

# Computing for Medicine

Lecture 12: FHIR and Analytics

# Example: Patient Resource

```
<Patient xmlns="http://hl7.org/fhir">
  <id value="glossy"/>
  <meta>
    <lastUpdated value="2014-11-13T11:41:00+11:00"/>
  </meta>
  <text>
    <status value="generated"/>
    <div xmlns="http://www.w3.org/1999/xhtml">
      <p>Henry Levin the 7th</p>
      <p>MRN: 123456. Male, 24-Sept 1932</p>
    </div>
  </text>
  <extension url="http://example.org/StructureDefinition/trials">
    <valueCode value="renal"/>
  </extension>
  <identifier>
    <use value="usual"/>
    <type>
      <coding>
        <system value="http://hl7.org/fhir/v2/0203"/>
        <code value="MR"/>
      </coding>
    </type>
    <system value="http://www.goodhealth.org/identifiers/mrn"/>
    <value value="123456"/>
  </identifier>
  <active value="true"/>
  <name>
    <family value="Levin"/>
    <given value="Henry"/>
    <suffix value="The 7th"/>
  </name>
  <gender value="male"/>
  <birthDate value="1932-09-24"/>
  <careProvider>
    <reference value="Organization/2"/>
    <display value="Good Health Clinic"/>
  </careProvider>
</Patient>
```

Resource Identity & Metadata

Human Readable Summary

Extension with URL to definition

Standard Data:  
• MRN  
• Name  
• Gender  
• Birth Date  
• Provider

# Summary of the resource

Section	Purpose	Key Elements
Resource Identity & Metadata	Tracks and manages the version and identity of this patient resource.	<code>id, meta.lastUpdated</code>
Human Readable Summary	Ensures clinicians can read an intelligible summary of the patient's data.	<code>text, div, p</code>
Extension	Adds non-core data while keeping the structure valid.	<code>extension, url, valueCode</code>
Identifier	Holds formal IDs such as MRN.	<code>identifier, system, value</code>
Standard Demographics	Core patient attributes and provider linkage.	<code>name, gender, birthDate, careProvider</code>

# Example JSON for the Patient Resource

```
json

{
  "resourceType": "Patient",
  "id": "glossy",
  "meta": {
    "lastUpdated": "2014-11-13T11:41:00+11:00"
  },
  "text": {
    "status": "generated",
    "div": "<div xmlns=\"http://www.w3.org/1999/xhtml\"><p>Henry Levin the
  },
  "extension": [
    {
      "url": "http://example.org/StructureDefinition/trials",
      "valueCode": "renal"
    }
  ],
  "identifier": [
    {
      "value": "1234567890"
    }
  ]
}
```

Base	Individuals	Entities #1	Entities #2	Workflow	Management
	<ul style="list-style-type: none"> <li>Patient 1</li> <li>Practitioner 3</li> <li>PractitionerRole 2</li> <li>RelatedPerson 2</li> <li>Person 2</li> <li>Group 1</li> </ul>	<ul style="list-style-type: none"> <li>Organization 3</li> <li>OrganizationAffiliation 0</li> <li>HealthcareService 2</li> <li>Endpoint 2</li> <li>Location 3</li> </ul>	<ul style="list-style-type: none"> <li>Substance 2</li> <li>BiologicallyDerivedProduct 0</li> <li>Device 2</li> <li>DeviceMetric 1</li> </ul>	<ul style="list-style-type: none"> <li>Task 2</li> <li>Appointment 3</li> <li>AppointmentResponse 3</li> <li>Schedule 3</li> <li>Slot 3</li> <li>VerificationResult 0</li> </ul>	<ul style="list-style-type: none"> <li>Encounter 2</li> <li>EpisodeOfCare 2</li> <li>Flag 1</li> <li>List 1</li> <li>Library 2</li> </ul>
Clinical	<h3>Summary</h3> <ul style="list-style-type: none"> <li>AllergyIntolerance 3</li> <li>AdverseEvent 0</li> <li>Condition (Problem) 3</li> <li>Procedure 3</li> <li>FamilyMemberHistory 2</li> <li>ClinicalImpression 0</li> <li>DetectedIssue 1</li> </ul>	<h3>Diagnostics</h3> <ul style="list-style-type: none"> <li>Observation 1</li> <li>Media 1</li> <li>DiagnosticReport 3</li> <li>Specimen 2</li> <li>BodyStructure 1</li> <li>ImagingStudy 3</li> <li>QuestionnaireResponse 3</li> <li>MolecularSequence 1</li> </ul>	<h3>Medications</h3> <ul style="list-style-type: none"> <li>MedicationRequest 3</li> <li>MedicationAdministration 2</li> <li>MedicationDispense 2</li> <li>MedicationStatement 3</li> <li>Medication 3</li> <li>MedicationKnowledge 0</li> <li>Immunization 3</li> <li>ImmunizationEvaluation 0</li> <li>ImmunizationRecommendation 1</li> </ul>	<h3>Care Provision</h3> <ul style="list-style-type: none"> <li>CarePlan 2</li> <li>CareTeam 2</li> <li>Goal 2</li> <li>ServiceRequest 2</li> <li>NutritionOrder 2</li> <li>VisionPrescription 2</li> <li>RiskAssessment 1</li> <li>RequestGroup 2</li> </ul>	<h3>Request &amp; Response</h3> <ul style="list-style-type: none"> <li>Communication 2</li> <li>CommunicationRequest 2</li> <li>DeviceRequest 1</li> <li>DeviceUseStatement 0</li> <li>GuidanceResponse 2</li> <li>SupplyRequest 1</li> <li>SupplyDelivery 1</li> </ul>

# RMIM

```
def add_participant(self, entity_name, role_type):  
    entity = Entity(entity_name)  
    role = Role(entity, role_type)  
    participation = Participation(role, self.encounter)  
    self.encounter.add_participation(participation)
```

# Openly Available FHIR Servers: HAPI FHIR

Home Server: HAPI Test Server (R4 FHIR) Source Code About This Server

Options

Encoding (default) XML JSON

Pretty (default) On Off

Summary (none) true text data count

Server

Server Home/Actions

HFQL / SQL

Resources

Observation 4336934

Patient 4013050

Specimen 1876165

Composition 940493

Bundle 405630

Encounter 258353

Binary 225136

Location 220229

HAPI FHIR



You are accessing the public FHIR server **HAPI Test Server (R4 FHIR)**. This server is hosted elsewhere on the internet but is being accessed using the HAPI client implementation.

**This is not a production server!** Do not store any information here that contains personal health information or any other confidential information. This server will be regularly purged and reloaded with fixed test data.

Server	HAPI FHIR Test/Demo Server R4 Endpoint
Software	HAPI FHIR Server - 8.5.3-SNAPSHOT/e3a3c5f741/2025-08-28
FHIR Base	<a href="http://hapi.fhir.org/baseR4">http://hapi.fhir.org/baseR4</a>

Server Actions

Retrieve the server's **conformance** statement.

Conformance

Retrieve the update **history** across all resource types on the server.

History Since  Limit #  (opt)

Post a bundle containing multiple resources to the server and store all resources within a single atomic transaction.

Transaction Bundle\* (place transaction bundle body here)

<https://hapi.fhir.org/baseR4>

← 3.4 Client Configuration

3.5 Client Examples ▾

Powered by HAPI FHIR v8.5.6-SNAPSHOT

Showing documentation for 8.6.0



## WELCOME TO HAPI FHIR

- Table of Contents 0.0
- Changelog: 2025 0.1
- Changelog: 2024 0.2
- Changelog: 2023 0.3
- Changelog: 2022 0.4
- Changelog: 2021 0.5
- Changelog: 2020 0.6
- Changelog: 2019 0.7
- Changelog: 2018 0.8
- Changelog: 2017 0.9
- Changelog: 2016 0.10
- Changelog: 2015 0.11
- Changelog: 2014 0.12

## GETTING STARTED

- Introduction 1.0
- FHIR and HAPI FHIR Versions 1.1
- HAPI FHIR Modules 1.2
- Downloading and Importing 1.3
- FHIR R4B Support 1.4

## WORKING WITH THE FHIR MODEL

- Working With Resources 2.0
- Parsing and Serializing 2.1
- Resource References 2.2

## 3.5.1 Client Examples

This page contains examples of how to use the client to perform complete tasks. If you have an example you could contribute, we'd love to hear from you!

## 3.5.2 Transaction With Conditional Create

The following example demonstrates a common scenario: How to create a new piece of data for a Patient (in this case, an Observation) where the identifier of the Patient is known, but the ID is not.

