.11-SNAPSHOT-jdk11_app.jar -configfile qdata
/c7/tessera-config-09-7.json >> qdata/logs/tessera7.log 2>&1 &
Waiting until all Tessera nodes are running...
Node 1 is not yet listening on tm.ipc
Node 1 is not yet listening on http
Node 2 is not yet listening on tm.ipc
Node 2 is not yet listening on http
Node 3 is not yet listening on tm.ipc
Node 3 is not yet listening on http
Node 4 is not yet listening on tm.ipc
Node 4 is not yet listening on http
Node 5 is not yet listening on tm.ipc
Node 5 is not yet listening on http
Node 6 is not yet listening on tm.ipc
Node 6 is not yet listening on http
Node 7 is not yet listening on tm.ipc
Node 7 is not yet listening on http
Waiting until all Tessera nodes are running...
```

Figure 4-9 The activation script activating all seven Blockchain nodes

```
All Tessera nodes started
[*] Starting 7 Ethereum nodes with ChainID and NetworkId of 10
ARGS="--nodiscover --istanbul.blockperiod 5 --networkid $NETWORK_ID --syncmode
full --mine --minerthreads 1 --rpc --rpccorsdomain=* --rpcvhosts=* --rpcaddr 0.
0.0.0 --rpcapi admin,db,eth,debug,miner,net,shh,txpool,personal,web3,quorum,ist
anbul,quorumPermission --unlock 0 --password passwords.txt $QUORUM_GETH_ARGS"

basePort=21000
baseRpcPort=22000
for i in `seq 1 ${numNodes}`
do
    port=$(($basePort + ${i} - 1))
    rpcPort=$(($baseRpcPort + ${i} - 1))
    permissioned=
    if ! [[ -z "${STARTPERMISSION+x}" ]] ; then
        permissioned="--permissioned"
    fi

    PRIVATE_CONFIG=qdata/c${i}/tm.ipc nohup geth --datadir qdata/dd${i} ${ARGS}
 ${permissioned} --rpcport ${rpcPort} --port ${port} 2>>qdata/logs/${i}.log &
done

set +v

All nodes configured. See 'qdata/logs' for logs, and run e.g. 'geth attach qdat
a/dd1/geth.ipc' to attach to the first Geth node.
```

Figure 4-10 All seven Blockchain nodes successfully activated

For this implementation, we further created a script to reduce the complexity of the initiation and activation process of 7-Nodes Blockchain. The script "rebirth.sh" will delete all data of an existing set of 7-Nodes Blockchain and initiate a new genesis file to simplify the reset of the Blockchain during the development of the implementation as shown in Figure 4-11. The script "runmy7nodes.sh" will issue the whole command line required for activating the 7-Nodes Blockchain, simplify the Blockchain activation process as shown in Figure 4-12. These scripts will help reduce the complexity of command-line manipulation during the implementation.

```
#!/bin/bash
rm -R qdata/
mkdir qdata
./istanbul-init.sh
```

Figure 4-11 The content of "rebirth.sh" script

```
#!/bin/bash
./istanbul-start.sh tessera --tesseraOptions "--tesseraJar /home/pcep/
Documents/7NodesTry/tessera/tessera-dist/tessera-app/target/tessera-
app-0.11-SNAPSHOT-jdk11_app.jar"
```

Figure 4-12 The content of "runmy7nodes.sh" script

4.1.4 Compile and Deploy Smartcontract Solidity Code

For Smartcontract programming, Ethereum provides a web-based IDE for Solidity language that can compile, test, and deploy smart-contract to specific Ethereum node called “Remix” [47]. In the implementation, Remix was accessed using Google Chrome from the main machine and the session was saved locally using Remix client to avoid unexpected issues on the Solidity code. Each Smartcontract is validated, and test deployed within the IDE before actual implementation on the test machine. The Smartcontract ready for implementation will be compiled using the Solidity compiler provided by Remix community-based plugin. Each compiled solidity code gives ABI code and Binary code which will be used on Web3js code to interact with the Blockchain Smartcontract.

After successfully compiled the Solidity Smartcontract code, the ABI code and Byte code will be automatically generated by the IDE as shown in Figure 4-13. These codes can be copied and passed into Web3js code by assigning a variable to store the code as its value like the example shown in Figure 4-14 and Figure 4-15. Noted that Byte code defines the behavior of the Smartcontract and only required on the first deployment of the Smartcontract which required only once in this implementation, while ABI code define interface for communicating with the Smartcontract and always required every time the native program communicate with Smartcontract.

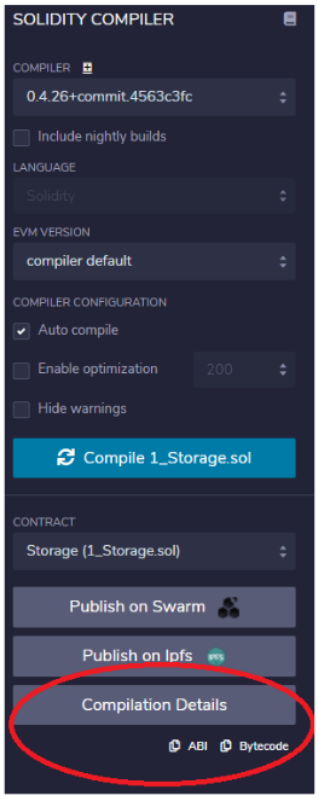


Figure 4-13 ABI Code and Byte code generated can be copied and passed directly (From red circle) into Web3js code [47]

```
var abi =
        [
         {
          "inputs": [],
          "name": "checkLastID",
          "outputs": [
           {
            "internalType": "uint256",
            "name": "",
            "type": "uint256"
           }
          ],
          "stateMutability": "view",
          "type": "function"
         },
         {
          "inputs": [
           {
            "internalType": "uint256",
            "name": "Docid",
            "type": "uint256"
           }
          ],
          "name": "retreiveFull",
          "outputs": [
           {
            "internalType": "string",
            "name": "",
            "type": "string"
           }
          ],
          "stateMutability": "view",
          "type": "function"
         },
         {
          "inputs": [
           {
            "internalType": "uint256",
            "name": "Docid",
            "type": "uint256"
           }
          ],
          "name": "retreiveSearch",
          "outputs": [
           {
            "internalType": "string",
            "name": "",
            "type": "string"
```

```
      }
    ],
    "stateMutability": "view",
    "type": "function"
  },
  {
    "inputs": [
      {
        "internalType": "uint256",
        "name": "Docid",
        "type": "uint256"
      },
      {
        "internalType": "string",
        "name": "searchJSON",
        "type": "string"
      },
      {
        "internalType": "string",
        "name": "fullJSON",
        "type": "string"
      }
    ],
    "name": "store",
    "outputs": [],
    "stateMutability": "nonpayable",
    "type": "function"
  }
];
```

Figure 4-14 ABI code (brown color) assigned into variable "abi"

```
var bytecode =
"0x6080604052348015610010576000080fd5b50612fbd80610020600039600 0f3fe608060405234801561
001057600080fd5b506004361061006257600035 60e01c80636ca1115d1461006757578063862723df1461
067c578063902a9f0814610eed5780639896e0c8146116c7578063d07d9f9b14611778578063f092186d1
4611b30575b600080fd5b61067a600480360361016081101561007e57600080fd5b81019080803590602
0019092919080359060200190640100000000811115610 0a557600080fd5b82018360208201111561 00b
757600080fd5b80359060200191846001830284011164010000000083111715610 0d957600080fd5b919
08080601f01602080910402602001604051908101604052809392919081815260200183838082843760 0
081840152601f19601f820116905080830192505050505050505091929192908035906020019064010000
00081111561013c57600080fd5b82018360208201111561014e57600080fd5b80359060200191846001 8
302840111640100000000831117156101 7057600080fd5b91908080601f01602080910402602001604 05
190810160405280939291908181526020018383808284376000818401526 01f19601f820116905080830
192505050505050505091929192908035906020019064010000000081111561 01d357600080fd5b8201836
020820111156101e557600080fd5b80359060200191846001830284011164010000000083111715610 20
757600080fd5b91908080601f01602080910402602001604051908101604052809392919081815260200
183838082843760008184015260 1f19601f820116905080830192505050505050505091929192908035906
020019064010000000081111561026a57600080fd5b82018360208201111561027c57600080fd5b80359
060200191846001830284011164010000000083111715610 29e57600080fd5b91908080601f016020809
104026020016040519081016040528093929190818152602001838380828437600081840152601f19601
f820116905080830192505050505050505091929192908035906020019064010000000081111561 0301576
00080fd5b82018360208201111561031357600080fd5b80359060200191846001830284011164010000
000831117156103357600080fd5b91908080601f01602080910402602001604051908101604052809 39
291908181526020018383808284376000818401526 01f19601f820116905080830192505050505050509
192919290803590602001906401000000008111156103985 7600080fd5b82018360208201111561 03aa5
7600080fd5b80359060200191846001830284011164010000000083111715610 3cc57600080fd5b91908
080601f01602080910402602001604051908101604052809392919081815260200183838082843760008
1840152601f19601f820116905080830192505050505050505091929192908035906020019064010000000
081111561042f57600080fd5b82018360208201111561044157600080fd5b80359060200191846001830
284011164010000000083111715610 46357600080fd5b91908080601f01602080910402602001604 519
081016040528093929190818152602001838380828437600081840152601f19601f82011690508083019
250505050505050919292192908035906020019064010000000081111156104c657600080fd5b820183602
0820111156104d857600080fd5b80359060200191846001830284011164010000000083111715610 4fa5
7600080fd5b91908080601f01602080910402602001604051908101604052809392919081815260200 18
383808284376000818401526 01f19601f820116905080830192505050505050505091929192908035906 02
0019064010000000081111561055d57600080fd5b82018360208201111561056f57600080fd5b8035906
020019184600183028401116401000000008311171561059157600080fd5b91908080601f01602080910
402602001604051908101604052809392919081815260200183838082843760008184015260 1f19601f8
201169050808301925050505050505091929192908035906020019064010000000081111561 05f457600
080fd5b82018360208201111561060657600080fd5b80359060200191846001830284011164010000000
083111715610628 57600080fd5b91908080601f01602080910402602001604051908101604052809392";
```

Figure 4-15 Byte code (brown color) assigned into variable "bytecode"

4.1.5 Deploy Smartcontract into Blockchain

Before begin registering health documents metadata into Blockchain Smartcontract, the Smartcontract must be first deployed to act as a contract format for the entire Blockchain ledger for the implementation. Figure 4-16 show the Web3js script that simply deploys the Smartcontract into the Blockchain ledger. It required both the Byte code and ABI code received from the Solidity compiler to be completed. Once the Smartcontract is deployed, any later communication with Smartcontract will only need ABI code to act as an interface for the communication. This Smartcontract deployment process only required once at the initiation of the Blockchain ledger and no longer needs to be performed ever again for the rest of the Blockchain ledger life cycle.

```
a = eth.accounts[0]
web3.eth.defaultAccount = a;

// abi and bytecode generated from simplestorage.sol:
// > solcjs --bin --abi simplestorage.sol
/*var abi =
[{"constant":true,"inputs":[],"name":"storedData","outputs":[{"name":"","type":"uint256"}],"payable":f
alse,"type":"function"},{"constant":false,"inputs":[{"name":"x","type":"uint256"}],"name":"set","outputs
":[],"payable":false,"type":"function"},{"constant":true,"inputs":[],"name":"get","outputs":[{"name":"ret
Val","type":"uint256"}],"payable":false,"type":"function"},{"inputs":[{"name":"initVal","type":"uint256"}
],"payable":false,"type":"constructor"}];*/

var abi = [...];

var bytecode = [...];

var simpleContract = web3.eth.contract(abi);
var simple = simpleContract.new(42, {from:web3.eth.accounts[0], data: bytecode, gas: 0x47b760},
function(e, contract) {
        if (e) {
                console.log("err creating contract", e);
        } else {
                if (!contract.address) {
                        console.log("Contract transaction send: TransactionHash: " +
contract.transactionHash + " waiting to be mined...");
                } else {
                        console.log("Contract mined! Address: " + contract.address);
                        console.log(contract);
                }
        }
});
```

Figure 4-16 The Web3js script for Smartcontract deploy

4.1.6 Prepare NodeJS Coding Environment

For the coding process of all non-Blockchain native Javascript programs, the coding environment must be provided essential coding components. In this implementation, we use NodeJS as a compiler and coding environment for all Javascript programs. NodeJS is available at [48]. The essential coding components node module can be installed using Node Package Manager (NPM) which comes together with NodeJS [49]. The coding of program for native side of XDS Actor require node module name "Web3" (Web3js) [50], "xml2js" [51], "fs" [52], "net" [53], "util" [54], "moment" [55], and "cryptr" [56] which can be installed using the command-line as shown in Figure 4-17.

```
npm install (with no args, in package dir)
npm install [<@scope>/]<name>
npm install [<@scope>/]<name>@<tag>
npm install [<@scope>/]<name>@<version>
npm install [<@scope>/]<name>@<version range>
npm install <alias>@npm:<name>
npm install <git-host>:<git-user>/<repo-name>
npm install <git repo url>
npm install <tarball file>
npm install <tarball url>
npm install <folder>

aliases: npm i, npm add
common options: [-P|--save-prod|-D|--save-dev|-O|--save-optional|--save-peer] [-E|--
save-exact] [-B|--save-bundle] [--no-save] [--dry-run]
```

Figure 4-17 "npm install" command-line [49]

4.2 XDS Actors

As we have seen from HL7 and FHIR, current healthcare information exchanged related standards are mostly web-based protocol. Additionally, development of IT infrastructure to support healthcare operation require the capability to handle a huge amount of transaction in a limited amount of time so, it requires our system implementation to be able to handle multitask properly. With asynchronous nature and compatibility with website integration, Javascript is one of the best choices for our implementation of this work. In this implementation, we adopt the "Node.js" variant of Javascript as it was made to build scalable network applications that handle many connections concurrently. Furthermore, Node.js also providing simple access to community-made node modules which offer a wide variety of useful APIs for software development which may reduce difficulty in our implementation further.

All actors within IHE XDS Profile communicate with each other using XML message transaction. As we utilize Javascript as main programming language for the implementation, these XML messages need to be interpreted into programming object to allow simpler handling method within the program. JSON (Javascript Object Notation) is a lightweight data-exchange format for programming objects that was created specifically for this purpose. It is simple to read and write for humans, and it is simple to parse or generate for machines. That mean, all XML message transactions sent to XDS Document Registry actor program will be converted into JSON. For this implementation, we utilize NodeJS “xml2js” module for the task.

To connect our program to Ethereum smart contract, we can use Ethereum API tools which is Web3 [57] as a middle. Web3 allows smart contract control through preferred programming language and transitions logic and variables from the language to Solidity. Web3 provides a programming API for Javascript called "Web3JS" which allows the Javascript program to interact with Ethereum based smart-contract. The API can be accessed using the node module provided via Node.js.

4.2.1 XDS Document Repository Actor

4.2.1.1 Interpret IHE ITI-42 Transaction

Figure 4-18, showing XML language code snippet of Registry Document Set-b [ITI-42] transaction sample. The code composing of two main sections. The first section

labeled with “lcm:SubmitObjectRequest” is where XML schematic information are located and the label also act as marker which tell interpreter program to recognize it as ITI-42 transaction. The second section starts from label “rim:RegistryObjectList” following with “rim:ExtrinsicObject” contain all information regarding corresponding health document. This section is where all Metadata attributes of the document are located. If the Document Registry Actor successfully received the transaction, they must return response as shown in Figure 4-18. The response transaction included only XML schematic information, message UUID number, and status type “successful” as shown in Figure 4-19. This response will let the Repository finish its process and end messaging attempt. Figure 4-20 show the actual code snippet using for the implementation.

```
<lcm:SubmitObjectsRequest xmlns:xsi="..XML schematic information..">
        <rim:RegistryObjectList>

                <rim:ExtrinsicObject id="..DocumentEntry identifier name or label.."
                        mimeType="text/xml"
                        objectType="urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1">
                        ...
                        ..DocumentEntry META-data attributes object list..
                        ...
                </rim:ExtrinsicObject>

                <rim:RegistryPackage id="..SubmissionSet identifier name or label..">
                        ...
                        ..SubmissionSet or Folder META-data attributes object list..
                        ...
                </rim:RegistryPackage>

                <rim:Classification id="..Classification identifier name or label.."
                        classifiedObject="..SubmissionSet ID which it belonged to.."
                        classificationNode="urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd"/>

                <rim:Association id="..Association identifier name or label.."
                        associationType="..Association Type.."
                        sourceObject="..SubmissionSet ID which it belonged to.."
                        targetObject="..Target Document Entry ID..">
                        ...
                        ..Association META-data attributes objext list..
                        ...
                </rim:Association>
        </rim:RegistryObjectList>
</lcm:SubmitObjectsRequest>
```

Figure 4-18 Pseudocode represents general format of Register Document Set-b [ITI - 42] [58]

```
<rs:RegistryResponse xmlns:rs="urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0 ../../schema/ebRS/rs.xsd"
        status="urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Success"/>
```

Figure 4-19 XML Code snippet of Registry Document Set-b Response transaction sample [58]

```
<lcm:SubmitObjectsRequest xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0 ../../schema/ebRS/lcm.xsd"
xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0" xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0" xmlns:rs="urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0">
        <rim:RegistryObjectList>
                <rim:ExtrinsicObject id="Document01" mimeType="text/xml"
objectType="urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1">
                        <rim:Slot name="creationTime">
                                <rim:ValueList>
                                        <rim:Value>20051224</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="languageCode">
                                <rim:ValueList>
                                        <rim:Value>en-us</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="serviceStartTime">
                                <rim:ValueList>
                                        <rim:Value>200412230800</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="serviceStopTime">
                                <rim:ValueList>
                                        <rim:Value>200412230801</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="sourcePatientId">
                                <rim:ValueList>
                                        <rim:Value>ST-
1000^^^&amp;1.3.6.1.4.1.21367.2003.3.9&amp;ISO</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="sourcePatientInfo">
                                <rim:ValueList>
                                        <rim:Value>PID-3|ST-
1000^^^&amp;1.3.6.1.4.1.21367.2003.3.9&amp;ISO</rim:Value>
                                        <rim:Value>PID-5|Doe^John^^^</rim:Value>
                                        <rim:Value>PID-7|19560527</rim:Value>
                                        <rim:Value>PID-8|M</rim:Value>
                                        <rim:Value>PID-11|100 Main
St^^Metropolis^Il^44130^USA</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
```

```
<rim:Name>
    <rim:LocalizedString value="Physical"/>
</rim:Name>
<rim:Description/>
<rim:Classification id="cl01" classificationScheme="urn:uuid:93606bcf-9494-
43ec-9b4e-a7748d1a838d" classifiedObject="Document01">
    <rim:Slot name="authorPerson">
        <rim:ValueList>
            <rim:Value>Gerald Smitty</rim:Value>
        </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
        <rim:ValueList>
            <rim:Value>Cleveland Clinic</rim:Value>
            <rim:Value>Parma Community</rim:Value>
        </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
        <rim:ValueList>
            <rim:Value>Attending</rim:Value>
        </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
        <rim:ValueList>
            <rim:Value>Orthopedic</rim:Value>
        </rim:ValueList>
    </rim:Slot>
</rim:Classification>
<rim:Classification id="cl02" classificationScheme="urn:uuid:41a5887f-8865-
4c09-adf7-e362475b143a" classifiedObject="Document01" nodeRepresentation="History and Physical">
    <rim:Slot name="codingScheme">
        <rim:ValueList>
            <rim:Value>Connect-a-thon classCodes</rim:Value>
        </rim:ValueList>
    </rim:Slot>
    <rim:Name>
        <rim:LocalizedString value="History and Physical"/>
    </rim:Name>
</rim:Classification>
<rim:Classification id="cl03" classificationScheme="urn:uuid:f4f85eac-e6cb-
4883-b524-f2705394840f" classifiedObject="Document01"
nodeRepresentation="1.3.6.1.4.1.21367.2006.7.101">
```

```
                    <rim:Slot name="codingScheme">
                        <rim:ValueList>
                            <rim:Value>Connect-a-thon
confidentialityCodes</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                    <rim:Name>
                        <rim:LocalizedString value="Clinical-Staff"/>
                    </rim:Name>
                </rim:Classification>
                <rim:Classification id="cl04" classificationScheme="urn:uuid:a09d5840-386c-
46f2-b5ad-9c3699a4309d" classifiedObject="Document01" nodeRepresentation="CDAR2/IHE 1.0">
                    <rim:Slot name="codingScheme">
                        <rim:ValueList>
                            <rim:Value>Connect-a-thon formatCodes</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                    <rim:Name>
                        <rim:LocalizedString value="CDAR2/IHE 1.0"/>
                    </rim:Name>
                </rim:Classification>
                <rim:Classification id="cl05" classificationScheme="urn:uuid:f33fb8ac-18af-
42cc-ae0e-ed0b0bdb91e1" classifiedObject="Document01" nodeRepresentation="Outpatient">
                    <rim:Slot name="codingScheme">
                        <rim:ValueList>
                            <rim:Value>Connect-a-thon
healthcareFacilityTypeCodes</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                    <rim:Name>
                        <rim:LocalizedString value="Outpatient"/>
                    </rim:Name>
                </rim:Classification>
                <rim:Classification id="cl06" classificationScheme="urn:uuid:cccf5598-8b07-
4b77-a05e-ae952c785ead" classifiedObject="Document01" nodeRepresentation="General Medicine">
                    <rim:Slot name="codingScheme">
                        <rim:ValueList>
                            <rim:Value>Connect-a-thon
practiceSettingCodes</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                    <rim:Name>
                        <rim:LocalizedString value="General Medicine"/>
                    </rim:Name>
                </rim:Classification>
```

```
                <rim:Classification id="cl07" classificationScheme="urn:uuid:f0306f51-975f-
434e-a61c-c59651d33983" classifiedObject="Document01" nodeRepresentation="34108-1">
                        <rim:Slot name="codingScheme">
                                <rim:ValueList>
                                        <rim:Value>LOINC</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Name>
                                <rim:LocalizedString value="Outpatient Evaluation And
Management"/>
                        </rim:Name>
                </rim:Classification>
                <rim:ExternalIdentifier id="ei01" registryObject="Document01"
identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427" value="SELF-
5^^^&amp;1.3.6.1.4.1.21367.2005.3.7&amp;ISO">
                        <rim:Name>
                                <rim:LocalizedString value="XDSDocumentEntry.patientId"/>
                        </rim:Name>
                </rim:ExternalIdentifier>
                <rim:ExternalIdentifier id="ei02" registryObject="Document01"
identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.3.6.1.4.1.21367.2005.3.9999.32">
                        <rim:Name>
                                <rim:LocalizedString value="XDSDocumentEntry.uniqueId"/>
                        </rim:Name>
                </rim:ExternalIdentifier>
        </rim:ExtrinsicObject>
        <rim:RegistryPackage id="SubmissionSet01">
                <rim:Slot name="submissionTime">
                        <rim:ValueList>
                                <rim:Value>20041225235050</rim:Value>
                        </rim:ValueList>
                </rim:Slot>
                <rim:Name>
                        <rim:LocalizedString value="Physical"/>
                </rim:Name>
                <rim:Description>
                        <rim:LocalizedString value="Annual physical"/>
                </rim:Description>
                <rim:Classification id="cl08" classificationScheme="urn:uuid:a7058bb9-b4e4-
4307-ba5b-e3f0ab85e12d" classifiedObject="SubmissionSet01">
                        <rim:Slot name="authorPerson">
                                <rim:ValueList>
                                        <rim:Value>Sherry Dopplemeyer</rim:Value>
                                </rim:ValueList>
```

```
                    </rim:Slot>
                    <rim:Slot name="authorInstitution">
                        <rim:ValueList>
                            <rim:Value>Cleveland Clinic</rim:Value>
                            <rim:Value>Berea Community</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                    <rim:Slot name="authorRole">
                        <rim:ValueList>
                            <rim:Value>Primary Surgon</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                    <rim:Slot name="authorSpecialty">
                        <rim:ValueList>
                            <rim:Value>Orthopedic</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                </rim:Classification>
                <rim:Classification id="cl09" classificationScheme="urn:uuid:aa543740-bdda-
424e-8c96-df4873be8500" classifiedObject="SubmissionSet01" nodeRepresentation="History and
Physical">
                    <rim:Slot name="codingScheme">
                        <rim:ValueList>
                            <rim:Value>Connect-a-thon
contentTypeCodes</rim:Value>
                        </rim:ValueList>
                    </rim:Slot>
                    <rim:Name>
                        <rim:LocalizedString value="History and Physical"/>
                    </rim:Name>
                </rim:Classification>
                <rim:ExternalIdentifier id="ei03" registryObject="SubmissionSet01"
identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.3.6.1.4.1.21367.2005.3.9999.33">
                    <rim:Name>
                        <rim:LocalizedString value="XDSSubmissionSet.uniqueId"/>
                    </rim:Name>
                </rim:ExternalIdentifier>
                <rim:ExternalIdentifier id="ei04" registryObject="SubmissionSet01"
identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832" value="3670984664">
                    <rim:Name>
                        <rim:LocalizedString value="XDSSubmissionSet.sourceId"/>
                    </rim:Name>
                </rim:ExternalIdentifier>
```

```
                        <rim:ExternalIdentifier id="ei05" registryObject="SubmissionSet01"
identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446" value="SELF-
5^^^&amp;1.3.6.1.4.1.21367.2005.3.7&amp;ISO">
                                <rim:Name>
                                        <rim:LocalizedString value="XDSSubmissionSet.patientId"/>
                                </rim:Name>
                        </rim:ExternalIdentifier>
                </rim:RegistryPackage>
                <rim:Classification id="cl10" classifiedObject="SubmissionSet01"
classificationNode="urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd"/>
                <rim:Association id="as01" associationType="HasMember"
sourceObject="SubmissionSet01" targetObject="Document01">
                        <rim:Slot name="SubmissionSetStatus">
                                <rim:ValueList>
                                        <rim:Value>Original</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                </rim:Association>
        </rim:RegistryObjectList>
</lcm:SubmitObjectsRequest>
```

Figure 4-20 XML Code snippet of Registry Document Set-b [ITI-42] transaction sample [58]

4.2.1.2 XDS Document Repository Actor Simulating Program

For document registering, XDS Document Repository Actor register document Metadata attributes into XDS Document Registry Actor using IHE ITI-42 transaction. XDS Document Registry Actor then interprets the transaction into a programmable object before check if the transaction is ITI-42. Then, the actor proceeds to pass the retrieved object into Blockchain smart-contract and publish it into a Blockchain ledger. Figure 4-21 showing the Javascript code snippet for the native side of XDS Document Repository Actor.

```
var hrstart = null;
var net = require('net');
var fs = require("fs");
var util = require("util");
var xml2js = require('xml2js');
var parseString = xml2js.parseString;
var builder = new xml2js.Builder;
const readline = require("readline");
const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout
});
var HOST = '127.0.0.1';
var PORT = 65519;
var client = new net.Socket();
client.connect(PORT, HOST, function() {
  rl.question("Choose documents: ", function(docNum) {
    console.log('CONNECTED TO: ' + HOST + ':' + PORT);
    var docChosen = 'SingleDocumentEntry' + docNum + '.xml';
    fs.readFile(docChosen, function(err, buf) {
      if (err) console.log(err);
      var text = buf.toString();
      client.write(text);
      hrstart = process.hrtime();
      console.log('Sent: \n' + text + '\n');
    });
    rl.close();
  });
});
client.on('data', function(data) {
  var hrend = process.hrtime(hrstart);
  console.log('==============================');
  console.log('Respond received: ' + data);
  console.info('Execution time (hr): %ds %dms', hrend[0], hrend[1] / 1000000);
  console.log('==============================');
  client.destroy();
});
client.on('close', function() {
  console.log('Connection closed');
});
```

Figure 4-21 Javascript Code Snippet of XDS Document Repository Actor

4.2.2 XDS Document Consumer Actor

4.2.2.1 Interpret IHE ITI-18 Transaction

Figure 4-22, showing XML language code snippet of RegistryStoredQueryRequest [ITI-18] transaction sample. The code composing of 3 main sections. The first section labeled “query:AdhocQueryRequest” is where XML schematic information are located and the label also act as marker which tell interpreter program to recognize it as ITI-18 transaction. The second section labeled “query:ResponseOption” mark the expected format of query result that will return to Document Consumer. The third section start from label “rim:AdhocQuery” contain all search keywords issued by Document Consumer. These search keywords are selected Metadata attributes and its value. When Document Registry Actor received the transaction, they will use search keyword provided to search for registry with matched Metadata attributes value then return the result to Document Consumer Actor as response transaction following Figure 4-23 and Figure 4-24. With header labeled “query:AdhocQueryResponse”, the transaction contain search result depend on query type specified in ITI-18 transaction. If the query expected for “LeafClass” as result, the response would return Metadata attributes of all matched result in detailed as shown in Figure 4-23. Otherwise, if the query expected for “ObjectList”, the response would return object reference number of all matched result as shown in Figure 4-24. These two types of response specifically selected depend on search behavior of Document Consumer Actor’s user. The query which specified “LeafClass” as its search result must provide keyword which unique to its corresponding document, such as document unique ID or object reference UUID. At the same time, “ObjectList” are used to search for wide range of document with generic search keyword and value where discovery of document existent is the main goal. Figure 4-25 showing the actual ITI-18 transaction using in the implementation while Figure 4-26 showing its actual response transaction.

