



Healthcare Practical Guide (eHealth - HL7)

-
- [Document Control](#)
 - [Introduction to eHealth Systems](#)
 - [HL7 Standards](#)
 - [HL7 Integration Scenarios](#)
 - [HL7 Interface](#)
 - [HL7 Interface Specifications](#)
 - [HL7 Message Types](#)
 - [HL7 Message Structure](#)
 - [HL7 Soup Example](#)
 - [MuleSoft in eHealth Systems](#)
 - [API-led Connectivity Approach](#)
 - [MuleSoft Healthcare Toolkit](#)
 - [Technical Challenges](#)
 - [Appendixes](#)
 - [Appendix 1: Acronyms](#)
 - [Appendix 2: References and Tools](#)

1 Document Control

Author	Amr Salah
Scope	eHealth systems (HL7 Integration)
Focus	eHealth systems and healthcare
Version	Draft

2 Introduction to eHealth Systems

E-health simply means the application of the latest information and communication technologies in all health-related fields such as collecting, storing, restoring, analysing and managing the information, unifying the electronic health records, disseminating and sharing medical information, surgeries and healthcare remotely, in addition to smart e-healthcare cards. The aim is to achieve stronger and more effective communication with patients and upgrade healthcare services and the entire healthcare sectors. It is all about digitising healthcare systems and records.

The promise of **eHealth** is comprehensive: better quality of care, more effective and efficient care, better service to the **patient**, more control for the **patient** (shared decision making, self-management **support**) and growing availability of high-quality data for quality assurance, education and research.

Due to the level of complexity into eHealth systems, Let's explore a bit more about the technical landscape for eHealth and the available integration standards we are dealing with before we jump into what MuleSoft offers to this domain.

Acronyms

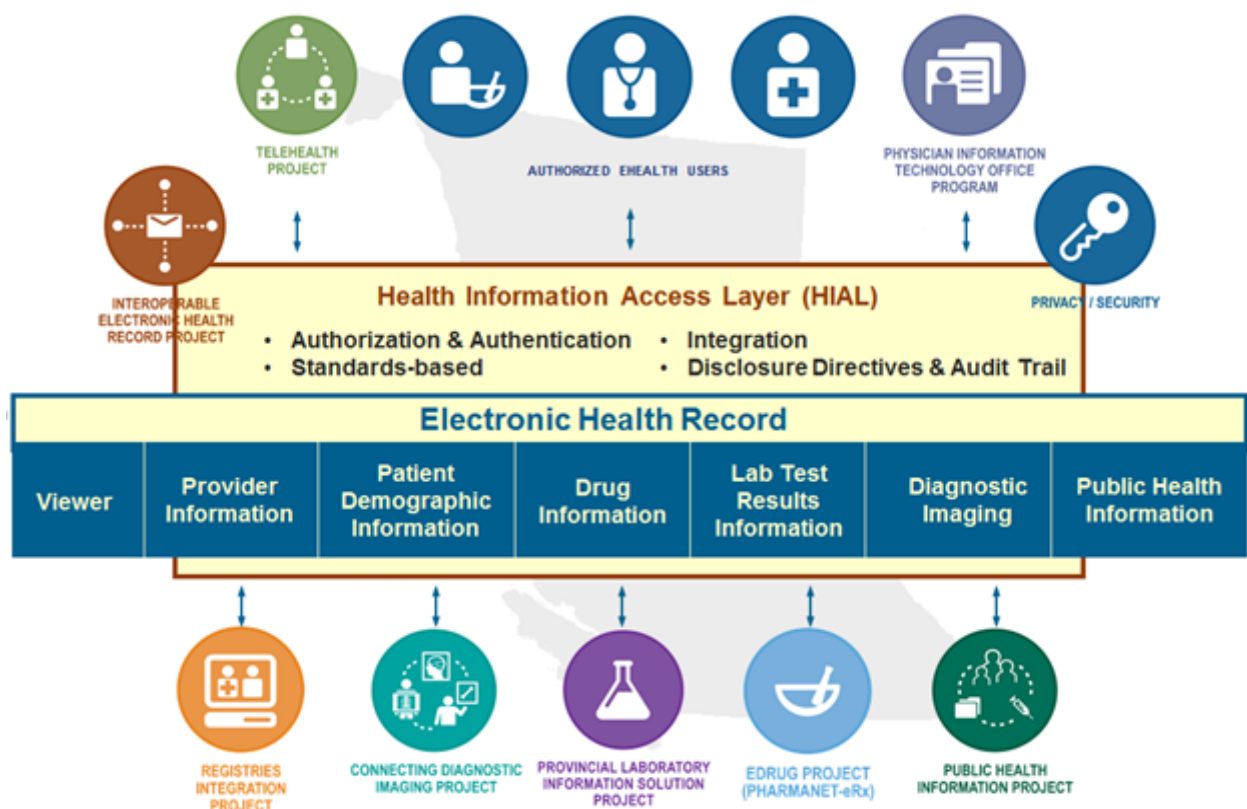
Term	Description
eHealth	eHealth is a broad term, and refers to the use of information and communications technologies in healthcare.
HIS (Health Information Systems)	<p>is essentially a computer system that can manage all the information to allow health care providers to do their jobs effectively.</p> <p>Nowadays, HIS consists of multiple integrated systems serve different purposes. For instance:</p> <ul style="list-style-type: none">• Electronic Medical Record (EMR),• Hospital Information System (HIS),• Radiology Information System (RIS),• Laboratory Information System (LIS),• Dietary, PACS, Emergency Department, transcription,• and many others <p>All systems are integrated together using a predefined international standards called "HL7".</p>
HL7	Set of international standards created by Health Level Seven International for transfer of clinical and administrative data between software applications.

MuleSoft uses HL7 standards to enable the integration between health systems by building few connectors complied and aligned with HL7 structure. Later, will talk more about How to integrate systems using MuleSoft connectors, but now let's deep dive into HL7 and understand how it works.