In this scenario, we want to look up the Patient record and reference it from the newly created Observation. In the event that no Patient record already exists with the given identifier, a new one will be created and the Observation will reference it. This is known in FHIR as a [Conditional Create](#).

### JSON:

```
{  
  "resourceType": "Bundle",  
  "type": "transaction",  
  "entry": [ {  
    "fullUrl": "urn:uuid:3bc44de3-069d-442d-829b-f3ef68cae371",  
    "resource": {  
      "resourceType": "Patient",  
      "identifier": [ {  
        "system": "http://acme.org/mrns",  
        "value": "12345"  
      } ],  
      "name": {  
        "given": "John",  
        "family": "Doe",  
        "prefix": "Dr.",  
        "suffix": "MD"  
      },  
      "telecom": [ {  
        "system": "phone",  
        "value": "+1 555-1234",  
        "use": "home"  
      } ]  
    }  
  }]
```

<https://hapifhir.io/hapi-fhir/docs/client/examples.html>

On this page:

[Client Examples](#)[Transaction With Conditional Create](#)[Fetch all Pages of a Bundle](#)[Create Composition and Generate Document](#)

# Hands On: Explore ClinFHIR & HAPI FHIR

### Search for Patient

Enter name for patient search

Enter Id of patient on this server

Enter Identifier of patient

Cancel

clinFHIR Launcher

For Implementers  For developers

Patient Viewer

Graph Builder 2

Add new patient

<https://fhir.s37vcloskatz.static-test-account.isccloud.io/>

If the server is protected by OAuth2 / SMART enter the Access Token

Bundle Visualizer

Various displays for the contents of a bundle. Bundles can be pasted into the viewer and optionally saved in the data server.

Server Query

Supports ad hoc queries against any FHIR server. Includes a simple query builder. The response can be displayed as Json or a Tree view, and FHIRPath is supported.

Implementation Guide Browser

Display the contents of an Implementation Guide, and the relationships between the contents of the Guide.

Scenario Builder

The Scenario Builder is used to join together the resources needed to represent a specific clinical scenario. It can use Core Resource types, Profiles and Logical models as it does this. The intention is to help people understand how resources can tell a clinical story, and to validate that the resource types available (including profiles) are sufficient.

Note that the builder still has issues with more complex resource types - this is a work in progress

Resource Validator

Validate a resource, or bundle of resources, by calling one or more validation servers.

<http://clinfhir.com/>

supported by  InterSystems®  
Creative data technology

Current servers

Data Server	InterSystems IRIS R4
Conformance Server	Public HAPI R4 server
Terminology Server	OntoserverR4 (terminology)

Add Server

FHIR Links (open in new tab)

R4 Specification (current)	<a href="#">Hay on FHIR</a>
STU-3 Specification	<a href="#">FHIR Chat</a>
STU-2 Specification	<a href="#">FHIR.org</a>
FHIR wiki	<a href="#">Clinicians Workshop</a>

clinFHIR Videos (open in new tab)

Scenario Builder	<a href="#">Other links</a>
Adding structured data	<a href="#">SNOMED browser</a>
Logical Modeler	<a href="#">SHRIMP (Terminology browser)</a>
Logical Modeler and Scenario Builder	
RESTful query tool	

*Note that some of these videos may describe earlier versions, so may not completely match the current functionality.*

Thanks to [InterSystems](#) for supporting the development of clinFHIR.

# FHIR Scope of Models: General to Specific

Scope: General Message Definitions to Implementable Message Specifications

- A single domain-wide model (reference model), e.g. HL7 v3/RIM, ISO 13606, FHIR Resource definitions
- Technology-independent specifications constraining the domain wide model, e.g. HL7 v3/RMIMs, CDA Templates, FHIR Profiles and archetypes.
- Implementable message specifications, mappings from technology-independent message specifications into the selected syntax, such as XML or JSON.

# Reference Information Model: General Scope

## HL7 v3/RIM (Reference Information Model):

- **RIM** is the core, abstract model used by HL7 v3. It provides a standardized, conceptual framework to represent healthcare data across different systems and domains.
- It defines the building blocks (like classes, attributes, and relationships) that are used to create healthcare-related messages, documents, and interactions.
- The RIM is highly abstract and not directly implementable. It serves as the foundation upon which more specific models and standards are built.
- Its purpose is to ensure consistency and a unified approach to handling healthcare information.

# Tying the RIM to the Storyboarding

Remember: The RIM contains abstract concepts.

Let's consider the following example of a patient encounter.

- **Entity:** A person (e.g., the patient or the healthcare provider).
- **Role:** The function of the entity in a particular context (e.g., "Patient" or "Doctor").
- **Act:** An action or event that happens (e.g., "Encounter", "Observation").
- **Participation:** The relationship between a role and an act (e.g., a patient participating in a doctor visit).

# Refined Message Information Model: Specific

- **RMIM** is a specialization or refinement of the RIM. It tailors the abstract concepts in the RIM to a particular use case or domain, such as lab results, patient records, or clinical encounters.
- RMIMs are more specific and closer to implementation. They define how data should be structured and exchanged in a given context based on the general principles of the RIM.
- Each RMIM refines the abstract RIM by selecting and constraining the appropriate classes, attributes, and relationships for the specific message or data exchange scenario.

# List out the objects at RIM level

## Entity (Person)

- Role: Patient
- Role: Healthcare Provider (Doctor)

## Act (Encounter)

- Participation: Patient (participates in Encounter)
- Participation: Doctor (participates in Encounter)

# List out the details at RMIM level

EncounterMessage (Message Header)

- Act: Encounter (type: Consultation)
- Participation: Patient (John Doe)
- Participation: Doctor (Dr. Smith)
- Observation: Blood Pressure (120/80 mmHg)

# Python code stub for RIM

```
# HL7 v3 RIM Abstract Classes
```

```
class Entity:
```

```
    def __init__(self, name):
```

```
        self.name = name
```

```
class Role:
```

```
    def __init__(self, entity, role_type):
```

```
        self.entity = entity
```

```
        self.role_type = role_type
```

```
class Act:
```

```
    def __init__(self, act_type):
```

```
        self.act_type = act_type
```

```
        self.participations = []
```

# RMIM

```
class EncounterMessage:  
  
    def __init__(self, encounter_type):  
  
        self.encounter = Act(encounter_type)  
  
        self.observations = []
```

# India on FHIR

MAY 23, 2022 • 8 MIN READ

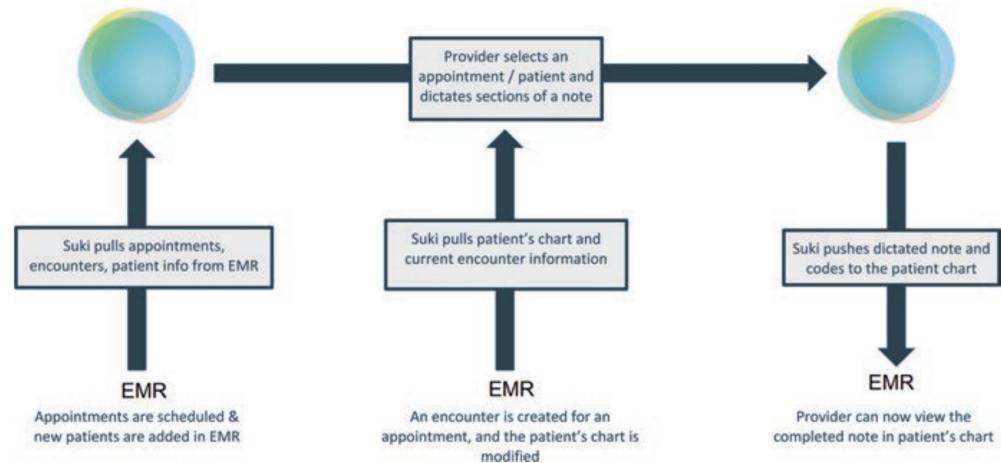
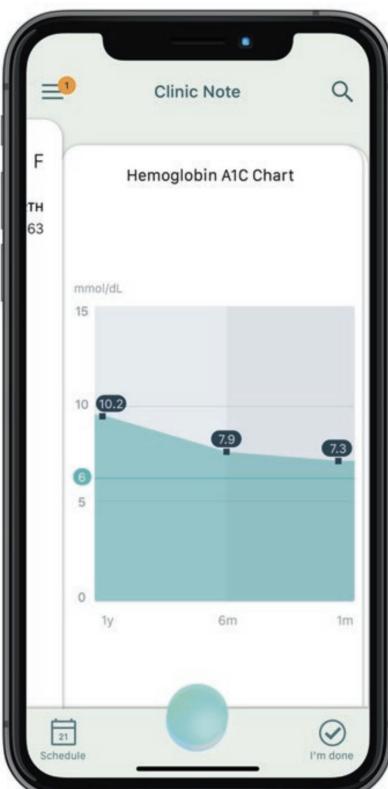
## Ayushman Bharat Digital Mission: Ushering in the UPI moment of healthcare