```
<query:AdhocQueryRequest xmlns:xsi="..XML schematic information..">
        <query:ResponseOption returnComposedObjects="true"
                returnType="(LeafClass or ObjectList)"/>
        <rim:AdhocQuery id=" urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d ">
                ...
                <rim:Slot name="..Search keyword label of META-data attributes..">
                        <rim:ValueList>
                                <rim:Value>..META-data attributes value..</rim:Value>
                        </rim:ValueList>
                </rim:Slot>
                ...
                ..List of search keywords from Document Consumer Actor..
                ...
        </rim:AdhocQuery>
</query:AdhocQueryRequest>
```

Figure 4-22 Pseudocode represents general format of Registry Stored Query Request [ITI - 18] [58]

```
<query:AdhocQueryResponse xmlns:xsi="..XML information schematic..">
        <rim:RegistryObjectList>
                ...
                <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
                        xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
                        id="..Object Reference UUID.."/>
                ...
                ..Object reference of all matched result..
                ...
        </rim:RegistryObjectList>
</query:AdhocQueryResponse>
```

Figure 4-23 Pseudocode represents general format of Query Response included "Object Reference" of search results [58]

```
<query:AdhocQueryResponse xmlns:xsi="..XML information schematic..">
        <rim:RegistryObjectList>
                <rim:ExtrinsicObject xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
                        xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
                        id="..Identification UUID.."
                        isOpaque="false"
                        mimeType="text/xml"
                        objectType="..Object Type UUID.."
                        status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
                        ..
                        //Document generic information META-data attributes
                        <rim:Slot name="..META-data attributes type name..">
                                <rim:ValueList>
                                        <rim:Value>..META-data attributes value..</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        ...
                        //META-data attributes "title"
                        <rim:Name>
                                <rim:LocalizedString charset="UTF-8"
                                        value="..META-data attributes value.."  xml:lang="en-us"/>
                        </rim:Name>
                        ...
                        //META-data attributes "comment"
                        <rim:Description/>
                        ...
                        //Communication protocol-based META-data attributes
                        <rim:Classification classificationScheme="..META-data attributes type UUID.."
                                classifiedObject="..Classified Object UUID.."
                                id="..Identification UUID.."
                                nodeRepresentation="..Representing Node.."
                                objectType="..Object Type UUID..">
                                <rim:Slot name="..META-data attributes type name..">
                                        <rim:ValueList>
                                                <rim:Value>..META-data attributes value..</rim:Value>
                                        </rim:ValueList>
                                </rim:Slot>
                                <rim:Name>
                                        <rim:LocalizedString charset="UTF-8"
                                                value="..Representing Node Detail.."
                                                xml:lang="en-us"/>
                                </rim:Name>
                                <rim:Description/>
                        </rim:Classification>
                        ...
```

```
                    ...
                    //External identifier-based META-data attributes
                    <rim:ExternalIdentifier id="..META-data attributes type UUID.."
                            registryObject="..Registry Object UUID.."
                            identificationScheme="..Identification scheme UUID.."
                            objectType="ExternalIdentifier"
                            value="..META-data attributes value..">
                            <rim:Name>
                                    <rim:LocalizedString charset="UTF-8"
                                            value="..META-data attributes type name.."
                                            xml:lang="en-us"/>
                            </rim:Name>
                            <rim:Description/>
                    </rim:ExternalIdentifier>
                    ...
                    ..List of Document Entry META-data attributes of matched result..
                    ...
            </rim:ExtrinsicObject>
    </rim:RegistryObjectList>
</query:AdhocQueryResponse>
```

Figure 4-24 Pseudocode represents general format of Query Response included "Leaf Class" of search result [58]

```
<query:AdhocQueryRequest xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0 ../../schema/ebRS/query.xsd"
xmlns:query="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0" xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0" xmlns:rs="urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0">
        <query:ResponseOption returnComposedObjects="true" returnType="LeafClass"/>
        <rim:AdhocQuery id="urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d">
                <rim:Slot name="$XDSDocumentEntryPatientId">
                        <rim:ValueList>

        <rim:Value>st3498702^^^&amp;1.3.6.1.4.1.21367.2005.3.7&amp;ISO</rim:Value>
                        </rim:ValueList>
                </rim:Slot>
                <rim:Slot name="$XDSDocumentEntryStatus">
                        <rim:ValueList>
                                <rim:Value>('urn:oasis:names:tc:ebxml-
regrep:ResponseStatusType:Approved')</rim:Value>
                        </rim:ValueList>
                </rim:Slot>
                <rim:Slot name="$XDSDocumentEntryCreationTimeFrom">
                        <rim:ValueList>
                                <rim:Value>200412252300</rim:Value>
                        </rim:ValueList>
                </rim:Slot>
                <rim:Slot name="$XDSDocumentEntryCreationTimeTo">
                        <rim:ValueList>
                                <rim:Value>200501010800</rim:Value>
                        </rim:ValueList>
                </rim:Slot>
                <rim:Slot name="$XDSDocumentEntryHealthcareFacilityTypeCode">
                        <rim:ValueList>
                                <rim:Value>('Emergency Department')</rim:Value>
                        </rim:ValueList>
                </rim:Slot>
        </rim:AdhocQuery>
</query:AdhocQueryRequest>
```


Figure 4-25 XML Code Snippet of RegistryStoredQueryRequest [ITI-18] Transaction Sample [58]

```
<query:AdhocQueryResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0 ../../schema/ebRS/query.xsd"
xmlns:query="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0" xmlns:rim="urn:oasis:names:tc:ebxml-
regrep:xsd:rim:3.0" status="Success">
        <rim:RegistryObjectList>
                <rim:ExtrinsicObject xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:08a15a6f-5b4a-42de-8f95-
89474f83abdf" isOpaque="false" mimeType="text/xml" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
                        <rim:Slot name="URI">
                                <rim:ValueList>
                                        <rim:Value>http://localhost:8080/XDS/Repository/08a15a6f-
5b4a-42de-8f95-89474f83abdf.xml</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="authorInstitution">
                                <rim:ValueList>
                                        <rim:Value>Fairview Hospital</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="creationTime">
                                <rim:ValueList>
                                        <rim:Value>200412261119</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="hash">
                                <rim:ValueList>

        <rim:Value>4cf4f82d78b5e2aac35c31bca8cb79fe6bd6a41e</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="languageCode">
                                <rim:ValueList>
                                        <rim:Value>en-us</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="serviceStartTime">
                                <rim:ValueList>
                                        <rim:Value>200412230800</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
                        <rim:Slot name="serviceStopTime">
                                <rim:ValueList>
                                        <rim:Value>200412230801</rim:Value>
                                </rim:ValueList>
                        </rim:Slot>
```

```
</rim:Slot>
<rim:Slot name="size">
	<rim:ValueList>
		<rim:Value>54449</rim:Value>
	</rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
	<rim:ValueList>
		<rim:Value>jd12323^^^wsh</rim:Value>
	</rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
	<rim:ValueList>
		<rim:Value>PID-3|pid1^^^domain</rim:Value>
		<rim:Value>PID-5|Doe^John^^^</rim:Value>
		<rim:Value>PID-7|19560527</rim:Value>
		<rim:Value>PID-8|M</rim:Value>
		<rim:Value>PID-11|100 Main
St^^Metropolis^Il^44130^USA</rim:Value>
	</rim:ValueList>
</rim:Slot>
<rim:Name>
	<rim:LocalizedString charset="UTF-8" value="Sample document 1"
xml:lang="en-us"/>
</rim:Name>
<rim:Description/>
<rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-
e362475b143a" classifiedObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
id="urn:uuid:ac872fc0-1c6e-439f-84d1-f76770a0ccdf" nodeRepresentation="Education"
objectType="Urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification">
	<rim:Slot name="codingScheme">
		<rim:ValueList>
			<rim:Value>Connect-a-thon classCodes</rim:Value>
		</rim:ValueList>
	</rim:Slot>
	<rim:Name>
		<rim:LocalizedString charset="UTF-8" value="Education"
xml:lang="en-us"/>
	</rim:Name>
	<rim:Description/>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-
f2705394840f" classifiedObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
id="urn:uuid:f1a8c8e4-3593-4777-b7e0-8b0773378705" nodeRepresentation="C"
objectType="Urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification">
```

```
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>Connect-a-thon
confidentialityCodes</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString charset="UTF-8" value="Celebrity"
xml:lang="en-us"/>
</rim:Name>
<rim:Description/>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-
9c3699a4309d" classifiedObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
id="urn:uuid:b6e49c73-96c8-4058-8c95-914d83bd262a" nodeRepresentation="CDAR2/IHE 1.0"
objectType="Urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>Connect-a-thon formatCodes</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString charset="UTF-8" value="CDAR2/IHE 1.0"
xml:lang="en-us"/>
</rim:Name>
<rim:Description/>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-
ed0b0bdb91e1" classifiedObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
id="urn:uuid:61e2b376-d74a-4984-ac21-dcd0b8890f9d" nodeRepresentation="Emergency Department"
objectType="Urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>Connect-a-thon
healthcareFacilityTypeCodes</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString charset="UTF-8" value="Assisted Living"
xml:lang="en-us"/>
</rim:Name>
<rim:Description/>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-
ae952c785ead" classifiedObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
```

id="urn:uuid:fb7677c5-c42f-485d-9010-dce0f3cd4ad5" nodeRepresentation="Cardiology"
objectType="Urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>Connect-a-thon
practiceSettingCodes</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString charset="UTF-8" value="Cardiology"
xml:lang="en-us"/>
</rim:Name>
<rim:Description/>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-
c59651d33983" classifiedObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
id="urn:uuid:0a8a8ed9-8be5-4a63-9b68-a511adee8ed5" nodeRepresentation="34098-4"
objectType="Urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>LOINC</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString charset="UTF-8" value="Conference
Evaluation Note" xml:lang="en-us"/>
</rim:Name>
<rim:Description/>
</rim:Classification>
<rim:ExternalIdentifier id="urn:uuid:db9f4438-ffff-435f-9d34-d76190728637"
registryObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
objectType="ExternalIdentifier" value="st3498702^^^&1.3.6.1.4.1.21367.2005.3.7&ISO">
<rim:Name>
<rim:LocalizedString charset="UTF-8"
value="XDSDocumentEntry.patientId" xml:lang="en-us"/>
</rim:Name>
<rim:Description/>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier id="urn:uuid:c3fcbf0e-9765-4f5b-abaa-b37ac8ff05a5"
registryObject="urn:uuid:08a15a6f-5b4a-42de-8f95-89474f83abdf"
identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
objectType="ExternalIdentifier" value="1.3.6.1.4.1.21367.2005.3.99.1.1010">
<rim:Name>
<rim:LocalizedString charset="UTF-8"
value="XDSDocumentEntry.uniqueId" xml:lang="en-us"/>

```
                    </rim:Name>
                    <rim:Description/>
                </rim:ExternalIdentifier>
            </rim:ExtrinsicObject>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:41a5887f-8865-4c09-adf7-
e362475b143a"/>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:f4f85eac-e6cb-4883-b524-
f2705394840f"/>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:a09d5840-386c-46f2-b5ad-
9c3699a4309d"/>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:f33fb8ac-18af-42cc-ae0e-
ed0b0bdb91e1"/>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:cccf5598-8b07-4b77-a05e-
ae952c785ead"/>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:f0306f51-975f-434e-a61c-
c59651d33983"/>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:58a6f841-87b3-4a3e-92fd-
a8ffeff98427"/>
            <rim:ObjectRef xmlns:q="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" id="urn:uuid:2e82c1f6-a085-4c72-9da3-
8640a32e42ab"/>
        </rim:RegistryObjectList>
</query:AdhocQueryResponse>
```


Figure 4-26 XML Code Snippet of RegistryStoredQueryResponse Transaction Sample [58]

4.2.2.2 XDS Document Consumer Actor Simulating Program

For document query, XDS Document Consumer query for document Metadata attributes stored within XDS Document Registry Blockchain providing search operation type and some Metadata attributes value as search keyword via ITI-18 transaction. XDS Document Registry will check if the transaction is ITI-18 before performing search operation matching specified search type using provided keyword Metadata attributes value. The search operation will be performed by consequently call for each registered smart-contracts until all contracts with matched attributes value were found. XDS Document Registry Actor then returns all query result in XML format following specification for ITI-18 responding. Upon receiving the query response, XDS Document Consumer then interprets the transaction and displays the result to the user in a human-understandable format.

Following IHE XDS Profile, XDS Document Consumer actor is where the user specifies search keyword values of Metadata attributes for the system to query for matching document exist within XDS Affinity Domain. For this implementation, we design that the user interface will take the form of a command-line program that can be run via Windows command prompt or Linux terminal. The program will prompt the user to specify search type, including META-attributes value, and specify the value. The actor then accepts these values to create an XML message following ITI-18 format before sending it to a local or accessible XDS Document Registry actor to query for matching document and start search operation. Figure 4-31 showing the Javascript code snippet for the native-side of the XDS Document Consumer Actor program.

The command-line interface of XDS Document Consumer begins with prompt the user to input registry query types or choose to quit the program as shown in Figure 4-27. The user will need to specify digit numbers corresponding to the choice. Then, the user will be prompt to input essential metadata attributes required for the query type (i.e., FindDocuments will require attributes included DocumentEntryPatientId and DocumentEntryStatus as shown in Figure 4-28) before prompt to input other metadata attributes as optional depending on the information the user known as shown in Figure 4-29. When there are no more metadata attribute values to add, the user can choose to start a query for the document as shown in Figure 4-30. The XDS Document Consumer

Actor program will then accept the input and assort it into the ITI-18 transaction before sending it to XDS Document Registry Actor.

```
==================================
|| XDS Consumer Actor Interface ||
==================================
Please select query type
1) FindDocuments
2) FindSubmissionSets
3) FindFolders
4) GetAll
5) GetDocuments
6) GetFolders
7) GetAssociations
8) GetDocumentsAndAssociations
9) GetSubmissionSets
10) GetSubmissionSetAndContents
11) GetFolderAndContents
12) GetFoldersForDocument
13) GetRelatedDocuments
14) FindDocumentsByReferenceId
#) Quit
(Specify number): 1
```

Figure 4-27 The program prompt user to input query type

```
Keywords require: XDSDocumentEntryPatientId
Value: 1234
Keywords require: XDSDocumentEntryStatus
Value: Approved
```

Figure 4-28 The program prompt user to input essential metadata attribute values

(In case of FindDocuments query type)

```
==========================
Query type: FindDocuments
Query keywords:
$XDSDocumentEntryPatientId = 1234
$XDSDocumentEntryStatus = Approved
==========================
Available optional keywords:
0) No more optional keywords
1) XDSDocumentEntryClassCode
2) XDSDocumentEntryTypeCode
3) XDSDocumentEntryPracticeSettingCode
4) XDSDocumentEntryCreationTime
5) XDSDocumentEntryServiceStartTime
6) XDSDocumentEntryServiceStopTime
7) XDSDocumentEntryHealthcareFacilityTypeCode
8) XDSDocumentEntryEventCodeList
9) XDSDocumentEntryConfidentialityCode
10) XDSDocumentEntryAuthorPerson
11) XDSDocumentEntryFormatCode
12) XDSDocumentEntryType
#) Quit
Select keywords (specify number): 10
Keyword: XDSDocumentEntryAuthorPerson
Value: Jennifer
```

Figure 4-29 The program prompt user to input optional metadata attributes

```
==========================
Query type: FindDocuments
Query keywords:
$XDSDocumentEntryPatientId = 1234
$XDSDocumentEntryStatus = Approved
$XDSDocumentEntryAuthorPerson = Jennifer
==========================
Available optional keywords:
0) No more optional keywords
1) XDSDocumentEntryClassCode
2) XDSDocumentEntryTypeCode
3) XDSDocumentEntryPracticeSettingCode
4) XDSDocumentEntryCreationTime
5) XDSDocumentEntryServiceStartTime
6) XDSDocumentEntryServiceStopTime
7) XDSDocumentEntryHealthcareFacilityTypeCode
8) XDSDocumentEntryEventCodeList
9) XDSDocumentEntryConfidentialityCode
10) XDSDocumentEntryAuthorPerson
11) XDSDocumentEntryFormatCode
12) XDSDocumentEntryType
#) Quit
Select keywords (specify number): 0
==========================
All keywords set...
==========================
Query type: FindDocuments
Query keywords:
$XDSDocumentEntryPatientId = 1234
$XDSDocumentEntryStatus = Approved
$XDSDocumentEntryAuthorPerson = Jennifer
==========================
```

Figure 4-30 The user chooses to start the query after input all known attributes

After the query has been sent, the XDS Document Consumer will wait for the response from XDS Document Registry Actor. When the response is received, the XDS Document Consumer then shows the metadata attributes the value of the query result in the terminal or just terminates the program if there is no matched result registered.

```
var hrstart = null;
var hrend = null;
var net = require('net');
var fs = require("fs");
var util = require("util");
var xml2js = require('xml2js');
var parseString = xml2js.parseString;
var builder = new xml2js.Builder;

const readline = require("readline");
const rl = readline.createInterface({
   input: process.stdin,
   output: process.stdout
});

var HOST = '127.0.0.1';
var PORT = 65519;
var client = new net.Socket();

var prepXDSAtt = {
     DocumentEntry: {
      author: [{
        authorPerson: 'N/A',
        authorInstitution: [],
        authorRole: 'N/A',
        authorSpecialty: 'N/A'
      }],
      availabilityStatus: 'N/A',
      classCode: {
        codingScheme: 'N/A',
        displayName: 'N/A'
      },
      comment: 'N/A',
      confidentialityCode: {
        codingScheme: 'N/A',
        displayName: 'N/A'
      },
      creationTime: 'N/A',
      entryUUID: 'N/A',
      eventCodeList: [],
      formatCode: {
        codingScheme: 'N/A',
        displayName: 'N/A'
      },
```

```
      hash: 'N/A',
      healthcareFacilityTypeCode: {
        codingScheme: 'N/A',
        displayName: 'N/A'
      },
      homeCommunityId: 'N/A',
      languageCode: 'N/A',
      legalAuthenticator: 'N/A',
      limitedMetadata: 'N/A',
      mimeType: 'N/A',
      objectType: 'N/A',
      patientId: 'N/A',
      practiceSettingCode: {
        codingScheme: 'N/A',
        displayName: 'N/A'
      },
      referenceIdList: 'N/A',
      repositoryUniqueId: 'N/A',
      serviceStartTime: 'N/A',
      serviceStopTime: 'N/A',
      size: 'N/A',
      sourcePatientId: 'N/A',
      sourcePatientInfo: [],
      title: 'N/A',
      typeCode: {
         codingScheme: 'N/A',
         displayName: 'N/A'
      },
      uniqueId: 'N/A',
      URI: 'N/A'
    }
  }
var documentEntryUUID = {
  //-----Document Entry----------------------------------
  DocumentEntry: 'urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1',
  author: 'urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d',
  classCode: 'urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a',
  confidentialityCode: 'urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f',
  formatCode: 'urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d',
  healthcareFacilityTypeCode: 'urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1',
  practiceSettingCode: 'urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead',
  typeCode: 'urn:uuid:f0306f51-975f-434e-a61c-c59651d33983',
  patientId: 'urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427',
  uniqueId: 'urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab',
  eventCodeList: 'urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4'
}
```

```
var submissionSetUUID = {
   //------SubmissionSet Attributes---------------------------------
   author: 'urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d',
   contentTypeCodes: 'urn:uuid:aa543740-bdda-424e-8c96-df4873be8500',
   uniqueId: 'urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8',
   sourceId: 'urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832',
   patientId: 'urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446',
   limitedMetadata: 'urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd'
}

var queryType = null;
var requiredKeywords = [];
var optionalKeywords = [];
var inputKeywords = [];

var keywordCount = 0;

var queryTypeList = ['', 'FindDocuments', 'FindSubmissionSets', 'FindFolders',
                                                'GetAll', 'GetDocuments', 'GetFolders', 'GetAssociations',
                                                'GetDocumentsAndAssociations', 'GetSubmissionSets',
                                                'GetSubmissionSetAndContents', 'GetFolderAndContents',
                                                'GetFoldersForDocument', 'GetRelatedDocuments',
'FindDocumentsByReferenceId'];
var queryTypeUUID = {
        FindDocuments: 'urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d',
        FindSubmissionSets: 'urn:uuid:f26abbcb-ac74-4422-8a30-edb644bbc1a9'
}



var timeKeyList = ['$XDSDocumentEntryCreationTimeFrom', '$XDSDocumentEntryCreationTimeTo',
                                                '$XDSDocumentEntryServiceStartTimeFrom',
'$XDSDocumentEntryServiceStartTimeTo',
                                                '$XDSDocumentEntryServiceStopTimeFrom',
'$XDSDocumentEntryServiceStartTimeTo'];
var availableKeywords = {
        FindDocuments: {
                required: ['XDSDocumentEntryPatientId', 'XDSDocumentEntryStatus'],
                optional: ['XDSDocumentEntryClassCode', 'XDSDocumentEntryTypeCode',
                                                'XDSDocumentEntryPracticeSettingCode',
'XDSDocumentEntryCreationTime',
                                                'XDSDocumentEntryServiceStartTime',
'XDSDocumentEntryServiceStopTime',
                                                'XDSDocumentEntryHealthcareFacilityTypeCode',
'XDSDocumentEntryEventCodeList',
                                                'XDSDocumentEntryConfidentialityCode',
'XDSDocumentEntryAuthorPerson',
                                                'XDSDocumentEntryFormatCode', 'XDSDocumentEntryType']
        },

}
```

```
var queryXML = {
            "query:AdhocQueryRequest": {
             "$": {
              "xmlns:query": "urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0",
              "xmlns:rim": "urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0",
              "xmlns:rs": "urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0",
              "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
              "xsi:schemaLocation": "urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0
../../schema/ebRS/query.xsd"
            },
            "query:ResponseOption": [
             {
              "$": {
               "returnComposedObjects": "true",
               "returnType": "LeafClass" //This should be determined by number of results
              }
             }
            ],
            "rim:AdhocQuery": [
             {
              "$": {
               "id": " urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d " //This is UUIDd of each query
type
              },
              "rim:Slot": [] //Query keyword
             }
            ]
           }
          }

function Main () {
        console.log('\n==================================');
        console.log('|| XDS Consumer Actor Interface ||');
        console.log('==================================');
        getQueryType();
}

function getQueryType () {
        console.log('Please select query type');
        console.log('1) FindDocuments');
        console.log('2) FindSubmissionSets');
        console.log('3) FindFolders');
        console.log('4) GetAll');
        console.log('5) GetDocuments');
        console.log('6) GetFolders');
        console.log('7) GetAssociations');
        console.log('8) GetDocumentsAndAssociations');
```

```
        console.log('9) GetSubmissionSets');
        console.log('10) GetSubmissionSetAndContents');
        console.log('11) GetFolderAndContents');
        console.log('12) GetFoldersForDocument');
        console.log('13) GetRelatedDocuments');
        console.log('14) FindDocumentsByReferenceId');
        console.log('#) Quit');
        rl.question("(Specify number): ", function(queryTypeInput) {
                var queryTypeInteger = parseInt(queryTypeInput, 10);
                if (queryTypeInput && queryTypeInteger){
                        queryType = queryTypeList[queryTypeInteger];
                        console.log('Query Type: ' + queryType + '\n');
                        requiredKeywords = availableKeywords[queryType]['required'];
                        optionalKeywords = availableKeywords[queryType]['optional'];
                }
                else if (queryTypeInput == '#' || queryTypeInput == 'quit' || queryTypeInput == 'Quit'){
                        console.log('Quit...');
                        process.exit();
                }
                else {
                        console.log('Error');
                        process.exit();
                }
                inputKeywords.push(queryType);
                keywordCount = 0;
                getRequiredKeywords();
        });
}

function getRequiredKeywords () {
        console.log('Keywords require: ' + requiredKeywords[keywordCount]);
        var addedHeader = '$' + requiredKeywords[keywordCount];
        rl.question('Value: ', function(requireKeyInput) {
                inputKeywords.push([addedHeader, requireKeyInput]);
                keywordCount++;
                if (keywordCount >= requiredKeywords.length){
                        showAllKeywords();
                        getOptionalKeywords();
                }
                else {
                        getRequiredKeywords();
                }
        });
};
```

```
function getOptionalKeywords () {
	console.log('Available optional keywords: ');
	console.log('0) No more optional keywords');
	for (i = 0; i < optionalKeywords.length; i++){ //Show all available optional keywords
		var count = i+1;
		console.log(count + ') ' + optionalKeywords[i]);
	}
	console.log('#) Quit')
	rl.question('Select keywords (specify number): ', function(selectedOpt) { //Promt user for
optional keyword by specifying number
		if (selectedOpt == '#'){ //'#' Mark as program terminate
			process.exit();
		}
		else if (selectedOpt == '0') { //'0' Mark as user approve that all known keywords included
			console.log('==========================');
			console.log('All keywords set...');
			showAllKeywords();
			createXML();
			return rl.close();
		}
		else { //Otherwise
			var selectedOpt = parseInt(selectedOpt, 10);
			var optionMarker = selectedOpt - 1;
			console.log('Keyword: ' + optionalKeywords[optionMarker]);
			if (selectedOpt && optionMarker >= 0 && optionMarker <
optionalKeywords.length){ //Check if user input is a number and the number is in available range
				var selectedOptKeywords = '$' + optionalKeywords[optionMarker];
				var replicateCheck = null;
				var timeKeyCheck = null;
				for (j = 1; j < inputKeywords.length; j++){ //Check for any replicated
keyword specified
					if (inputKeywords[j][0] == selectedOptKeywords ||
inputKeywords[j][0] == selectedOptKeywords + 'From'){
						replicateCheck = 1;
						var replicatedKeywordPos = j;
					}
					if (timeKeyList.includes(selectedOptKeywords + 'From')){
						timeKeyCheck = 1;
					}
				}
				if (replicateCheck && !timeKeyCheck){ //If found any relicated keyword
then ask if user want to replace the value && the keyword is not a time keyword
					console.log('Query keywords set already contain ' +
selectedOptKeywords);
					rl.question('Overwrite the keyword? (y/n): ',
function(overwriteConfirm) {
						if (overwriteConfirm == 'y' || overwriteConfirm == 'Y' ||
overwriteConfirm == 'yes' || overwriteConfirm == 'Yes'){ //Ask for user to specify yes or else
```

```
                                                rl.question('Replace with value: ',
function(optionalKeyInput) {
                                                        if
(inputKeywords[replicatedKeywordPos][0] == selectedOptKeywords){ //Second check if the keyword
really replicated

        inputKeywords[replicatedKeywordPos][1] = optionalKeyInput;
                                                                showAllKeywords();
                                                                getOptionalKeywords();
                                                        }
                                                });
                                        }
                                        else { //If user not confirm on overwrite the keyword,
just skip overwriting
                                                console.log('Overwrite cancelled...');
                                                showAllKeywords();
                                                getOptionalKeywords();
                                        }
                                }});
                        }
                        else if (replicateCheck && timeKeyCheck){ //If found any relicated
keyword then ask if user want to replace the value && the keyword is a time keyword
                                console.log('Query keywords set already contain ' +
selectedOptKeywords);
                                rl.question('Overwrite the keyword? (y/n): ',
function(overwriteConfirm) {
                                        if (overwriteConfirm == 'y' || overwriteConfirm == 'Y' ||
overwriteConfirm == 'yes' || overwriteConfirm == 'Yes'){ //Ask for user to specify yes or else
                                                rl.question('Replace time value from
(YYYYMMDDhhmmss): ', function(optionalKeyInputFrom) {
                                                        rl.question('Replace time value to
(YYYYMMDDhhmmss): ', function(optionalKeyInputTo) {

        inputKeywords[replicatedKeywordPos][1] = optionalKeyInputFrom;

        inputKeywords[replicatedKeywordPos + 1][1] = optionalKeyInputTo;
                                                                showAllKeywords();
                                                                getOptionalKeywords();
                                                        });
                                                });
                                        }
                                        else { //If user not confirm on overwrite the keyword,
just skip overwriting
                                                console.log('Overwrite cancelled...');
                                                showAllKeywords();
                                                getOptionalKeywords();
                                        }
                                }});
```

```
                                        }
                                        else if (!replicateCheck && timeKeyCheck){ //If non of any replicated
were found, then add more keyword into query set && the keyword is a time keyword
                                                rl.question('Time value from (YYYYMMDDhhmmss): ',
function(optionalKeyInputFrom) {
                                                        rl.question('Time value to (YYYYMMDDhhmmss): ',
function(optionalKeyInputTo) {
                                                                inputKeywords.push([selectedOptKeywords +
'From', optionalKeyInputFrom]);
                                                                inputKeywords.push([selectedOptKeywords +
'To', optionalKeyInputTo]);
                                                                showAllKeywords();
                                                                getOptionalKeywords();
                                                        });
                                                });
                                        }
                                        else { //If non of any replicated were found, then add more keyword
into query set
                                                rl.question('Value: ', function(optionalKeyInput) {
                                                        inputKeywords.push([selectedOptKeywords,
optionalKeyInput]);
                                                        showAllKeywords();
                                                        getOptionalKeywords();
                                                });
                                        }
                                }
                                else { //If user try to input anything that not available, force to try again
                                        console.log('Error, try again...');
                                        getOptionalKeywords();
                                }
                        }
                });
}

function showAllKeywords () {
        console.log('==========================');
        console.log('Query type: ' + inputKeywords[0]);
        console.log('Query keywords: ');
        for (i = 1; i < inputKeywords.length; i++){
                console.log(inputKeywords[i][0] + ' = ' + inputKeywords[i][1]);
        }
        console.log('==========================');
}

function createXML () { //Assort keywords into ITI-18 XML format

        queryXML['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['$']['id'] =
queryTypeUUID[inputKeywords[0]];
```

```
        var slot = queryXML['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['rim:Slot'];

        for (i = 1; i < inputKeywords.length; i++){
                var rimSlot = {
            "$": {
                "name": inputKeywords[i][0]
            },
            "rim:ValueList": [
                {
                        "rim:Value": [
                          inputKeywords[i][1]
                        ]
                }
            ]
          }

          slot.push(rimSlot);
        }

        sendQuery();
}

function sendQuery () {
        client.connect(PORT, HOST, function() {
    console.log('CONNECTED TO: ' + HOST + ':' + PORT);
    // Write a message to the socket as soon as the client is connected, the server will receive it as
message from the client
    var queryXMLrebuilt = builder.buildObject(queryXML);
    hrstart = process.hrtime();
    client.write(queryXMLrebuilt);
    console.log('Query Sent...');
  });

  // Add a 'data' event handler for the client socket
  // data is what the server sent to this socket
  client.on('data', function(data) {
    // Close the client socket completely
    if (data.includes('ACK from ')){
      console.log('Respond received: ' + data);
      hrend = process.hrtime(hrstart);
      console.info('Execution time (hr): %ds %dms', hrend[0], hrend[1] / 1000000);
    }
    else {
      console.log('==============================\nQuery response received: ');
      var dataIn = data.toString();
      parseString(dataIn, function (err, result) {
        var eventCodeListCount = 0;
        if (err) throw err;
```

```
            var bodyExtrinsicObject =
result['query:AdhocQueryResponse']['rim:RegistryObjectList'][0]['rim:ExtrinsicObject'][0];
            if (bodyExtrinsicObject['$']['objectType'] == documentEntryUUID.DocumentEntry){
              //Scanning object within DocumentEntry "Classification"
              if (bodyExtrinsicObject['$']['id']){
                prepXDSAtt.DocumentEntry.entryUUID = bodyExtrinsicObject['$']['id'];
              }
              if (bodyExtrinsicObject['$']['mimeType']){
                prepXDSAtt.DocumentEntry.mimeType = bodyExtrinsicObject['$']['mimeType'];
              }
              if (bodyExtrinsicObject['$']['objectType']){
                prepXDSAtt.DocumentEntry.objectType = bodyExtrinsicObject['$']['objectType'];
              }
              if (bodyExtrinsicObject['$']['status']){
                                        prepXDSAtt.DocumentEntry.availabilityStatus =
bodyExtrinsicObject['$']['status'];
                                  }
              for (var i = 0; i < bodyExtrinsicObject['rim:Classification'].length; i++){
                 //Detect DocumentEntry > author (Set)
                 if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.author){
                    if (i != 0) { //If there are more than one author for the Doc, add more author object into
array
                       prepXDSAtt.DocumentEntry.author.push({
                                                  authorPerson: 'N/A',
                                                  authorInstitution: [],
                                                  authorRole: 'N/A',
                                                  authorSpecialty: 'N/A'
                                               });
                    }
                    //Assign each element of the author
                    for (var j = 0; j < bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'].length; j++){
                       if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorPerson'){
                          prepXDSAtt.DocumentEntry.author[i].authorPerson =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                       }
                       if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorInstitution'){
                          prepXDSAtt.DocumentEntry.author[i].authorInstitution =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'];
                       }
                       if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] == 'authorRole'){
                          prepXDSAtt.DocumentEntry.author[i].authorRole =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                       }
                       if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorSpecialty'){
```

```
                    prepXDSAtt.DocumentEntry.author[i].authorSpecialty =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                }
            }
        }
        //Detect DocumentEntry > classCode
        if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.classCode){
            prepXDSAtt.DocumentEntry.classCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
            if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                prepXDSAtt.DocumentEntry.classCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
            }
        }
        //Detect DocumentEntry > confidentialityCode
        if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.confidentialityCode){
            prepXDSAtt.DocumentEntry.confidentialityCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
            if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                prepXDSAtt.DocumentEntry.confidentialityCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
            }
        }
        //Detect DocumentEntry > formatCode
        if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.formatCode){
            prepXDSAtt.DocumentEntry.formatCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
            if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                prepXDSAtt.DocumentEntry.formatCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
            }
        }
        //Detect DocumentEntry > healthcareFacilityTypeCode
        if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.healthcareFacilityTypeCode){
            prepXDSAtt.DocumentEntry.healthcareFacilityTypeCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
            if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                prepXDSAtt.DocumentEntry.healthcareFacilityTypeCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
            }
```

```
            }
            //Detect DocumentEntry > practiceSettingCode
            if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.practiceSettingCode){
              prepXDSAtt.DocumentEntry.practiceSettingCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
              if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                prepXDSAtt.DocumentEntry.practiceSettingCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
              }
            }
            //Detect DocumentEntry > eventCode
            if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.eventCodeList){
              prepXDSAtt.DocumentEntry.eventCodeList.push({
                                        codingScheme: 'N/A',
                                        displayName: 'N/A'
                                      });
              prepXDSAtt.DocumentEntry.eventCodeList[eventCodeListCount].displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
              if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                prepXDSAtt.DocumentEntry.eventCodeList[eventCodeListCount].codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
              }
              eventCodeListCount++;
            }
            //Detect DocumentEntry > TypeCode
            if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.typeCode){
              prepXDSAtt.DocumentEntry.typeCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
              if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                prepXDSAtt.DocumentEntry.typeCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
              }
            }
          }
          //Scanning object within DocumentEntry "Descriptor" which usually be "comment"
          for (var i = 0; i < bodyExtrinsicObject['rim:Description'].length; i++){
            prepXDSAtt.DocumentEntry.comment = bodyExtrinsicObject['rim:Description'][i];
          }
          for (var i = 0; i < bodyExtrinsicObject['rim:Name'].length; i++){
            prepXDSAtt.DocumentEntry.title =
bodyExtrinsicObject['rim:Name'][i]['rim:LocalizedString'][0]['$']['value'];
          }
```

```
            //Scanning object within DocumentEntry "ExternalIdentifier"
            for (var i = 0; i < bodyExtrinsicObject['rim:ExternalIdentifier'].length; i++){
              //Detect DocumentEntry > patientId
              if (bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['identificationScheme'] ==
documentEntryUUID.patientId){
                prepXDSAtt.DocumentEntry.patientId =
(bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['value']);
              }
              if (bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['identificationScheme'] ==
documentEntryUUID.uniqueId){
                prepXDSAtt.DocumentEntry.uniqueId =
(bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['value']);
              }
            }
            //Scannig object within DocumentEntry "Slot"
            for (var i = 0; i < bodyExtrinsicObject['rim:Slot'].length; i++){
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'size'){
                prepXDSAtt.DocumentEntry.size =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'repositoryUniqueId'){
                prepXDSAtt.DocumentEntry.repositoryUniqueId =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'hash'){
                prepXDSAtt.DocumentEntry.hash =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'creationTime'){
                prepXDSAtt.DocumentEntry.creationTime =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'languageCode'){
                prepXDSAtt.DocumentEntry.languageCode =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'serviceStartTime'){
                prepXDSAtt.DocumentEntry.serviceStartTime =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'serviceStopTime'){
                prepXDSAtt.DocumentEntry.serviceStopTime =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
              if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'sourcePatientId'){
                prepXDSAtt.DocumentEntry.sourcePatientId =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
              }
```

```
                if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'sourcePatientInfo'){
                  prepXDSAtt.DocumentEntry.sourcePatientInfo =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'];;
                }
              }
            }
            console.log(util.inspect(prepXDSAtt));
            hrend = process.hrtime(hrstart);
            console.log('==============================');
           console.info('Execution time (hr): %ds %dms', hrend[0], hrend[1] / 1000000);
           console.log('==============================');
         });
         client.destroy();
         rl.close();
      }
   });

   // Add a 'close' event handler for the client socket
   client.on('close', function() {
      console.log('Connection closed');
   });
}

Main();
```

Figure 4-31 Javascript Code Snippet of XDS Document Consumer Actor

4.2.3 XDS Document Registry Actor

Following Section 3.5.1, the XDS Document Registry Actor program must be able to communicate with the simulated XDS Document Repository actor and XDS Document Consumer actor. At the same time, the software will need to act as the middle between the local XDS system and the Blockchain ledger. The completed process flow of the XDS Document Registry Actor is shown in pseudocode Figure 4-32. The Actor will wait until receiving the XML message transaction and react differently to ITI-42 and ITI-18 transactions.

