

Overview:

When a patient shifts from one provider to another, the accuracy of the medical records that get transmitted has an inverse bearing on the redundancy and complexity of actions that follow. Seamless interoperability holds the key to the riddle. We undertook this project to put in place an approach that can be undertaken to affect seamless exchange of patients' medical records across all key healthcare stakeholders. It was done using data and communication standards devised by designated bodies to be followed universally across the industry along with a healthcare integration engine.

Client details:

Name: Confidential | Type: Healthcare | Location: US

Technologies:

- Docker, Docker Compose
- Google Cloud
- Java, Spring Boot
- Message Queue RabbitMQ



Project Description:

With Interoperability being considered as the silver-bullet for the healthcare industry, efforts are underway to make it an integral part of the ecosystem. Widespread adoption of common communication standards is expected to provide the foundation for this to be realized.

This project is an exhibition of patient data interoperability, implemented using the most universally accepted standards for data and communication. This project addresses one of the most fundamental needs of patients – of getting their medical records made available to providers as they move from one to the other. This is of critical importance for specialist referrals and for care transition in general. Details of the project are as follows:

Entity 1: Provider A which uses OpenMRS

Entity 2: Provider B which uses Google Healthcare APIs

Goal: Exchange data between Provider A and Provider B using NextGen Connect

About OpenMRS:

It is a leading open source enterprise EMR system platform. It comes with a FHIR module, which exposes patient related data via REST API calls. Some of the entities are listed below:

Allergy Intolerance	Family Member History	Medication Request	Person
Condition	Group	Observation	Practitioner
Diagnostic Report	Location	Operation Definition	Procedure Request
Encounter	Medication	Patient	Related Person

We picked a few related entities from the above list to process, i.e. Patient, Allergy Intolerance, Condition, Observation, Diagnostic Report, Encounter

About Google Cloud Healthcare API:

It is a fully managed and secure cloud development environment for health care applications that support HIPAA compliance. It facilitates the exchange of data between health care applications and



solutions built on Google Cloud. Organizations leveraging it can take in data from a wide range of systems and perform desired actions on them.

We setup the Google Healthcare APIs, i.e. FHIR Stores within datasets, and were able to perform CRUD operations using postman. It was used as the destination for the data sent via NextGen Connect.

About NextGen Connect:

It is a cross-platform integration engine used extensively in the healthcare industry that facilitates bidirectional exchanges of different types of messages. This is done by creating multiple channels for transferring data from source to destinations. There are many types of connectors for Source/Destination, for e.g. Channel Reader/Writer, DICOM Listener/Sender, Database Reader/Writer, File Reader/Writer, HTTP Listener/Sender, JMS Listener/Sender, JavaScript Reader/Writer, TCP Listener/Sender, Web Service (SOAP) Listener/Sender. We used HTTP Listener/Sender for this project.

Solution

- The process of data transfer begins with a "Request" getting raised through the Message Broker (RabbitMQ). A "Request" comprises of a "Patient ID" and "Names of Resources" for the corresponding Patient ID that are to be transferred e.g. allergy, condition, observation. These requests are shared in JSON format.
- The Worker Agent fetches such a request from the Message Broker and sends it to OpenMRS (Provider A) with the required details: Patient ID, Names of Resources to be fetched. The Worker Agent is an application developed using Java, Spring boot.
- After pulling the required data from OpenMRS, the Worker Agent sends it to NextGen Connect.
- NextGen Connect allows channels to be created within to transfer data. One such is specifically linked to Google Healthcare API and acts as the conduit for transferring this data pulled from OpenMRS.
- With this approach, data can be transferred from a single source (e.g. Provider A using OpenMRS) to multiple destinations. This can be achieved by creating dedicated channels within NextGen Connect that get mapped to each of the target destinations. For e.g. Provider C (refer Architecture diagram)



Architecture:



Screenshots



Screenshot 1: Patient Data in OpenMRS



Overview		
	Connections Channels Exchanges Queues Admin	User guest Log out
Publish mes	ssage	
essage will be p	published to the default exchange with routing key interopQueue, routing it to this queue.	
elivery mode:	1 - Non-persistent ♥ Message published.	
Headers: ?	Content-Type = applica = Close	
roperties: ?	-	
	/	
		Ν
ıblish message		Ç₂
ublish message Get message	es	C ₂