 **AYUSHMAN BHARAT**  
**DIGITAL MISSION**

Building digital health ecosystem<sup>+</sup>

# Suki, show me the medications of Mr. Sinha



# India's FHIR Stack (ABDM)



- ▶ Building blocks
- ▼ Overview of FHIR framework

## Architecture

- FHIR components & roles
- FHIR flows
- APIs and Standards
- ▶ Implementer's guide
- ▶ Preparation of data and packaging
- ▶ How do I participate?
- ▶ Postman Collection

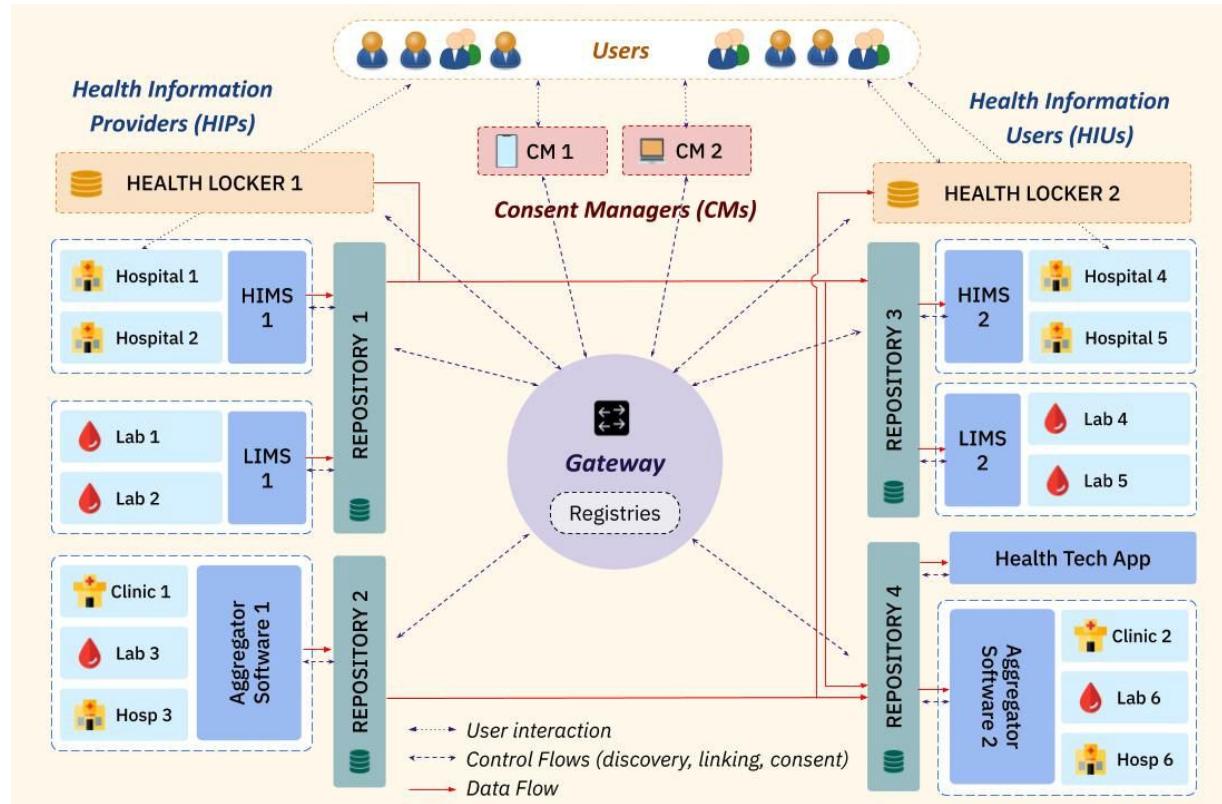
## ABDM digital health architecture

### Overview of architecture

The FHIR architecture is inline with the concepts described in the Niti Aayog National Health Stack (NHS) strategy and approach concept paper<sup>1</sup> published in July 2018.

As stated in the National Health Digital Mission (ABDM) operational strategy, the architecture is modelled as a Federated Architecture, where management and data access occurs in a 'federated' manner – where multiple entities will manage health data about users. The federated architecture will enable current and future health information flowing through the system, for example between providers and patients, wearables and EHR/EMRs, consumers and physicians, labs, institutions and payers. A centralized approach, where collation of data in National Repositories for democracy like ours with a 1.35 billion population will be prohibitively expensive and increase friction with the ecosystem, would be a single point of failure, with any security breach resulting in unimaginable compromise. A federated system allows data to sit at the source and be accessible on demand.

# India's FHIR Stack (ABDM)





# FHIR Implementation Guide for ABDM

CI Build v2.0.1



Home Profiles Terminology Examples Downloads



## FHIR Profiles for ABDM Health Data Interchange Specifications

### Table of Contents

This Implementation guide is published and maintained by NRCeS for NHA.

This version is based on the ABDM Health Data Interchange Specifications 1.0 and updated periodically.

Narrative Content

Detailed Descriptions

Mappings

XML

JSON

Turtle

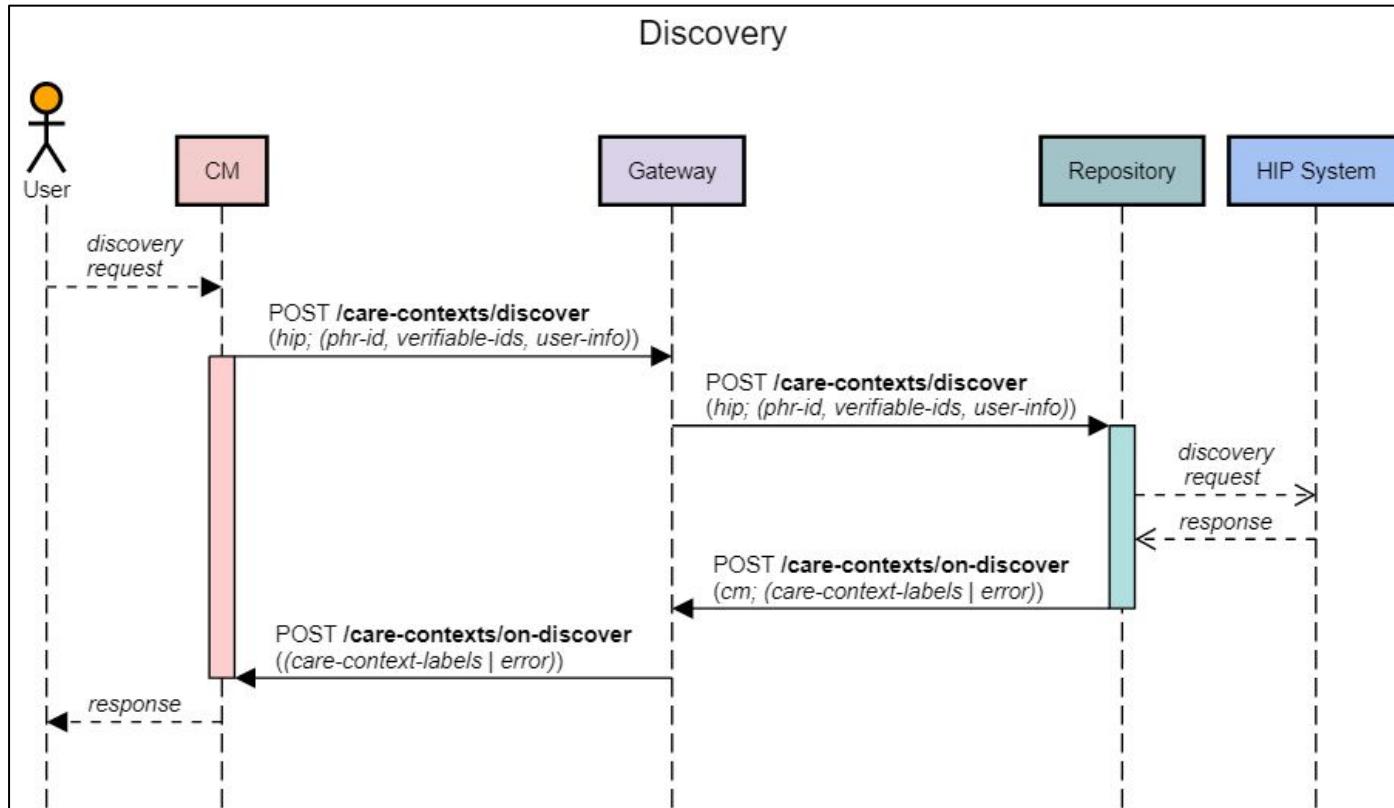
## Definitions for the StructureDefinition-PrescriptionRecord Profile.