```
Main() {
        while (True) {
                var ReceivedMessage = Wait_For_XMLMessage();
                if (ReceivedMessage == "ITI-42"){

                        var SmartcontractCompatible = Received_ITI_42(XMLMessage)

                        var registerStatus = DocumentRegistering_Into_Blockchain(SmartcontractCompatible);

                        Send_Response_To_XDSDocumentRepositoryActor(registerStatus);
                }
                else if (ReceivedMessage == "ITI-18"){

                        var SearchKeywords = Received_ITI_18(XMLMessage);

                        var SearchResult = DocumentSearch_Within_Blockchain(SearchKeywords);

                        var ITI_18_Response = SortResult_Into_ITI18ResponseFormat(SearchResult);
                        Send_Response_To_XDSDocumentConsumerActor(ITI_18_Response);
                }
        }
}
```

Figure 4-32 Pseudocode showing the process flow of the XDS Document Registry Actor

Unlike the XDS Document Repository Actor and XDS Document Consumer Actor, the XDS Document Registry was made to act as the medium between the native system and the Blockchain Smartcontract. This section will break down the component of the XDS Document Registry Actor in a pattern different from the rest where it separated by its main function interacting with health document metadata which included Document Registering and Document Search. Each section will break down into Javascript native program part and Smartcontract part.

4.2.3.1 Implementing Document Register Function

4.2.3.1.1 Native-Side Javascript Program

For this implementation, the XDS Document Registry Actor will open a TCP connection to receive the transaction on a specified port. Upon receiving the ITI-42 transaction, the Actor then converts the XML message into JSON using xml2js. When ITI-42 was interpreted into JSON, the actor then assorts the object and sorts it into the Smartcontract compatible format. Figure 4-33 is the pseudocode showing the process flow for the Javascript program handling the ITI-42 transaction until converted into the Smartcontract compatible format. Figure 4-34 to Figure 4-39 showing the Javascript code snippet of XDS Document Registry which is the part dealing with ITI-42 transaction and Document Registering Function.

```
var Received_ITI_42(XMLMessage){
        var JSON_attributes = InterpretXMLtoJSON(XMLMessage);
        var Assorted_JSON = AssortMetadataAttributes(JSON_attributes);
        var SmartcontractCompatible = SortInto_SmartcontractCompatibleFormat(Assorted_JSON);
        return SmartcontractCompatible;
}
```

Figure 4-33 The pseudocode showing the process flow of XDS Document Registry Actor for Document Registering Function

```
var hrstart = null;
var hrend = null;
var fs = require("fs");
var net = require('net');
var util = require("util");
var xml2js = require('xml2js');
var parseString = xml2js.parseString;
var builder = new xml2js.Builder;
var moment = require('moment');
const Cryptr = require('cryptr');
const cryptr = new Cryptr('XDSDomainSharedSecretKey');

var Web3 =  require('web3');
var web3 = new Web3("qdata/dd1/geth.ipc", net);

var HOST = '127.0.0.1';
var PORT = 65519;

var netServer = null;
var netSocket = null;


//Net socket wait for any messages====================
netServer = net.createServer(function(sock) {
	netSocket = sock;
	// We have a connection - a socket object is assigned to the connection automatically
	console.log('CONNECTED: ' + sock.remoteAddress +':'+ sock.remotePort);

	// Add a 'data' event handler to this instance of socket
	sock.on('data', function(data) {
		console.log('Received data....');
		hrstart = process.hrtime();
		sock.write('ACK from ' + sock.remoteAddress + '\n'); //Write ACK back to sender
		processData(data); //converting XML to JSON based on Module "xml-js"
		//console.log(result); // show the result of xml to js conversion
	});
	// Add a 'close' event handler to this instance of socket
	sock.on('close', function(data) {
		console.log('CLOSED: ' + sock.remoteAddress +' '+ sock.remotePort);
	});
}).listen(PORT, HOST);
console.log('XDS Document Registry Actor listening on ' + HOST +':'+ PORT);
```

Figure 4-34 Javascript Code Snippet of XDS Document Registry Actor

Node Module import declaration and TCP Socket message receiver section

```
//ProcessData interprete any xmlMessages came through Netsocket ====================
function processData (dataIn) {
  console.log('XML:\n' + dataIn);
  parseString(dataIn, function (err, result) {
    if (err) throw err;
    console.log('\nConverted to object: ');
    console.log('----------------------\n' + util.inspect(result) + '\n--------------------');
    /*
    fs.writeFile("queryReceived.json", stringXDSAttrib, function(err, data) {
      if (err) console.log(err);
      console.log("Successfully Written to File. ");
    });
    */
    if (Object.keys(result)[0] == 'query:AdhocQueryRequest') {
      console.log('Query requested...');
      documentQuery(result);
    }
    else{
            if (Object.keys(result)[0] == 'soapenv:Envelope'){
                    if (result['soapenv:Envelope']['soapenv:Header'][0]['wsa:Action'][0]['_'] ==
'urn:ihe:iti:2007:RegisterDocumentSet-b'){
                            console.log('RegisterDocumentSet-b...');
                            registerDocumentSetb(result);
                    }
            }
    }
  });
}
```

Figure 4-35 XDS Document Registry Actor

This section checks if receiving message is ITI-42 or ITI-18 identified by its header

```
//RegisterDocumentSet-b
//----------------------------------------------
//Declare main object to store all META-data attributes essential for search operation
function registerDocumentSetb (inputAttributes) {
        var prepXDSAtt = {
          DocumentEntry: {
            author: [{
              authorPerson: 'N/A',
              authorInstitution: [],
              authorRole: 'N/A',
              authorSpecialty: 'N/A'
            }],
            availabilityStatus: 'N/A',
            classCode: {
              codingScheme: 'N/A',
              displayName: 'N/A'
            },
            comment: 'N/A',
            confidentialityCode: {
              codingScheme: 'N/A',
              displayName: 'N/A'
            },
            creationTime: 'N/A',
            entryUUID: 'N/A',
            eventCodeList: [],
            formatCode: {
              codingScheme: 'N/A',
              displayName: 'N/A'
            },
            hash: 'N/A',
            healthcareFacilityTypeCode: {
              codingScheme: 'N/A',
              displayName: 'N/A'
            },
            homeCommunityId: 'N/A',
            languageCode: 'N/A',
            legalAuthenticator: 'N/A',
            limitedMetadata: 'N/A',
            mimeType: 'N/A',
            objectType: 'N/A',
            patientId: 'N/A',
            practiceSettingCode: {
              codingScheme: 'N/A',
              displayName: 'N/A'
            },
```

```
  referenceIdList: 'N/A',
  repositoryUniqueId: 'N/A',
  serviceStartTime: 'N/A',
  serviceStopTime: 'N/A',
  size: 'N/A',
  sourcePatientId: 'N/A',
  sourcePatientInfo: [],
  title: 'N/A',
  typeCode: {
     codingScheme: 'N/A',
     displayName: 'N/A'
  },
  uniqueId: 'N/A',
  URI: 'N/A'
 },
 SubmissionSet: {
  author: [{
     authorPerson: 'N/A',
     authorInstitution: [],
     authorRole: 'N/A',
     authorSpecialty: 'N/A'
   }],
  availabilityStatus: 'N/A',
  comments: 'N/A',
  contentTypeCodes: {
   codingScheme: 'N/A',
   displayName: 'N/A'
  },
  entryUUID: 'N/A',
  homeCommunityId: 'N/A',
  intendedRecipient: 'N/A',
  limitedMetadata: 0,
  patientId: 'N/A',
  sourceId: 'N/A',
  submissionTime: 'N/A',
  title: 'N/A',
  uniqueId: 'N/A'
 },
 Folder: {
  availabilityStatus: 'N/A',
  codeList: 'N/A',
  comments: 'N/A',
  entryUUID: 'N/A',
  homeCommunityId: 'N/A',
  lastUpdateTime: 'N/A',
  limitedMetadata: 'N/A',
  patientId: 'N/A',
  title: 'N/A',
  uniqueId: 'N/A'
 },

 Association: {
  associationType: 'N/A',
  sourceObject: 'N/A',
  targetObject: 'N/A',
  SubmissionSetStatus: 'N/A'
 }
}
```

Figure 4-36 XDS Document Registry Actor

Declaration of JSON variable to store all Metadata attributes by its position in the format

```
//Define UUID number of each META-data attributes
var documentEntryUUID = {
   //-----Document Entry---------------------------------
   DocumentEntry: 'urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1',
   author: 'urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d',
   classCode: 'urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a',
   confidentialityCode: 'urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f',
   formatCode: 'urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d',
   healthcareFacilityTypeCode: 'urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1',
   practiceSettingCode: 'urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead',
   typeCode: 'urn:uuid:f0306f51-975f-434e-a61c-c59651d33983',
   patientId: 'urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427',
   uniqueId: 'urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab',
   eventCodeList: 'urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4'
}

var submissionSetUUID = {
   //------SubmissionSet Attributes--------------------------------
   author: 'urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d',
   contentTypeCodes: 'urn:uuid:aa543740-bdda-424e-8c96-df4873be8500',
   uniqueId: 'urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8',
   sourceId: 'urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832',
   patientId: 'urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446',
   limitedMetadata: 'urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd'
}
```

Figure 4-37 XDS Document Registry Actor

Define variable of each Metadata attribute UUID label following IHE ITI Framework

```
function assignAll (rXDSAttribute, myCallback) {
  //Define variable for shorter object accessing
  var sEnvelope = rXDSAttribute['soapenv:Envelope'];
  //inside Enveope
  var s$ = sEnvelope['$'];
  var sBody = sEnvelope['soapenv:Body'][0];
  var sHeader = sEnvelope['soapenv:Header'][0];
  //inside Envelope>Header
  var wsaTo = sHeader['wsa:To'];
  var wsaMessageID = sHeader['wsaMessageID'];
  var wsaAction = sHeader['wsaAction'];
  //inside Envelope>Body
  var lcmSubmitObjectsRequest = sBody['lcm:SubmitObjectsRequest'][0];
  //inside Envelope>Body>lcm:SubmitObjectsRequest
  var bodyRegistryObjectList = lcmSubmitObjectsRequest['rim:RegistryObjectList'][0];
  //inside Envelope>Body>lcm:SubmitObjectsRequest>rim:RegistryObjectList
  var bodyExtrinsicObject = bodyRegistryObjectList['rim:ExtrinsicObject'][0];
  var bodyRegistryPackage = bodyRegistryObjectList['rim:RegistryPackage'][0];
  var bodyClassification = bodyRegistryObjectList['rim:Classification'][0];
  var bodyAssociation = bodyRegistryObjectList['rim:Association'][0];

  var eventCodeListCount = 0; //Document may have more than one eventCodeList, so it need
counter


  //Detect DocumentEntry
  if (bodyExtrinsicObject['$']['objectType'] == documentEntryUUID.DocumentEntry){
    //Scanning object within DocumentEntry "Classification"
    if (bodyExtrinsicObject['$']['id']){
        prepXDSAtt.DocumentEntry.entryUUID = bodyExtrinsicObject['$']['id'];
    }
    if (bodyExtrinsicObject['$']['mimeType']){
        prepXDSAtt.DocumentEntry.mimeType = bodyExtrinsicObject['$']['mimeType'];
    }
    if (bodyExtrinsicObject['$']['objectType']){
        prepXDSAtt.DocumentEntry.objectType = bodyExtrinsicObject['$']['objectType'];
    }
    if (bodyExtrinsicObject['$']['status']){
        prepXDSAtt.DocumentEntry.availabilityStatus = bodyExtrinsicObject['$']['status'];
    }
```

```
            for (var i = 0; i < bodyExtrinsicObject['rim:Classification'].length; i++){
              //Detect DocumentEntry > author (Set)
              if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.author){
                if (i != 0) { //If there are more than one author for the Doc, add more author object into
array
                  prepXDSAtt.DocumentEntry.author.push({
                                    authorPerson: 'N/A',
                                    authorInstitution: [],
                                    authorRole: 'N/A',
                                    authorSpecialty: 'N/A'
                                  });
                }
                //Assign each element of the author
                for (var j = 0; j < bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'].length; j++){
                  if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorPerson'){
                    prepXDSAtt.DocumentEntry.author[i].authorPerson =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                  }
                  if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorInstitution'){
                    prepXDSAtt.DocumentEntry.author[i].authorInstitution =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'];
                  }
                  if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] == 'authorRole'){
                    prepXDSAtt.DocumentEntry.author[i].authorRole =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                  }

                  if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorSpecialty'){
                    prepXDSAtt.DocumentEntry.author[i].authorSpecialty =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                  }
                }
              }
              //Detect DocumentEntry > classCode
              if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.classCode){
                prepXDSAtt.DocumentEntry.classCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                  prepXDSAtt.DocumentEntry.classCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                } }
```

```
                //Detect DocumentEntry > confidentialityCode
                if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.confidentialityCode){
                    prepXDSAtt.DocumentEntry.confidentialityCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                    if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                        prepXDSAtt.DocumentEntry.confidentialityCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                    }
                }
                //Detect DocumentEntry > formatCode
                if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.formatCode){
                    prepXDSAtt.DocumentEntry.formatCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                    if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                        prepXDSAtt.DocumentEntry.formatCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                    }
                }
                //Detect DocumentEntry > healthcareFacilityTypeCode
                if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.healthcareFacilityTypeCode){
                    prepXDSAtt.DocumentEntry.healthcareFacilityTypeCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                    if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                        prepXDSAtt.DocumentEntry.healthcareFacilityTypeCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                    }
                }
                //Detect DocumentEntry > practiceSettingCode
                if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.practiceSettingCode){
                    prepXDSAtt.DocumentEntry.practiceSettingCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                    if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                        prepXDSAtt.DocumentEntry.practiceSettingCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                    }
                }
```

```
                //Detect DocumentEntry > eventCode
                if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.eventCodeList){
                  prepXDSAtt.DocumentEntry.eventCodeList.push({
                                        codingScheme: 'N/A',
                                        displayName: 'N/A'
                                      });
                  prepXDSAtt.DocumentEntry.eventCodeList[eventCodeListCount].displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                  if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                    prepXDSAtt.DocumentEntry.eventCodeList[eventCodeListCount].codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                  }
                  eventCodeListCount++;
                }
                //Detect DocumentEntry > TypeCode
                if (bodyExtrinsicObject['rim:Classification'][i]['$']['classificationScheme'] ==
documentEntryUUID.typeCode){
                  prepXDSAtt.DocumentEntry.typeCode.displayName =
bodyExtrinsicObject['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                  if (bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                    prepXDSAtt.DocumentEntry.typeCode.codingScheme =
bodyExtrinsicObject['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                  }
                }
              }

              //Scanning object within DocumentEntry "Descriptor" which usually be "comment"
              for (var i = 0; i < bodyExtrinsicObject['rim:Description'].length; i++){
                prepXDSAtt.DocumentEntry.comment = bodyExtrinsicObject['rim:Description'][i];
              }

              for (var i = 0; i < bodyExtrinsicObject['rim:Name'].length; i++){
                prepXDSAtt.DocumentEntry.title =
bodyExtrinsicObject['rim:Name'][i]['rim:LocalizedString'][0]['$']['value'];
              }
```

```
//Scanning object within DocumentEntry "ExternalIdentifier"
for (var i = 0; i < bodyExtrinsicObject['rim:ExternalIdentifier'].length; i++){
  //Detect DocumentEntry > patientId
  if (bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['identificationScheme'] ==
documentEntryUUID.patientId){
    prepXDSAtt.DocumentEntry.patientId =
(bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['value']);
  }
  if (bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['identificationScheme'] ==
documentEntryUUID.uniqueId){
    prepXDSAtt.DocumentEntry.uniqueId =
(bodyExtrinsicObject['rim:ExternalIdentifier'][i]['$']['value']);
  }
}
//Scannig object within DocumentEntry "Slot"
for (var i = 0; i < bodyExtrinsicObject['rim:Slot'].length; i++){
  if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'size'){
    prepXDSAtt.DocumentEntry.size =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
  }
  if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'repositoryUniqueId'){
    prepXDSAtt.DocumentEntry.repositoryUniqueId =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
  }
  if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'hash'){
    prepXDSAtt.DocumentEntry.hash =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
  }
  if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'creationTime'){
    prepXDSAtt.DocumentEntry.creationTime =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
  }
  if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'languageCode'){
    prepXDSAtt.DocumentEntry.languageCode =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
  }
  if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'serviceStartTime'){
    prepXDSAtt.DocumentEntry.serviceStartTime =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
  }
  if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'serviceStopTime'){
    prepXDSAtt.DocumentEntry.serviceStopTime =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
  }
```

```
                if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'sourcePatientId'){
                  prepXDSAtt.DocumentEntry.sourcePatientId =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
                }
                if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'sourcePatientInfo'){
                  var plaintext = bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'];
                  var encryptedString = cryptr.encrypt(plaintext);
        prepXDSAtt.DocumentEntry.sourcePatientInfo = 'Anonymized';
        //Also replace the attributes within full XDSAttributes object with encrypted attribute

        rXDSAttribute['soapenv:Envelope']['soapenv:Body'][0]['lcm:SubmitObjectsRequest'][0]['rim:Regi
stryObjectList'][0]['rim:ExtrinsicObject'][0]['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'] = encryptedString;
                }
              }
            }

            if (bodyRegistryPackage){
              for (var i = 0; i < bodyRegistryPackage['rim:Classification'].length; i++){
                if (bodyRegistryPackage['rim:Classification'][i]['$']['classificationScheme'] ==
submissionSetUUID.author){
                  if (i != 0) { //If there are more than one author for the Doc, add more author object into
array
                    prepXDSAtt.SubmissionSet.author.push({
                                    authorPerson: 'N/A',
                                    authorInstitution: [],
                                    authorRole: 'N/A',
                                    authorSpecialty: 'N/A'
                                  });
                  }
                  //Assign each element of the author
                  for (var j = 0; j < bodyRegistryPackage['rim:Classification'][i]['rim:Slot'].length; j++){
                    if (bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorPerson'){
                      prepXDSAtt.SubmissionSet.author[i].authorPerson =
bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                    }

                    if (bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorInstitution'){
                      prepXDSAtt.SubmissionSet.author[i].authorInstitution =
bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'];
                    }
```

```
                    if (bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorRole'){
                        prepXDSAtt.SubmissionSet.author[i].authorRole =
bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                    }
                    if (bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['$']['name'] ==
'authorSpecialty'){
                        prepXDSAtt.SubmissionSet.author[i].authorSpecialty =
bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][j]['rim:ValueList'][0]['rim:Value'][0];
                    }
                }
            }

            if (bodyRegistryPackage['rim:Classification'][i]['$']['classificationScheme'] ==
submissionSetUUID.contentTypeCodes){
                prepXDSAtt.SubmissionSet.contentTypeCodes.displayName =
bodyRegistryPackage['rim:Classification'][i]['rim:Name'][0]['rim:LocalizedString'][0]['$']['value'];
                if (bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][0]['$']['name'] ==
'codingScheme'){
                    prepXDSAtt.SubmissionSet.contentTypeCodes.codingScheme =
bodyRegistryPackage['rim:Classification'][i]['rim:Slot'][0]['rim:ValueList'][0]['rim:Value'][0];
                }
            }
        }

        for (var i = 0; i < bodyRegistryPackage['rim:Description'].length; i++){
            prepXDSAtt.SubmissionSet.comment =
bodyRegistryPackage['rim:Description'][i]['rim:LocalizedString'][0]['$']['value'];
        }

        for (var i = 0; i < bodyRegistryPackage['rim:Name'].length; i++){
            prepXDSAtt.SubmissionSet.title =
bodyRegistryPackage['rim:Name'][i]['rim:LocalizedString'][0]['$']['value'];
        }

        for (var i = 0; i < bodyRegistryPackage['rim:ExternalIdentifier'].length; i++){
            if (bodyRegistryPackage['rim:ExternalIdentifier'][i]['$']['identificationScheme'] ==
submissionSetUUID.uniqueId){
                prepXDSAtt.SubmissionSet.uniqueId =
bodyRegistryPackage['rim:ExternalIdentifier'][i]['$']['value'];
            }
            if (bodyRegistryPackage['rim:ExternalIdentifier'][i]['$']['identificationScheme'] ==
submissionSetUUID.sourceId){
                prepXDSAtt.SubmissionSet.sourceId =
bodyRegistryPackage['rim:ExternalIdentifier'][i]['$']['value'];
            }
```

```
                if (bodyRegistryPackage['rim:ExternalIdentifier'][i]['$']['identificationScheme'] ==
submissionSetUUID.patientId){
                    prepXDSAtt.SubmissionSet.patientId =
bodyRegistryPackage['rim:ExternalIdentifier'][i]['$']['value'];
                }
            }

            for (var i = 0; i < bodyRegistryPackage['rim:Slot'].length; i++){
                if (bodyRegistryPackage['rim:Slot'][i]['$']['name'] == 'submissionTime'){
                    prepXDSAtt.SubmissionSet.submissionTime =
bodyRegistryPackage['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'][0];
                }
            }
        }
        console.log('--------------------------------');
        console.log(prepXDSAtt);
        //fs.writeFileSync("extractedObject", util.inspect(prepXDSAtt));
        myCallback(prepXDSAtt, rXDSAttribute);
    }

    assignAll(inputAttributes, invokeContract);
}
```

Figure 4-38 XDS Document Registry Actor

This section interprets and assort Metadata attribute value from ITI-42 to JSON

```
//-----------------------Smartcontract interact function
async function invokeContract(XDSMETADataAttributes, rawXDSAttr){
 //web3.eth.defaultAccount = web3.eth.personal.getAccounts().then(console.log);
 let acc = await web3.eth.personal.getAccounts();
  if (acc.err) {console.log(acc.err);}
  else {console.log('Accounts available on this node:\n' + acc);}

 console.log('-------------------------------------------');
 var deployerAccount = acc[0];
 console.log('Deploying with account:' + deployerAccount);
 var abi =
 [
  {
   "inputs": [],
   "name": "checkLastID",
   "outputs": [
    {
     "internalType": "uint256",
     "name": "",
     "type": "uint256"
    }
   ],
   "stateMutability": "view",
   "type": "function"
  },
  {
   "inputs": [
    {
     "internalType": "uint256",
     "name": "Docid",
     "type": "uint256"
    }
   ],
   "name": "retreiveFull",
   "outputs": [
    {
     "internalType": "string",
     "name": "",
     "type": "string"
    }
   ],
   "stateMutability": "view",
   "type": "function"
  },
```

```
{
  "inputs": [
    {
      "internalType": "uint256",
      "name": "Docid",
      "type": "uint256"
    }
  ],
  "name": "retreiveSearch",
  "outputs": [
    {
      "internalType": "string",
      "name": "",
      "type": "string"
    }
  ],
  "stateMutability": "view",
  "type": "function"
},
{
  "inputs": [
    {
      "internalType": "uint256",
      "name": "Docid",
      "type": "uint256"
    },
    {
      "internalType": "string",
      "name": "searchJSON",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "fullJSON",
      "type": "string"
    }
  ],
  "name": "store",
  "outputs": [],
  "stateMutability": "nonpayable",
  "type": "function"
}
];

var contractAddress = require('./contractAddress.js');
console.log('Contract Address: ' + contractAddress);
```

```
  var myContract = new web3.eth.Contract(abi, contractAddress, {
    from: deployerAccount,
      gas: 30000000
  });

  var Docid = 0;
  var inputSearch = JSON.stringify(XDSMETADataAttributes);
  var inputFull = JSON.stringify(rawXDSAttr);

  myContract.methods.store(Docid, inputSearch, inputFull).send({
    from: deployerAccount
  }).then(function(receipt){
    console.log(receipt);
    console.log('====================\nTransaction success...');
    hrend = process.hrtime(hrstart);
          console.info('Execution time (hr): %ds %dms', hrend[0], hrend[1] / 1000000);
          console.log('====================');
  });
}
```

Figure 4-39 XDS Document Registry Actor

This section passes JSON into Smart Contract as single string variable

4.2.3.1.2 Smartcontract

In the implementation, Smartcontract was designed to store string values and will return the stored value when called by the corresponding Smartcontract function. The prepared JSON must be converted into a string variable before entering Smartcontract. This is due to the limit of the Ethereum Smartcontract which can cover a limited number of programming variables so, we simplify our program to avoid that limit by storing the whole JSON in string form as a single variable. Because the Ethereum Blockchain requires a particular amount of gas to run the Smartcontract, the length of the variable could produce an error if there isn't enough gas available. That means, we need to increase the limit amount of gas for executing Smartcontract from the default value as shown in Figure 4-40. Although this change may not affect the implementation, it may affect the network where its member prefers to use actual cryptocurrency like Ether to maintain Blockchain where an increase in required gas may accelerate the depletion of currency circulating in the network and severely reduce the maintainability of the Blockchain. Figure 4-41 showing the process flow of Smartcontract function related to Document Registering.

```
var myContract = new web3.eth.Contract(abi, contractAddress, {
  from: deployerAccount,
    gas: 30000000
});
```

Figure 4-40 Specified gas value spplying Ethereum Smartcontract execution

```
var DocumentRegistering_Into_Blockchain() {
        var lastID = Invoke_CheckLastID_SmartcontractFunction();
        var NewID = lastID + 1;
        Invoke_DocumentRegister_SmartcontractFunction(NewID);
        var registerStatus
        if (not Error) {
                registerStatus = "Success";
        }
        return registerStatus;
}
```

Figure 4-41 The pseudocode showing the process flow of Document Register Smartcontract

By these Smartcontract design, XDS Document Registry actors can keep Metadata attributes of each document by storing them as JSON string variable inside Blockchain using one Smartcontract per document. At the same time, the actor can perform search operation by sequentially call upon each published Smartcontract one-by-one until the result was found or until the last in the case which no matching result. It must be noted that publishing of Smartcontract always requires gas to execute the task while calling for the variable value stored in the Smartcontract is not consuming any Blockchain resource. Figure 4-42 showing the Solidity code snippet highlighting (green color) the part relating to Document Registering Function.

It must be noted that at the beginning of the XDS Blockchain network, the base Smartcontract must be initially deployed into the Blockchain to allow further registration of health documents for the rest of the Blockchain ledger.

```
pragma solidity >=0.4.22 <0.7.0; //SPDX-License-Identifier: UNLICENSED

contract Storage {

  struct Document_Registry {
        string searchAttributes; //Storing META-Data Attributes for search operation
        string fullAttributes; //Storing META-Data Attributes for return result
  }
  uint DocID = 0; //This variable is for selecting Document ID, always reset to 0
  uint lastDoc; //This variable keep track for the latest Document ID being used
  mapping (uint => Document_Registry) docreg;
  //Assign variable "documentregist" to map value of struct Document_Registry
  //Constructor Document_Registry store JSON string for search and for return as full response

  //Store string JSON
  function store(uint Docid, string memory searchJSON, string memory fullJSON) public {
    if (Docid > 0){
      DocID = Docid; //If Docid was specified (not 0) replacing string JSON of existing Docid
              //This probably require some kind of authentication
    }
    else{
      lastDoc++; //If Docid was not specified, create new ID
      DocID = lastDoc;
    }
    docreg[DocID].searchAttributes = searchJSON;
    docreg[DocID].fullAttributes = fullJSON;
  }

  //Check the lastest ID being used
  function checkLastID() public view returns (uint) {
    return lastDoc;
  }
```

Figure 4-42 Solidity Code Snippet of Smart Contract

(Highlight - green color) related to Document Registering Function

4.2.3.2 Implementing Document Search Function

4.2.3.2.1 Native-Side Javascript Program

Like IHE ITI-42 transaction handling, the XDS Document Registry actors also wait for ITI-18 on the TCP channel. The received transaction will be converted into JSON while matching UUID with its corresponding metadata attributes as declared in Figure 4-43. The transaction is specified with the header "QueryResponse" and compose of Metadata attributes value input by Document Consumer. These values will be used in search operation which will seek for the Smartcontract with matching metadata attribute values. After the result was found, the actor then proceeds to create a response XML message following the ITI-18 format. Figure 4-44 showing the pseudocode describing the process flow for the native-side Javascript program related to Document Search. Figure 4-45 to Figure 4-49 show the Javascript code snippet for native-side of XDS Document Registry handling ITI-18 transaction and Document Search function.