3 HL7 Standards

What is HL7?

Health Level Seven or **HL7** refers to a set of international standards created by Health Level Seven International for transfer of clinical and administrative data between software applications used by various healthcare providers. HL7 and its members provide a framework (and related standards) for the exchange, integration, sharing, and retrieval of electronic health information. These standards define how information is packaged and communicated from one party to another, setting the language, structure and data types required for seamless integration between systems.



HL7 messages transmit data between disparate systems.

HL7 standards are grouped and classified into multiple sections as following:

Standards	Description
Section 1: Primary Standards	Primary standards are considered the most popular standards integral for system integrations, inter-operability and compliance. Our most frequently used and in-demand standards are in this category.
Section1: Primary Standards	Description
Section 1a: Clinical Document Architecture (CDA®)	Clinical Document Architecture (CDA®) Products
Section 1b: EHR - Electronic Health Records	These standards provide functional models and profiles that enable the constructs for management of electronic health records.
Section 1c: FHIR®-Fast Health Interop Resources	FHIR® – Fast Health Interop Resources
Section 1d: Version 2 (V2)	Version 2 (V2) - The most commonly used version.
Section 1e: Version 3 (V3)	HL7 Version 3 (V3) - a suite of specifications based on HL7's Reference Information Model (RIM)
Section 1f: Arden Syntax	The Arden Syntax is a formalism for representing procedural clinical knowledge in order to facilitate the sharing of computerised health knowledge bases among personnel, information systems and institutions
Section 1g: CCOW	HL7 Clinical Context Management Specification (CCOW) is aimed at facilitating the integration of applications at the point of use, as a standard for both internal applications programming and runtime environment infrastructure that complements HL7's traditional emphasis on data interchange and enterprise workflow.
Section 1h: Cross-paradigm/ Domain Analysis Models	Cross-paradigm/ Logical Level Standards e.g. Domain Analysis Models
Section 2: Clinical and Administrative Domains	Messaging and document standards for clinical specialties and groups are found in this section. These standards are usually implemented once primary standards for the organisation are in place.
Section 3: Implementation Guides	This section is for implementation guides and/or support documents created to be used in conjunction with an existing standard. All documents in this section serve as supplemental material for a parent standard.
Section 4: Rules and References	Technical specifications, programming structures and guidelines for software and standards development.

Above standards together form end to end integration best practices for health systems and they are used widely to deliver highly integrated solutions between those systems.

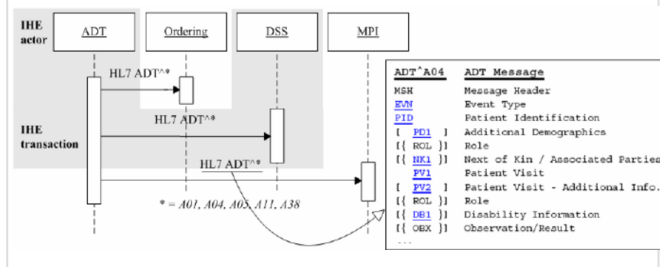
Let's have a look at the following scenarios to understand what type of integrations between systems might be required:

4 HL7 Integration Scenarios

Scenario 1

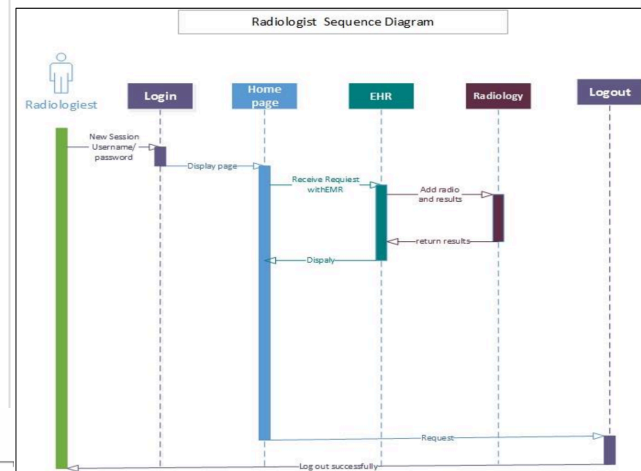
Scenario 1: Patient registration requires interactions between multiple systems as shown below

- Admit Dis-charge Transfer (ADT): carry patient demographic information
- Decision Support Service (DSS): takes in patient data as the input and provides back patient-specific assessments and recommendations
- Master Patient Index (MPI) : assessment of whether a candidate patient record is a match or not



Scenario 2

Scenario 2: Radiologist loads patient data into systems which requires integration between multiple systems as shown below:



4.1 The communications between those systems happen by sending HL7 messages to their interface as displayed below. In next section will explore more about the interfaces and message structure.

HL7 - **Version 2 (V2)** is the most commonly used across the health system nowadays, so let's explore the HL7 version 2 in more details and understand the interface, message types and structures.

5 HL7 Interface

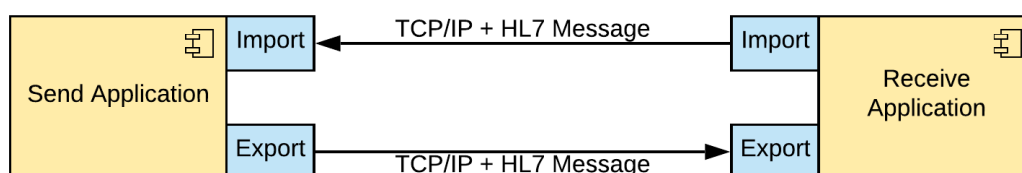
HL7 Interface/integration interface definition consists of two parts

- **First**, HL7 is **Health Level Seven**. It is an organisation of volunteers who define the data specifications for various message types (e.g., **ADT**, **ORM**, **ORU**, etc.). These HL7 specification documents provide the framework in which to communicate patient information between healthcare organisations. There are several **versions of HL7** – versions 2.x and version 3. Today, the different versions under the HL7 version 2.x family are the most commonly used.
- The “interface” is defined as an “interconnection” or “interaction” or “communication” between applications or systems. Healthcare application vendors need to provide a means within their application for this interaction to take place.≠