### 1. Composition

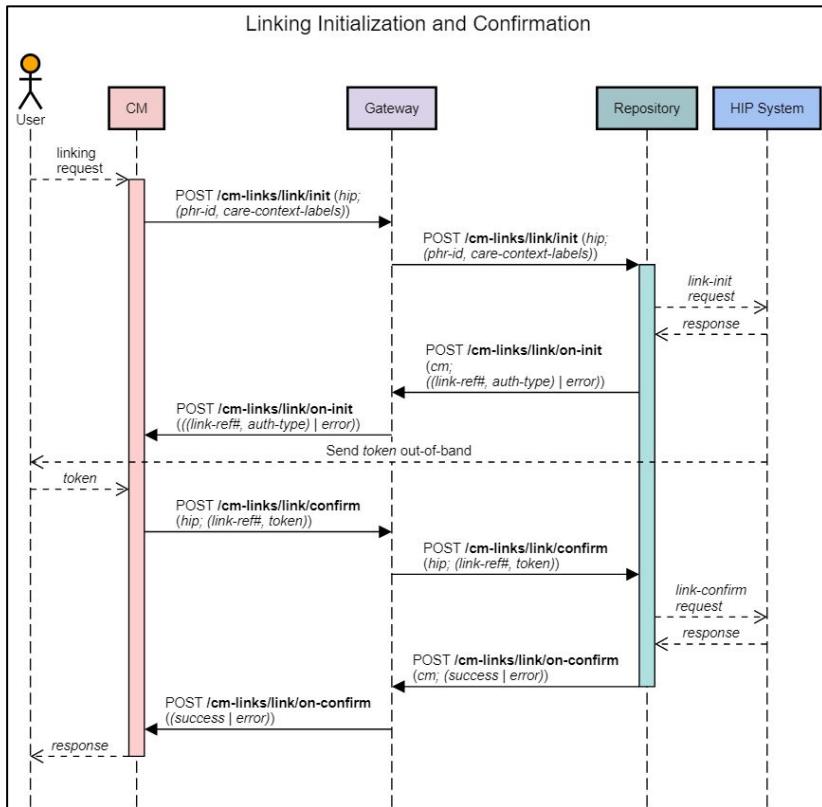
**Definition** A set of healthcare-related information that is assembled together into a single logical package that provides a single coherent statement of meaning, establishes its own context and that has clinical attestation with regard to who is making the statement. A Composition defines the structure and narrative content necessary for a document. However, a Composition alone does not constitute a document. Rather, the Composition must be the first entry in a Bundle where Bundle.type=document, and any other resources referenced from Composition must be included as subsequent entries in the Bundle (for example Patient, Practitioner, Encounter, etc.).

A set of healthcare-related information that is assembled together into a Composition to represent a physician's order for the preparation and administration of a drug or device for a patient. A Composition defines the structure, it does not actually contain the content: rather the full content of a document (for example Patient, Practitioner, Organization, MedicationRequest, etc.) is contained in a Bundle, of which the Composition is the first resource contained.

# Discovery of Patient Information



# Linking to Hospital (HIP)



# Health ID and Consent Management Framework

## Health data consent manager (HDCM)

HDCM plays the role of fiduciary or trustee with which a patient signs up to begin with. The HDCM does the following:

- Helps to create or discover existing ABDM ABHA Number
- Manages patient linkages to providers
- Manages consent for access to health information
- Helps discover patient information
- Monitors information exchange between HIP and HIU

All the above are done via a Patient App. In future, there will be other channels for patients to manage their health data.

## Health information gateway

Gateway is the hub that mediates and connects Consent Manager(s), Health Repository Providers and HIUs in the network. Its primary job is to allow for discovery, routing in the network. The gateway does the following:

- Connects and validates the HDCMs and health repositories (servicing HIUs and HIPs) to the network.
- Enables routing of information.
- Authenticates connected systems within the FHR Framework (provides a signed authentication token for FHR framework including the gateway communicates via asynchronous APIs over HTTPs channel.

## Registries

There are various National level Registries that are integrated with the network.

- Health Facility registry (HFR)
- Doctor or practitioner registry (DigiDoctor)
- HDCM and Health Repository Provider (HPR) Registry

The HDCM and HRP registries are currently maintained by the ABDM Gateway.

## Health repository

A health repository is the connector or bridge through which any HIU or HIP can connect to the network. The health repository allows HIPs and HIUs to access patient data for a specific period of time. The sandbox environment provides a basic HIU system, representing a doctor's interface. The sandbox environment also provides a mock HIP system using which you can test linkages and access to health information from multiple sources.

# Health ID and Consent Management Framework

## Registering as a patient using the Patient App

The following sections explain how a patient can register via consent manager app. Once you've registered as a patient, you will be able to search hospitals and choose to link your patient care contexts for the selected one. This will enable access to medical records for a HIU or for the patient herself.

### Mobile number verification

To register via the patient app, the first step is to register using a digitally verifiable identifier. As of now, Aadhaar and mobile numbers are such identifiers. In future, more digital identifiers would be supported.

Perform the following tasks to begin your registration process on the Consent Manager app:

1. Install the consent manager application on your phone.  
[Download .apk file ↴](#).
2. Open the Consent Manager app and click on the **Register** button to register as a patient.
3. You will be asked to enter your mobile phone number, in order to verify your identity. Make sure that you enter the mobile number which you generally provide at the hospitals that you visit.
4. Enter your mobile number. Take care to enter valid phone number.
5. Click the **Verify** button to authenticate your mobile number.
6. You will receive an SMS from the ABDM Consent Manager, along with an OTP (One Time Password).
7. Enter the OTP in the Consent Manager app.
8. If your OTP is validated, you will be directed to a Registration screen. If your OTP is not validated, you will be prompted to enter a valid OTP

## HIP services

The following responsibilities are expected to be carried out by HIP services in the FHIR Framework context:

- Ability to discover and link patient's care contexts within an organization's system.
- Allow for consented means of exchange of patient's health information in a machine readable format (FHIR), conforming to standards setup by ABDM.
- Provide a secure and safe means of data transfer from organization-specific HIS (Health Information System) instance to a validated and consented requester HIU, through the ABDM Gateway.

## HIU services

The following capabilities are expected of HIU services in the FHR context:

- Ability to search for and identify a patient using her ABHA Number. HIU systems can search for patients linked within the ABDM Gateway network, and across diverse Gateway networks in future.
- Ability to request and receive patient data in a safe and secure manner, manage data lifecycle, and enable secure data storage and access.

## Health locker services

Health locker provides a personal digital storage for users/patients. The following capabilities are expected of Health locker services in the FHR context:

- Sign up users/patients in their system.
- Allow users to upload their health documents to their locker.
- Ability to fetch users health documents from the HIPs based on patient consent.
- Ability to Provide Health Documents, both self uploaded and fetched from HIPs, to a valid requestor.
- Allow user/patient access to their health documents (through their locker apps or other channels).
- Provide secure means for health document access and storage.

### What is a Patient App?

While building and testing your health repository in integration with the sandbox's network components, you'll need to test patient flows as well. For example:

- Signup of patient with Consent Manager (CM)
- Linking patient's CM account with HIP's patient reference and health care contexts
- Granting or revoking consents

You can do all of the above using APIs as well, but the easiest way would be to use the reference patient Android App. [Download .apk file ↗](#).