```
//DocumentQuery==========
function documentQuery (inputAttributes) {
        var requestUUID = {
         FindDocuments: 'urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d',
         FindSubmissionSets: 'urn:uuid:f26abbcb-ac74-4422-8a30-edb644bbc1a9',
         FindFolders: 'urn:uuid:958f3006-baad-4929-a4de-ff1114824431',
         GetAll: 'urn:uuid:10b545ea-725c-446d-9b95-8aeb444eddf3',
         GetDocuments: 'urn:uuid:5c4f972b-d56b-40ac-a5fc-c8ca9b40b9d4',
         GetFolders: 'urn:uuid:5737b14c-8a1a-4539-b659-e03a34a5e1e4',
         GetAssociations: 'urn:uuid:a7ae438b-4bc2-4642-93e9-be891f7bb155',
         GetDocumentsAndAssociations: 'urn:uuid:bab9529a-4a10-40b3-a01f-f68a615d247a',
         GetSubmissionSets: 'urn:uuid:51224314-5390-4169-9b91-b1980040715a',
         GetSubmissionSetAndContents: 'urn:uuid:e8e3cb2c-e39c-46b9-99e4-c12f57260b83',
         GetFolderAndContents: 'urn:uuid:b909a503-523d-4517-8acf-8e5834dfc4c7',
         GetFoldersForDocument: 'urn:uuid:10cae35a-c7f9-4cf5-b61e-fc3278ffb578',
         GetRelatedDocuments: 'urn:uuid:d90e5407-b356-4d91-a89f-873917b4b0e6',
         FindDocumentsByReferenceId: 'urn:uuid:12941a89-e02e-4be5-967c-ce4bfc8fe492'
        }
```

Figure 4-43 XDS Document Registry Actor

Define variable of query request type UUID label following IHE ITI Framework

```
var Received_ITI_18(XMLMessage){
        var JSON_attributes = InterpretXMLtoJSON(XMLMessage);
        var Assorte_JSON = AssortMetadataAttributes(JSON_attributes);
        var SearchKeywords = SortInto_SearchKeywordsFormat(Assorted_JSON);
        return SearchKeywords;
}
```

Figure 4-44 The process flow for the native-side Javasceipt program

related Document Search Function

```
function specifyRequestType (requestedJSON, myCallback) {
  //Specify responseOption
  console.log('Specify responseOption');
  var responseOption = '';
  if
(requestedJSON['query:AdhocQueryRequest']['query:ResponseOption'][0]['$']['returnComposedObjects'
] == 'true'){
    console.log('returnComposedObjects = true');
    if (requestedJSON['query:AdhocQueryRequest']['query:ResponseOption'][0]['$']['returnType']
== 'LeafClass') {
      responseOption = 'LeafClass';
      console.log('Request response as ' + responseOption + '...');
    }
    else if
(requestedJSON['query:AdhocQueryRequest']['query:ResponseOption'][0]['$']['returnType'] ==
'ObjectRef'){
      responseOption = 'ObjectRef';
      console.log('Request response as ObjectRef...');
    }
  }
  //Specify queryType -> revert UUID to human understandable word
  console.log(requestedJSON['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['$']['id']);
  for (i = 0; i < Object.entries(requestUUID).length; i++){ //Cycle through requestUUID object to
check request types
    if (requestedJSON['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['$']['id'] ==
Object.entries(requestUUID)[i][1]){
      var requestType = Object.keys(requestUUID)[i];
      console.log('Query Type: ' + requestType);
    } }
  //Define search keyword array to meet specification of each request type
  var searchKeyword = [];
  if (requestType == 'FindDocuments'){
    var rimSlot = requestedJSON['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['rim:Slot'];
    console.log(util.inspect(rimSlot));
    for (i = 0; i < rimSlot.length; i++){ //Assign attributes in rim:Slot to search keyword array
      if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryPatientId'){
        searchKeyword.push(['DocumentEntry', 'patientId',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
      }
      else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryClassCode'){
        searchKeyword.push(['DocumentEntry', ['classCode', 'displayName'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
      }
      else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryTypeCode'){
        searchKeyword.push(['DocumentEntry', ['typeCode', 'displayName'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
      }
```

```
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryPracticeSettingCode'){
              searchKeyword.push(['DocumentEntry', ['practiceSettingCode', 'displayName'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryCreationTimeFrom'){
              searchKeyword.push(['DocumentEntry', ['creationTime', 'From'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryCreationTimeTo'){
              searchKeyword.push(['DocumentEntry', ['creationTime', 'To'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryServiceStartTimeFrom'){
              searchKeyword.push(['DocumentEntry', ['serviceStartTime', 'From'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryServiceStartTimeTo'){
              searchKeyword.push(['DocumentEntry', ['serviceStartTime', 'To'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryServiceStopTimeFrom'){
              searchKeyword.push(['DocumentEntry', ['serviceStopTime', 'From'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryServiceStopTimeTo'){
              searchKeyword.push(['DocumentEntry', ['serviceStopTime', 'To'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryHealthcareFacilityTypeCode'){
              searchKeyword.push(['DocumentEntry', ['healthcareFacilityTypeCode', 'displayName'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryEventCodeList'){
              searchKeyword.push(['DocumentEntry', 'eventCodeList',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryConfidentialityCode'){
              searchKeyword.push(['DocumentEntry', ['confidentialityCode', 'displayName'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryAuthorPerson'){
              searchKeyword.push(['DocumentEntry', ['author', 'authorPerson'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
            }
            else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryFormatCode'){
              searchKeyword.push(['DocumentEntry', ['formatCode', 'displayName'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]); }
```

```
                else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryStatus'){
                  searchKeyword.push(['DocumentEntry', 'availabilityStatus',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                }
                else if (rimSlot[i]['$']['name'] == '$XDSDocumentEntryType'){
                  searchKeyword.push(['DocumentEntry', 'objectType',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                }  }  }
            else if (requestType == 'FindSubmissionSets'){ //Need to add case in DocSearch***
                    var rimSlot =
requestedJSON['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['rim:Slot'];
            console.log(util.inspect(rimSlot));
            for (i = 0; i < rimSlot.length; i++){
                    if (rimSlot[i]['$']['name'] == '$XDSSubmissionSetPatientId'){
                            searchKeyword.push(['SubmissionSet', 'patientId',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                    }
                    else if (rimSlot[i]['$']['name'] == '$XDSSubmissionSetSourceId'){
                            searchKeyword.push(['SubmissionSet', 'sourceId',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                    }
                    else if (rimSlot[i]['$']['name'] == '$XDSSubmissionSetSubmissionTimeFrom'){ //***
                            searchKeyword.push(['SubmissionSet', ['submissionTime', 'From'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                    }
                    else if (rimSlot[i]['$']['name'] == '$XDSSubmissionSetSubmissionTimeTo'){ //***
                            searchKeyword.push(['SubmissionSet', ['submissionTime', 'To'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                    }
                    else if (rimSlot[i]['$']['name'] == '$XDSSubmissionSetAuthorPerson'){ //***
                            searchKeyword.push(['SubmissionSet', ['author', 'authorPerson'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                    }
                    else if (rimSlot[i]['$']['name'] == '$XDSSubmissionSetContentType'){
                            searchKeyword.push(['SubmissionSet', ['contentTypeCodes', 'displayName'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                    }
                    else if (rimSlot[i]['$']['name'] == '$XDSSubmissionSetStatus'){
                            searchKeyword.push(['SubmissionSet', 'availabilityStatus',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                    }
             }
            }
            else if (requestType == 'FindFolders'){
                    var rimSlot =
requestedJSON['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['rim:Slot'];
                    console.log(util.inspect(rimSlot));
```

```
                for (i = 0; i < rimSlot.length; i++){
                        if (rimSlot[i]['$']['name'] == '$XDSFolderPatientId'){
                                searchKeyword.push(['Folder', 'patientId',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                        }
                        else if (rimSlot[i]['$']['name'] == '$XDSFolderLastUpdateTimeFrom'){ //***
                                searchKeyword.push(['Folder', ['lastUpdateTime', 'From'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                        }
                        else if (rimSlot[i]['$']['name'] == '$XDSFolderLastUpdateTimeTo'){ //***
                                searchKeyword.push(['Folder', ['lastUpdateTime', 'To'],
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                        }
                        else if (rimSlot[i]['$']['name'] == '$XDSFolderCodeList'){ //***
                                searchKeyword.push(['Folder', 'codeList',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                        }
                        else if (rimSlot[i]['$']['name'] == '$XDSFolderStatus'){
                                searchKeyword.push(['Folder', 'availabilityStatus',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                        }
                }
        }
        else if (requestType == 'GetAll'){
                var rimSlot =
requestedJSON['query:AdhocQueryRequest']['rim:AdhocQuery'][0]['rim:Slot'];
                console.log(util.inspect(rimSlot));
                for (i = 0; i < rimSlot.length; i++){
                        if (rimSlot[i]['$']['name'] == '$patientId'){
                                searchKeyword.push(['DocumentEntry', 'patientId',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                                searchKeyword.push(['SubmissionSet', 'patientId',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                                searchKeyword.push(['Folder', 'patientId',
rimSlot[i]['rim:ValueList'][0]['rim:Value'][0]]);
                        }
                } }
        searchKeyword.push([requestType, responseOption]);
        console.log(searchKeyword);
        myCallback(searchKeyword, invokeContract);
       }
       specifyRequestType(inputAttributes, checkLastID);
}
```

Figure 4-45 XDS Document Registry Actor

Identify query request type following received ITI-18 header and assort search keyword

```
async function checkLastID (sK, myCallback) {
 //web3.eth.defaultAccount = web3.eth.personal.getAccounts().then(console.log);
 console.log('Checking for latest ID...');
 let acc = await web3.eth.personal.getAccounts();
  if (acc.err) {console.log(acc.err);}
  else {console.log('Accounts available on this node:\n' + acc);}

 var deployerAccount = acc[0];
 console.log('Originally deployed with account:' + deployerAccount);
 var abi =
 [
  {
   "inputs": [],
   "name": "checkLastID",
   "outputs": [
    {
     "internalType": "uint256",
     "name": "",
     "type": "uint256"
    }
   ],
   "stateMutability": "view",
   "type": "function"
  },
  {
   "inputs": [
    {
     "internalType": "uint256",
     "name": "Docid",
     "type": "uint256"
    }
   ],
   "name": "retreiveFull",
   "outputs": [
    {
     "internalType": "string",
     "name": "",
     "type": "string"
    }
   ],
   "stateMutability": "view",
   "type": "function"
  },
```

```
{
 "inputs": [
  {
   "internalType": "uint256",
   "name": "Docid",
   "type": "uint256"
  }
 ],
 "name": "retreiveSearch",
 "outputs": [
  {
   "internalType": "string",
   "name": "",
   "type": "string"
  }
 ],
 "stateMutability": "view",
 "type": "function"
},
{
 "inputs": [
  {
   "internalType": "uint256",
   "name": "Docid",
   "type": "uint256"
  },
  {
   "internalType": "string",
   "name": "searchJSON",
   "type": "string"
  },
  {
   "internalType": "string",
   "name": "fullJSON",
   "type": "string"
  }
 ],
 "name": "store",
 "outputs": [],
 "stateMutability": "nonpayable",
 "type": "function"
}
];

var contractAddress = require('./contractAddress.js');
console.log('Contract Address: ' + contractAddress);
```

```
var myContract = new web3.eth.Contract(abi, contractAddress, {
  from: deployerAccount,
    gas: 30000000
});

console.log('Calling contract...');
myContract.methods.checkLastID().call({
  from: deployerAccount
}, (err,res) => {
  if (err) {
    console.log(err);
  }else{
    console.log('Found, the lastest document ID is ' + res);
    console.log('----------------------------------------');
    myCallback(sK, res, matchMaker);
  }
});
}
```

Figure 4-46 XDS Document Registry Actor

Check for the latest document ID published in Blockchain before beginning search operation

```
//Invoke each contract for keyword search
async function invokeContract(sK, maxDoc, myCallback){
 //web3.eth.defaultAccount = web3.eth.personal.getAccounts().then(console.log);
 console.log('Search keywords received...\nMoving on to search function...');
 let acc = await web3.eth.personal.getAccounts();
  if (acc.err) {console.log(acc.err);}
  else {console.log('Accounts available on this node:\n' + acc);}

 var deployerAccount = acc[0];
 console.log('Originally deployed with account:' + deployerAccount);
 var abi =
 [
  {
   "inputs": [],
   "name": "checkLastID",
   "outputs": [
    {
     "internalType": "uint256",
     "name": "",
     "type": "uint256"
    }
   ],
   "stateMutability": "view",
   "type": "function"
  },
  {
   "inputs": [
    {
     "internalType": "uint256",
     "name": "Docid",
     "type": "uint256"
    }
   ],
   "name": "retreiveFull",
   "outputs": [
    {
     "internalType": "string",
     "name": "",
     "type": "string"
    }
   ],
   "stateMutability": "view",
   "type": "function"
  },
```

```
{
  "inputs": [
    {
      "internalType": "uint256",
      "name": "Docid",
      "type": "uint256"
    }
  ],
  "name": "retreiveSearch",
  "outputs": [
    {
      "internalType": "string",
      "name": "",
      "type": "string"
    }
  ],
  "stateMutability": "view",
  "type": "function"
},
{
  "inputs": [
    {
      "internalType": "uint256",
      "name": "Docid",
      "type": "uint256"
    },
    {
      "internalType": "string",
      "name": "searchJSON",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "fullJSON",
      "type": "string"
    }
  ],
  "name": "store",
  "outputs": [],
  "stateMutability": "nonpayable",
  "type": "function"
}
];

var contractAddress = require('./contractAddress.js');
console.log('Contract Address: ' + contractAddress);
```

```
var myContract = new web3.eth.Contract(abi, contractAddress, {
  from: deployerAccount,
    gas: 30000000
});

console.log('The latest document ID is no.:' + maxDoc);
//callRegSearch while run along Docid from 1 to latest (known using checkID)
console.log('Begin contract search...');
var traceDocid = 0;
for (Docid = 1; Docid <= maxDoc; Docid++){
  traceDocid++;
  myContract.methods.retreiveSearch(Docid).call({
    from: deployerAccount
  }, (err,res) => {
    if (err) {
      console.log(err);
    }else{
      var XDSattributes = JSON.parse(res);
      myCallback(XDSattributes, sK, traceDocid, fullContract);
    }
  });
 }
}
```

Figure 4-47 XDS Document Registry Actor

Begin search operation by sequentially check each published contract one-by-one

```
//Compare keyword with JSON called from smartcontract
function matchMaker (searchXDSAtt, sK, Docid, myCallback){
 //searchXDSAtt = XDS Object called from smartcontract
 //sK = search keyword received from ITI-18
 console.log('----------\nSmartcontract called...');

 var matchedCount = 0;
 var timeAttributes = {
  creationTime: {
   From: null,
   To: null
  },
  serviceStartTime: {
   From: null,
   To: null
  },
  serviceStopTime: {
   From: null,
   To: null
  }
 }

 for (i = 0; i < sK.length - 1; i++){
  var keyCount = i+1;
  if (Array.isArray(sK[i][1])){ //check if attributes have sub-attributes i.e. author, classCode, etc.
   if (sK[i][1][0] == 'author') { //author specific case
    if (searchXDSAtt[sK[i][0]][sK[i][1][0]][0][sK[i][1][1]] == sK[i][2]){
     matchedCount++;
     console.log('Keyword ' + keyCount + ' matched...');
    }
    else {
     console.log('Keyword ' + keyCount + ' unmatched...');
     break;
    }
   }
   else if (sK[i][1][0] == 'creationTime') {
    timeAttributes[sK[i][1][0]][sK[i][1][1]] = sK[i][2];
    if (timeAttributes[sK[i][1][0]]['From'] && timeAttributes[sK[i][1][0]]['To']){
     var dateTimeFrom = moment.utc(timeAttributes[sK[i][1][0]]['From'],
'YYYYMMDDHHmmss');
     var dateTimeTo = moment.utc(timeAttributes[sK[i][1][0]]['To'], 'YYYYMMDDHHmmss');
     var dateTimeTarget = moment.utc(searchXDSAtt[sK[i][0]][sK[i][1][0]],
'YYYYMMDDHHmmss');
```

```
                if (moment(dateTimeTarget).isBetween(dateTimeFrom, dateTimeTo, undefined, '[]')){
                  matchedCount++; //match count 2 times due to the attributes require 2 search keywords
                  matchedCount++;
                  console.log(sK[i][1][0] + ' at Keyword ' + keyCount + ' matched...');
                }
              }
            }
            else if (sK[i][1][0] == 'serviceStartTime' || sK[i][1][0] == 'serviceStopTime') {
              if (searchXDSAtt[sK[i][0]][sK[i][1][0]] != 'N/A'){ //check if current Document have
serviceTime attributes present
                timeAttributes[sK[i][1][0]][sK[i][1][1]] = sK[i][2];
                if (timeAttributes[sK[i][1][0]]['From'] && timeAttributes[sK[i][1][0]]['To']){
                  var dateTimeFrom = moment.utc(timeAttributes[sK[i][1][0]]['From'],
'YYYYMMDDHHmmss');
                  var dateTimeTo = moment.utc(timeAttributes[sK[i][1][0]]['To'], 'YYYYMMDDHHmmss');
                  var dateTimeTarget = moment.utc(searchXDSAtt[sK[i][0]][sK[i][1][0]],
'YYYYMMDDHHmmss');
                  if (moment(dateTimeTarget).isBetween(dateTimeFrom, dateTimeTo, undefined, '[]')){
                    matchedCount++; //match count 2 times due to the attributes require 2 search
keywords
                    matchedCount++;
                    console.log(sK[i][1][0] + ' at Keyword ' + keyCount + ' matched...');
                  }
                }
              }
            }
            else {
              if (searchXDSAtt[sK[i][0]][sK[i][1][0]][sK[i][1][1]] == sK[i][2]){
                matchedCount++;
                console.log('Keyword ' + keyCount + ' matched...');
              }
              else {
                console.log('Keyword ' + keyCount + ' unmatched...');
                break;
              }
            }
          }
          else {
            if (Array.isArray(sK[i][2])){
              if (searchXDSAtt[sK[i][0]][sK[i][1]][0] == sK[i][2][0]){
                matchedCount++;
                console.log('Keyword ' + keyCount + ' matched...');
              }
              else if (searchXDSAtt[sK[i][0]][sK[i][1]][0] != sK[i][2][0]) {
                console.log('Keyword ' + keyCount + ' unmatched...');
                break;
              }}
```

```
            else {
              if (searchXDSAtt[sK[i][0]][sK[i][1]] == sK[i][2]){
                matchedCount++;
                console.log('Keyword ' + keyCount + ' matched...');
              }
              else {
                console.log('Keyword ' + keyCount + ' unmatched...');
                break;
              }
            }
          }
        }

        if (matchedCount == sK.length - 1){
          console.log('All matched, successfully... \nReturn document as search
result:\n========================================');
          console.log(searchXDSAtt.DocumentEntry);
          console.log('========================================');
          myCallback(Docid, sK, responseToUser);
        }
        else {
          console.log('Unmatched...');
        }
      }

      async function fullContract (Docid, sK, myCallback){
        //web3.eth.defaultAccount = web3.eth.personal.getAccounts().then(console.log);
        console.log('Calling for full document...');
        let acc = await web3.eth.personal.getAccounts();
          if (acc.err) {console.log(acc.err);}
          else {console.log('Accounts available on this node:\n' + acc);}

        var deployerAccount = acc[0];
        console.log('Originally deployed with account:' + deployerAccount);
        var abi =
        [
          {
            "inputs": [],
            "name": "checkLastID",
            "outputs": [
              {
                "internalType": "uint256",
                "name": "",
                "type": "uint256"
              }
            ],
```

```
  "stateMutability": "view",
  "type": "function"
 },
 {
  "inputs": [
   {
    "internalType": "uint256",
    "name": "Docid",
    "type": "uint256"
   }
  ],
  "name": "retreiveFull",
  "outputs": [
   {
    "internalType": "string",
    "name": "",
    "type": "string"
   }
  ],
  "stateMutability": "view",
  "type": "function"
 },
 {
  "inputs": [
   {
    "internalType": "uint256",
    "name": "Docid",
    "type": "uint256"
   }
  ],
  "name": "retreiveSearch",
  "outputs": [
   {
    "internalType": "string",
    "name": "",
    "type": "string"
   }
  ],
  "stateMutability": "view",
  "type": "function"
 },
 {
  "inputs": [
   {
    "internalType": "uint256",
    "name": "Docid",
    "type": "uint256"
   },
```

```
      {
        "internalType": "string",
        "name": "searchJSON",
        "type": "string"
      },
      {
        "internalType": "string",
        "name": "fullJSON",
        "type": "string"
      }
    ],
    "name": "store",
    "outputs": [],
    "stateMutability": "nonpayable",
    "type": "function"
  }
];

var contractAddress = require('./contractAddress.js');
console.log('Contract Address: ' + contractAddress);

var myContract = new web3.eth.Contract(abi, contractAddress, {
  from: deployerAccount,
    gas: 30000000
});

console.log('Returning document: ' + Docid);
        myContract.methods.retreiveFull(Docid).call({
        from: deployerAccount
        }, (err,res) => {
                if (err) {
                        console.log(err);
                } else {
                        var XDSattributes = JSON.parse(res);
                        myCallback(XDSattributes, sK);
                }
        });
}
```

Figure 4-48 XDS Document Registry Actor

Check if value of Metadata attributes in each publish contract matched with search keyword before summarize search result.

```
function responseToUser (rXDSattribute, sK) {
        //Define variable for shorter object accessing
        var sEnvelope = rXDSattribute['soapenv:Envelope'];
        //inside Enveope
        var s$ = sEnvelope['$'];
        var sBody = sEnvelope['soapenv:Body'][0];
        var sHeader = sEnvelope['soapenv:Header'][0];
        //inside Envelope>Header
        var wsaTo = sHeader['wsa:To'];
        var wsaMessageID = sHeader['wsaMessageID'];
        var wsaAction = sHeader['wsaAction'];
        //inside Envelope>Body
        var lcmSubmitObjectsRequest = sBody['lcm:SubmitObjectsRequest'][0];
        //inside Envelope>Body>lcm:SubmitObjectsRequest
        var bodyRegistryObjectList = lcmSubmitObjectsRequest['rim:RegistryObjectList'][0];
        //inside Envelope>Body>lcm:SubmitObjectsRequest>rim:RegistryObjectList
        var bodyExtrinsicObject = bodyRegistryObjectList['rim:ExtrinsicObject'][0];
        var bodyRegistryPackage = bodyRegistryObjectList['rim:RegistryPackage'][0];
        var bodyClassification = bodyRegistryObjectList['rim:Classification'][0];
        var bodyAssociation = bodyRegistryObjectList['rim:Association'][0];
        //Decrypt sourcePatientInfo -> So encryption just prevent those who look directly into
contract to read the attribute
        //Alternatively, this can be left encrypted while require Document Consumer to decrypt
by themselves
        for (var i = 0; i < bodyExtrinsicObject['rim:Slot'].length; i++) {
                if (bodyExtrinsicObject['rim:Slot'][i]['$']['name'] == 'sourcePatientInfo') {
                        var encryptedString =
bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'];
        var plaintext = cryptr.decrypt(encryptedString);
    //Also replace the attributes within full XDSAttributes object with encrypted attribute
    bodyExtrinsicObject['rim:Slot'][i]['rim:ValueList'][0]['rim:Value'] = plaintext.split(',');
                }
        }
        console.log('=========================================\nReturn type: ' +
sK[sK.length-1] + '\n=========================================');
        var responseJSON = {
                "query:AdhocQueryResponse": {
                        "$": {
                                "status": "Success",
                                "xmlns:query": "urn:oasis:names:tc:ebxml-
regrep:xsd:query:3.0",
                                "xmlns:rim": "urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0",
                                "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
                                "xsi:schemaLocation": "urn:oasis:names:tc:ebxml-
regrep:xsd:query:3.0 ../../schema/ebRS/query.xsd"
                        },
```

```
                                {
                                        "rim:ExtrinsicObject": [bodyExtrinsicObject]
                                }
                        ]
                }
        }
        console.log(util.inspect(responseJSON));

        if (netServer && netSocket) {
                console.log('Responsing query result: ');
                var responseXML = builder.buildObject(responseJSON);
    //var regex = /\r?\n|\r/g;
    //var responseXML = responseXML.replace(regex,'');
    netSocket.write(responseXML);
    console.log('----------------------');
    console.log(responseXML);
        }
        console.log('========================================\nDone!');
        hrend = process.hrtime(hrstart);
        console.log('Execution time (hr): %ds %dms', hrend[0], hrend[1] / 1000000);
        console.log('========================================');
}
```

Figure 4-49 XDS Document Registry

Gather search result and response back to Document Consumer Actor

4.2.3.2.2 Smartcontract

Like the Document Register function, the XDS Document Registry Actor must be able to respond to the Document Search query from the Document Consumer by returning the metadata attributes of the registered document matched with the query to the consumer. In a traditional database, this can be done by utilizing a query of a relational (SQL) database. However, for Blockchain, the structure of stored data is different from relational databases but similar to NoSQL. That means search operation will need to rely on a sequential search algorithm. The program will need to take a look at all published transactions one-by-one from the first until the result was found. Each transaction will require the program to call on smart-contract for reviewing the stored value before comparing it with the specified value used for search. When all of the values called from the Smartcontract are matched with the value specified for search, the value called will be marked as a search result which will be returned to XDS Document Consumer Actor via ITI-18 format. Figure 4-50 showing the pseudocode describing the process flow for the Smartcontract function related to Document Search. Figure 4-51 showing the Solidity code snippet highlighting (green color) the part relating to Document Search Function.

```
var DocumentSearch_Within_Blockchain(SearchKeywords) {
        var SearchResult;
        var lastID = Invoke_CheckLastID_SmartcontractFunction();
        for (i = 0; i < lastID; i++) {
                var StoredValue = Invoke_ReadStoredValue_SmartcontractFunction(i);
                if (SearchKeywords == StoredValue) {
                        SearchResult = StoredValue;
                }
        }
        return SearchResult;
}
```

Figure 4-50 The pseudocode showing the process flow of Smartcontract function related to Document Search

```
pragma solidity >=0.4.22 <0.7.0; //SPDX-License-Identifier: UNLICENSED

contract Storage {

    struct Document_Registry {
            string searchAttributes; //Storing META-Data Attributes for search operation
            string fullAttributes; //Storing META-Data Attributes for return result
    }
    uint DocID = 0; //This variable is for selecting Document ID, always reset to 0
    uint lastDoc; //This variable keep track for the latest Document ID being used
    mapping (uint => Document_Registry) docreg;
    //Assign variable "documentregist" to map value of struct Document_Registry
    //Constructor Document_Registry store JSON string for search and for return as full response

    //Store string JSON
    function store(uint Docid, string memory searchJSON, string memory fullJSON) public {
        if (Docid > 0){
            DocID = Docid; //If Docid was specified (not 0) replacing string JSON of existing Docid
                        //This probably require some kind of authentication
        }
        else{
            lastDoc++; //If Docid was not specified, create new ID
            DocID = lastDoc;
        }
        docreg[DocID].searchAttributes = searchJSON;
        docreg[DocID].fullAttributes = fullJSON;
    }

    //Check the lastest ID being used
    function checkLastID() public view returns (uint) {
        return lastDoc;
    }

    //Call for string JSON for Search Program
    function retreiveSearch(uint Docid) public view returns (string memory) {
        return docreg[Docid].searchAttributes;
    }

    //Call for string JSON for Result Return Program
    function retreiveFull(uint Docid) public view returns (string memory) {
        return docreg[Docid].fullAttributes;
    }
}
```

Figure 4-51 Solidity Code Snippet of Smartcontract

(Highlight - green color) related to Document Search Function

4.3 Implementation Result

As mentioned in Section 4.1.1, the environment of the test machine is running on Linux Ubuntu (64-bit) version 18.10 with 8 GB RAM and 100 GB storage dynamically shared from the host machine.

As mentioned in Section 4.1.3, the Blockchain network environment for the implementation will be demonstrated using the 7-Nodes Example Blockchain. That means there will be seven IBFT Blockchain nodes simulated within the environment of the test machine. The test sample will be monitored from node number one (which would later refer as "1st Node").

To indicate that the implementation has the compatibility to operate with XDS Actors in a common XDS system, we created transaction samples based on the example provided by the IHE ITI framework. Transaction samples provided by the framework are including ITI-42 Register Document Set-b transaction, ITI-18 Registry Stored Query transaction and its corresponding response transaction. However, there is much more specification provided in the framework that far exceeds the element provided within the example so, there may consist of an element that differs from the transaction used in the actual system which may cause an error and need further adjustment in the actual deployment.

The main group of XDS Actors in the implementation are XDS Document Registry Actor, XDS Document Repository Actor, and XDS Document Consumer Actor as shown in Figure 4-52. All files corresponding to the implementation must be included within the directory that contains the 7-Node Examples as shown in Figure 4-52 in the bottom right window. As mentioned in Section 4.2.3 that the XDS Document Registry Actor is the actor that acts as the medium between the XDS system and the Blockchain network. At the same time, the main function and process triggered in the implementation including Document Register and Document Search (Document Query) are centered around the XDS Document Registry. The XDS Document Registry will always active and standby for incoming transactions as shown in Figure 4-52 top left terminal and Figure 4-53.

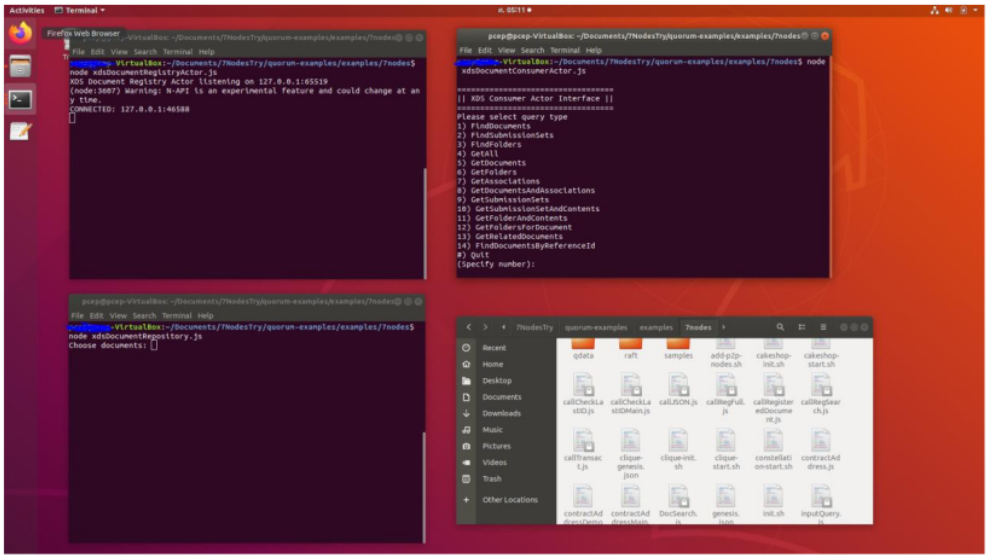


Figure 4-52 All XDS Actors activated via its terminal
(Top Left) XDS Document Registry Actor
(Top Right) XDS Document Consumer Actor
(Bottom Left) XDS Document Repository Actor
(Bottom Right) The folder containing all files related to the XDS Blockchain

```
node xdsDocumentRegistryActor.js
XDS Document Registry Actor listening on 127.0.0.1:65519
(node:3553) Warning: N-API is an experimental feature and could change at an
y time.
```

Figure 4-53 XDS Document Registry Actor standby and wait for incoming XML Messages.

As mentioned in Section 4.2.1 that the XDS Document Repository Actor is the actor that triggers the Document Register function registering new health document metadata attributes set into the Blockchain ledger. The Document Register function is triggered once the XDS Document Repository Actor program was called to register new health document metadata set into the XDS Affinity Domain network as shown in Figure 4-54 and Figure 4-55. The XDS Document Registry Actor then received the Document Register transaction and proceeds to broadcast the transaction into the Blockchain network publishing the transaction into the Blockchain ledger through a Block validation process as shown in Figure 4-56. After the transaction was successfully published into the Blockchain ledger, the XDS Document Registry Actor response back to the Document Repository Actor to report that the process was completed as shown in Figure 4-57 which terminating the XDS Document Repository Actor program.