5.1 HL7 Interface Specifications

Healthcare vendors will use the HL7 interface specifications for the various message types. HL7 interface includes:

- An export endpoint for the sending application
- An import endpoint for the receiving application
- A method of moving data between the two endpoints (**TCP/IP** communication is the most common transport)



HL7 Example

```
MSH|^~\&|MegaReg|XYZHospC|SuperOE|XYZImgCtr|20060529090131-0500||ADT^A01^ADT_A01|01052901|P|2.5
EVN||200605290901|||200605290900
PID||56782445^^^UAREg^PI||KLEINSAMPLE^BARRY^Q^JR||19620910|M||2028-9^^HL70005^RA99113^^XYZ|260
GOODWIN CREST DRIVE^^BIRMINGHAM^AL^35209^^M-NICKELL'S PICKLES^10000 W 100TH
AVE^BIRMINGHAM^AL^35200^^O|||||0105I30001^^^99DEF^AN
PV1||I|W^389^1^UABH^^^3||||12345^MORGAN^REX^J^^^MD^0010^UAMC^L|
|67890^GRAINGER^LUCY^X^^^MD^0010^UAMC^L|MED||||A0||13579^POTTER^SHERMAN^T^^^MD^0010^UAMC^L|
|||||||||||||||||200605290900 OBX|1|NM|^Body Height||1.80|m^Meter^ISO+||||F
OBX|2|NM|^Body Weight||79|kg^Kilogram^ISO+||||F AL1|1|^ASPIRIN DG1|1||786.50^CHEST PAIN,
UNSPECIFIED^I9|||A
```

The message structure is part of the interface as the message content has a composite key (version, message type, message sub-type) to define the message trigger event. The message trigger event gives the system information about what is the message about and how to process incoming data.

For example:

The following table shows that the message coming with a type (ADT) and sub-type (ADT-A04) on version 2.5.1 means register a patient.

For the full list of versions, messages types and events please [visit this link](#).

Version		Message Type	Message Sub-type	Event/Trigger
2.1	2.5.1	ACK	ADT-A01	Admit/Visit Notification
2.2	2.6	ADT	ADT-A02	Transfer a Patient
2.3	2.7	DFT	ADT-A03	Discharge/End Visit
2.3.1	2.8	MDM	ADT-A04	Register a Patient
2.4	2.8.1	ORU	ADT-A05	Pre-Admit a Patient
2.5	2.8.2	RDE	ADT-A06	Charge an OutputPatient to an Inpatient

Now let's understand more about the HL7 message types and structure.

5.2 HL7 Message Types

As shown above message type defines how the system is going to process the message and there are many message types and trigger events. For example ADT (Admit Discharge Transfer) type has triggers and events in this range ADT-A01 to ADT-A17 and so on. Below is a list of them most common and used messages

Most commonly used HL7 message types include:

- ACK – General acknowledgement
- ADT – Admit, Discharge, Transfer
- BAR – Add/change billing account
- DFT – Detailed financial transaction
- MDM – Medical document management
- MFN – Master files notification
- ORM – Order (Pharmacy/treatment)
- ORU – Observation result (unsolicited)
- QRY – Query, original mode
- RAS – Pharmacy/treatment administration
- RDE – Pharmacy/treatment encoded order
- RGV – Pharmacy/treatment give
- SIU – Scheduling information unsolicited
- OBX - Observation Segment

Each of the above message types and subtypes has different message format, however, some of them are sharing the same format e.g. ADT-A01 to ADT-A08 . In next section we will discuss the message structure.

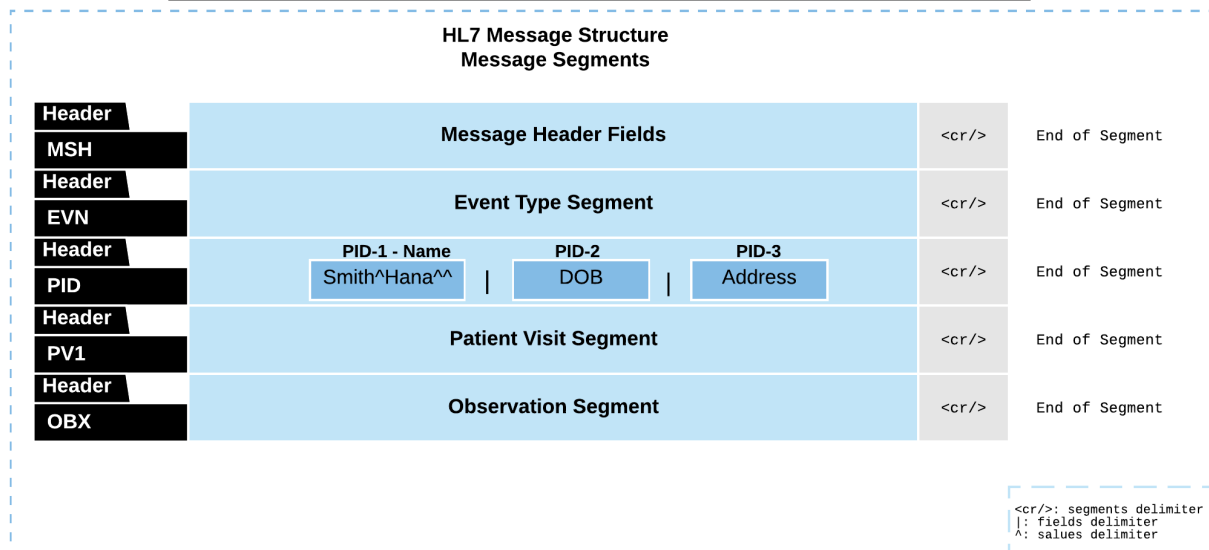
5.3 HL7 Message Structure

As shown in below diagram, HL7 message consists of

- Multiple segments delimited by a new line <cr/>.
- Each segment consists of
 - Header which holds the segment type and;
 - Segment body consists of multiple fields delimited by | character.
- Each field has a name (SegmentType-Index). E.g. PID segment has multiple fields in the following order (PID-1, PID-2, PID-3)
- Each field has multiple values delimited by ^ character.
- Having multiple Fields delimiters together (e.g. |||) means there are multiple empty fields