ABHA Number Service

Health Facility Registry

DigiDoctor

Consent Manager and Gateway

ABHA Mobile Application

HIU Application

HIP Application

UHI Apps

HCX Integrators

▼ Overview of FHR framework

Architecture

FHR components & roles

FHR flows

APIs and Standards

▼ Implementer's guide

What is already available

What resources can I use

ABDM Sandbox integration & exit process

Building HIP

Facility share profile

Building HIU

Building Health Locker

Building ABHA Mobile Application

▼ Building UHI Applications

UHI Protocol

Terminology

UHI Architecture Overview

Registering your UHI App

Sandbox Gateway, EUA & HSPA

## Building UHI EUA App

This details out the steps to build your own EUA app including

- Sign up / Sign in with PHR Address
- Select a UHI service
  - Teleconsultations
    - Search HSPAs for Doctor
    - Display search results
    - Book selected Doctor/Facility
    - Collect Payment if required
    - Confirm Booking
    - Exchange Messages with Doctor
    - Share health records
    - Setup WebRTC (teleconsults)
    - Initiate Call (teleconsults)
    - Get final prescription
  - Appointment booking
    - Search HSPAs for Facility/Doctor
    - Display search results
    - Book selected Facility/Doctor
    - Collect Payment if required
    - Confirm Booking
  - Ambulance booking
    - Search HSPAs for Facility/Doctor
    - Display search results
    - Book selected Facility/Doctor
    - Collect Payment if required
    - Confirm Booking

[Link for UHI APIs](#)

# Example Apps

Citizen/Patient Services
Single, Secure Health id to all citizens
Personal Health Record
Singel (National) Health Portal
App Store
Specialized Services for Remote Areas/ Disadvantaged Groups
NDHM Call Centre
Digital Referrals & Consultations
Online Appointments
e-Prescription Service
Digital Child Health
National "Opt-out" (for privacy)
Services by / for Healthcare Providers/ Professionals
Summary Care Record
Open Platform to access Emergency Services
Technology for Practitioner (GP) Transformation
Digital Referrals, Case Transfers
Clinical Decision Support (CDS)
Digital Pharmacy & pharmacy Supply Chain
Hospital Digitization (HIS)
Digital Diagnostics
Technical Services
Architecture & Interoperability
Health Information Exchange
Standards
Health Network
Data & Cyber Security
Information Governance

The image displays three screenshots of a mobile health application interface, likely for the National Digital Health Mission (NDHM). The interface is designed for both iOS and Android devices, as indicated by the different icons at the bottom.

- Left Screenshot:** Shows the main menu with sections like Citizen/Patient Services, Services by / for Healthcare Providers/ Professionals, and Technical Services.
- Middle Screenshot:** Shows the "My Health Records" screen with a "LINK RECORD" button. It lists immunizations and prescriptions from Max Health Care on 5th Aug, 2019. A detailed view of an immunization record for Tetanus Toxoid is shown, including lot number, route, reason, occurrence date, dose number, and manufacturer.
- Right Screenshot:** Shows the "Discharge Summary - Bacillus..." screen. It details an inpatient visit for Max Health Care on 01 Oct 2019, diagnosed with Bacterial infection due to Bacillus. It includes sections for Presenting Problems (Vomiting, Brief loss of consciousness, Swinging fever), Prescribed medications during Admission, Procedures Performed, and Diagnostic reports.



- ▶ Building blocks
- ▶ Overview of FHIR framework
- ▶ Implementer's guide
- ▼ Preparation of data and packaging
  - The main envelope
  - Diagnostic reports as FHIR
  - DiagnosticReport
- Data encryption and decryption**
  - How to start testing the health repositories
- ▶ How do I participate?
- ▶ Postman Collection

## Data encryption and decryption as requester HIU

The following abbreviations are used in this section:

- ECDH: Elliptic-curve Diffie–Hellman Key Exchange
- AES–GCM: Advanced Encryption Standard–Galois/Counter Mode
- DHPK: Elliptic-curve Diffie–Hellman public key
- DHSK: Elliptic-curve Diffie–Hellman secret/private key
- P and U is annotation for system
- DHK(U,P): Elliptic-curve Diffie–Hellman Key
- Rand: Random String

Information shared as part of the data flow will be secured using an encryption mechanism that ensures perfect forward secrecy. This means that even if any of the key materials stored at HIPs, HIUs or HDCM clients (either long-term private keys or session keys) are compromised, it would not be possible to decipher data that was previously exchanged. The encryption mechanism uses Elliptic-curve Diffie–Hellman Key Exchange (ECDH), which is used in many Internet protocols such as SSH and TLS for establishing shared secret keys between remote parties.

### Data encryption for HDCM client

The following points detail the process behind data encryption for HDCM client:

1. When creating a data request, the HIU does the following:
  - a. Creates a set of Elliptic-curve Diffie–Hellman (ECDH) parameters
  - b. Generates a **ECDH key pair ( $dhsk(U)$ ,  $dhpk(U)$ )** (which is a short-term public-private key pair)
  - c. Generates a **32-byte random value**,  $rand(U)$  which is also called **nonce**.
2. The HIU sends these values to HDCM, along with the data request via a digitally-signed API call.
3. HDCM forwards the request to the HIP, again via a digitally-signed API call.
4. HIP checks whether the consent artefact is valid, and whether the data being requested is in keeping with the terms of the artefact. If the artefact is valid, the HIP does the following:
  - Generates a fresh ECDH public-private key pair in the same group as specified by the HIU **(( $dhsk(P)$ ,  $dhpk(P)$ )**

AUG 2, 2022 • 9 MIN READ

## Case Study - Implementing ABDM for the Government of Andhra Pradesh



IMPLEMENTING  
**ABDM**  
FOR THE ANDHRA PRADESH GOVERNMENT

CASE STUDY

The Ministry of Health and Welfare, Andhra Pradesh approached us to help with implementing all the 3 milestones for [Ayushman Bharat Digital Mission](#). Most of the ABDM APIs were built on top of FHIR, and they found out about Medblocks through our content.

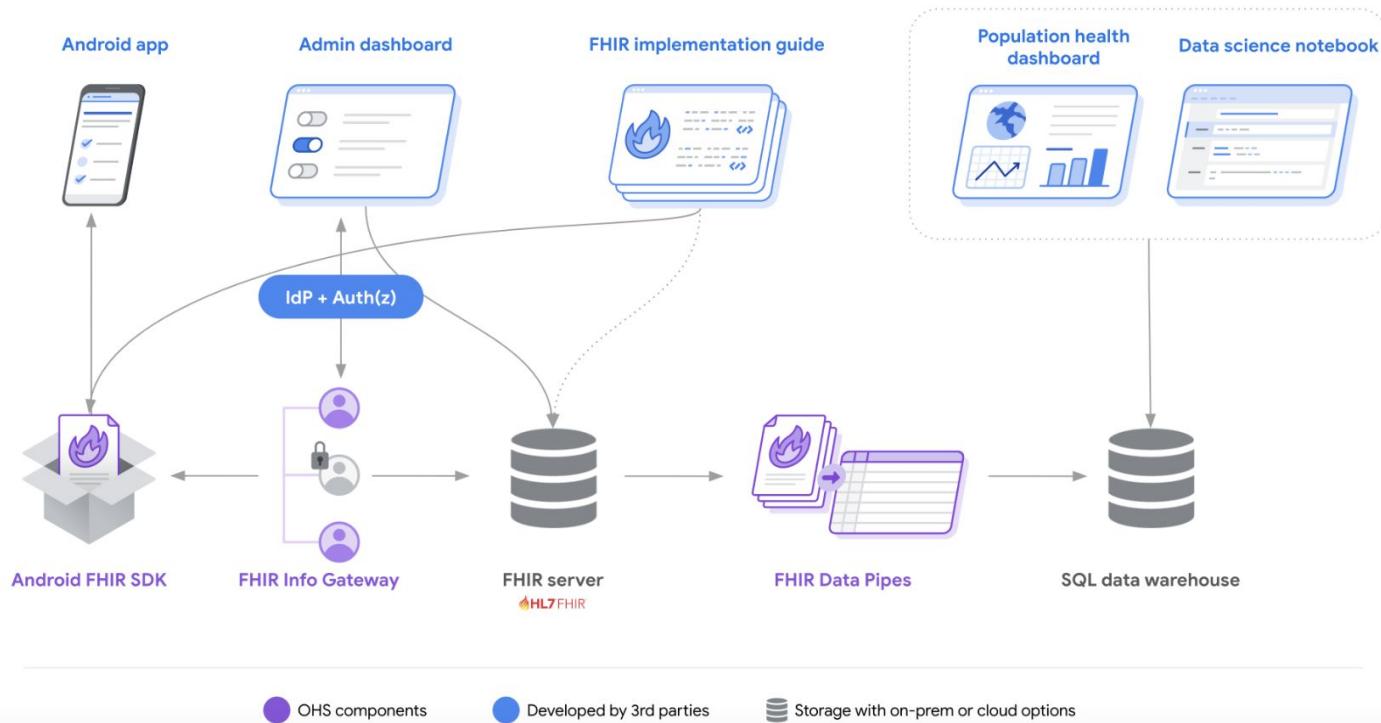
<https://medblocks.org/abdm-for-the-ap-government-a-honest-review/>

# Health ID



Three screenshots illustrating the Ayushman Bharat Digital Mission platform. The top screenshot shows a web browser window titled 'ABHA number' with fields for 'ABHA Address' (n123@bx), 'Authentication Mode' (Mobile OTP), and 'OTP' (279224). The middle screenshot shows a 'Recent Encounters' list for a user named Nishtha Mahajan, listing several encounters from 13 minutes ago to 4 weeks ago. The bottom screenshot shows a mobile application interface with tabs for 'My Records', 'Linked Facility', 'Consents', and 'Nearby'. It displays a patient ID (n123@bx) and a recent encounter record for 'ENC-00035 2022-5-14 - OFD'.