```
node xdsDocumentRepository.js
Choose documents: 1
```

Figure 4-54 XDS Document Repository prompt for health document number to register

```
CONNECTED TO: 127.0.0.1:65519
Sent:
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
   <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
      <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
      <wsa:MessageID soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7
B91554413514373</wsa:MessageID>
      <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDoc
umentSet-b</wsa:Action>
   </soapenv:Header>
   <soapenv:Body>
      <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:x
sd:lcm:3.0">
         <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:
xsd:rim:3.0">
            <rim:ExtrinsicObject id="Document01" mimeType="text/plain" objec
tType="urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1" status="urn:oasis:name
s:tc:ebxml-regrep:StatusType:Approved">
               <rim:Slot name="size">
                  <rim:ValueList>
```

Figure 4-55 XDS Document Repository sent ITI-42 transaction to XDS Document Registry

```
-----------------------------------------
Deploying with account:0xed9d02e382b34818e88B88a309c7fe71E65f419d
Contract Address: 0xF380286a425Fe5107ad8D755E407317c6e965Ad2
CLOSED: 127.0.0.1 46588
{ blockHash: '0x9c13b1870a68e0734fa426b543761fad7cfae59452145affba000a9106a7
956e',
  blockNumber: 44880,
  contractAddress: null,
  cumulativeGasUsed: 9420604,
  from: '0xed9d02e382b34818e88b88a309c7fe71e65f419d',
  gasUsed: 9420604,
  logsBloom: '0x00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000',
  status: true,
  to: '0xf380286a425fe5107ad8d755e407317c6e965ad2',
  transactionHash: '0xe07312f8cbdf5d18a2f35aeabd281c802698f09e01ae1099d1dacf
c3bbf327b3',
  transactionIndex: 0,
  events: {} }
====================
Transaction success...
```

Figure 4-56 XDS Document Registry received ITI-42 transaction and

successfully registered the metadata set into the Blockchain while wait for other XML Messages

```
==============================
Respond received: ACK from 127.0.0.1

Execution time (hr): 0s 4.232416ms
==============================
Connection closed
```

Figure 4-57 XDS Document Repository received response from XDS Document Registry then terminate

As mentioned in Section 4.2.2 that the XDS Document Consumer Actor is the actor that triggers the Document Query function receiving user input search keyword values and asks the XDS Document Registry Actor to search for an existing registered document with the matching metadata attributes value as specified in the search keywords. Once called, the XDS Document Consumer Actor program will prompt the user for document query type (Figure 4-58) then prompt for essential search keyword values (Figure 4-59) and followed by the addition of optional search keyword values (Figure 4-60). After all search keyword values from the user were set, the XDS Document Consumer Actor program then proceeds to assort the input into the ITI-18 transaction format and send it to the XDS Document Registry Actor as shown in Figure 4-61. Upon receiving the transaction, the XDS Document Registry Actor program then uses the provided search keywords to search the Blockchain ledger for the metadata set with a matching value as shown in Figure 4-62 and Figure 4-63. Whether the matching result was found or not, after searching on all registered metadata set, the XDS Document Registry Actor then response search result to the XDS Document Consumer Actor as shown in Figure 4-64. The XDS Document Consumer Actor then interprets the response and displays the search result to the user as shown in Figure 4-65.

```
==================================
|| XDS Consumer Actor Interface ||
==================================
Please select query type
1) FindDocuments
2) FindSubmissionSets
3) FindFolders
4) GetAll
5) GetDocuments
6) GetFolders
7) GetAssociations
8) GetDocumentsAndAssociations
9) GetSubmissionSets
10) GetSubmissionSetAndContents
11) GetFolderAndContents
12) GetFoldersForDocument
13) GetRelatedDocuments
14) FindDocumentsByReferenceId
#) Quit
(Specify number):
```

Figure 4-58 XDS Document consumer Actor prompt the user for input

```
(Specify number): 1
Query Type: FindDocuments

Keywords require: XDSDocumentEntryPatientId
Value: IHEBLUE-2736^^^&1.3.6.1.4.1.27829.13.20.3000&ISO
Keywords require: XDSDocumentEntryStatus
Value: urn:oasis:names:tc:ebxml-regrep:StatusType:Approved
```

Figure 4-59 XDS Document Consumer Actor prompt for essential search keyword values

```
==========================
Query type: FindDocuments
Query keywords:
$XDSDocumentEntryPatientId = IHEBLUE-2736^^^&1.3.6.1.4.1.27829.13.20.3000&ISO
$XDSDocumentEntryStatus = urn:oasis:names:tc:ebxml-regrep:StatusType:Approved
==========================
Available optional keywords:
0) No more optional keywords
1) XDSDocumentEntryClassCode
2) XDSDocumentEntryTypeCode
3) XDSDocumentEntryPracticeSettingCode
4) XDSDocumentEntryCreationTime
5) XDSDocumentEntryServiceStartTime
6) XDSDocumentEntryServiceStopTime
7) XDSDocumentEntryHealthcareFacilityTypeCode
8) XDSDocumentEntryEventCodeList
9) XDSDocumentEntryConfidentialityCode
10) XDSDocumentEntryAuthorPerson
11) XDSDocumentEntryFormatCode
12) XDSDocumentEntryType
#) Quit
Select keywords (specify number):
```

Figure 4-60 XDS Document Consumer Actor prompt for optional search keyword values

```
==========================
All keywords set...
==========================
Query type: FindDocuments
Query keywords:
$XDSDocumentEntryPatientId = IHEBLUE-2736^^^&amp;1.3.6.1.4.1.21367.13.20.3000&am
p;ISO
$XDSDocumentEntryStatus = urn:oasis:names:tc:ebxml-regrep:StatusType:Approved
==========================
CONNECTED TO: 127.0.0.1:65519
Query Sent...
Respond received: ACK from 127.0.0.1

Execution time (hr): 0s 10.777685ms
```

Figure 4-61 XDS Document Consumer Actor sent ITI-18 transaction

to XDS Document Registry and wait for response

```
Received data....
XML:
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<query:AdhocQueryRequest xmlns:query="urn:oasis:names:tc:ebxml-regrep:xsd:qu
ery:3.0" xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" xmlns:rs="u
rn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0" xmlns:xsi="http://www.w3.org/2001
/XMLSchema-instance" xsi:schemaLocation="urn:oasis:names:tc:ebxml-regrep:xsd
:query:3.0 ../../schema/ebRS/query.xsd">
  <query:ResponseOption returnComposedObjects="true" returnType="LeafClass"/
>
  <rim:AdhocQuery id="urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d">
    <rim:Slot name="$XDSDocumentEntryPatientId">
      <rim:ValueList>
        <rim:Value>IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;
amp;ISO</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="$XDSDocumentEntryStatus">
      <rim:ValueList>
        <rim:Value>urn:oasis:names:tc:ebxml-regrep:StatusType:Approved</rim:
Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:AdhocQuery>
</query:AdhocQueryRequest>

Converted to object:
-----------------------
{ 'query:AdhocQueryRequest':
   { '$':
      { 'xmlns:query': 'urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0',
        'xmlns:rim': 'urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0',
        'xmlns:rs': 'urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0',
        'xmlns:xsi': 'http://www.w3.org/2001/XMLSchema-instance',
        'xsi:schemaLocation': 'urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0
 ../../schema/ebRS/query.xsd' },
     'query:ResponseOption': [ [Object] ],
     'rim:AdhocQuery': [ [Object] ] } }
--------------------
Query requested...
Specify responseOption
returnComposedObjects = true
Request response as LeafClass...
urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d
Query Type: FindDocuments
[ { '$': { name: '$XDSDocumentEntryPatientId' },
    'rim:ValueList': [ [Object] ] },
  { '$': { name: '$XDSDocumentEntryStatus' },
==============================
```

Figure 4-62 XDS Document Registry received ITI-18 transaction then interpret the message

```
Checking for latest ID...
Accounts available on this node:
0xed9d02e382b34818e88B88a309c7fe71E65f419d
Originally deployed with account:0xed9d02e382b34818e88B88a309c7fe71E65f419d
Contract Address: 0xF380286a425Fe5107ad8D755E407317c6e965Ad2
Calling contract...
Found, the lastest document ID is 12
----------------------------------------
Search keywords received...
Moving on to search function...
Accounts available on this node:
0xed9d02e382b34818e88B88a309c7fe71E65f419d
Originally deployed with account:0xed9d02e382b34818e88B88a309c7fe71E65f419d
Contract Address: 0xF380286a425Fe5107ad8D755E407317c6e965Ad2
The latest document ID is no.:12
Begin contract search...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
```

```
----------------------------------------
Search keywords received...
Moving on to search function...
Accounts available on this node:
0xed9d02e382b34818e88B88a309c7fe71E65f419d
Originally deployed with account:0xed9d02e382b34818e88B88a309c7fe71E65f419d
Contract Address: 0x1932c48b2bF8102Ba33B4A6B545C32236e342f34
The latest document ID is no.:2
Begin contract search...
----------
Smartcontract called...
Keyword 1 unmatched...
Unmatched...
----------
Smartcontract called...
Keyword 1 matched...
Keyword 2 matched...
All matched, successfully...
Return document as search result:
========================================
{ author:
   [ { authorPerson: '^Tymoteusz^Mccabe^^^',
       authorInstitution: [Array],
       authorRole: 'Attending',
       authorSpecialty: 'Neurology' },
     { authorPerson: '^Stevie^Lamb^^^',
       authorInstitution: [Array],
       authorRole: 'Neurologist',
       authorSpecialty: 'Neurology' } ],
  availabilityStatus: 'urn:oasis:names:tc:ebxml-regrep:StatusType:Approved',
  classCode:
   { codingScheme: '1.3.6.1.4.1.19376.1.2.6.1',
     displayName: 'Treatment Plan or Protocol' },
  comment: '',
  confidentialityCode:
   { codingScheme: '2.16.840.1.113883.5.25',
     displayName: 'Restricted' },
  creationTime: '20070101',
  entryUUID: 'Document02',
  eventCodeList:
   [ { codingScheme: '1.3.6.1.4.1.21367.2017.3',
       displayName: 'Foundational Connectathon Read-Access Policy' },
     { codingScheme: '1.3.6.1.4.1.21367.2017.3',
       displayName: 'FULL ACCESS TO ALL POLICY' } ],
  formatCode:
   { codingScheme: '1.3.6.1.4.1.19376.1.2.3',
     displayName: 'urn:ihe:iti:bppc:2007' },
  hash: '8cd7c25aa2526918fef504fea46b79a3ebf123db',
  healthcareFacilityTypeCode:
   { codingScheme: '2.16.840.1.113883.6.96',
     displayName: 'Hospital-trauma center' },
  homeCommunityId: 'N/A',
  languageCode: 'en-us',
```

Figure 4-63 XDS Document Registry then begin search operation over Smartcontract

If the matching result were found, it will be response back to XDS Document Consumer.

```
Keyword 1 matched...
Keyword 2 matched...
All matched, successfully...
Return document as search result:
========================================
{ author:
   [ { authorPerson: '^Tymoteusz^Mccabe^^^',
       authorInstitution: [Array],
       authorRole: 'Attending',
       authorSpecialty: 'Neurology' },
     { authorPerson: '^Stevie^Lamb^^^',
       authorInstitution: [Array],
       authorRole: 'Neurologist',
       authorSpecialty: 'Neurology' } ],
  availabilityStatus: 'urn:oasis:names:tc:ebxml-regrep:StatusType:Approved',
  classCode:
   { codingScheme: '1.3.6.1.4.1.19376.1.2.6.1',
     displayName: 'Treatment Plan or Protocol' },
  comment: '',
  confidentialityCode:
   { codingScheme: '2.16.840.1.113883.5.25',
     displayName: 'Restricted' },
  creationTime: '20070101',
  entryUUID: 'Document02',
  eventCodeList:
   [ { codingScheme: '1.3.6.1.4.1.21367.2017.3',
       displayName: 'Foundational Connectathon Read-Access Policy' },
     { codingScheme: '1.3.6.1.4.1.21367.2017.3',
       displayName: 'FULL ACCESS TO ALL POLICY' } ],
```

```
========================================
Calling for full document...
Accounts available on this node:
0xed9d02e382b34818e88B88a309c7fe71E65f419d
Originally deployed with account:0xed9d02e382b34818e88B88a309c7fe71E65f419d
Contract Address: 0x1932c48b2bF8102Ba33B4A6B545C32236e342f34
Returning document: 2
========================================
Return type: FindDocuments,LeafClass
========================================
{ 'query:AdhocQueryResponse':
   { '$':
      { status: 'Success',
        'xmlns:query': 'urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0',
        'xmlns:rim': 'urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0',
        'xmlns:xsi': 'http://www.w3.org/2001/XMLSchema-instance',
        'xsi:schemaLocation': 'urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0 ../../schem
a/ebRS/query.xsd' },
     'rim:RegistryObjectList': [ [Object] ] } }
Responsing query result:
----------------------
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<query:AdhocQueryResponse status="Success" xmlns:query="urn:oasis:names:tc:ebxml-regrep:
xsd:query:3.0" xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" xmlns:xsi="http:/
/www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:oasis:names:tc:ebxml-regrep
:xsd:query:3.0 ../../schema/ebRS/query.xsd">
  <rim:RegistryObjectList>
    <rim:ExtrinsicObject id="Document02" mimeType="text/plain" objectType="urn:uuid:7edc
a82f-054d-47f2-a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:App
roved">
```

```
========================================
Done!
Execution time (hr): 0s 445.282259ms
========================================
CLOSED: 127.0.0.1 41192
```

Figure 4-64 XDS Document Registry responding search result back to XDS Document Consumer

```
Query response received:
{ DocumentEntry:
   { author: [ [Object], [Object] ],
     availabilityStatus: 'urn:oasis:names:tc:ebxml-regrep:StatusType:Approved',
     classCode:
      { codingScheme: '1.3.6.1.4.1.19376.1.2.6.1',
        displayName: 'Treatment Plan or Protocol' },
     comment: '',
     confidentialityCode:
      { codingScheme: '2.16.840.1.113883.5.25',
        displayName: 'Restricted' },
     creationTime: '20070101',
     entryUUID: 'Document02',
     eventCodeList: [ [Object], [Object] ],
     formatCode:
      { codingScheme: '1.3.6.1.4.1.19376.1.2.3',
        displayName: 'urn:ihe:iti:bppc:2007' },
     hash: '8cd7c25aa2526918fef504fea46b79a3ebf123db',
     healthcareFacilityTypeCode:
      { codingScheme: '2.16.840.1.113883.6.96',
        displayName: 'Hospital-trauma center' },
     homeCommunityId: 'N/A',
     languageCode: 'en-us',
     legalAuthenticator: 'N/A',
     limitedMetadata: 'N/A',
     mimeType: 'text/plain',
     objectType: 'urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1',
     patientId: 'IHEBLUE-2736^^^&1.3.6.1.4.1.27829.13.20.3000&ISO',
     practiceSettingCode:
      { codingScheme: '1.3.6.1.4.1.21367.2017.3',
        displayName: 'Neurology' },
     referenceIdList: 'N/A',
     repositoryUniqueId: '1.19.6.24.110.42.1',
     serviceStartTime: '200701011800',
     serviceStopTime: '200701011900',
     size: '4',
     sourcePatientId: '6b12add33^^^&1.3.4.5&ISO',
     sourcePatientInfo:
      [ 'PID-3|pid1^^^&1.2.3&ISO',
        'PID-5|Emile^Sheehan^^^',
        'PID-7|19560527',
        'PID-8|M',
        'PID-11|3094  Glen St^^Paducah^KY^42003^USA' ],
     title: 'DocB',
     typeCode:
      { codingScheme: '2.16.840.1.113883.6.1',
        displayName: 'LABORATORY REPORT.TOTAL' },
     uniqueId: '1.2.42.17115670011797.30',
     URI: 'N/A' } }
==============================
Execution time (hr): 0s 497.036633ms
==============================
Connection closed
```

Figure 4-65 XDS Document Consumer received search result and display it to the user

4.4 Evaluation

The goal of the evaluation is to test the functionalities and the performance of the implemented system. This including the Document Register function and Document Query function. The evaluation will indicate the compatibility of the implemented system to an actual healthcare operation environment. The implemented system should be able to sustain a huge amount of data that continuously flows through the system without a failure. Each setup will be tested with the mockup transactions and measure the processing time required for the system to complete the specific process.

There are ten transactions created for the experiment as test samples. Figure 4-66 and Figure 4-67 show an example of the mockup transaction created for the experiment and its full content can be further inspected in the Appendix Section B. In each transaction have its metadata attributes modified varied in each transaction for the test as shown in Table 4-1. Each transaction will have about the same file size but different metadata attributes values to prove functionalities of the Document Query function. All experiments will be tested with these transaction samples resulting as 10 times test for each setup ("transaction number #n" will be referred to as "Document#n").

Table 4-1 Modified metadata attributes in each sample transaction

Metadata Attributes	Modify Pattern
DocumentEntry.patientId	1.3.6.1.4.1.xxxx.13.20.3000&ISO Replace “xxxx” with 5 digits random number
DocumentEntry.repositoryUniqueId	1.19.6.24.xxx.42.1 Replace “xxx” with random number
DocumentEntry.hash	Replace with md5 random hash number
DocumentEntry.creationTime	Replace Date Time differ from other transaction
DocumentEntry.serviceStartTime	Replace Date Time differ from other transaction
DocumentEntry.serviceStopTime	Replace Date Time differ from other transaction
DocumentEntry.sourcePatientId	xxxxxxxxx^^^&1.3.4.5&ISO Replace “xxxxxxxxx” with random value
DocumentEntry.sourcePatientInfo	Random generated personal info.
DocumentEntry.author	Random generated personal info.
DocumentEntry.healthcareFacilityTypeCode	Randomly chosen from Facility Type and Practice Setting list
DocumentEntry.practiceSettingCode	

```xml
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
   <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
      <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
      <wsa:MessageID soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
      <wsa:Action soapenv:mustUnderstand="true">urn:ihe:itl:2007:RegisterDocumentSet-b</wsa:Action>
   </soapenv:Header>
   <soapenv:Body>
      <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
         <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
            <rim:ExtrinsicObject id="Document01" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1"
status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
               <rim:Slot name="size">
                  <rim:ValueList>
                     <rim:Value>4</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="repositoryUniqueId">
                  <rim:ValueList>
                     <rim:Value>1.19.6.24.109.42.1</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="hash">
                  <rim:ValueList>
                     <rim:Value>e543712c0e10501972de13a5bfcbe826c49feb75</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="creationTime">
                  <rim:ValueList>
                     <rim:Value>20061224</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="languageCode">
                  <rim:ValueList>
                     <rim:Value>en-us</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="serviceStartTime">
                  <rim:ValueList>
                     <rim:Value>200612230800</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="serviceStopTime">
                  <rim:ValueList>
                     <rim:Value>200612230900</rim:Value>
                  </rim:ValueList>
```


Figure 4-66 Part of transaction samples content [59]

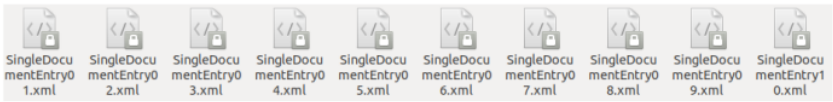


Figure 4-67 Ten of mockup transactions generated for the experiment

4.4.1 Functionalities Test

4.4.1.1 Functionalities under the normal circumstance

As mentioned in Section 4.3, The main functions of the implemented system are Document Register and Document Query. So, this setup will test the functionalities of each function under the normal circumstance where there are seven active Blockchain nodes and there is one node performing the task.

4.4.1.1.1 Document Register Functionalities

The criteria to check if the Document Register function correctly is the registered document must be discoverable within the Blockchain ledger when its smartcontract was called. This smartcontract must return metadata attribute values of its corresponding document when it was called by the system.

```
Choose documents: 01
CONNECTED TO: 127.0.0.1:65519
Sent:
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
   <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
      <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
      <wsa:MessageID soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373<
      <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:
   </soapenv:Header>
   <soapenv:Body>
      <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
         <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
            <rim:ExtrinsicObject id="Document01" mimeType="text/plain" objectType="urn:uuid:
86c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
               <rim:Slot name="size">
                  <rim:ValueList>
                     <rim:Value>4</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="repositoryUniqueId">
                  <rim:ValueList>
                     <rim:Value>1.19.6.24.109.42.1</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="hash">
                  <rim:ValueList>
                     <rim:Value>e543712c0e10501972de13a5bfcbe826c49feb75</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="creationTime">
                  <rim:ValueList>
```

Figure 4-68 The XDS Document Repository initiate Document Register function

```
XDS Document Registry Actor listening on 127.0.0.1:65519
(node:4311) Warning: N-API is an experimental feature and could change at any time.
CONNECTED: 127.0.0.1:35778
Received data....
XML:
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
   <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
      <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
      <wsa:MessageID soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</
      <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:A
   </soapenv:Header>
   <soapenv:Body>
      <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
         <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
            <rim:ExtrinsicObject id="Document01" mimeType="text/plain" objectType="urn:uuid:7
2-a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
               <rim:Slot name="size">
                  <rim:ValueList>
                     <rim:Value>4</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
               <rim:Slot name="repositoryUniqueId">
                  <rim:ValueList>
                     <rim:Value>1.19.6.24.109.42.1</rim:Value>
                  </rim:ValueList>
               </rim:Slot>
```

Figure 4-69 The XDS Document Registry received ITI-42 transaction

Figure 4-68 show the XDS Document Repository Actor initiate Document Register function to register Document01 into the Blockchain ledger. Then, in Figure 4-69, the XDS Document Registry Actor received a transaction from the XDS Document Repository Actor and publish the metadata attributes set of the document into the Blockchain ledger. In Figure 4-70, the XDS Document Registry Actor signal that it successfully published the metadata attributes set into the Blockchain ledger using smartcontract.

```
Accounts available on this node:
0xed9d02e382b34818e88B88a309c7fe71E65f419d
-----------------------------------------
Deploying with account:0xed9d02e382b34818e88B88a309c7fe71E65f419d
Contract Address: 0x1932c48b2bF8102Ba33B4A6B545C32236e342f34
CLOSED: 127.0.0.1 35778
{ blockHash: '0xc09af905916fde64b932276e3f9d431b76a35187b26acf3b4777d1c94aceb6b8',
  blockNumber: 30,
  contractAddress: null,
  cumulativeGasUsed: 9450604,
  from: '0xed9d02e382b34818e88b88a309c7fe71e65f419d',
  gasUsed: 9450604,
  logsBloom: '0x0000000000
```

```
--------------------------------
{ DocumentEntry:
   { author: [ [Object], [Object] ],
     availabilityStatus: 'urn:oasis:names:tc:ebxml-regrep:StatusType:Approved',
     classCode:
      { codingScheme: '1.3.6.1.4.1.19376.1.2.6.1',
        displayName: 'Treatment Plan or Protocol' },
     comment: '',
     confidentialityCode:
      { codingScheme: '2.16.840.1.113883.5.25',
        displayName: 'Restricted' },
     creationTime: '20061224',
     entryUUID: 'Document01',
     eventCodeList: [ [Object], [Object] ],
     formatCode:
      { codingScheme: '1.3.6.1.4.1.19376.1.2.3',
        displayName: 'urn:ihe:iti:bppc:2007' },
     hash: 'e543712c0e10501972de13a5bfcbe826c49feb75',
     healthcareFacilityTypeCode:
      { codingScheme: '2.16.840.1.113883.6.96',
        displayName: 'Private home-based care' },
     homeCommunityId: 'N/A',
     languageCode: 'en-us',
     legalAuthenticator: 'N/A',
     limitedMetadata: 'N/A',
     mimeType: 'text/plain',
     objectType: 'urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1',
     patientId: 'IHEBLUE-2736^^^&1.3.6.1.4.1.21367.13.20.3000&ISO',
     practiceSettingCode:
      { codingScheme: '1.3.6.1.4.1.21367.2017.3',
        displayName: 'Pathology' },
     referenceIdList: 'N/A',
     repositoryUniqueId: '1.19.6.24.109.42.1',
     serviceStartTime: '200612230800',
     serviceStopTime: '200612230900',
     size: '4',
     sourcePatientId: '89765a87b^^^&1.3.4.5&ISO',
     sourcePatientInfo: 'Anonymized',
     title: 'DocA',
     typeCode:
      { codingScheme: '2.16.840.1.113883.6.1',
        displayName: 'LABORATORY REPORT.TOTAL' },
     uniqueId: '1.2.42.20190405034511.30',
     URI: 'N/A' },
```


Figure 4-71 The smartcontract return metadata attributes correctly when called

4.4.1.1.2 Document Query Functionalities

The criteria to check if the Document Register function correctly is when users input corrected search keywords that match one of the documents registered within the Blockchain ledger, the XDS Document Registry actor must be able to find the matching document and return the search result to the XDS Document Consumer Actor to display it to the user correctly.

```
==================================
|| XDS Consumer Actor Interface ||
==================================
Please select query type
1) FindDocuments
2) FindSubmissionSets
3) FindFolders
4) GetAll
5) GetDocuments
6) GetFolders
7) GetAssociations
8) GetDocumentsAndAssociations
9) GetSubmissionSets
10) GetSubmissionSetAndContents
11) GetFolderAndContents
12) GetFoldersForDocument
13) GetRelatedDocuments
14) FindDocumentsByReferenceId
#) Quit
(Specify number): 5
Query Type: GetDocuments

Please select the required keyword
1) XDSDocumentEntryEntryUUID
2) XDSDocumentEntryUniqueId
Value: 1
Keyword: XDSDocumentEntryEntryUUID
Value: Document01
==========================
Query type: GetDocuments
Query keywords:
$XDSDocumentEntryEntryUUID = Document01
==========================
Available optional keywords:
0) No more optional keywords
1) XDSDocumentEntryHomeCommunityId
#) Quit
Select keywords (specify number): 0
==========================
All keywords set...
==========================
```

Figure 4-72 The XDS Document Consumer initiate Document Query

Figure 4-72 show the XDS Document Consumer user interface that prompts the user to input search keywords matching the document they are seeking. Figure 4-72 also show the list of search keywords that match with the Document01.

```
XDS Document Registry Actor listening on 127.0.0.1:65519
(node:4528) Warning: N-API is an experimental feature and could change at any time.
CONNECTED: 127.0.0.1:35780
Received data....
XML:
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<query:AdhocQueryRequest xmlns:query="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0" xmlns:rim
="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0" xmlns:rs="urn:oasis:names:tc:ebxml-regrep:xsd:r
s:3.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:oasis:nam
es:tc:ebxml-regrep:xsd:query:3.0 ../../schema/ebRS/query.xsd">
  <query:ResponseOption returnComposedObjects="true" returnType="LeafClass"/>
  <rim:AdhocQuery id="urn:uuid:5c4f972b-d56b-40ac-a5fc-c8ca9b40b9d4">
    <rim:Slot name="$XDSDocumentEntryEntryUUID">
      <rim:ValueList>
        <rim:Value>Document01</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:AdhocQuery>
</query:AdhocQueryRequest>

Converted to object:
----------------------
{ 'query:AdhocQueryRequest':
   { '$':
      { 'xmlns:query': 'urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0',
        'xmlns:rim': 'urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0',
        'xmlns:rs': 'urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0',
        'xmlns:xsi': 'http://www.w3.org/2001/XMLSchema-instance',
        'xsi:schemaLocation': 'urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0 ../../schema/ebRS
/query.xsd' },
     'query:ResponseOption': [ [Object] ],
     'rim:AdhocQuery': [ [Object] ] } }
--------------------
Query requested...
Specify responseOption
returnComposedObjects = true
Request response as LeafClass...
```

Figure 4-73 The XDS Document Registry Actor received the query

Figure 4-73 show the XDS Document Registry Actor received the query from the XDS Document Consumer Actor and initiate search program. In Figure 4-74, the XDS Document Consumer found the matching document as the search keywords provided. The XDS Document Registry Actor then returns the search result to the XDS Document Consumer Actor as also shown in Figure 4-74.

```
========================================
LeafClass
1 Documents matched....
Matched result included (Document ID):1
Calling for full document....
Accounts available on this node:
0xed9d02e382b34818e88B88a309c7fe71E65f419d
Originally deployed with account:0xed9d02e382b34818e88B88a309c7fe71E65f419d
Contract Address: 0x1932c48b2bF8102Ba33B4A6B545C32236e342f34
Returning document: 1
Responsing as LeafClass
========================================
Return type: GetDocuments,LeafClass
========================================
```

Figure 4-74 The XDS Document Registry Actor found the matching document and about to return the result to the XDS Document Consumer Actor

Figure 4-75 show the XDS Document Consumer Actor received the search result returned from the XDS Document Registry Actor and display the result to the user. In this case, the result show metadata attributes set for the Document01 correctly. This proves that the Document Query is functioning correctly.