- Having multiple values delimiters together (e.g. ^^) means there are multiple empty values.

```
MSH|^~\&|MegaReg|XYZHospC|SuperOE|XYZImgCtr|20060529090131-0500||ADT^A01^ADT_A01|01052901||P2.5
EVN||200605290901|||200605290900
PID||56782445^^^UReg^PI||KLEINSAMPLE^BARRY^Q^JR||19620910|M||2028-9^^HL70005^RA99113^^XYZ|260
GOODWIN CREST DRIVE^^BIRMINGHAM^AL^35209^^M-NICKELL'S PICKLES^10000 W 100TH
AVE^BIRMINGHAM^AL^35200^^0|||0105I30001^^^99DEF^AN
PV1||I|W^389^1^UABH^^^3|||12345^MORGAN^REX^J^^MD^0010^UAMC^L|
|67890^GRAINGER^LUCY^X^^MD^0010^UAMC^L|MED|||AO||13579^POTTER^SHERMAN^T^^MD^0010^UAMC^L|
||||||||||||||||200605290900 OBX|1|NM|^Body Height||1.80|m^Meter^ISO+||||F
OBX|2|NM|^Body Weight||79|kg^Kilogram^ISO+||||F AL1|1|^ASPIRIN DG1|1||786.50^CHEST PAIN,
UNSPECIFIED^I9|||A
```



Each message have different structure and segment types depends on the message type. Please visit this [link](#) for more details about the different message types and segments.

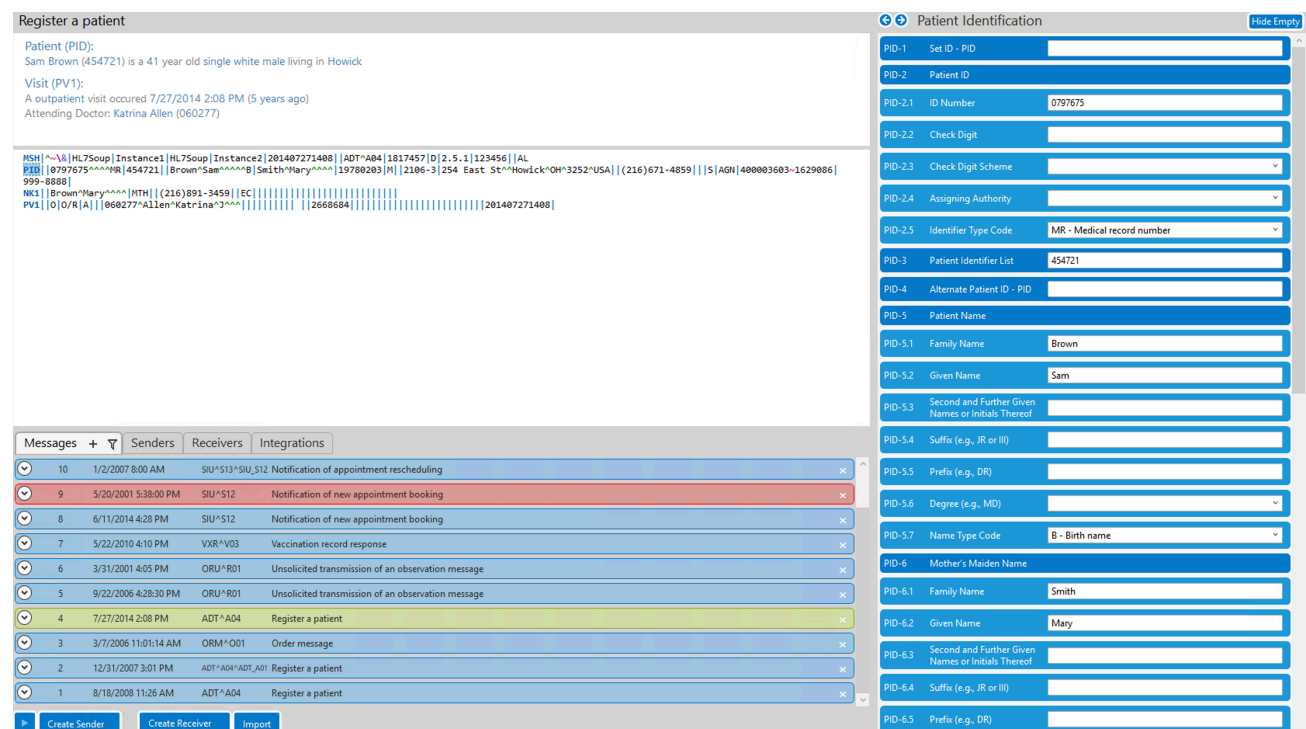
There are many tools available to help you analyse and understand the different messages and segments and provide full details about each message.

- [7Edit](#)
- [HL7 Inspector](#)
- [HL7 Soup](#)

5.4 HL7 Soup Example

In the below photo, HL7 Soup analyses the message and shows each segment fields by clicking on the header. Also you can click on the below list to generate different message formats.

HL7 Interface



Exported by obd-svc

6 MuleSoft in eHealth Systems

MuleSoft provides two main things for eHealth systems:

1. API-led connectivity approach
2. MuleSoft Healthcare Toolkit

6.1 API-led Connectivity Approach

API-led connectivity is a methodical way to connect data to applications through reusable and purposeful **APIs**. You can learn more about **API-led Connectivity** [here](#).