# Google on FHIR: Open OHS End to End FHIR



# Real World Implementations of FHIR Resources

# How UK is Building PHRs on FHIR

The screenshot shows a website header with a blue bar. On the left, there's a house icon followed by the text "Care Connect API | FHIR®". To the right of the header are navigation links: "Nav", "Care Connect", "FHIR", and "Feedback". A dropdown menu is open under the "Care Connect" link, listing three items: "HL7 UK Care Connect Profiles", "NHS Digital Profiles", and "Care Connect Reference Implementation". Below the header, on the left, is the INTEROPen logo (a teal circle with a white stylized figure) and the text "INTEROPen". To the right of the logo is the word "Build | FHIR". At the bottom of the page, there's a red vertical bar with the text "Summary: How to create a basic FHIR Patient server using open source tools".

Supported by **NHS**  
Digital

Build | FHIR

Summary: How to create a basic FHIR Patient server using open source tools

- HL7 UK Care Connect Profiles
- NHS Digital Profiles
- Care Connect Reference Implementation

[Overview](#)[Repositories 198](#)[Projects](#)[Packages](#)[People 4](#) Find a repository...[Type](#) [Language](#) [Sort](#) 

## STU3-FHIR-Assets

[Public](#)

Repository for development of NHS Digital STU3 FHIR assets.



10

8

1

0

Updated 19 minutes ago

## prm-repo-pds-adaptor

[Public](#)

The PDS-Adaptor is a web service that will communicate with the PDS using the FHIR API to maintain patient details.



0

0

0

0

Updated 13 hours ago

[nhsconnect / careconnect-examples](#)

Public

Notifications

[Code](#)[Issues](#)[Pull requests](#)

3

[Actions](#)[Projects](#)[Security](#)[Insights](#)[master](#) ▾

4 branches

0 tags

[Go to file](#)[Code](#) ▾

KevinMayfield Merge remote-tracking branch 'origin/master' ...

e82bddb on 25 Jun 2019 273 commits

[UHSDiagnosticReports](#)

Github warning

3 years ago

[ccri-edms-server](#)

Copied over EDMS (alfresco) example from ccri-document

4 years ago

[ccri-eolc](#)

Changes for DWP

3 years ago

[ccri-interopen-examples](#)

Combined all doc uploads into one module

3 years ago

# Step 1: Building a FHIR Server

If you followed the pre-requisites and whether you have chosen IntelliJ or Eclipse, you should have a basic FHIR server. To test and confirm, start POSTMan and enter this URL

GET http://127.0.0.1:8183/STU3/metadata

You should see a FHIR **ConformanceStatement** returned from the server.

The screenshot shows the Postman application window. The top navigation bar includes File, Edit, View, Help, New, Import, Runner, and a workspace dropdown set to "My Workspace". The status bar indicates "IN SYNC". The left sidebar shows a "History" tab with three collections: "Mitre" (3 requests), "Postman Ec..." (37 requests), and "FHIR Message" (6 requests). The main workspace displays a GET request to "http://127.0.0.1:8183/STU3/metadata". The request details panel shows the method as "GET", the URL as "http://127.0.0.1:8183/STU3/metadata", and a "Send" button. Below the URL, the "Authorization" tab is selected, showing "Inherit auth from parent". A note states: "The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)". The right side of the workspace shows a message: "This request is not inheriting any authorization helper at the moment. Save it in a collection to use the parent's authorization helper."

[https://nhsconnect.github.io/CareConnectAPI/build\\_patient\\_server.html](https://nhsconnect.github.io/CareConnectAPI/build_patient_server.html)

# Step 2: Adding a Patient

The screenshot shows the Postman application interface. At the top, there is a header bar with 'POST' dropdown, URL 'http://127.0.0.1:8183/STU3/Patient', 'Params' button, 'Send' button, and 'Save' button. Below the header, there are tabs for 'Authorization', 'Headers (1)', 'Body' (which is selected and highlighted in orange), 'Pre-request Script', and 'Tests'. On the far right, there are 'Cookies' and 'Code' buttons. Under the 'Body' tab, there are four options: 'form-data', 'x-www-form-urlencoded', 'raw' (which is selected and highlighted in orange), and 'binary'. Below these options is a dropdown menu set to 'Text'. The main body area contains an XML payload for a patient resource, numbered from 1 to 26. The XML includes meta-information like last updated and profile, and extensions for ethnic category and NHS number verification status.

```
1 <Patient xmlns="http://hl7.org/fhir">
2   <id value="1"/>
3   <meta>
4     <lastUpdated value="2018-04-03T13:59:40.080+00:00"/>
5     <profile value="https://fhir.hl7.org.uk/STU3/StructureDefinition/CareConnect-Patient-1"/>
6   </meta>
7   <extension url="https://fhir.hl7.org.uk/STU3/StructureDefinition/Extension-CareConnect-EthnicCategory-1">
8     <valueCodeableConcept>
9       <coding>
10         <system value="https://fhir.hl7.org.uk/STU3/CodeSystem/CareConnect-EthnicCategory-1"/>
11         <code value="A"/>
12         <display value="British, Mixed British"/>
13       </coding>
14     </valueCodeableConcept>
15   </extension>
16   <identifier>
17     <extension url="https://fhir.hl7.org.uk/STU3/StructureDefinition/Extension-CareConnect-NHSNumberVerificationStatus-1">
18       <valueCodeableConcept>
19         <coding>
20           <system value="https://fhir.hl7.org.uk/STU3/CodeSystem/CareConnect-NHSNumberVerificationStatus-1"/>
21           <code value="01"/>
22           <display value="Number present and verified"/>
23         </coding>
24       </valueCodeableConcept>
25     </extension>
26   <system value="https://fhir.nhs.uk/Td/nhs-number"/>
```

# Storing in a Database

The screenshot shows the MongoDB Compass interface connected to a standalone MongoDB instance at localhost:27017. The database is 'ccri-fhirStarter' and the collection is 'idxPatient'. There is one document in the collection.

**Document Details:**

```
_id: ObjectId("5ac47fd723598f6af80ff1fe")
_class: "uk.nhs.careconnect.fhirstarter.entities.PatientEntity"
dateOfBirth: 1998-03-13 00:00:00.000
gender: "FEMALE"
identifiers: Array
  0: Object
    system: "https://fhir.nhs.uk/Id/nhs-number"
    value: "9876543210"
  1: Object
    system: "https://fhir.leedsth.nhs.uk/Id/pas-number"
    value: "ABC8650149"
  2: Object
    system: "https://fhir.leedsth.nhs.uk/Id/PPMIIdentifier"
    value: "1"
telecoms: Array
names: Array
  0: Object
addresses: Array
  0: Object
    city: "Nottingham"
    county: "Derbyshire"
    use: "WORK"
    postcode: "NG10 1ZZ"
    lines: Array
      0: "Field Jardin"
      1: "Long Eaton"
```

# What is SMART on FHIR

**SMART on FHIR** stands for **Substitutable Medical Applications, Reusable Technologies on Fast Healthcare Interoperability Resources**.

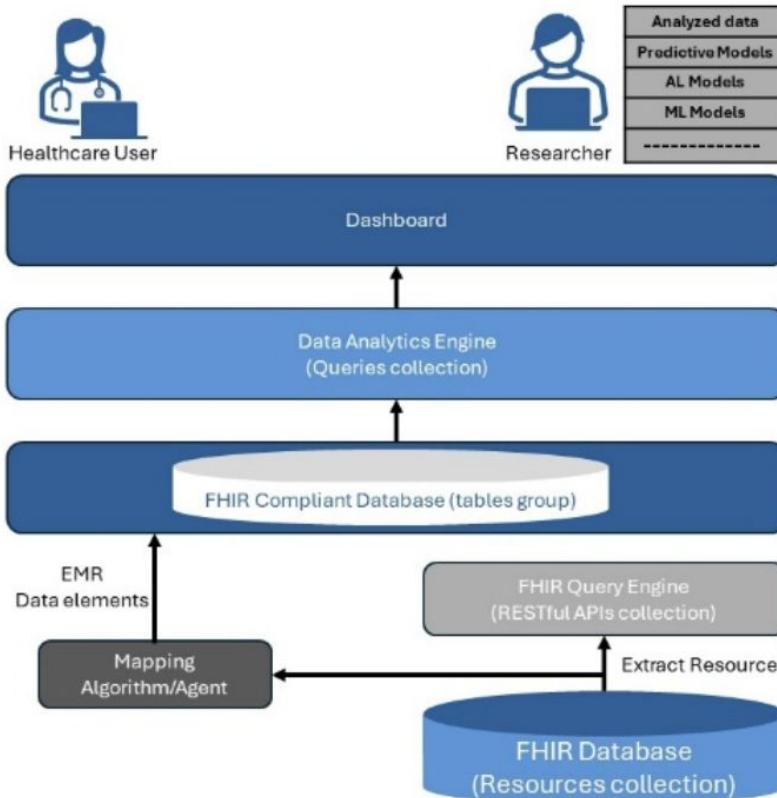
It's a **framework that allows third-party apps to connect securely to electronic health record (EHR) systems** — like Epic, Cerner, or Allscripts — in a standardized, interoperable way.