```
==============================
Query response received:
{ DocumentEntry:
   { author: [ [Object], [Object] ],
     availabilityStatus: 'urn:oasis:names:tc:ebxml-regrep:StatusType:Approved',
     classCode:
      { codingScheme: '1.3.6.1.4.1.19376.1.2.6.1',
        displayName: 'Treatment Plan or Protocol' },
     comment: '',
     confidentialityCode:
      { codingScheme: '2.16.840.1.113883.5.25',
        displayName: 'Restricted' },
     creationTime: '20061224',
     entryUUID: 'Document01',
     eventCodeList: [ [Object], [Object] ],
     formatCode:
      { codingScheme: '1.3.6.1.4.1.19376.1.2.3',
        displayName: 'urn:ihe:iti:bppc:2007' },
     hash: 'e543712c0e10501972de13a5bfcbe826c49feb75',
     healthcareFacilityTypeCode:
      { codingScheme: '2.16.840.1.113883.6.96',
        displayName: 'Private home-based care' },
     homeCommunityId: 'N/A',
     languageCode: 'en-us',
     legalAuthenticator: 'N/A',
     limitedMetadata: 'N/A',
     mimeType: 'text/plain',
     objectType: 'urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1',
     patientId: 'IHEBLUE-2736^^^&1.3.6.1.4.1.21367.13.20.3000&ISO',
     practiceSettingCode:
      { codingScheme: '1.3.6.1.4.1.21367.2017.3',
        displayName: 'Pathology' },
     referenceIdList: 'N/A',
     repositoryUniqueId: '1.19.6.24.109.42.1',
     serviceStartTime: '200612230800',
     serviceStopTime: '200612230900',
     size: '4',
     sourcePatientId: '89765a87b^^^&1.3.4.5&ISO',
     sourcePatientInfo:
      [ 'PID-3|pid1^^^&1.2.3&ISO',
        'PID-5|Doe^John^^^',
        'PID-7|19560527',
        'PID-8|M',
```


Figure 4-75 The XDS Document Consumer received the query result and display it to the user

4.4.1.2 Functionalities on the different setup of active node numbers.

On the hypothesis that a changed number of active nodes may cause a change in the functionalities of the XDS Blockchain, to test the functionalities, the experiment performs by varying the number of nodes and test if the system could perform any function. Start from the setup with seven active nodes as the reference, followed by six, five, four, three, two, and one active node for the variation.

To create a situation where there are fewer than seven active nodes, the 7-Nodes Example configuration must be modified to meet the situation. In Quorum 7-Nodes Example, the number of active nodes was defined in the "permissioned-nodes.json" file as shown in Figure 4-54. Each active node represents by its "enode" identifier number together with its communicating TCP port number as shown in Figure 4-55. The node will become inactive if its enode number was excluded from the file. That means, in each setup for a specific number of nodes, enode number of inactive nodes must be excluded from the file before initiate the 7-Nodes Blockchain network. At the same time, the "numNodes" variable value in the "istanbul-init.sh" file must be set to the specific number of active nodes (i.e., numNodes = 6 for six active nodes) as shown in Figure 4-56 to declare that there will be file directory prepared for the specified number of nodes at the initiation of the 7-Node Blockchain network.

After the configuration was modified, the 7-Nodes Example will automatically generate Blockchain nodes to meet with the specified number upon the initiation of the Blockchain ledger. That means every time the configuration file was modified for a new setup, the Blockchain network must be reset and re-initiated to ensure that there is no conflict with another setup which could affect the accuracy of the experiment result.

```
Open ▾   *permissioned-nodes.json   Save
~/Documents/7NodesTry/quorum-examples/examples/7nodes
[
  "enode://ac6b1096ca56b9f6d004b779ae3728bf83f8e2245@127.0.0.1:21000?discport=0&raftport=50401",
  "enode://0ba6b9f606a43a95edc6247cdb1c1e105145817be@127.0.0.1:21001?discport=0&raftport=50402",
  "enode://579f786d4e2830bbcc02815a27e8a9bacccc9605d@127.0.0.1:21002?discport=0&raftport=50403",
  "enode://3d9ca5956b38557aba991e31cf510d4df641dce9c@127.0.0.1:21003?discport=0&raftport=50404",
  "enode://3701f007bfa4cb26512d7df18e6bbd202e8484a6e@127.0.0.1:21004?discport=0&raftport=50405",
  "enode://eacaa74c4b0e7a9e12d2fe5fee6595eda841d6d99@127.0.0.1:21005?discport=0&raftport=50406",
  "enode://239c1f044a2b03b6c4713109af036b775c5418fe4@127.0.0.1:21006?discport=0&raftport=50407"
]
JSON ▾   Tab Width: 8 ▾   Ln 8, Col 53   INS
```

Figure 4-76 Content of "permissioned-nodes.json" file define active nodes

(node ids truncated for simpler explanation) [33]

```
"enode://ac6b1096ca56b9f6d004b779ae3728bf83f8e2245@127.0.0.1:21000?discport=0&raftport=50401",
```

Figure 4-77 Single node id represent single active node

```
#!/bin/bash
set -u
set -e

function usage() {
  echo ""
  echo "Usage:"
  echo "    $0 [--istanbulTools] [--numNodes numberOfNodes]"
  echo ""
  echo "Where:"
  echo "    --istanbulTools will perform initialisation from data generated using"
  echo "      istanbul-tools (note that permissioned-node.json and istanbul-genesis.json"
  echo "      files will be overwritten)"
  echo "    numberOfNodes is the number of nodes to initialise (default = $numNodes)"
  echo ""
  exit -1
}

istanbulTools="false"
numNodes=6
while (( "$#" )); do
    case "$1" in
        --istanbulTools)
            istanbulTools="true"
            shift
            ;;
        --numNodes)
            re='^[0-9]+$'
            if ! [[ $2 =~ $re ]] ; then
                echo "ERROR: numberOfNodes value must be a number"
                usage
            fi
            numNodes=$2
            shift 2
            ;;
```

Figure 4-78 "numNodes" variable in "istanbul-init.sh" file reassigned with new value

Table 4-2 shows that the system result from the implementation can function normally with 7, 6, and 5 active nodes. The system stops functioning when there are active nodes lesser than 5.

Table 4-2 Functionalities experiment result

	7 Nodes	6 Nodes	5 Nodes	4 Nodes	3 Nodes	2 Nodes	1 Nodes
Smartcontract Deployment	✓	✓	✓	×	×	×	×
Document Register	✓	✓	✓	×	×	×	×
Document Query	✓	✓	✓	×	×	×	×

After looking into the root cause of the result, it is turn out that the 7-Nodes Example cannot resolve the Block validation process cycle in a situation with fewer than 5 active nodes. Figure 4-79 shows the content of the log file monitoring the 7-Nodes example process where the upper red rectangle shows the beginning of the network while the lower red rectangle shows the process stopped at the initiation of the first Block validation process. The log shows that the Block validation process cycle which would repeat continuously and endlessly simply stopped at the first Block and there is no further upcoming process appeared in the log or none of any error notification appeared.

Noteworthy, this is not even related to the condition of IBFT consensus where it requires at least 2/3 of all nodes to vote for the same Block version to complete the validation process. The system should be able to operate normally without error even there are fewer than 5 active nodes when all active nodes agreeing on the same Block version and there is no "bad actor node" present in the system to propose a falsified Block version that interrupts the vote

We think that a possible reason could be that the 7-Nodes example may not be developed for the situation with 4,3,2, or 1 active node as it was only built to aid Smartcontract developers to easily deploy and test their Smartcontract in the proper 7 active nodes simulated environment closely similar to the actual Blockchain network.

```
Open ▾   1.log   ~/Documents/7NodesTry/quorum-examples/examples/7nodes/qdata/logs   Save
nohup: appending output to 'nohup.out'
INFO [07-07|14:26:31.914] Maximum peer count                       ETH=25 LES=0 total=25
INFO [07-07|14:26:31.915] Starting peer-to-peer node               instance=Geth/v1.8.18-stable-a603e745(quorum-v2.4.0)/linux-amd64/go1.10.4
INFO [07-07|14:26:31.915] Allocated cache and file handles         database=/home/pcep/Documents/7NodesTry/quorum-examples/examples/7nodes/
qdata/dd1/geth/chaindata cache=768 handles=2048
INFO [07-07|14:26:32.001] Initialised chain configuration          config="{ChainID: 10 Homestead: 0 DAO: <nil> DAOSupport: false EIP150: 0
EIP155: 0 EIP158: 0 Byzantium: 0 IsQuorum: true Constantinople: 0 TransactionSizeLimit: 64 MaxCodeSize: 35 Engine: istanbul}"
INFO [07-07|14:26:32.001] Initialising Ethereum protocol           versions="[63 62]" network=10
INFO [07-07|14:26:32.002] Loaded most recent local header          number=0 hash=85c72c…e7aec8 td=1 age=52y3mo5d
INFO [07-07|14:26:32.002] Loaded most recent local full block      number=0 hash=85c72c…e7aec8 td=1 age=52y3mo5d
INFO [07-07|14:26:32.002] Loaded most recent local fast block      number=0 hash=85c72c…e7aec8 td=1 age=52y3mo5d
INFO [07-07|14:26:32.002] Regenerated local transaction journal    transactions=0 accounts=0
INFO [07-07|14:26:32.042] IPC endpoint opened                      url=/home/pcep/Documents/7NodesTry/quorum-examples/examples/7nodes/qdata/
dd1/geth.ipc
INFO [07-07|14:26:32.043] HTTP endpoint opened                     url=http://
0.0.0.0:22000                                                      cors=* vhosts=*
WARN [07-07|14:26:32.043] ------------------------------------------------------------------------
WARN [07-07|14:26:32.043] Referring to accounts by order in the keystore folder is dangerous!
WARN [07-07|14:26:32.043] This functionality is deprecated and will be removed in the future!
WARN [07-07|14:26:32.043] Please use explicit addresses! (can search via `geth account list`)
WARN [07-07|14:26:32.043] ------------------------------------------------------------------------
INFO [07-07|14:26:32.058] New local node record                    seq=1 id=7577115fd9184a27 ip=127.0.0.1 udp=0 tcp=21000
INFO [07-07|14:26:32.058] Started P2P networking                   self="enode://
ac6b1096ca56b9f6d004b779ae3728bf83f8e22453404cc3cef16a3d9b96608bc67c4b30db88e0a5a6c6390213f7acbe1153ff6d23ce57380104288ae19373ef@127.0.0.1:21
discport=0"
WARN [07-07|14:26:32.058] Removing static dial candidate           id=0x8c5d60         addr=127.0.0.1:21000 err="is self"
INFO [07-07|14:26:35.605] Unlocked account                         address=0xed9d02e382b34818e88B88a309c7fe71E65f419d
INFO [07-07|14:26:35.605] Transaction pool price threshold updated price=1000000000
INFO [07-07|14:26:35.605] Transaction pool price threshold updated price=1000000000
INFO [07-07|14:26:35.605] Commit new mining work                   number=1 sealhash=013766…01880d uncles=0 txs=0 gas=0 fees=0
elapsed=67.609µs
INFO [07-07|14:28:22.279] Setting new local account                address=0xed9d02e382b34818e88B88a309c7fe71E65f419d
INFO [07-07|14:28:22.279] Submitted contract creation              fullhash=0x047145f40bd5476d3fc7a6fd6cad9ac9306254d81e7b074d77d6261ad7ffab75 to=0x1932c48b2bF8102Ba33B4A6B545C32236e342f34
INFO [07-07|14:28:23.617] Commit new mining work                   number=1 sealhash=36eb9a…e89085 uncles=0 txs=1 gas=459309 fees=0
elapsed=384.074µs
Plain Text ▾   Tab Width: 8 ▾   Ln 27, Col 53   INS
```

Figure 4-79 The 7-Node Example system log

It also must be noted that it is possible for the case that contains an even number of nodes may cause failure in Block validation attempt due to the majority of Block approval from nodes not meet the least requirement of 2/3 of all nodes in the network. For example, if ten nodes are participating, five nodes proposed Block named 'ABC' while another five nodes proposed Block named 'CBA'. This example will result in failed Block validation attempt. If this happened, the network was expected to try another attempt until the majority meet the least requirement of 2/3 and all the remaining transactions were successfully added into the valid Block and published into the Blockchain ledger. However, the evaluation of the implementation will not cover the case due to complications in creating the “bad actor” node. We assume that the feature was not directly included with Ethereum, Quorum, or its 7-Nodes Example as it was not created for a malicious purpose so, it is too difficult for us to create the situation that half of the participate nodes attempt to propose falsified Block for the evaluation.

4.4.2 Performance Test

On the hypothesis that a changed number of active nodes or change in the amount of XDS transactions entering the system may cause a change in the performance of the XDS Blockchain, the performance test will be performed by triggering the Document Register function and/or the Document Query function depends on the setup. The prepared transaction samples will be tested on the setup with a specific number of nodes then measure the time the system took to complete the process. Additionally, to increase the accuracy of experiment results and reduce the effect from other factors outside the consideration, the experiment will be repeated ten times on each transaction sample. The final performance result of each transaction will be derived from the average values of these ten times repeated.

4.4.2.1 Test if the number of transactions would cause a change in performance of the XDS Document Registry to perform the Document Register function.

This setup is performed by trigger the XDS Document Repository Actor to start the Document Register function which will send the ITI-42 transaction containing metadata attributes set of the selected document (Doc01 - Doc10) to the XDS Document Registry Actor. The processing time the XDS Document Registry took to complete the process publishing the metadata attributes set into the Blockchain ledger will be measured for the experiment result.

The "Single Node" line was the averaged processing time monitored at the 1st Node performing the Document Register function when there is only one node performing the function. The "All Nodes" line was the average processing time monitored at the 1st Node performing the Document Register function when all active nodes performed the function within the same time. Both setups measure the processing time the node used to complete the Document Register function since the moment it received the ITI-42 transaction from the XDS Document Repository actor.

Figure 4-80 shows that when all nodes perform the Document Register function at the same time, it took a longer time to complete the function when compared to the situation when there is only a single node perform the Document Register function. The raw results can be further inspected in Appendix Section C.

Evidently, the nature of the 7-Nodes Example which its Blockchain nodes need to share processing units of the same host machine could be the main factor affecting the performance. The more processing threads executed together at once mean the more processor resource of the host machine divided for each thread. So, the shared processing unit environment may be affecting the performance of each node resulting in a difference in performance as shown in Figure 4-80.

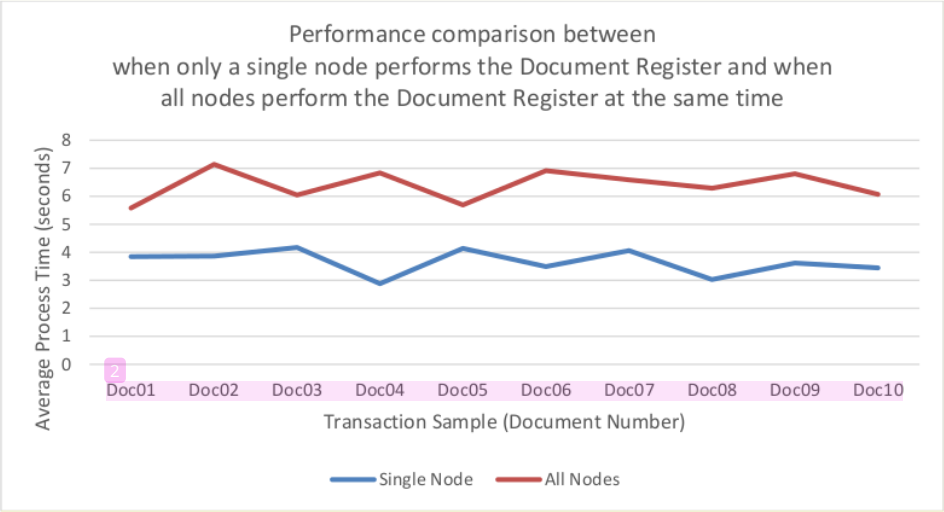


Figure 4-80 Performance comparison of the system performing the Document Register function between when performed by only a single node and when performed by all nodes at the same time (without triggering the Document Query function at the time)

4.4.2.2 Test if the number of search keywords would cause a change in performance of Document Query function.

This setup is performed by trigger the XDS Document Consumer Actor to start the Document Query function which prompts for input from the user. The user then selected for “FindDocument” query type and inputs metadata attributes values of each transaction sample (Doc01 – Doc10) into the program. The program then proceeds to query for the matching metadata set registered in the Blockchain ledger from the XDS Document Registry Actor in the 1st Node (only one active node received the query). The processing time since the moment the XDS Document Consumer Actor sent the ITI-18 transaction to the XDS Document Registry Actor until the XDS Document Consumer received the search result from the XDS Document Registry will be measured for the experiment result. The "minimum" search keywords will include only two attribute values while the "maximum" search keywords will include fifteen attribute values. The setup for “Single Node” and “All Nodes” is like Section 4.4.2.1.

The Document Query with the "FindDocument" query type performed by a single node using minimum keywords took an average process time of 285.2455158 milliseconds to complete the function. FindDocument with maximum keywords took an average process time of 287.7405654 milliseconds to complete the function. Performed by all active nodes, minimum keywords took an average process time of 754.1227332 milliseconds to complete the function while maximum keywords took an average process time of 767.9703297 milliseconds to complete the function. Figure 4-81 and Figure 4-82 show no significant difference between the Document Query with maximum search keywords compared to the Document Query with minimum search keywords. That means the amount of search keywords input is not affecting the performance of the Document Query function. A full version of the experiment result can be further inspected in the Appendix section.

The Document Query function program only used provided search keywords to search for the matching transaction registered in the Blockchain ledger. A greater number of search keywords may take a little more process to finish the search, but the change will not add more significant processing time to the whole process which was expected not to affect the performance of the system.

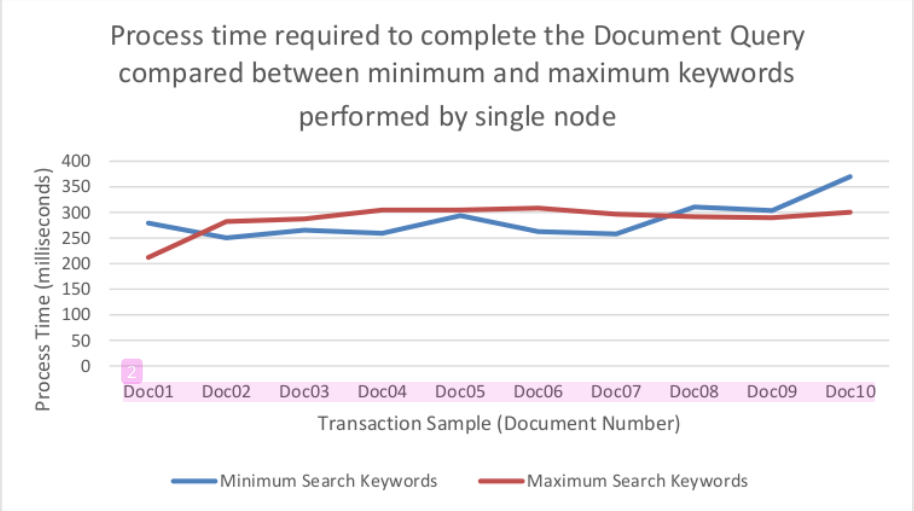


Figure 4-81 Processing time to complete Document Query compared between minimum keywords and maximum keywords performed by single node

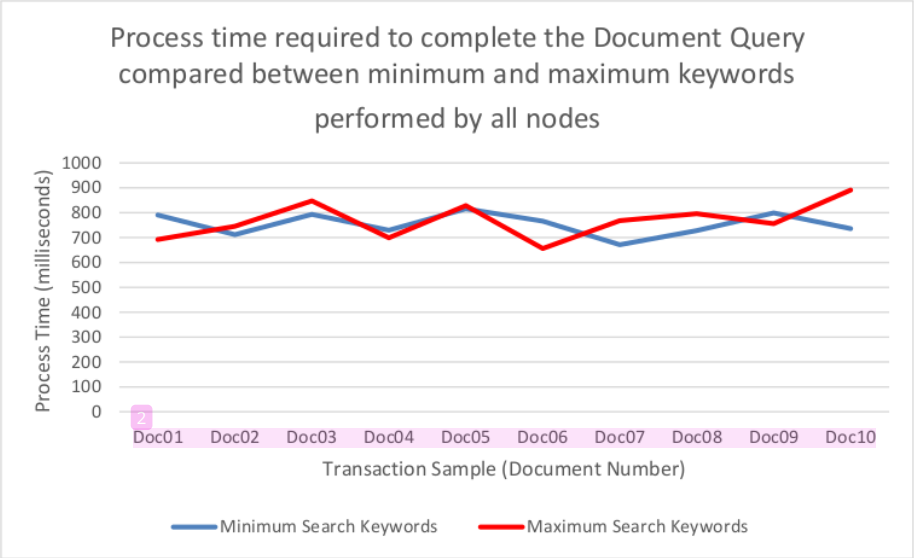


Figure 4-82 Processing time to complete Document Query compared between minimum keywords and maximum keywords performed by all nodes

4.4.2.3 Test if different query types would cause a change in performance of Document Query function.

This setup is performed by trigger the XDS Document Consumer Actor to start the Document Query function which prompts for input from the user. The user then selected for query type and inputs metadata attributes values of each transaction sample (Doc01 – Doc10) into the program. The program then proceeds to query for the matching metadata set registered in the Blockchain ledger from the XDS Document Registry Actor in the 1st Node (only one active node received the query). The processing time since the moment the XDS Document Consumer Actor sent the ITI-18 transaction to the XDS Document Registry Actor until the XDS Document Consumer received the search result from the XDS Document Registry will be measured for the experiment result. The test begins with the "FindDocument" query type with minimum keywords before the "GetDocument" query type with required keyword (1 keyword). The result then compares the performance of each query type. The setup for “Single Node” and “All Nodes” situation is like Section 4.4.2.1.

The Document Query with the "FindDocument" query type performed by a single node using minimum keywords took an average process time of 285.2455158 milliseconds to complete the function. The "GetDocument" took an average process time of 266.3474373 milliseconds to complete the function. Performed by all active nodes, the "FindDocument" took an average process time of 754.1227332 milliseconds to complete the function while the "GetDocument" took an average process time of 765.7761302 milliseconds to complete the function. Figure 4-83 and Figure 4-84 show no significant difference between the Document Query with the "FindDocument" query type compared to the Document Query with the "GetDocument" query type. That means the query type is not affecting the performance of the Document Query function. A full version of the experiment result can be further inspected in the Appendix section.

The different query types only change the end content of the search result while remain operates on the same base algorithm, so the change will not add more significant processing time to the whole process which was expected not to affect the performance of the system.

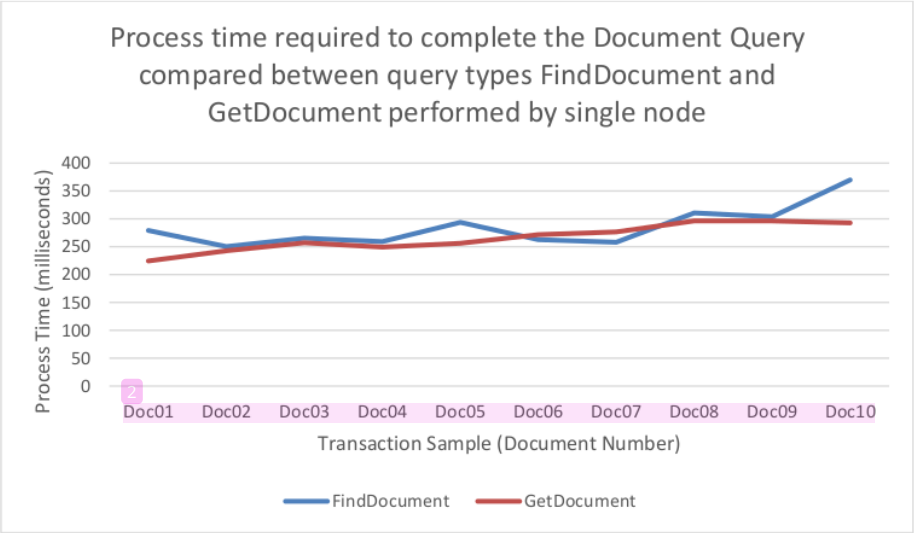


Figure 4-83 Processing time to complete Document Query compared between FindDocument and GetDocument performed by single node

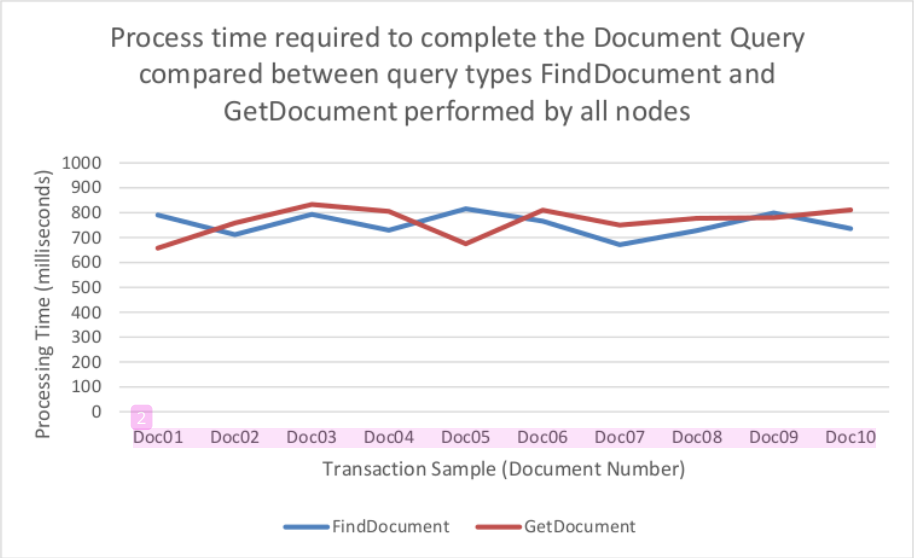


Figure 4-84 Processing time to complete Document Query compared between FindDocument and GetDocument performed by all nodes

4.4.2.4 Test if the number of transactions would cause a change in performance of Document Query function.

This setup compares all graphs resulting from Section 4.4.2.2 and 4.4.2.3 to show the overall difference between the case when there is only one single node (1st Node) perform the Document Query function and when all active nodes perform the Document Query function (inspect process time from 1st Node).

Figure 4-85 shows a significant difference between the setup when the Document Query function is performed by a single node and the setup when the Document Query function is performed by all active nodes. When transactions are entering all nodes at the same time, the system took a significantly longer time to complete the Document Query function when compared to the processing time when there is only one transaction from a single node performing the function. That means the number of transactions is affecting the performance of the Document Query function. A full version of the experiment result can be further inspected in the Appendix section.

Like Section 4.4.2.1, this could be the effect from the 7-Nodes Example which its Blockchain nodes need to share processing units of the same host machine. This may be the main factor affecting the performance when all nodes need to perform the Document Query function at once at the same time. So, the shared processing unit environment may be affecting the performance of each node resulting in a difference in performance as shown in Figure 4-85.

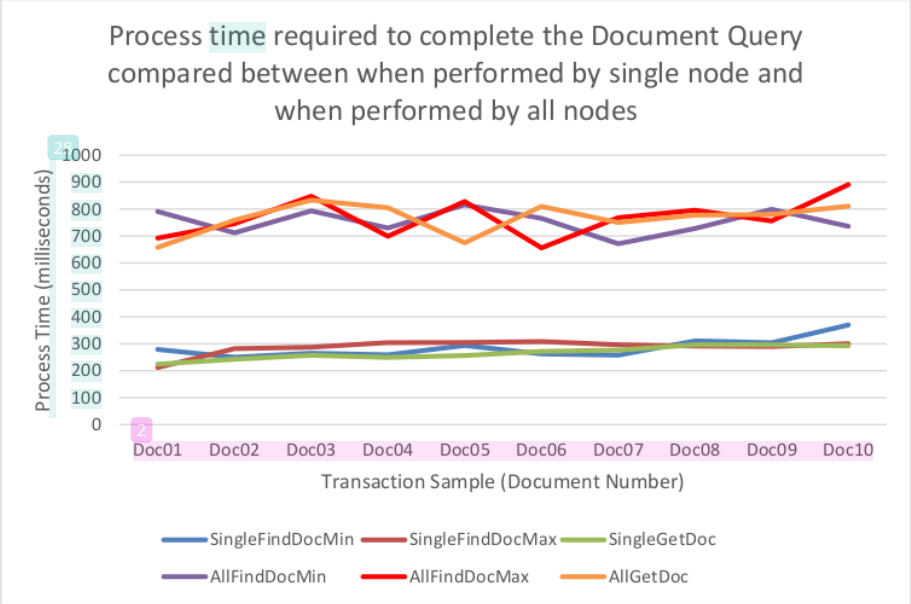


Figure 4-85 Processing time to complete Document Query compared between when performed by single node and when performed by all nodes

CHAPTER V
DISCUSSION AND CONCLUSION

This chapter comprises two main sections including the discussion and conslusion.

5.1 Discussion

Amongst many metadata attributes, there are attributes that value can expose patients' confidential information into the Blockchain network. For example, "DocumentEntry.sourcePatientInfo" contains demographic information of the source patient to whose medical record the document belongs. The attribute can contain multiple values such as patient name and address. So, in the actual adoption of the proposed concept, these metadata attributes values must be anonymized before entering the Blockchain ledger. This can be done by replacing the value with its hash counterpart when the attributes entering the Blockchain ledger via the Document Register function. At the same time, when the attributes were required for the search operation as search keywords input, the XDS Document Consumer will only need to hash the input value and use the hash to allow the XDS Document Registry actor to search for the matching hash value registered within the Document Registry Blockchain. This allows the concept to maintain confidentiality of patients' data while preserving the functionalities of the concept.

During the study on the IHE Profiles standard, we found that the complexity of the IHE standard gave difficulty for interpretation. There are a lot of components including transaction formats, data structures, and specific protocols that are specifically designed for each IHE Profile that published separately in different volumes of IHE Framework documents. Some part of the document was deprecated or obsoleted in the present. There are a limited number of transaction samples available from the framework which makes it difficult for a beginner to develop a system that fully complies with the standard. These make the standard implementation the most time-consuming process.

For Ethereum, the usage of Smartcontract was limited by its gas requirement. Every task performed by Smartcontract requires gas which represents the computational power of the network to perform. That means there will be more factors for the platform to be addressed before the actual adoption. We suggest that the adopted network must be able to provide enough amount of gas or disregard the component from the consideration by integrating it into the common agreement amongst the networks.

5.2 Conclusion

In this work, we achieve the system prototype for the XDS Blockchain which acts as the medium for health document sharing amongst the XDS Affinity Domain Network. The prototype was designed to be compatible with normal XDS Actors while also act as a medium for a common XDS network to interface with the IBFT Blockchain ledger. This enhances the IHE XDS.b Profile with the Blockchain characteristics while appreciating the network to further share their health document to further benefit from the network for both operational interoperability and cyber-security. However, due to the ever-changing nature of the software platform, further adoption of the implementation will need to be updated as the platform released the newer version of the software to avoid version conflict of the source code.

In the implementation of the proposed concept, we excluded an XDS On-Demand Document Repository Actor and XDS Patient Identity Feed actor to reduce the complexity of the concept demonstration. For future work, those XDS Actors should also be implemented to the XDS Blockchain concept too. The XDS On-Demand Document Repository would enhance the benefit of the XDS Blockchain as it provides On-Demand health document type which gave flexibility and a wider range of usability of shared health documents to healthcare operation. At the same time, The Patient Identity Feed Actor would aid the member of the XDS Affinity Domain Blockchain network by establishing the medium identifier for all members to seamlessly share their health documents. The Patient Identity Feed Actor may even further integrate into the Smartcontract and eliminate the need for centralized identity feed in the network. Eliminate the cost which would be spent on maintaining the Patient Identity Feed Actor for the network. Furthermore, the Smartcontract also has the potential to become the exchange medium for ITI-43 transactions where the XDS Document Consumer

negotiates with XDS Document Repository for retrieving actual health documents, allow health documents exchanging activities in the network to be recorded in the Blockchain ledger which could be further used in the incident investigation during the cyber-incident. These would maximize the potential of Blockchain technology implemented on the Cross-Enterprise Document Sharing Profile.

Other than the Cross-Enterprise Document Sharing Profile, the IHE IT Infrastructure is providing much more profiles and various tools for use in achieving healthcare interoperability. There remain a lot more possibilities of using the framework to maximize the potential of Blockchain technology and the future technology to come.

As Blockchain technology still has a long way of development and research path to go through, the concept proposed in this work also could be further developed into a more advanced version for actual adoption in the future.

REFERENCE

[1] Weinelt B. Digital Transformation of Industries. Logistics Industry, http://reports.weforum.org/digital-transformation/wp-content/blogs.dir/94/mp/files/pages/files/digital-enterprise-narrative-final-january-2016.pdf (2016).

[2] Marcelo A, Medeiros D, Ramesh K, et al. Transforming Health Systems Through Good Digital Health Governance. *adb Sustain Dev Work Pap Ser* 2018; 1–15.

[3] Shaw T, Hines M, Kielly C. *Impact of Digital Health on the Safety and Quality of Health Care*, https://www.safetyandquality.gov.au/wp-content/uploads/2018/02/Report-The-Impact-of-Digital-Health-on-Safety-and-Quality-of-Healthcar....pdf (2000).

[4] Cisco. The Digitization of the Healthcare Industry: Using Technology to Transform Care. *Cisco* 2017; 1: 12.

[5] Bullhound G. *Digital healthcare*. 2015.

[6] Meskó B, Drobni Z, Bényei É, et al. Digital health is a cultural transformation of traditional healthcare. *mHealth* 2017; 3: 38–38.

[7] Carestream Health. Interoperability : Connecting the Healthcare Enterprise to Deliver Responsive Patient Care. 2015; 1–9.

[8] PolicyMedical. Interoperability in Healthcare: To Have or Not to Have, https://www.policymedical.com/interoperability-healthcare/ (accessed 22 September 2018).

[9] Interoperability DH. Digital Healthcare Interoperability.

[10] Healthcare Information and Management Systems Society. Definition of Interoperability. *Himss* 2013; 2013.

[11] Oracle. Interoperability : A Key to Meaningful Use. *Solutions*, http://www.oracle.com/us/industries/healthcare/interoperability-wp-188782.pdf (2010).

[12] HIMSS. What is Interoperability?, https://www.himss.org/library/interoperability-standards/what-is-interoperability (accessed 27 April 2019).

[13] Paige Goodhew. Why Healthcare Interoperability Matters | Redox, https://www.redoxengine.com/blog/why-healthcare-interoperability-matters/ (accessed 27 April 2019).

[14] Dr.David Hay. Why is interoperability so important for healthcare organisations? | Orion Health, https://orionhealth.com/global/knowledge-hub/blogs/why-is-interoperability-so-important-for-healthcare-organisations/ (accessed 27 April 2019).

[15] Le Bris A, Asri W El. STATE OF CYBERSECURITY & CYBER THREATS IN HEALTHCARE ORGANIZATIONS Applied Cybersecurity Strategy for Managers. *ESSEC Bus Sch* 2017; 13.

[16] Healthcare IT News. The biggest healthcare breaches of 2017, https://www.healthcareitnews.com/slideshow/biggest-healthcare-breaches-2017-so-far?page=1 (accessed 11 September 2018).

[17] HIPAA Journal. Largest Healthcare Data Breaches of 2018, https://www.hipaajournal.com/largest-healthcare-data-breaches-of-2018/ (accessed 27 April 2019).

[18] Healthcare IT News. The biggest healthcare data breaches of 2018 (so far), https://www.healthcareitnews.com/projects/biggest-healthcare-data-breaches-2018-so-far (accessed 27 April 2019).

[19] IHE International Inc. About IHE, https://www.ihe.net/about_ihe/ (accessed 11 September 2018).

[20] IHE International Inc. IHE Process, https://www.ihe.net/about_ihe/ihe_process/ (accessed 11 September 2018).

[21] IHE International Inc. Profiles, https://www.ihe.net/resources/profiles/ (accessed 17 September 2018).

[22] IHE International Inc. IHE IT Infrastructure (ITI) Technical Framework Volume 1 Integration Profiles. *Int J Healthc Technol Manag* 2008; 1: 1–177.

[23] dkorolyk. What Is The Difference Between XDS,XDS.a,XDS.b and XDS-I?,

http://healthcareitsystems.com/2012/05/22/what-is-the-difference-between-xds-xds-a-xds-b-and-xds-i/ (2012, accessed 17 February 2019).

[24] Luke MN, Lee SJ, Pekarek Z, et al. Blockchain in Electricity: a Critical Review of Progress to Date. 2018; 1–36.

[25] PwC. a Catalyst for New Approaches in Insurance.

[26] Zheng Z, Xie S, Dai H, et al. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. *Proc - 2017 IEEE 6th Int Congr Big Data, BigData Congr 2017* 2017; 557–564.

[27] Yaga D, Mell P, Roby N, et al. Blockchain Technology Overview (NISTIR-8202). *Draft NISTIR* 2018; 59.

[28] ethereum/devp2p: Ethereum peer-to-peer networking specifications, https://github.com/ethereum/devp2p (accessed 3 June 2021).

[29] Buterin V. *A NEXT GENERATION SMART CONTRACT & DECENTRALIZED APPLICATION PLATFORM*.

[30] Solidity Programming Language | The Solidity language portal is a comprehensive information page for the Solidity programming language. It features documentation, binaries, blog, resources & more., https://soliditylang.org/ (accessed 3 June 2021).

[31] Quorum. GoQuorum, https://github.com/ConsenSys/quorum (2020, accessed 3 June 2021).

[32] Morgan JP. Quorum | J.P. Morgan, https://www.jpmorgan.com/global/Quorum (2017, accessed 26 April 2019).

[33] Quorum 7-nodes Example, https://github.com/ConsenSys/quorum-examples/tree/master/examples/7nodes (accessed 3 June 2021).

[34] Peterson K, Deeduvanu R, Kanjamala P, et al. A Blockchain-Based Approach to Health Information Exchange Networks. *Mayo Clin* 2016; 10.

[35] Ekblaw A, Azaria A, Halamka JD, et al. A Case Study for Blockchain in Healthcare: " MedRec " prototype for electronic health records and medical research data. *IEEE Technol Soc Mag* 2016; 1–13.

[36] Zyskind G, Nathan O, Pentland AS. Decentralizing privacy: Using Blockchain to Protect Personal Data. *Proc - 2015 IEEE Secur Priv Work SPW 2015* 2015; 180–184.

[37] Li H, Zhu L, Shen M, et al. Blockchain-Based Data Preservation System for Medical Data. *J Med Syst* 2018; 42: 1–13.

[38] Tanwar S, Parekh K, Evans R. Blockchain-based electronic healthcare record system for healthcare 4.0 applications. *J Inf Secur Appl* 2020; 50: 102407.

[39] Sultan K, Ruhi U, Lakhani R. *CONCEPTUALIZING BLOCKCHAINS: CHARACTERISTICS & APPLICATIONS*, https://arxiv.org/ftp/arxiv/papers/1806/1806.03693.pdf (2018, accessed 29 October 2018).

[40] ethereum/go-ethereum: Official Go implementation of the Ethereum protocol, https://github.com/ethereum/go-ethereum (accessed 5 June 2021).

[41] Installing Geth | Go Ethereum, https://geth.ethereum.org/docs/install-and-build/installing-geth#install-on-ubuntu-via-ppas (accessed 10 June 2021).

[42] Command-line Options | Go Ethereum, https://geth.ethereum.org/docs/interface/command-line-options (accessed 10 June 2021).

[43] Install - GoQuorum, https://docs.goquorum.consensys.net/en/stable/HowTo/GetStarted/Install/ (accessed 10 June 2021).

[44] ConsenSys/tessera: Tessera - Enterprise Implementation of Quorum's transaction manager, https://github.com/ConsenSys/tessera (accessed 5 June 2021).

[45] ConsenSys/constellation: Peer-to-peer encrypted message exchange, https://github.com/ConsenSys/constellation (accessed 5 June 2021).

[46] quorum-examples/README.md at master · ConsenSys/quorum-examples, https://github.com/ConsenSys/quorum-examples/blob/master/README.md (accessed 10 June 2021).

[47] Remix - Ethereum IDE, https://remix.ethereum.org/ (accessed 8 March 2021).

[48] Node.js, https://nodejs.org/en/ (accessed 10 June 2021).

[49] npm, https://www.npmjs.com/ (accessed 10 June 2021).

[50] web3 - npm, https://www.npmjs.com/package/web3 (accessed 10 June 2021).

[51] xml2js - npm, https://www.npmjs.com/package/xml2js (accessed 10 June 2021).

[52] fs - npm, https://www.npmjs.com/package/fs (accessed 10 June 2021).

[53] net - npm, https://www.npmjs.com/package/net (accessed 10 June 2021).

[54] util - npm, https://www.npmjs.com/package/util (accessed 10 June 2021).

[55] moment - npm, https://www.npmjs.com/package/moment (accessed 10 June 2021).

[56] cryptr - npm, https://www.npmjs.com/package/cryptr (accessed 10 June 2021).

[57] web3.eth.Contract — web3.js 1.0.0 documentation, https://web3js.readthedocs.io/en/v1.3.4/web3-eth-contract.html (accessed 8 March 2021).

[58] IHE International Inc. IHE IT Infrastructure Technical Framework: Volume 2b (ITI TF-2b): Transactions Part B – Sections 3.29 – 3.64. 2.

[59] IHE International Inc. IT Infrastructure Technical Framework: Volume 3 (ITI TF-3) Cross-Transaction Specifications and Content Specifications. 3.

APPENDIX

A. Metadata Attributes used in IHE ITI-42 and ITI-18 represent corresponding health document [59]

Table A-1 SubmissionSet

SubmissionSet Metadata Attributes	Description
author	The humans and/or machines that authored the SubmissionSet. This attribute contains the sub-attributes: authorInstitution, authorPerson, authorRole, authorSpecialty, authorTelecommunication.
availabilityStatus	The lifecycle status of the SubmissionSet.
comments	Comments associated with the SubmissionSet.
contentTypeCode	The code specifying the type of clinical activity that resulted in placing the associated content in the SubmissionSet.
entryUUID	A globally unique identifier used to manage the entry.
homeCommunityId	A globally unique identifier for a community.
intendedRecipient	The organizations or persons for whom the SubmissionSet is intended.
limitedMetadata	A flag that the associated SubmissionSet was created using the less rigorous metadata requirements as defined for the Metadata-Limited Document Source.
patientId	The patientId represents the primary subject of care of the SubmissionSet.
sourceId	Identifier of the entity that contributed the SubmissionSet.
submissionTime	Point in time at the creating entity when the SubmissionSet was created
title	The title of the SubmissionSet.
uniqueId	Globally unique identifier for the SubmissionSet assigned by the creating entity.

Table A-2 Folder

Folder Metadata Attributes	**Description**
availabilityStatus	The lifecycle status of the Folder.
codeList	The set of codes specifying the type of clinical activities that resulted in placing DocumentEntry objects in the Folder.
comments	Comments associated with the Folder.
entryUUID	A globally unique identifier used to manage the entry.
homeCommunityId	A globally unique identifier for a community.
lastUpdateTime	Most recent point in time that the Folder has been modified.
limitedMetadata	A flag that the associated Folder was created using the less rigorous metadata requirements as defined for the Metadata-Limited Document Source.
patientId	The patientId represents the primary subject of care of the Folder.
title	The title of the Folder
uniqueId	Globally unique identifier for the Folder.

Table A-3 DocumentEntry

DocumentEntry Metadata Attributes	Description
author	The humans and/or machines that authored the document. This attribute contains the sub-attributes: authorInstitution, authorPerson, authorRole, authorSpecialty and authorTelecommunication.
availabilityStatus	The lifecycle status of the DocumentEntry
classCode	The code specifying the high-level use classification of the document type (e.g., Report, Summary, Images, Treatment Plan, Patient Preferences, Workflow).
comment	Comments associated with the document.
confidentialityCode	The code specifying the level of confidentiality of the documented.
creationTime	The time the author created the document.
entryUUID	A globally unique identifier used to manage the entry.
eventCodeList	This list of codes represents the main clinical acts, such as a colonoscopy or an appendectomy, being documented.
formatCode	The code specifying the detailed technical format of the document.
hash	The hash of the contents of the document.
healthcareFacility TypeCode	This code represents the type of organizational setting of the clinical encounter during which the documented act occurred.
homeCommunityId	A globally unique identifier for a community.
languageCode	Specifies the human language of character data in a document.
legalAuthenticator	Represents a participant within an authorInstitution who has legally authenticated or attested the document.
limitedMetadata	Indicates whether the DocumentEntry was created using the less rigorous requirements of metadata as defined for the Metadata-Limited Document Source.

mimeType	MIME type of the document.
objectType	The type of DocumentEntry (e.g., On-Demand DocumentEntry).
patientId	The patientId represents the subject of care of the document.
practiceSettingCode	The code specifying the clinical specialty where the act that resulted in the document was performed (e.g., Family Practice, Laboratory, Radiology).
referenceIdList	A list of Identifiers related to the document
repositoryUniqueId	The globally unique identifier of the repository where the document can be accessed.
serviceStartTime	The start time of the service being documented.
serviceStopTime	The stop time of the service being documented.
size	Size in bytes of the document.
sourcePatientId	The sourcePatientId represents the subject of care's medical record identifier (e.g., Patient Id) in the local patient identifier domain of the creating entity.
sourcePatientInfo	This attribute contains demographic information of the source patient to whose medical record this document belongs.
title	The title of the document.
typeCode	The code specifying the precise type of document from the user perspective (e.g., LOINC code).
uniqueId	Globally unique identifier assigned to the document by its creator.
URI	The URI for the document.

B. Full content of transaction sample used in Section 4.3 and 4.4.

These are full content of transaction samples created for the experiment in Section 4.3 and 4.4 referred as Document#n (01 - 10). The XML format following the format for IHE ITI-42 transaction [58, 59] used for the Document Register function.