Let's have a look at how API-led connectivity approach for eHealth scenario could look like

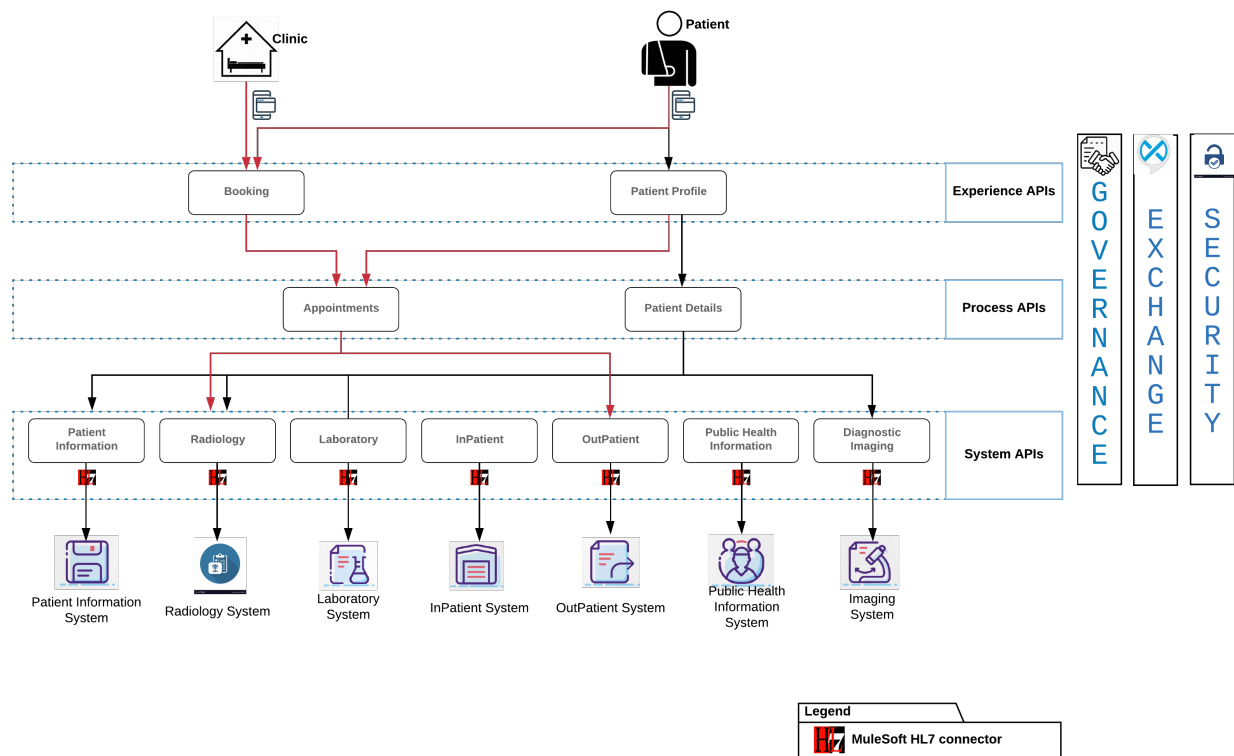
Below diagram shows a sample of digitising a user experience for two main functionalities by building an integration between hospital information systems :

1. Enable the patient to view the profile (incl. laboratory test results, radiology results, diagnostic images and information)
2. Enable clinics to view the patient profile and schedule an appointment

API-led provided the following features:

- All APIs are secured and can be self-discovered through exchange with respect to security.
- Provides a great level of reusability between APIs.
- Enhanced the user experience and the business ROI by providing a quick way to consume and easy way to developer a new experience API.

This is not the only architecture can be used to integrate between systems as there are multiple integration patterns can fit into API-led connectivity for more reliability and performance (e.g. EDA). So by the end it's all about the requirement and criteria.



In each of the above System APIs, HL7 connector is used to integrate with one of health information systems. So let's talk more about MuleSoft connectors and their usage.

6.2 MuleSoft Healthcare Toolkit

Healthcare Toolkit includes the following elements.

Type	Description
Connector	HL7 EDI - Electronic Data Interchange (EDI) schemas, configuration, and operations.
API Documentation	HL7 Technical Reference
Connector	HL7 MLLP Reference - Minimal Lower Layer Protocol (MLLP)
Testing	Connector Testing with TestPanel
Release Notes	HL7 EDI Connector Release Notes , HL7 MLLP Connector Release Notes

6.2.1 HL7 EDI Connector

HL7 EDI lets you convert HL7 ER7 messages to and from DataWeave-compatible representations using lists and maps.

HL7 EDI includes:

- HL7 ER7 message-reading, message-validation, and message-writing
- Integration with DataSense and DataWeave

- HL7 message packs for v2.1, v2.2, v2.3.1, v2.3, v2.4, v2.5, v2.5.1, v2.6, 2.7, 2.7.1, 2.8, and 2.8.1.
- The ability to define your own schemas or customise the base HL7 schemas

Check the [documentation](#) for more information about it.

6.2.2 HL7 MLLP Connector

Premium

The Mule HL7 MLLP transport provides connectivity and parsing functionality for HL7 MLLP messages.

Check the [documentation](#) for more information about it.

Build Your First Project with HL7 Connectors

Perquisites:

1. Download and Setup [Java JDK](#) - for running Hapi Testpanel and Anypoint studio
2. Download [hapi-testpanel-2.3-dist.zip](#) - This is used to simulate the experience of health information systems, transmit, send and receive messages to your application
3. Download and setup [Anypoint Studio](#) - MuleSoft Developers IDE
4. Setup hapi-testpanel
 - a. Extract the zip file
 - b. run the following commands to run Hapi

Run Hapi Testpanel

```
cd hapi-testpanel-2.3
java -jar hapi-testpanel-2.3-jar-with-dependencies
```

Implementation Steps:

The application is available [here](#) to download.