Essentially, it's the “**app store for healthcare data**”:

- SMART defines **how apps launch and authenticate** within an EHR.
- FHIR defines **how data are structured and exchanged**.

Together, they make it possible for apps to **access patient data in a standardized format**, securely and consistently across different health systems.

# FHIR Analytics and How US is Using It

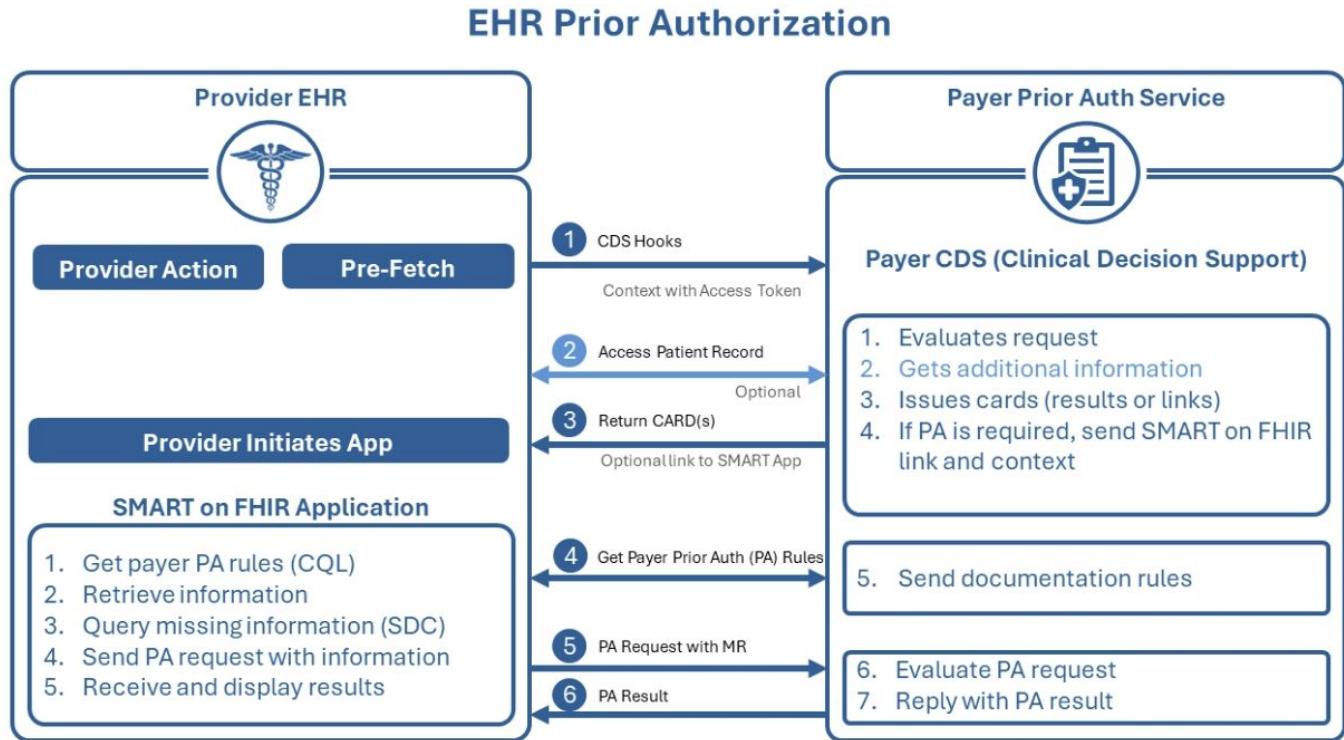


# What SMART Does

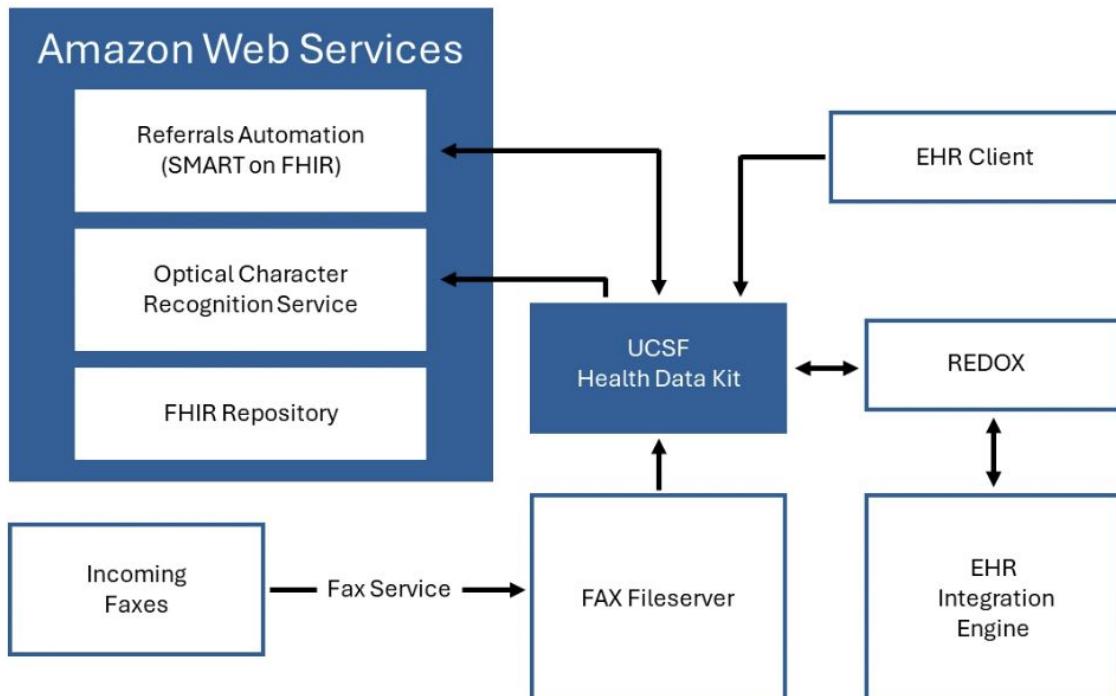
- Adds an authorization and app-launch layer on top of FHIR.
- Defines how apps are installed, authenticated, and granted permissions to read or write data.
- Uses OAuth 2.0 and OpenID Connect for secure user login and token-based data access.
- Apple Health uses SMART on FHIR to pull health data from hospitals
- Epic App Orchard, Cerner CODE, and Allscripts Developer Program all rely on SMART on FHIR.
- CMS Interoperability and Prior Authorization Rule (CMS-0057-F) (effective 2026–2027) also builds upon FHIR APIs for payer-provider data exchange.