```xml
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document01" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.109.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>e543712c0e10501972de13a5bfcbe826c49feb75</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20061224</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200612230800</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200612230900</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>89765a87b^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Doe^John^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|M</rim:Value>
    <rim:Value>PID-11|100 Main St^^Metropolis^Il^44130^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
      <rim:LlocalizedString value="DocA"></rim:LocalizedString>
    </rim:Name>
    <rim:Description></rim:Description>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Smitty^Gerald^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Cleveland Clinic</rim:Value>
          <rim:Value>Parma Community</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Attending</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Orthopedic</rim:Value>
        </rim:ValueList>
      </rim:Slot>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
```

```
        <rim:Value>^Dopplemeyer^Sherry^^^</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Cleveland Clinic</rim:Value>
        <rim:Value>Berea Community</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Primary Surgon</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Orthopedic</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document01" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document01" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>2.16.840.1.113883.5.25</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Restricted"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document01" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document01" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>2.16.840.1.113883.6.96</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Private home-based care"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document01" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="Pathology"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document01" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document01" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
        </rim:ValueList>
      </rim:Slot>
```

```
      <rim:Name>
        <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document01" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document01">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.20190405034511.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document01">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
```

```
      <rim:Value>20041225235050</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Physical"></rim:LocalizedString>
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
  </rim:Description>
  <rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
    <rim:Slot name="authorPerson">
      <rim:ValueList>
        <rim:Value>^Dopplemeyer^Sherry^^^</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Cleveland Clinic</rim:Value>
        <rim:Value>Berea Community</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Primary Surgon</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Orthopedic</rim:Value>
      </rim:ValueList>
    </rim:Slot>
```

```
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.96</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
      <rim:Name>
        <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
      <rim:Name>
        <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
      <rim:Name>
        <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:RegistryPackage>
```

```
        <rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">

        </rim:Classification>

        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document01" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">

          <rim:Slot name="SubmissionSetStatus">

            <rim:ValueList>

              <rim:Value>Original</rim:Value>

            </rim:ValueList>

          </rim:Slot>

        </rim:Association>

      </rim:RegistryObjectList>

    </lcm:SubmitObjectsRequest>

  </soapenv:Body>

</soapenv:Envelope>
```


Appendix B-1 Full content of transaction sample “Tx1”

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document02" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.110.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>8cd7c25aa2526918fef504fea46b79a3ebf123db</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070101</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200701011800</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200701011900</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>6b12add33^^^&amp;1.3.4.5&amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;1.2.3&amp;ISO</rim:Value>
    <rim:Value>PID-5|Emile^Sheehan^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|M</rim:Value>
    <rim:Value>PID-11|3094  Glen St^^Paducah^KY^42003^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
      <rim:LocalizedString value="DocB"></rim:LocalizedString>
    </rim:Name>
    <rim:Description></rim:Description>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document02" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Tymoteusz^Mccabe^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Kentucky Hospital</rim:Value>
          <rim:Value>Paducah Community</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Attending</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Neurology</rim:Value>
        </rim:ValueList>
      </rim:Slot>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document02" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
```

```
        <rim:Value>^Stevie^Lamb^^^</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Kentucky Hospital</rim:Value>
        <rim:Value>Paducah Community</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Neurologist</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Neurology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document02" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document02" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>2.16.840.1.113883.5.25</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Restricted"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document02" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document02" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>2.16.840.1.113883.6.96</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Hospital-trauma center"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document02" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="Neurology"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document02" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document02" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
        </rim:ValueList>
      </rim:Slot>
```

```
      <rim:Name>
        <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document02" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.27829.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document02">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.17115670011797.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document02">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
```

```
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Physical"></rim:LocalizedString>
    </rim:Name>
    <rim:Description>
      <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
    </rim:Description>
    <rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Stevie^Lamb^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Kentucky Hospital</rim:Value>
          <rim:Value>Paducah Community</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Neurologist</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Neurology</rim:Value>
        </rim:ValueList>
      </rim:Slot>
```

```
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>2.16.840.1.113883.6.96</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
  <rim:Name>
    <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
  </rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
  <rim:Name>
    <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
  </rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
  <rim:Name>
    <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
  </rim:Name>
</rim:ExternalIdentifier>
</rim:RegistryPackage>
```

```
        <rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">

        </rim:Classification>

        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document02" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">

          <rim:Slot name="SubmissionSetStatus">

            <rim:ValueList>

              <rim:Value>Original</rim:Value>

            </rim:ValueList>

          </rim:Slot>

        </rim:Association>

      </rim:RegistryObjectList>

    </lcm:SubmitObjectsRequest>

  </soapenv:Body>

</soapenv:Envelope>
```


Appendix B-2 Full content of transaction sample “Tx2”

```xml
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document03" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.111.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>824d0d3f18b2008b911415101ff75abe954de8dc</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070215</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200702151000</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200702151100</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>818f14933^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Alissa^Lugo^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|F</rim:Value>
    <rim:Value>PID-11|147 Cambridge Court^^Springdale^AR^72764^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
      <rim:LocalizedString value="DocC"></rim:LocalizedString>
    </rim:Name>
    <rim:Description></rim:Description>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document03" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Sharna^Hood^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Springdale Clinic</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Attending</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Physiotherapy</rim:Value>
        </rim:ValueList>
      </rim:Slot>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document03" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Romana^Woodward^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Springdale Clinic</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Physical therapist</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Physiotherapy</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document03" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document03" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.5.25</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Restricted"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document03" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document03" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.6.96</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Hospital-rehabilitation"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document03" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Rehabilitation"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document03" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document03" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document03" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.56658.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document03">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.21345401333474.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document03">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Romana^Woodward^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Springdale Clinic</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Physical therapist</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Physiotherapy</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
            <rim:Slot name="codingScheme">
              <rim:ValueList>
                <rim:Value>2.16.840.1.113883.6.96</rim:Value>
              </rim:ValueList>
            </rim:Slot>
            <rim:Name>
              <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
            </rim:Name>
          </rim:Classification>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
        </rim:RegistryPackage>
        <rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
        </rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document03" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-3 Full content of transaction sample “Tx3”

```xml
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document04" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.112.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>3f1c9ea8a6743efa9f289ecc1719196098bfcba6</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070225</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200702250800</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200702250900</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>057a0fed7^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Vivek^Trevino^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|M</rim:Value>
    <rim:Value>PID-11|2061 Abia Martin Drive^^Westbury^NY^11590^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
        <rim:LocalizedString value="DocD"></rim:LocalizedString>
      </rim:Name>
      <rim:Description></rim:Description>
      <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document04" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
        <rim:Slot name="authorPerson">
          <rim:ValueList>
            <rim:Value>^Bear^Ryan^^^</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Slot name="authorInstitution">
          <rim:ValueList>
            <rim:Value>Westbury Hospital</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Slot name="authorRole">
          <rim:ValueList>
            <rim:Value>Attending</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Slot name="authorSpecialty">
          <rim:ValueList>
            <rim:Value>Urology</rim:Value>
          </rim:ValueList>
        </rim:Slot>
      </rim:Classification>
      <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document04" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
        <rim:Slot name="authorPerson">
          <rim:ValueList>
            <rim:Value>^Sameeha^Mustafa^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Westbury Hospital</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Primary Surgon</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Urology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document04" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document04" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.5.25</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Restricted"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document04" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document04" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.6.96</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Hospital-long term care"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document04" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Urological oncology"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document04" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document04" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document04" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.50877.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document04">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.05195861265360.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document04">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Sameeha^Mustafa^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Westbury Hospital</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Primary Surgon</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Urology</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.6.96</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
</rim:RegistryPackage>
<rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
</rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document04" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-4 Full content of transaction sample “Tx4”

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document05" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.113.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>b7ba19a806546ab34d1b0df2569889d2aee9092d</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070302</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200703021300</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200703021530</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>2cf8cd9ce^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Rui^Corbett^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|M</rim:Value>
    <rim:Value>PID-11|3870 Ocala Street^^Winter Park^FL^32789^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
    <rim:LocalizedString value="DocE"></rim:LocalizedString>
  </rim:Name>
  <rim:Description></rim:Description>
  <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document05" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
    <rim:Slot name="authorPerson">
      <rim:ValueList>
        <rim:Value>^Katya^Cairns^^^</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Winter Park Hospital</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Attending</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Nephrology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document05" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
    <rim:Slot name="authorPerson">
      <rim:ValueList>
        <rim:Value>^Anabel^Delarosa^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Winter Park Hospital</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Nephrologist</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Nephrology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document05" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document05" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.5.25</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Restricted"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document05" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document05" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.6.96</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Dialysis unit--hospital"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document05" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Nephrology"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document05" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document05" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document05" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;1.3.6.1.4.1.20232.13.20.3000&amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document05">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.62551796790265.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document05">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Anabel^Delarosa^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Winter Park Hospital</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Nephrologist</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Nephrology</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
            <rim:Slot name="codingScheme">
              <rim:ValueList>
                <rim:Value>2.16.840.1.113883.6.96</rim:Value>
              </rim:ValueList>
            </rim:Slot>
            <rim:Name>
              <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
            </rim:Name>
          </rim:Classification>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
        </rim:RegistryPackage>
        <rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
        </rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document05" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-5 Full content of transaction sample “Tx5”

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document06" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.114.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>ceb184540ee6704b82af22ea23c8c76a8bba4dc3</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070317</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200703170800</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200703170900</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>1ea6205bb^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Maira^Chambers^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|F</rim:Value>
    <rim:Value>PID-11|2352 Modoc Alley^^Preston^ID^83263^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
      <rim:LocalizedString value="DocF"></rim:LocalizedString>
    </rim:Name>
    <rim:Description></rim:Description>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document06" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^TJay^ODoherty^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Preston Hospital</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Attending</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Gynaecology</rim:Value>
        </rim:ValueList>
      </rim:Slot>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document06" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Lillia^Hood^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Preston Hospital</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Gynaecologist</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Gynaecology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document06" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document06" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.5.25</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Restricted"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document06" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document06" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.6.96</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Hospital-government"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document06" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Gynecological oncology"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document06" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document06" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document06" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.28694.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document06">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.20558236010995.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document06">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Lillia^Hood^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Preston Hospital</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Gynaecologist</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Gynaecology</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
            <rim:Slot name="codingScheme">
              <rim:ValueList>
                <rim:Value>2.16.840.1.113883.6.96</rim:Value>
              </rim:ValueList>
            </rim:Slot>
            <rim:Name>
              <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
            </rim:Name>
          </rim:Classification>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
          <rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;1.3.6.1.4.1.21367.13.20.3000&amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
            <rim:Name>
              <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
            </rim:Name>
          </rim:ExternalIdentifier>
        </rim:RegistryPackage>
        <rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
        </rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document06" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-6 Full content of transaction sample “Tx6”

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document07" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.115.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>2ea5f1f52a743115d722db1f4a3cc8c4c738b476</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070403</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200704030900</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200704031000</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>9926a4bf6^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Kaiya^Zamora^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|F</rim:Value>
    <rim:Value>PID-11|1269 Hall Street^^North Las Vegas^NV^89030^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
      <rim:LocalizedString value="DocG"></rim:LocalizedString>
    </rim:Name>
    <rim:Description></rim:Description>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document07" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Mujtaba^Palacios^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Nevada Hospital</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Attending</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Physiology</rim:Value>
        </rim:ValueList>
      </rim:Slot>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document07" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Jolie^Parker^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Nevada Hospital</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Primary Physiologist</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Physiology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document07" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document07" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.5.25</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Restricted"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document07" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document07" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.6.96</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Hospital-community"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document07" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="General practice"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document07" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document07" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document07" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.41480.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document07">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.55694976369743.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document07">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Jolie^Parker^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Nevada Hospital</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Primary Physiologist</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Physiology</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.6.96</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
</rim:RegistryPackage>
<rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
</rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document07" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-7 Full content of transaction sample “Tx7”

```xml
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document08" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.116.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>489e1c4afa021fc3a3fe3d267cb04c8debbdee4c</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070311</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200703111000</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200703111030</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>9ed9d95ba^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Mccauley^Stanley^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|M</rim:Value>
    <rim:Value>PID-11|4495 Bungalow Road^^Omaha^NE^68114^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
        <rim:LocalizedString value="DocH"></rim:LocalizedString>
      </rim:Name>
      <rim:Description></rim:Description>
      <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document08" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
        <rim:Slot name="authorPerson">
          <rim:ValueList>
            <rim:Value>^Mahima^Ho^^^</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Slot name="authorInstitution">
          <rim:ValueList>
            <rim:Value>Bungalow Clinic</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Slot name="authorRole">
          <rim:ValueList>
            <rim:Value>Attending</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Slot name="authorSpecialty">
          <rim:ValueList>
            <rim:Value>Physiology</rim:Value>
          </rim:ValueList>
        </rim:Slot>
      </rim:Classification>
      <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document08" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
        <rim:Slot name="authorPerson">
          <rim:ValueList>
            <rim:Value>^Sullivan^Mahoney^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Bungalow Clinic</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Physiologist</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Physiology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document08" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document08" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.5.25</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Restricted"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document08" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document08" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
<rim:Slot name="codingScheme">
<rim:ValueList>
<rim:Value>2.16.840.1.113883.6.96</rim:Value>
</rim:ValueList>
</rim:Slot>
<rim:Name>
<rim:LocalizedString value="Local community health center"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document08" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Clinical physiology"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document08" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document08" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document08" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.88518.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document08">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.79196258923277.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document08">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Sullivan^Mahoney^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Bungalow Clinic</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Physiologist</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Physiology</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
        <rim:Slot name="codingScheme">
          <rim:ValueList>
            <rim:Value>2.16.840.1.113883.6.96</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Name>
          <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
        </rim:Name>
      </rim:Classification>
      <rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
        <rim:Name>
          <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
        </rim:Name>
      </rim:ExternalIdentifier>
      <rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
        <rim:Name>
          <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
        </rim:Name>
      </rim:ExternalIdentifier>
      <rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
        <rim:Name>
          <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
        </rim:Name>
      </rim:ExternalIdentifier>
    </rim:RegistryPackage>
    <rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
    </rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document08" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-8 Full content of transaction sample “Tx8”

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document09" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.117.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>ed5ee56d4975cb3052526d66f3fbddc7210c2140</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070419</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200704190800</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200704190900</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>9a399c351^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5Loretta^Parker^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|F</rim:Value>
    <rim:Value>PID-11|1974 Lynn Street^^Dorchester^MA^02122^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
      <rim:LocalizedString value="DocI"></rim:LocalizedString>
    </rim:Name>
    <rim:Description></rim:Description>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document09" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Nuala^Ballard^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Massachusetts Health Institute</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Attending</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Gastroenterology</rim:Value>
        </rim:ValueList>
      </rim:Slot>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document09" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Safiyah^Madden^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Massachusetts Health Institute</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Gastroenterologist</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Gastroenterology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document09" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document09" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.5.25</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LlocalizedString value="Restricted"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document09" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document09" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.96</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="Hospital outpatient gastroenterology clinic"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document09" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Gastroenterology"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document09" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document09" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document09" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.91167.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document09">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.71294592707618.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document09">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Safiyah^Madden^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Massachusetts Health Institute</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Gastroenterologist</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Gastroenterology</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
        <rim:Slot name="codingScheme">
          <rim:ValueList>
            <rim:Value>2.16.840.1.113883.6.96</rim:Value>
          </rim:ValueList>
        </rim:Slot>
        <rim:Name>
          <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
        </rim:Name>
      </rim:Classification>
      <rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
        <rim:Name>
          <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
        </rim:Name>
      </rim:ExternalIdentifier>
      <rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
        <rim:Name>
          <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
        </rim:Name>
      </rim:ExternalIdentifier>
      <rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;1.3.6.1.4.1.21367.13.20.3000&amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
        <rim:Name>
          <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
        </rim:Name>
      </rim:ExternalIdentifier>
    </rim:RegistryPackage>
    <rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
    </rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
sourceObject="SubmissionSet01" targetObject="Document09" id="ID_1507585920_2"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-9 Full content of transaction sample “Tx9”

```xml
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
  <soapenv:Header xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <wsa:To soapenv:mustUnderstand="true">http://127.0.0.1:6969/</wsa:To>
    <wsa:MessageID
soapenv:mustUnderstand="true">urn:uuid:2311B77C122650C7B91554413514373</wsa:MessageID>
    <wsa:Action soapenv:mustUnderstand="true">urn:ihe:iti:2007:RegisterDocumentSet-b</wsa:Action>
  </soapenv:Header>
  <soapenv:Body>
    <lcm:SubmitObjectsRequest xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0">
      <rim:RegistryObjectList xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0">
        <rim:ExtrinsicObject id="Document10" mimeType="text/plain" objectType="urn:uuid:7edca82f-054d-47f2-
a032-9b2a5b5186c1" status="urn:oasis:names:tc:ebxml-regrep:StatusType:Approved">
          <rim:Slot name="size">
            <rim:ValueList>
              <rim:Value>4</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="repositoryUniqueId">
            <rim:ValueList>
              <rim:Value>1.19.6.24.118.42.1</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="hash">
            <rim:ValueList>
              <rim:Value>ff6c5cb547ffc4aad72161e8e63b3f1108ac8b4c</rim:Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="creationTime">
            <rim:ValueList>
              <rim:Value>20070510</rim:Value>
            </rim:ValueList>
```

```
</rim:Slot>
<rim:Slot name="languageCode">
  <rim:ValueList>
    <rim:Value>en-us</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStartTime">
  <rim:ValueList>
    <rim:Value>200705100800</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="serviceStopTime">
  <rim:ValueList>
    <rim:Value>200705100900</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientId">
  <rim:ValueList>
    <rim:Value>6e4efa3ec^^^&amp;amp;1.3.4.5&amp;amp;ISO</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Slot name="sourcePatientInfo">
  <rim:ValueList>
    <rim:Value>PID-3|pid1^^^&amp;amp;1.2.3&amp;amp;ISO</rim:Value>
    <rim:Value>PID-5|Evan^Griffith^^^</rim:Value>
    <rim:Value>PID-7|19560527</rim:Value>
    <rim:Value>PID-8|M</rim:Value>
    <rim:Value>PID-11|259 Froe Street^^Wheeling^WV^26003^USA</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
```

```
      <rim:LocalizedString value="DocJ"></rim:LocalizedString>
    </rim:Name>
    <rim:Description></rim:Description>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document10" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_1">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Harriette^Whitworth^^^</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorInstitution">
        <rim:ValueList>
          <rim:Value>Wheeling Hospital</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorRole">
        <rim:ValueList>
          <rim:Value>Attending</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Slot name="authorSpecialty">
        <rim:ValueList>
          <rim:Value>Cardiology</rim:Value>
        </rim:ValueList>
      </rim:Slot>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:93606bcf-9494-43ec-9b4e-a7748d1a838d"
classifiedObject="Document10" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_2">
      <rim:Slot name="authorPerson">
        <rim:ValueList>
          <rim:Value>^Zaine^Lord^^^</rim:Value>
```