- Create new project on Anypoint studio. Let's call it 'HL7-app-demo'
- From the right palette (Menu Palette), search on exchange on HL7 and add the following two modules:
 - HL7 MLLP listener
 - HL7 EDI

Search: HI7

Available modules		Selected modules	
Name	Publisher	Name	Version
HL7 MLLP Connector - M	MuleSoft	HL7 MLLP Connec...	2.1.1
HL7 Connector - Mule 4	MuleSoft	HL7 Connector - ...	4.0.2

Buttons: Add > < Remove

- Now you can expand the connector jars and explore the available schemas come with a connector for different versions, message types, segment types, events and triggers

```

▼ HL7 EDI [v4.0.2]
  ► edi-parser-2.2.3.jar - /Users/amr.salah/.m2/repository/org/mule/edi/edi-parser
  ▼ hl7-schemas-4.0.2.jar - /Users/amr.salah/.m2/repository/com/mulesoft/connec
    ► com.mulesoft.connectors.hl7
      ► com.mulesoft.flatfile.schema
        ► hl7.v2_1
        ► hl7.v2_2
        ► hl7.v2_3
        ► hl7.v2_3_1
        ► hl7.v2_4
        ► hl7.v2_5
        ► hl7.v2_5_1
        ► hl7.v2_6
        ► hl7.v2_7
        ► hl7.v2_7_1
        ► hl7.v2_8
        ▼ hl7.v2_8_1
          ► ACK.esl
          ► ADT_A01.esl
          ► ADT_A02.esl
          ► ADT_A03.esl
          ► ADT_A05.esl
          ► ADT_A06.esl
          ► ADT_A09.esl
          ► ADT_A12.esl
          ► ADT_A15.esl
          ► ADT_A16.esl
          ► ADT_A17.esl
          ► ADT_A20.esl
          ► ADT_A21.esl
          ► ADT_A24.esl
  
```

- each of above scheme ends with *.esl and is written into yaml syntax which is the scheme definition for **different events and triggers**.

Scheme Syntax example

Below is an example of scheme definition. Check comments below for clarification.

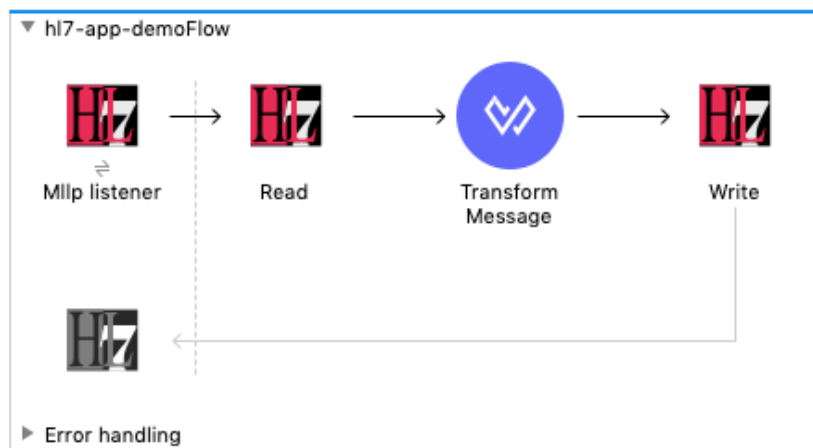
ADT_A02 Scheme Syntax

```

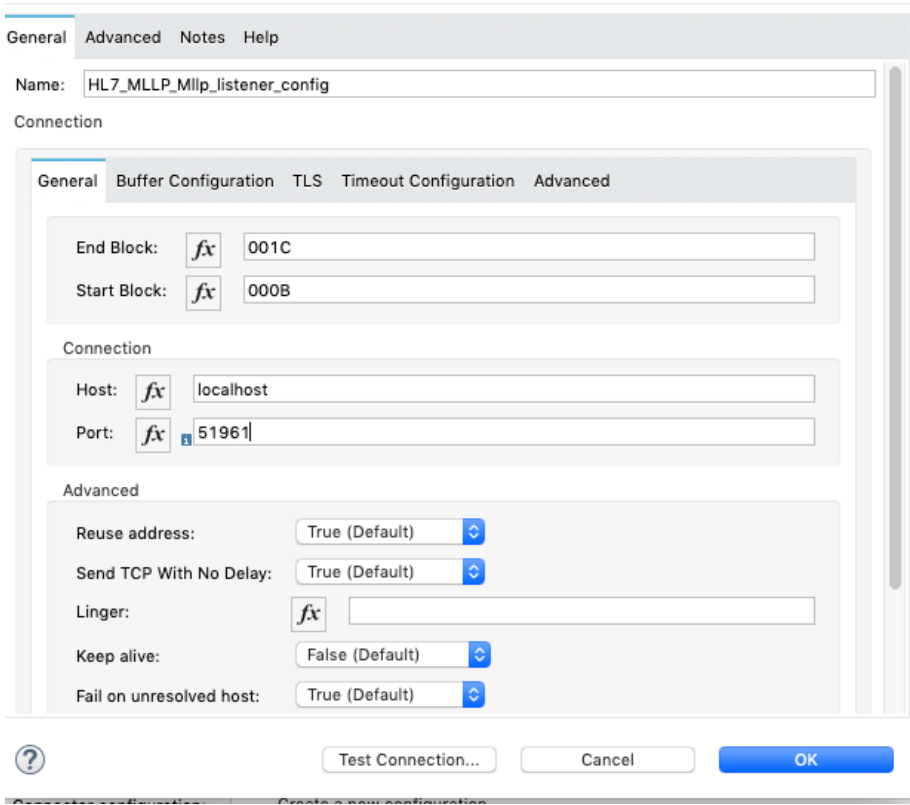
form: HL7 #the message is coming from HL7 system
version: '2.8.1' # message version
imports: [ '/hl7/v2_8_1/basedefs.es1' ] #import another scheme which has the common definition across all
schemas.
id: 'ADT_A02' # message Id
name: 'ADT_A02' # message name
data:
- { idRef: 'MSH', position: '01', usage: M } # 1st segment (type: MSH - Message header) and it's position
is 1
- { idRef: 'SFT', position: '02', usage: O, count: '>1' }
- { idRef: 'UAC', position: '03', usage: O }
- { idRef: 'EVN', position: '04', usage: M } # 4th segment (type: EVN - Event type and it's position is 4
- { idRef: 'PID', position: '05', usage: M }
- { idRef: 'PD1', position: '06', usage: O }
- { idRef: 'ARV', position: '07', usage: O, count: '>1' }
- { idRef: 'ROL', position: '08', usage: O, count: '>1' }
- { idRef: 'PV1', position: '09', usage: M }
- { idRef: 'PV2', position: '10', usage: O }
- { idRef: 'ARV', position: '11', usage: O, count: '>1' }
- { idRef: 'ROL', position: '12', usage: O, count: '>1' }
- { idRef: 'DB1', position: '13', usage: O, count: '>1' }
- { idRef: 'OBX', position: '14', usage: O, count: '>1' } # 14th segment (type: OBX - Observation segment
and can multiple entries) and it's position is 14
- { idRef: 'PDA', position: '15', usage: O }