Screenshot 2: Request from Message Broker





Screenshot 3: NextGen Channel

		nttps:	//interop-nextgen-co	innect.interop.iocainost	28443 - Mirth Connect	Administrator - (3.5	k(1)		
Mirth Connect 🛞								mith conne	
Daubhaund	Start Time:	12:40 PM	All Day REC	EIVED	Current Search-				
Dashboard		TRANSFORMED		Max Message Id: 9	Max Message Id: 9				
Liners	End Time:	ime: IIII 12:40 PM		Date Range: (any)	Results 1 - 9 of 9 Count				
Cattions	Text Search: QUEUED		Connectors: (any)			Page 1 of 1 Ge			
Aleste			π						
Events	Page Size:	20 Advanced Reset	Search	arch ERROR			< Prev Next >		
Extensions	14	Connector	Oratur	Received Date	Response Date	Emper	SOURCE	TYPE	
	10	Source	TRANSFARMED	2020-08-28 15-02-48-837	Kesponse Late	errors	SURCE	ine	
		Send HI 7 message to Store	SENT	2020-08-28 15:02:49:400	2020-08-28 15:02:51:043				
iessage Tasks 🛞		Source	TRANSFORMED	2020-08-28 15:02:06:657					
		Cand M 7 marrana to Chora	CENT	2020-06-28 15:02:07:160	2020-06-26 15:02:09:000		-		
Keresn	0.1	Course	TRANCEORMED	2020-00-20 15:00-51:014		-			
Send Message		Source	TRANSFORMED	2020-08-28 15:00:51:914	2020 00 20 10 01 20 120				
Import Messages		Seno HL7 message to store	SENI	2020-08-28 15:00:59:199	2020-00-28 15:01:38:128				
Export Results	80	Source	TKANSFORMED	2020-08-28 14:43:47:160	-	-	-	-	
Remove All Messages		Send HL/ message to store	SENI	2020-08-28 14:43:48:189	2020-08-28 14:43:49:409	-			
Remove Results	85	Source	TRANSFORMED	2020-08-28 14:42:09:823		-			
Remove Message	_	Exchange JWT for Access Token	SENT	2020-08-28 14:42:13:056	2020-08-28 14:42:15:497	-	-	-	
Reprocess Results		Send HL7 message to Store	SENT	2020-08-28 14:42:16:091	2020-08-28 14:43:08:545	-	-	-	
Reprocess Message	84	Source	TRANSFORMED	2020-08-28 14:39:02:943		-		-	
		Send HL7 message to Store	ERROR	2020-08-28 14:39:03:322	2020-08-28 14:39:04:145	Processing	-	-	
Other (A)	<u>a 1</u>	Source	TRANSFORMED	2020-08-28 14:38:12:336					
viner 🖉	Messages	Mappings							
Notifications (1)	Scope		Variable				Value		
View User API	Source	headers			{content-length=[2741], upgrade=[h2c], connection=[Upgrade, HTTP2-Settings], host=[interop-nextgen-con				
View Client API	Source	localPort			9002				
Help	Source	method			POST				
About Mirth Connect	Source	query							
Visit nextgen.com	Source	remotePort			50172				
Report Issue	Source	contextPath			/Cond	ition			
Jogout.	Source	uri			/Cond	ition			
a notion	Source	url			http://	Interop-nextgen-conne	t:9002/Condition		
	Source	protocol			HTTP/	HTTP/1.1			
	Source	localAddress			172.2	0.0.6			
	Source	parameters			0				
	Source	remoteAddress			172.2	172.20.0.8			
	Source	destinationSet			[3]				
	Channel	service account				mindfire-283914@appspot.gserviceaccount.com			
	Channel	DESTINATION ENDPOINT			https://	//healthcare.googleacis	.com/v1beta1/projects/mindfire-283914/h	ocations/us-central1/datasets/dhealth1/fhirSt-	
	Channel	nrivata kav							
		Annual							

Screenshot 4: NextGen Channel Messages

New Import Runner 🗔 🗸	品 My Workspace 🗸 🧍 Invite 🛛 📀 순 🕸 🗘 ♡ 🌔	Upgrade	
Filter		* ©	
History Collections APIs	 SearchGoogleEncounter Examples () * 	BUILD	
New Collection Trash	GET + https://healthcare.googleapis.com/v1beta1/projects/mindfire-283914/locations/us-central1/datasets/dhealth1/fhirSto Send	- Save	
ET worker			
iET sendQueue	Params Authorization Headers (7) Body Pre-request Script Tests Settings	Cookies	s Code
ET mirth	Body Cookies Headers (11) Test Results Cookies Headers (11) Test Results 26.87 KB	Save Respo	onse v
ET DeleteResouceGoogleCloud			
T SearchGoodeEncounter	Pretty Raw Preview Visualize JSON * 📅	1	Q
Searchooglechoonler	519 }		
DeleteGoogleEncounter	520),		
DeleteGooglePatient	S21 Seench : {		
SearchGooglePatient	523 X		
see a se	524		
GetGoogleAllergy	525 1.		
Delete George Alleren	526 "link": [
DeletegoogleAllergy	527 \$		
GetGoogleObservation	528 "relation": "search",		
	529 "url": "https://healthcare.googleapis.com/vlbetal/projects/mindfire-283914/locations/us-central1/datasets/dhealth	1/fhirStores/	
DeleteGoogleObservation	fhealth1/fhir/Encounter/?patient=8ea8d6c9-6657-43d4-8e77-de2287b05776"		
GetGoogleDiagnosticReport	530 >		
	531],		
EL DeleteGoogleDiagnosticReport	532 "resourceType": "Bundle",		
GetGoogleCondition	533 "total": 10,		
dergoogleconoidon	534 "type": "searchset"		- L
DeleteGoogleCondition	535 0		
	-		



Screenshot 5: Google APIs receiving transmitted Patient Data