# Prior Authorization Reform in US CMS (MEDICAID)

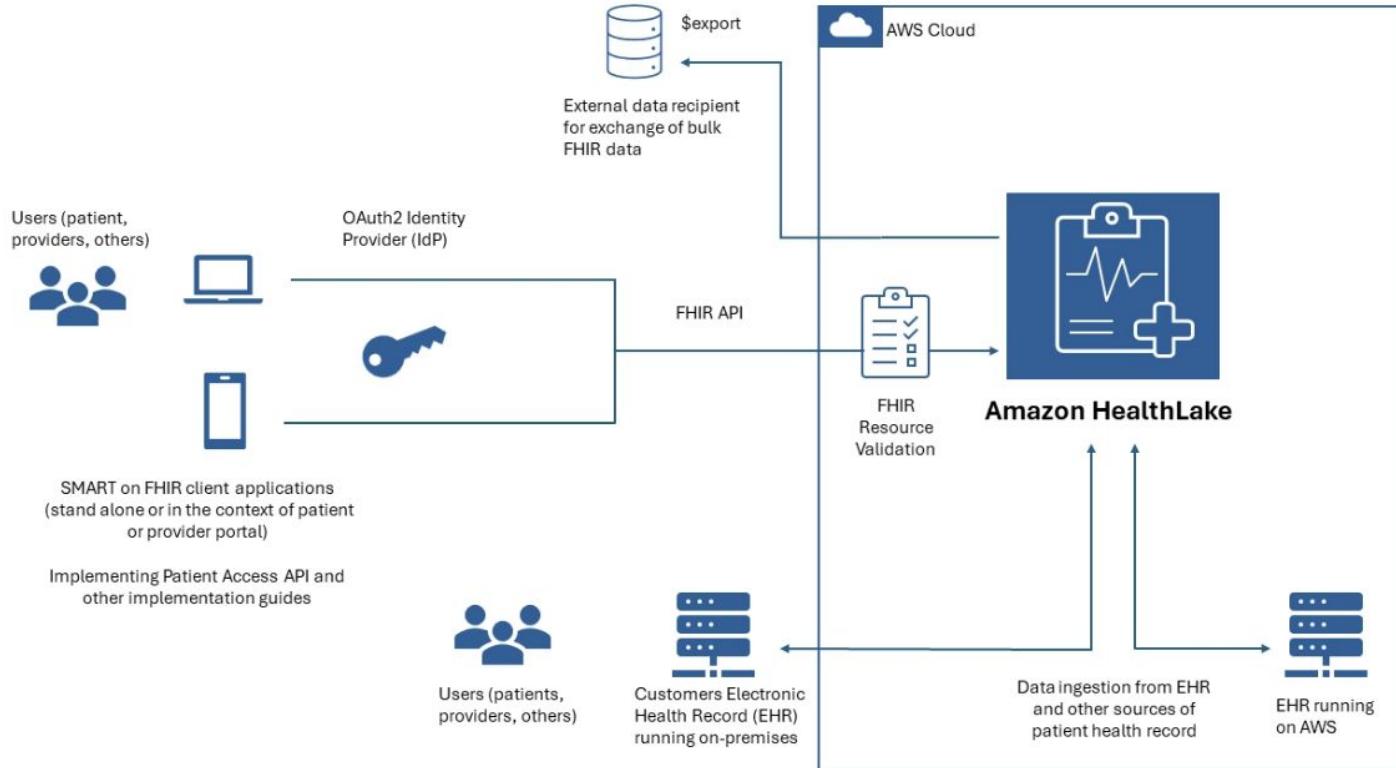
Starting in 2026, affected payers will be required to issue prior authorization decisions within **72 hours for urgent requests** and **seven calendar days for non-urgent requests**, thereby accelerating patient care.



# UCSF Uses AWS APIs for FHIR

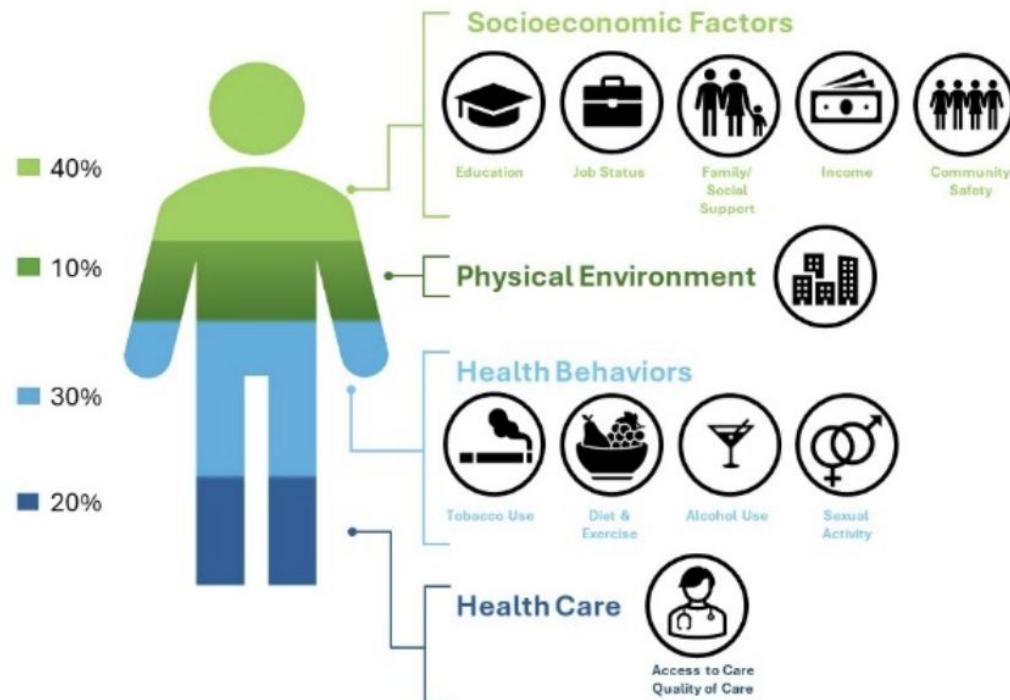


It supports functions such as conditional creation, deletion, update of resources, and recovery of deleted files. Additional capabilities include secure processing of clinical data in compliance with HIPAA, data analytics based on artificial intelligence, and bulk access to FHIR-enabled servers with role-based access control privileges.

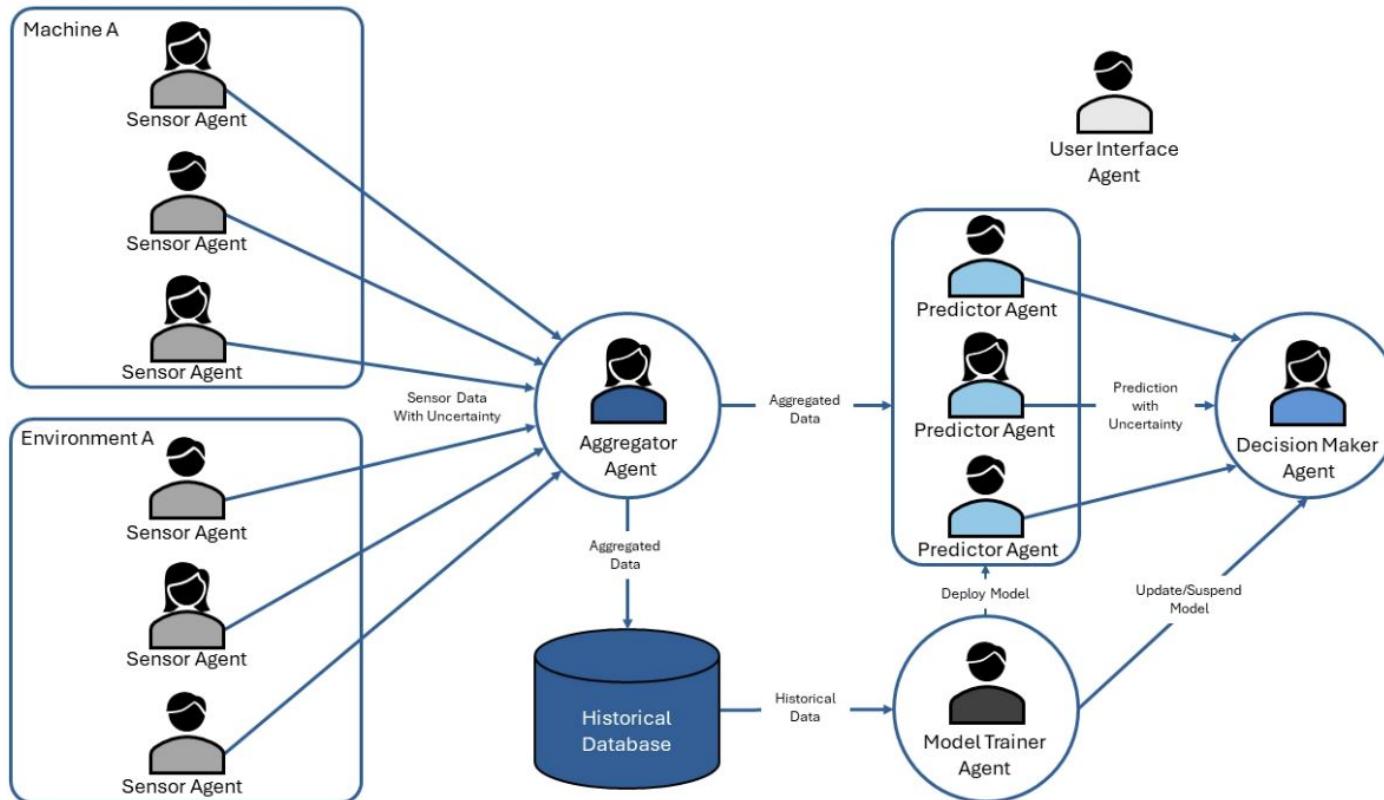


# FHIR allows you to integrate Determinants of Health

## What Goes Into Your Health?



# FHIR and Agentic AI





**Thanks for  
attending the class!**