```

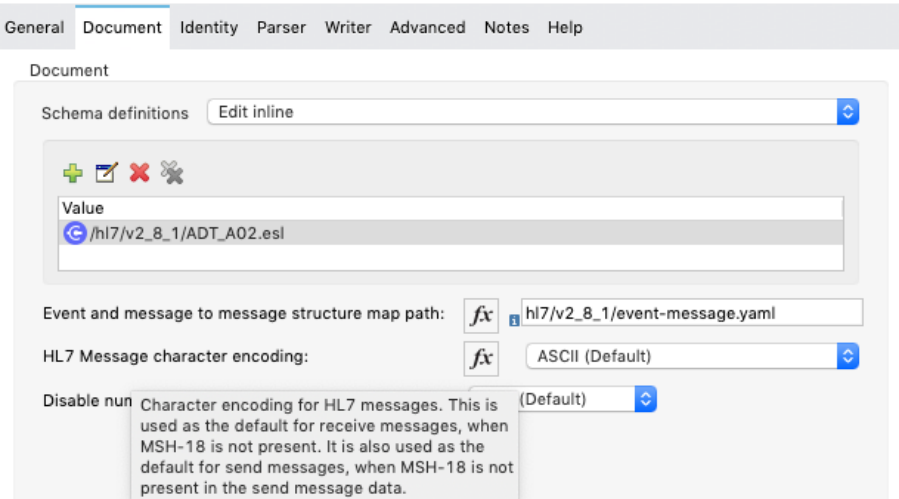
- Build the flow as following by dragging and dropping the connectors



- Configure Mllp Listner connection
 - Make sure Hapi test panel is up and running successfully as mentioned above
 - You are free to use the default port to connect or to change it



- Configure the HL7 EDI connection
 - Pass the scheme is going to be used to read and parse the HL7 incoming message
 - pass the event message as mentioned blow.
 - Event message: Each version of HL7 defines a different set of mappings from event type and message type to message structure. The default mappings are provided in the same jar as the standard HL7 schema definitions, in files named event-message.yaml.



- Run the application and monitor the logs

- Go back to Hapi test panel and Add a message then send it. From HAPI test panel logs you can see the message is sent successfully to the host

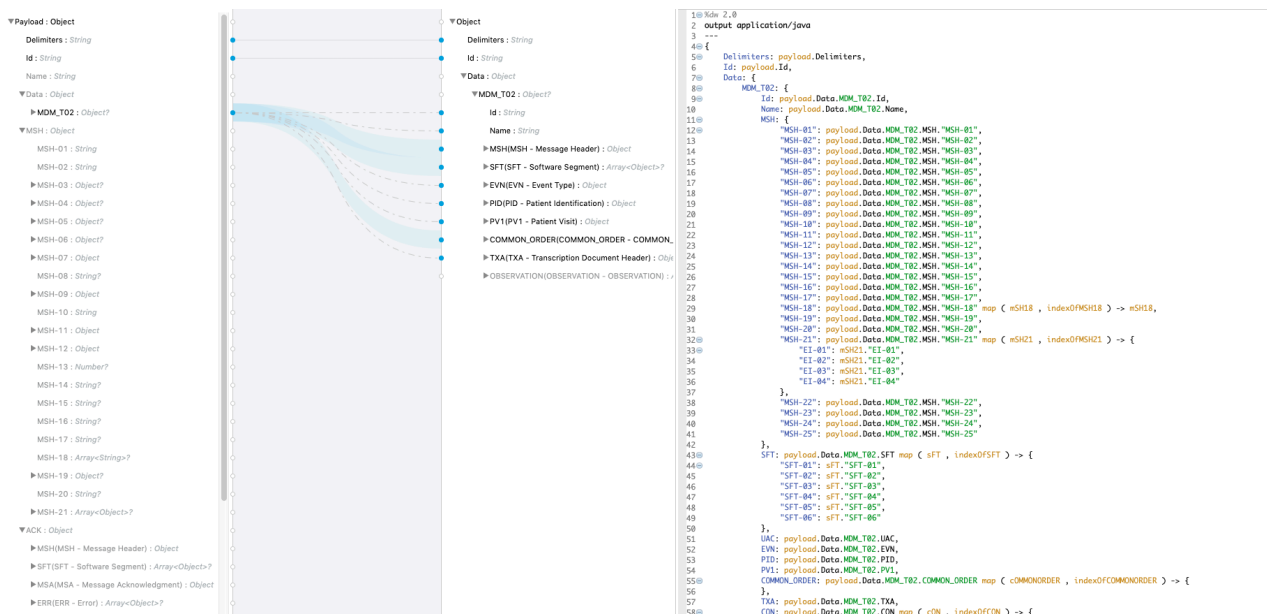
19:37:34.809	Info	Starting interface "localhost:51961"...
19:37:34.810	Outgoing	MSH ^~\& MedOne FACILITY A CARECENTER^HL7NOTES HFH 20060105180000 D61AFE1-B10E-11D5-8666-000 EVN T01 20060105180000 PID 1 112388^BS ESPARZA^MARIA PV1 1 O BS^15^15 TXA 1 CN TX 200601051800 50041^SMITH^CHRIS^M 200601051800 200601051800 200601051800 24567^
19:37:34.810	Info	Attempting outbound connection to localhost:51961
19:37:34.810	Info	Successfully connected to localhost:51961

- Go back to your mule app logs to trace the incoming message and process it accordingly before write it back to HL7 system.