```
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorInstitution">
      <rim:ValueList>
        <rim:Value>Wheeling Hospital</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorRole">
      <rim:ValueList>
        <rim:Value>Primary Surgon</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Slot name="authorSpecialty">
      <rim:ValueList>
        <rim:Value>Cardiology</rim:Value>
      </rim:ValueList>
    </rim:Slot>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:41a5887f-8865-4c09-adf7-e362475b143a"
classifiedObject="Document10" nodeRepresentation="PLANS" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_3">
    <rim:Slot name="codingScheme">
      <rim:ValueList>
        <rim:Value>1.3.6.1.4.1.19376.1.2.6.1</rim:Value>
      </rim:ValueList>
    </rim:Slot>
    <rim:Name>
      <rim:LocalizedString value="Treatment Plan or Protocol"></rim:LocalizedString>
    </rim:Name>
  </rim:Classification>
  <rim:Classification classificationScheme="urn:uuid:f4f85eac-e6cb-4883-b524-f2705394840f"
classifiedObject="Document10" nodeRepresentation="R" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_4">
```

```
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.5.25</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Restricted"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:a09d5840-386c-46f2-b5ad-9c3699a4309d"
classifiedObject="Document10" nodeRepresentation="urn:ihe:iti:bppc:2007"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_5">
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>1.3.6.1.4.1.19376.1.2.3</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="urn:ihe:iti:bppc:2007"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:f33fb8ac-18af-42cc-ae0e-ed0b0bdb91e1"
classifiedObject="Document10" nodeRepresentation="66280005" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_6">
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.6.96</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Hospital-Veterans' Administration"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
```

```
<rim:Classification classificationScheme="urn:uuid:cccf5598-8b07-4b77-a05e-ae952c785ead"
classifiedObject="Document10" nodeRepresentation="Practice-D" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_7">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Surgery-Cardiac surgery"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document10" nodeRepresentation="urn:connectathon:bppc:foundational:policy"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_8">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="Foundational Connectathon Read-Access Policy"></rim:LocalizedString>
  </rim:Name>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:2c6b8cb7-8b2a-4051-b291-b1ae6a575ef4"
classifiedObject="Document10" nodeRepresentation="urn:connectathon:policy:full-access"
objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Classification" id="id_9">
  <rim:Slot name="codingScheme">
    <rim:ValueList>
      <rim:Value>1.3.6.1.4.1.21367.2017.3</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Name>
    <rim:LocalizedString value="FULL ACCESS TO ALL POLICY"></rim:LocalizedString>
```

```
      </rim:Name>
    </rim:Classification>
    <rim:Classification classificationScheme="urn:uuid:f0306f51-975f-434e-a61c-c59651d33983"
classifiedObject="Document10" nodeRepresentation="11502-2" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_10">
      <rim:Slot name="codingScheme">
        <rim:ValueList>
          <rim:Value>2.16.840.1.113883.6.1</rim:Value>
        </rim:ValueList>
      </rim:Slot>
      <rim:Name>
        <rim:LocalizedString value="LABORATORY REPORT.TOTAL"></rim:LocalizedString>
      </rim:Name>
    </rim:Classification>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:58a6f841-87b3-4a3e-92fd-a8ffeff98427"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.00234.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_11" registryObject="Document10">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.patientId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
    <rim:ExternalIdentifier identificationScheme="urn:uuid:2e82c1f6-a085-4c72-9da3-8640a32e42ab"
value="1.2.42.44004336459043.30" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_12" registryObject="Document10">
      <rim:Name>
        <rim:LocalizedString value="XDSDocumentEntry.uniqueId"></rim:LocalizedString>
      </rim:Name>
    </rim:ExternalIdentifier>
  </rim:ExtrinsicObject>
  <rim:RegistryPackage id="SubmissionSet01" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:RegistryPackage">
    <rim:Slot name="submissionTime">
      <rim:ValueList>
        <rim:Value>20041225235050</rim:Value>
      </rim:ValueList>
```

```
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Physical"></rim:LocalizedString>
</rim:Name>
<rim:Description>
  <rim:LocalizedString value="Annual physical"></rim:LocalizedString>
</rim:Description>
<rim:Classification classificationScheme="urn:uuid:a7058bb9-b4e4-4307-ba5b-e3f0ab85e12d"
classifiedObject="SubmissionSet01" nodeRepresentation="" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_13">
  <rim:Slot name="authorPerson">
    <rim:ValueList>
      <rim:Value>^Zaine^Lord^^^</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorInstitution">
    <rim:ValueList>
      <rim:Value>Wheeling Hospital</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorRole">
    <rim:ValueList>
      <rim:Value>Primary Surgon</rim:Value>
    </rim:ValueList>
  </rim:Slot>
  <rim:Slot name="authorSpecialty">
    <rim:ValueList>
      <rim:Value>Cardiology</rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:Classification>
<rim:Classification classificationScheme="urn:uuid:aa543740-bdda-424e-8c96-df4873be8500"
classifiedObject="SubmissionSet01" nodeRepresentation="394747008" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification" id="id_14">
```

```
<rim:Slot name="codingScheme">
  <rim:ValueList>
    <rim:Value>2.16.840.1.113883.6.96</rim:Value>
  </rim:ValueList>
</rim:Slot>
<rim:Name>
  <rim:LocalizedString value="Health Authority"></rim:LocalizedString>
</rim:Name>
</rim:Classification>
<rim:ExternalIdentifier identificationScheme="urn:uuid:96fdda7c-d067-4183-912e-bf5ee74998a8"
value="1.2.42.20190405034511.31" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_15" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.uniqueId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:554ac39e-e3fe-47fe-b233-965d2a147832"
value="1.3.6.1.4.1.21367.4" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_16" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.sourceId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
<rim:ExternalIdentifier identificationScheme="urn:uuid:6b5aea1a-874d-4603-a4bc-96a0a7b38446"
value="IHEBLUE-2736^^^&amp;amp;1.3.6.1.4.1.21367.13.20.3000&amp;amp;ISO" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:ExternalIdentifier" id="id_17" registryObject="SubmissionSet01">
<rim:Name>
  <rim:LocalizedString value="XDSSubmissionSet.patientId"></rim:LocalizedString>
</rim:Name>
</rim:ExternalIdentifier>
</rim:RegistryPackage>
<rim:Classification classifiedObject="SubmissionSet01" classificationNode="urn:uuid:a54d6aa5-d40d-43f9-
88c5-b4633d873bdd" id="ID_1507585920_1" objectType="urn:oasis:names:tc:ebxml-
regrep:ObjectType:RegistryObject:Classification">
</rim:Classification>
```

```
        <rim:Association associationType="urn:oasis:names:tc:ebxml-regrep:AssociationType:HasMember"
 sourceObject="SubmissionSet01" targetObject="Document10" id="ID_1507585920_2"
 objectType="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Association">
          <rim:Slot name="SubmissionSetStatus">
            <rim:ValueList>
              <rim:Value>Original</rim:Value>
            </rim:ValueList>
          </rim:Slot>
        </rim:Association>
      </rim:RegistryObjectList>
    </lcm:SubmitObjectsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```


Appendix B-10 Full content of transaction sample “Tx10”

C. Experiment Result for Section 4.4.2

Raw data from Document Register performed by single node (1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	2.680730228	3.151614353	2.661581788	5.23734807	4.541333189	3.311541217	6.615839494	5.640675257	7.61848117	4.116580378
	4.372585358	6.394495098	2.216456838	2.758617684	3.49683507	4.551797695	2.835120983	2.684245476	3.399675776	2.324835152
	2.215025236	6.994962397	2.293773148	2.643010937	2.406690028	5.827773203	2.840869399	2.590497133	2.883382742	7.156620574
	4.824055267	2.954027515	6.703592989	2.954469282	6.88253523	2.816631988	2.436429147	3.630659028	2.65742554	2.540884756
	2.484757429	2.169813076	6.365844802	2.702545988	3.435864506	4.84419419	6.753343675	2.811426626	2.902904188	2.322985942
	6.590677217	2.213652235	7.39258051	2.504734101	6.337972488	2.796690239	3.162911041	2.670018159	3.341148195	2.818651467
	6.872279293	2.637137473	2.673112311	2.600542999	6.200746353	2.358182237	3.151458786	2.701559093	3.140090953	3.113011935
	2.891139425	2.444128947	2.262200883	2.530834861	3.155367566	2.715653198	3.14731784	2.62726818	3.91789401	3.364168868
	2.86458596	6.963518282	6.496214969	2.450075405	2.156458772	2.728363475	3.226092498	2.384504037	3.169569158	3.325036778
	2.666282349	2.71300873	2.660164094	2.419570722	2.796509696	2.931885299	6.444079624	2.521417623	3.148795557	3.358815869
Average	3.846211776	3.863635811	4.172552233	2.880175005	4.14103129	3.488271274	4.061346249	3.026227061	3.617936729	3.444159172

Raw data from Document Register performed by all nodes (Inspected from 1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	6.15673509	8.636553523	6.208584804	7.658494597	10.17554187	6.782308882	8.446496869	8.685503974	37.18996088	5.137618599
	5.568159795	8.423783065	7.550394017	3.375108275	5.93842267	7.388122313	6.85576471	6.431093627	5.278317013	6.930143429
	4.8226547	7.26993863	6.567001925	7.210125726	5.991954421	7.459163889	7.195278827	6.771607248	6.389931839	8.309826787
	4.266414741	7.741777947	2.152550718	7.500981409	2.576576136	6.966749908	5.745020729	7.256076198	8.115371773	2.87990031
	5.634260707	7.852191837	7.432675583	6.408232849	7.473243367	6.572272878	6.119250557	2.339184951	4.183155079	5.986352656
	6.362323947	5.403609266	4.715458635	5.223317914	7.238744547	7.254771298	5.356058012	8.60507121	6.558008262	6.827084324
	4.759756519	7.33146089	4.628819509	7.556689834	7.101153052	7.266370245	6.75297524	5.924425918	7.557353298	6.737186291
	5.290520474	7.556890435	6.683296205	7.644745588	3.478950159	8.367521301	5.922678925	4.760415615	7.765911402	6.991612926
	7.327435841	6.778447479	7.524533217	7.774795245	2.247268945	6.337485075	6.754915781	5.89435987	7.180417129	3.375356193
	5.61716245	4.394353324	6.96922061	7.997927276	4.702248188	4.726712626	6.719916971	6.207208647	7.813705233	7.500566555
Average	5.580542426	7.13890064	6.043253522	6.835041871	5.692410336	6.912147842	6.586835662	6.287494726	9.803213191	6.067564807

Raw data from Document Query with "FindDocument" using minimum keywords performed by single node (1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	494.996883	237.52462	241.538263	260.755375	276.452013	285.971111	247.696266	268.420053	288.454075	487.58561
	267.197056	249.66493	265.614844	240.78102	271.456253	276.485391	282.619828	273.417264	374.320159	294.558271
	251.310735	315.324432	336.590983	254.917368	333.34181	263.683099	266.273093	263.042896	297.824701	293.504961
	245.569823	245.990772	250.583109	261.688863	273.826076	254.117918	248.25325	390.454674	300.222271	522.552464
	267.280761	244.113696	244.522397	266.651406	229.287007	266.937748	259.31533	260.016936	373.965174	345.877843
	223.765689	235.635288	348.799713	293.298575	243.593881	253.547761	250.01695	271.644042	260.947825	287.109535
	288.267233	242.457123	231.83636	258.982063	218.281316	259.375997	237.060239	474.502183	262.803668	389.366736
	247.578886	262.135906	238.119741	273.389073	376.75633	267.931011	267.916718	251.207295	274.563584	346.263293
	265.379783	228.85958	245.389639	235.753094	271.254542	248.64113	266.565653	301.645934	343.806174	456.193364
	240.940362	241.100788	249.452284	244.864384	444.059807	249.462709	253.836941	351.862295	260.400106	275.378115
Average	279.2287211	250.2807135	265.2447333	259.1081221	293.8309035	262.6153875	257.9554268	310.6213572	303.7307737	369.8390192

Raw data from Document Query with "FindDocument" using maximum keywords performed by single node (1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	555.765608	359.726963	278.026292	350.760697	485.348443	249.07578	512.00073	366.880376	292.501565	299.411293
	172.980077	314.593751	290.54273	321.07353	414.373719	289.273132	286.38236	259.860533	292.414281	425.786404
	137.309275	282.838335	236.792215	274.765977	303.22754	439.7641	320.827793	283.738313	279.653015	299.505323
	141.77949	223.61948	240.996909	342.562729	274.981318	323.25329	266.417713	429.269847	256.543802	300.965208
	146.856924	262.311425	267.69796	262.620099	272.908782	279.202461	269.610117	265.765957	251.940379	268.305898
	255.949123	249.304082	231.953734	234.521422	248.765945	430.531544	276.471444	250.048337	261.422032	271.320609
	228.764771	219.021122	407.482412	368.939888	264.677993	272.961675	286.868046	267.011149	364.109721	261.485676
	212.257528	259.247968	400.164407	306.783935	246.76498	242.323931	257.711774	251.480835	308.53247	286.776725
	117.908425	255.87442	249.95314	236.974838	239.174016	281.423541	245.330103	250.722326	309.858141	307.910678
	151.042747	395.347859	270.61541	347.073838	297.214699	276.228674	242.550542	290.679669	279.02432	282.683939
Average	212.0613968	282.1885405	287.4225209	304.6076953	304.7437435	308.4038128	296.4170622	291.5457342	289.5999726	300.4151753

Raw data from Document Query with “GetDocument” performed by single node (1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	218.143233	232.369402	253.519308	263.826271	243.64457	239.71787	274.525688	316.157329	296.223803	278.405688
	231.395275	239.709668	248.899351	267.528636	259.373869	247.668887	276.274429	325.057731	297.255746	305.382313
	228.543667	241.950565	287.30651	233.591846	272.364098	309.449997	259.991917	276.295972	246.881567	293.835938
	252.71108	344.840129	267.155344	238.617145	241.913224	245.970238	267.57978	296.979961	412.135766	287.294625
	210.069598	213.38407	261.193244	268.358379	228.030849	262.102768	270.358105	306.629297	244.622504	309.331862
	215.76425	234.169374	314.09412	273.715701	248.463539	287.750826	257.192844	292.278526	314.89733	308.201521
	201.892842	244.251852	243.61846	241.863787	268.920226	258.815696	274.217217	312.418343	297.460206	292.334693
	234.955298	227.056787	247.153502	250.123466	260.216048	277.244005	286.629823	273.334891	288.952043	262.614331
	245.211191	240.43681	236.764309	240.655066	260.897583	287.689068	293.482992	281.474495	329.42525	320.02427
	205.795992	209.567478	211.00394	215.207939	278.258624	300.464044	304.631478	282.733342	234.790228	271.058969
Average	224.4482426	242.7736135	257.0708088	249.3488236	256.208263	271.6873399	276.4884273	296.3359887	296.2644443	292.848421

Raw data from Document Query with “FindDocument” using minimum keywords performed by all nodes (inspected from 1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	1693.847875	174.450997	974.685047	988.671246	123.201068	516.488333	157.054154	553.261786	178.37349	782.885974
	190.574229	116.942637	926.127169	992.150258	118.359081	989.850035	1119.605896	542.998017	1192.679766	783.464502
	982.532618	927.594576	954.205227	158.517954	671.211488	510.425455	117.916494	543.100668	176.535573	745.437295
	1229.94134	536.805356	959.388502	942.952798	1169.631677	995.857444	1146.329806	1132.065009	180.501482	796.391525
	1137.865444	532.492183	149.332681	549.426491	1348.020283	992.115531	515.940574	512.310313	1171.051106	649.991793
	110.941354	911.153214	944.631962	559.931537	1106.306557	1180.133021	514.325143	1136.714122	141.153126	834.908318
	141.29262	934.826098	990.525422	947.533017	963.820611	1174.858223	987.919178	513.427151	1144.516677	548.247252
	958.836501	938.226721	939.702294	997.513011	984.650428	140.477969	1104.573804	516.072714	1206.023369	920.56
	1318.81095	937.076888	167.71494	581.921442	852.390801	180.375286	529.983527	1298.033876	1410.202252	776.330333
	143.423079	1108.137495	927.7005	577.080806	818.491237	976.916887	519.273589	530.217444	1190.387414	522.442917
Average	790.806601	711.7706165	793.4013744	729.569856	815.6083231	765.7498184	671.2922165	727.82011	799.1424255	736.0659909

Raw data from Document Query with “FindDocument” using maximum keywords performed by all nodes (inspected from 1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	889.481616	878.642597	907.76357	978.551833	1301.73118	112.293142	512.317853	583.821185	628.82629	181.981268
	879.7481	872.733275	933.632871	930.935794	534.164857	1117.588122	1256.491206	1129.834413	564.009307	198.625476
	851.93658	864.55861	895.44492	130.389071	150.388397	981.129743	983.936187	593.763604	572.228281	163.558076
	137.250737	836.168661	124.314267	120.898709	967.587026	185.495192	558.465101	1158.411944	568.588794	1147.45904
	848.443372	406.848064	969.543639	920.630339	920.040638	957.545374	516.599448	1114.207724	538.942168	1481.866201
	837.528242	844.81616	920.286049	1134.451684	950.151186	191.559712	183.894745	1123.358185	1166.632363	1597.990998
	546.880426	853.123624	989.804203	11.191306	959.584527	954.468297	984.058988	581.602841	559.895458	1524.367422
	505.198656	496.694992	924.009025	923.713906	580.012737	955.890535	546.004171	557.079879	1154.896372	1289.213129
	524.157351	903.319474	893.833832	914.299059	950.001563	968.548805	1152.005163	519.522888	561.295489	181.694987
	904.617924	498.937968	916.177687	928.302999	968.822149	136.459457	985.318044	596.567396	1243.300878	1141.676177
Average	692.5243004	745.5843425	847.4810063	699.33647	828.248426	656.0978379	767.9090906	795.8170059	755.86154	890.8432774

Raw data from Document Query with “GetDocument” performed by all nodes (inspected from 1st Node)

	Doc01	Doc02	Doc03	Doc04	Doc05	Doc06	Doc07	Doc08	Doc09	Doc10
Process Time	856.455881	897.291284	899.30363	175.510782	1175.910604	1156.867019	1164.842629	727.972237	1204.354223	1357.300065
	901.156597	979.454412	959.608064	1157.522063	962.975557	997.706218	1212.370079	770.490537	192.277228	1262.230591
	117.285613	888.451099	905.586741	932.111924	143.517073	1213.541531	1152.529487	768.656069	169.494279	176.559986
	861.332369	911.827304	884.359066	908.359489	127.998597	157.157788	1105.572328	719.972403	1139.653751	175.921636
	134.14013	909.748749	917.113464	921.5171	188.115254	120.186768	197.546304	759.567029	1139.491125	187.628997
	903.909022	883.376998	923.326979	927.6562	989.765044	1106.642069	1361.588626	714.657488	131.673094	158.303964
	904.930935	945.261579	991.197965	927.684465	947.160071	121.975834	17.135781	1102.215304	158.524658	1258.043378
	846.321004	160.687148	920.892166	187.501745	1181.558639	997.672086	180.847255	716.599627	1158.034447	1171.883766
	514.252435	136.097837	918.546419	945.302002	116.640064	1120.316232	997.376457	735.250079	1136.189434	1183.859137
	537.067707	873.78231	11.066743	968.564782	916.133879	1106.706431	110.101743	756.853241	1372.958238	1176.977363
Average	657.6851693	758.597872	833.1001237	805.1730552	674.9774782	809.8771976	749.9910689	777.2234014	780.2650477	810.8708883

BIOGRAPHY

NAME	Petnathean Julled
DATE OF BIRTH	22 October 1992
PLACE OF BIRTH	Nakhon Si Thammarat, Thailand
INSTITUTIONS ATTENDED	Mahidol University, 2011-2015 Bachelor of Engineering (Biomedical) Master of Science (Cybersecurity and Information Assurance)

Thesis Final 9

ORIGINALITY REPORT

12%	9%	5%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Mahidol University Student Paper	6%
2	geofon.gfz-potsdam.de Internet Source	<1%
3	www.ihe.net Internet Source	<1%
4	www.tandfonline.com Internet Source	<1%
5	dokumen.pub Internet Source	<1%
6	www.mdpi.com Internet Source	<1%
7	Ram Machlev, Yoash Levron, Yuval Beck. "Modified Cross-Entropy Method for Classification of Events in NILM Systems", IEEE Transactions on Smart Grid, 2019 Publication	<1%
8	Submitted to Curtin University of Technology Student Paper	<1%

9	Davi R. Ortega, Poorna Subramanian, Petra Mann, Andreas Kjær et al. "Repurposing a macromolecular machine: Architecture and evolution of the F7 chemosensory system", Cold Spring Harbor Laboratory, 2019 Publication	<1 %
10	www.springerprofessional.de Internet Source	<1 %
11	Cheng-Yi Yang, Chien-Tsai Liu. "Developing IHE-Based PHR Cloud Systems", 2013 International Conference on Social Computing, 2013 Publication	<1 %
12	ihe.net Internet Source	<1 %
13	gazelle.ihe.net Internet Source	<1 %
14	github.com Internet Source	<1 %
15	www.srdc.com.tr Internet Source	<1 %
16	Hani Mohammed Alshahrani. "Droid-IoT: Detect Android IoT Malicious Applications Using ML and Blockchain", Computers, Materials & Continua, 2022 Publication	<1 %

17	Submitted to Napier University Student Paper	<1 %
18	Thibaud Ecarot, Benoît Fraikin, Luc Lavoie, Mark McGilchrist, Jean-François Ethier. "A Sensitive Data Access Model in Support of Learning Health Systems", Computers, 2021 Publication	<1 %
19	soliditylang.org Internet Source	<1 %
20	www.thaicentenarian.mahidol.ac.th Internet Source	<1 %
21	bmcmedinformdecismak.biomedcentral.com Internet Source	<1 %
22	Submitted to Kaplan University Student Paper	<1 %
23	Submitted to University of East London Student Paper	<1 %
24	journals.sagepub.com Internet Source	<1 %
25	www.researchgate.net Internet Source	<1 %
26	S. G. Langer, W. Tellis, C. Carr, M. Daly, B. J. Erickson, D. Mendelson, S. Moore, J. Perry, K. Shastri, M. Warnock, W. Zhu. "The RSNA	<1 %

Image Sharing Network", Journal of Digital Imaging, 2014
Publication

27	Submitted to State Fair Community College Student Paper	<1 %
28	www2.eecs.berkeley.edu Internet Source	<1 %
29	Submitted to Imperial College of Science, Technology and Medicine Student Paper	<1 %
30	eprints.utem.edu.my Internet Source	<1 %
31	Camden Thatcher, Subrata Acharya. "RxBlock: Towards the design of a distributed immutable electronic prescription system", Network Modeling Analysis in Health Informatics and Bioinformatics, 2020 Publication	<1 %
32	hdl.handle.net Internet Source	<1 %
33	"Advances in Cyber Security", Springer Science and Business Media LLC, 2021 Publication	<1 %
34	Submitted to Chamberlain College of Nursing Student Paper	<1 %

35	Yang, Cheng-Yi, and Chien-Tsai Liu. "Performance assessment and tuning for exchange of clinical documents cross healthcare enterprises", Computer Standards & Interfaces, 2016. Publication	<1 %
36	link.springer.com Internet Source	<1 %
37	nhic.gov.sa Internet Source	<1 %
38	Submitted to Colorado Technical University Online Student Paper	<1 %
39	Pradeep Sinha, Gaur Sunder, Prashant Bendale, Manisha Mantri, Atreya Dande. "Electronic Health Record", Wiley, 2012 Publication	<1 %
40	www.ejbi.org Internet Source	<1 %
41	H.K. Huang. "PACS - Based Multimedia Imaging Informatics", Wiley, 2018 Publication	<1 %
42	Submitted to Jacksonville University Student Paper	<1 %
43	Kemal Turksonmez, Marcin Furtak, Mike P. Wittie, David L. Millman. "Two Ways Gas Price	<1 %

Oracles Miss The Mark", 2021 IEEE International Conference on Omni-Layer Intelligent Systems (COINS), 2021
Publication

44	Submitted to University of Limerick Student Paper	<1 %
45	journals.pan.pl Internet Source	<1 %
46	ruor.uottawa.ca Internet Source	<1 %
47	Chien Hua Wu, Ruey Kei Chiu, Hong Mo Yeh, Da Wei Wang. "Implementation of a cloud-based electronic medical record exchange system in compliance with the integrating healthcare enterprise's cross-enterprise document sharing integration profile", International Journal of Medical Informatics, 2017 Publication	<1 %
48	Magda Foti, Manolis Vavalis. "What blockchain can do for power grids?", Blockchain: Research and Applications, 2021 Publication	<1 %
49	Submitted to Taylor's Education Group Student Paper	<1 %
50	Submitted to Queen Mary and Westfield College	<1 %

Student Paper

51	Submitted to Universiteit van Amsterdam Student Paper	<1 %
52	Submitted to Western Governors University Student Paper	<1 %
53	anandkumarsinha07.medium.com Internet Source	<1 %
54	Submitted to Melbourne Institute of Technology Student Paper	<1 %
55	www.pubmedcentral.nih.gov Internet Source	<1 %
56	"Business Process Management: Blockchain and Robotic Process Automation Forum", Springer Science and Business Media LLC, 2021 Publication	<1 %
57	"Introduction to Nursing Informatics", Springer Science and Business Media LLC, 2021 Publication	<1 %
58	Tomasz Hyla, Jerzy Pejaś. "eHealth Integrity Model Based on Permissioned Blockchain", Future Internet, 2019 Publication	<1 %
59	Submitted to University of Canterbury	

Student Paper

<1 %

60	docplayer.net Internet Source	<1 %
61	www.estandards-project.eu Internet Source	<1 %
62	Kedar Iyer, Chris Dannen. "Building Games with Ethereum Smart Contracts", Springer Science and Business Media LLC, 2018 Publication	<1 %
63	core.ac.uk Internet Source	<1 %
64	iris.wpro.who.int Internet Source	<1 %
65	www.ncbi.nlm.nih.gov Internet Source	<1 %
66	"Future Network Systems and Security", Springer Science and Business Media LLC, 2019 Publication	<1 %
67	Nicholas Nicholson, Andrea Perego. "Interoperability of population-based patient registries", Journal of Biomedical Informatics, 2020 Publication	<1 %

68	Vikram Dhillon, David Metcalf, Max Hooper. "Blockchain Enabled Applications", Springer Science and Business Media LLC, 2017 Publication	<1 %
69	academic.oup.com Internet Source	<1 %
70	archiv.ub.uni-heidelberg.de Internet Source	<1 %
71	harvest.usask.ca Internet Source	<1 %
72	healthcare.report Internet Source	<1 %
73	www.imperial.ac.uk Internet Source	<1 %
74	Jianguo Zhang, Kai Zhang, Yuanyuan Yang, Jianyong Sun, Tonghui Ling, Mingqing Wang, Peter Bak. "Implementation methods of medical image sharing for collaborative health care based on IHE XDS-I profile", Journal of Medical Imaging, 2015 Publication	<1 %
75	Jie Zhang, Nian Xue, Xin Huang. "A Secure System For Pervasive Social Network-Based Healthcare", IEEE Access, 2016 Publication	<1 %

76	Myron Anthony Godinho, Jitendra Jonnagaddala, Nachiket Gudi, Rubana Islam, Padmanesan Narasimhan, Siaw-Teng Liaw. "mHealth for Integrated People-Centred Health Services in the Western Pacific: A Systematic Review", International Journal of Medical Informatics, 2020 Publication	<1%
77	Tim Benson, Grahame Grieve. "Chapter 17 Sharing Documents and IHE XDS", Springer Science and Business Media LLC, 2016 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

Thesis Final 9

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42

PAGE 43

PAGE 44

PAGE 45

PAGE 46

PAGE 47

PAGE 48

PAGE 49

PAGE 50

PAGE 51

PAGE 52

PAGE 53

PAGE 54

PAGE 55

PAGE 56

PAGE 57

PAGE 58

PAGE 59

PAGE 60

PAGE 61

PAGE 62

PAGE 63

PAGE 64

PAGE 65

PAGE 66

PAGE 67

PAGE 68

PAGE 69

PAGE 70

PAGE 71

PAGE 72

PAGE 73

PAGE 74

PAGE 75

PAGE 76

PAGE 77

PAGE 78

PAGE 79

PAGE 80

PAGE 81

PAGE 82

PAGE 83

PAGE 84

PAGE 85

PAGE 86

PAGE 87

PAGE 88

PAGE 89

PAGE 90

PAGE 91

PAGE 92

PAGE 93

PAGE 94

PAGE 95

PAGE 96

PAGE 97

PAGE 98

PAGE 99

PAGE 100

PAGE 101

PAGE 102

PAGE 103

PAGE 104

PAGE 105

PAGE 106

PAGE 107

PAGE 108

PAGE 109

PAGE 110

PAGE 111

PAGE 112

PAGE 113

PAGE 114

PAGE 115

PAGE 116

PAGE 117

PAGE 118

PAGE 119

PAGE 120

PAGE 121

PAGE 122

PAGE 123

PAGE 124

PAGE 125

PAGE 126

PAGE 127

PAGE 128

PAGE 129

PAGE 130

PAGE 131

PAGE 132

PAGE 133

PAGE 134

PAGE 135

PAGE 136

PAGE 137

PAGE 138

PAGE 139

PAGE 140

PAGE 141

PAGE 142

PAGE 143

PAGE 144

PAGE 145

PAGE 146

PAGE 147

PAGE 148

PAGE 149

PAGE 150

PAGE 151

PAGE 152

PAGE 153

PAGE 154

PAGE 155

PAGE 156

PAGE 157

PAGE 158

PAGE 159

PAGE 160

PAGE 161

PAGE 162

PAGE 163

PAGE 164

PAGE 165

PAGE 166

PAGE 167

PAGE 168

PAGE 169

PAGE 170

PAGE 171

PAGE 172

PAGE 173

PAGE 174

PAGE 175

PAGE 176

PAGE 177

PAGE 178

PAGE 179

PAGE 180

PAGE 181

PAGE 182

PAGE 183

PAGE 184

PAGE 185

PAGE 186

PAGE 187

PAGE 188

PAGE 189

PAGE 190

PAGE 191

PAGE 192

PAGE 193

PAGE 194

PAGE 195

PAGE 196

PAGE 197

PAGE 198

PAGE 199

PAGE 200

PAGE 201

PAGE 202

PAGE 203

PAGE 204

PAGE 205

PAGE 206

PAGE 207

PAGE 208

PAGE 209

PAGE 210

PAGE 211

PAGE 212

PAGE 213

PAGE 214

PAGE 215

PAGE 216

PAGE 217

PAGE 218

PAGE 219

PAGE 220

PAGE 221

PAGE 222

PAGE 223

PAGE 224

PAGE 225

PAGE 226

PAGE 227

PAGE 228

PAGE 229

PAGE 230

PAGE 231

PAGE 232

PAGE 233

PAGE 234

PAGE 235

PAGE 236

PAGE 237

PAGE 238

PAGE 239

PAGE 240

PAGE 241

PAGE 242

PAGE 243

PAGE 244

PAGE 245

PAGE 246

PAGE 247

PAGE 248

PAGE 249

PAGE 250

PAGE 251

PAGE 252

PAGE 253

PAGE 254

PAGE 255

PAGE 256

PAGE 257

PAGE 258

PAGE 259

PAGE 260

PAGE 261

PAGE 262

PAGE 263

PAGE 264

PAGE 265

PAGE 266

PAGE 267

PAGE 268

PAGE 269

PAGE 270

PAGE 271

PAGE 272

PAGE 273

PAGE 274

PAGE 275

PAGE 276

PAGE 277

PAGE 278

PAGE 279

PAGE 280

PAGE 281

PAGE 282

PAGE 283

PAGE 284

PAGE 285

PAGE 286

PAGE 287

PAGE 288