Transform the message:

Dataweave 2.0 is complied with the HL7 standards and the object sensors made it easily for developers to transform the message from one scheme to another.

Luckily, Dataweave 2.0 enables the visual transformation as shown below



6.3 Technical Challenges

There are many challenges might face you as an architect or developer which is

- Defining a custom scheme to meet the customer standards
- Parsing an incoming message through different stream (e.g. JMS messaging system) which doesn't allow you to create an object sensor for the dataweave. In this case you could do a string manipulation to generate JSON object based on your understanding for the message structure and a list of predefined mapping rules.

Dataweave 2.0 - String Manipulation

```
%dw 2.0
output application/json
---
payload as String splitBy "\n" map (segment, segmentIdx) -> {
    ((segment splitBy "|")[0]): ((segment splitBy "|") - ((segment splitBy "|") [0] )
map(field, fieldIdx) -> {
    ((segment splitBy "|")[0] ++ "-" ++ fieldIdx ): field
    }
}
```

```
1 %dw 2.0
2 output application/json
3 ---
4 payload as String splitBy "\n" map (segment, segmentIdx) -> {
5
6 ((segment splitBy "|")[0]): ((segment splitBy "|") - ((segment splitBy "|") [0] ))
7 ((segment splitBy "|")[0] ++ "-" ++ fieldIdx ): field,
8
9 }
10
11 }
12
```

```
{
  "MSH": [
    {
      "MSH-0": "A~\&"
    },
    {
      "MSH-1": "MedOne"
    },
    {
      "MSH-2": "FACILITY A"
    },
    {
      "MSH-3": "CARECENTER^HL7NOTES"
    },
    {
      "MSH-4": "HFH"
    },
    {
      "MSH-5": "20060105180000"
    },
    {
      "MSH-6": " D61AFEF1-B10E-11D5-8666-0004ACD80749"
    },
    {
      "MSH-7": "MDM^T01^MDM_T02"
    },
    {
      "MSH-8": "20060105180000999999"
    },
    {
      "MSH-9": "T"
    },
    {
      "MSH-10": "2.3"
    },
    {
      "MSH-11": "5050"
    },
    {
      "MSH-12": "res"
    },
    {
      "MSH-13": "AL"
    },
    {
      "MSH-14": "AL"
    },
    {
      "MSH-15": "ABW"
    },
    {
      "MSH-16": "8859/1"
    },
    {
      "MSH-17": "QU"
    },
    {
      "MSH-18": "2.3"
    }
  ]
}
```

- other challenges requires extra work to understand about the schemas, event types, fields and possible values for each field.

7 Appendixes

7.1 Appendix 1: Acronyms

Keep remembering the following acronyms will help you to gain a better understanding about HL7 standards.

Acronym	Description
eHealth	eHealth is a broad term, and refers to the use of information and communications technologies in healthcare.
HIS (Health Information Systems)	<p>is essentially a computer system that can manage all the information to allow health care providers to do their jobs effectively.</p> <p>Nowadays, HIS consists of multiple integrated systems that serve different purposes. For instance:</p> <ul style="list-style-type: none"> • Electronic Medical Record (EMR), • Hospital Information System (HIS), • Radiology Information System (RIS), • Laboratory Information System (LIS), • Dietary, PACS, Emergency Department, transcription, • and many others <p>All systems are integrated together using a predefined international standard called "HL7".</p>
HL7	set of international standards created by Health Level Seven International for transfer of clinical and administrative data between software applications.
ACK	General acknowledgment
ADT	Admit, Discharge, Transfer
BAR	Add/change billing account
DFT	Detailed financial transaction
MDM	Medical document management
MFN	Master files notification
ORM	Order (Pharmacy/treatment)
ORU	Observation result (unsolicited)
QRY	Query, original mode
RAS	Pharmacy/treatment administration
RDE	Pharmacy/treatment encoded order
RGV	Pharmacy/treatment give
SIU	Scheduling information unsolicited
OBX	Observation Segment

7.2 Appendix 2: References and Tools

For more information please review the following links:

- <https://docs.mulesoft.com/healthcare-toolkit/3.1/>
- <https://hl7-definition.caristix.com/v2/HL7v2.5.1/Segments>
- <https://www.hl7soup.com/HL7TutorialUnderstandingHL7MessageStructure.html>
- <https://github.com/mulesoft-catalyst/ehealth-app-demo>

References:

- <https://corepointhealth.com/resource-center/hl7-resources/hl7-mdm-message/>
- <https://docs.mulesoft.com/healthcare-toolkit/3.1/>
- <https://hl7-definition.caristix.com/v2/HL7v2.5.1/Segments>
- <https://www.hl7soup.com/HL7TutorialUnderstandingHL7MessageStructure.html>

Tools

- <https://www.hl7inspector.com/>
- <http://try-it.caristix.com:9030/default.aspx>
- <https://hapifhir.github.io/hapi-hl7v2/hapi-testpanel/>
- <https://www.hl7soup.com/>
- <https://www.lucidchart.com/>
- <https://www.mulesoft.com/platform/studio>

Your feedback is always appreciated.