FHIR Server

Integrating it in your Architecture

Christiaan Knaap

01 September 2021

Stop contemplating

Start FHIRing

The Question

I want to integrate some systems.

Where does this FHIR Server go?

Me

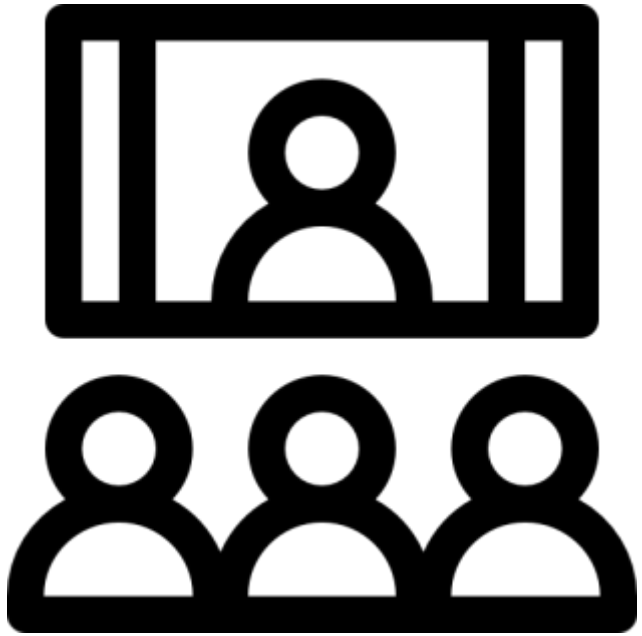
- Christiaan Knaap
- Firely
- 20 yr IT dev / analist / architect
- Lead dev of Firely Server
- christiaan@fire.ly
- chat.fhir.org

You

What are you?

A. Dev [ops]
B. Architect
C. Data scientist
D. Manager

(write in the chat)



FHIR Experience?

1. Do you mean fire?
2. I read about it
3. Experimented
4. Real world implementation

The answer you hope to get?

Is there a case or question that brought you here?

(Put a 1-liner in the chat)

Firely FHIR Product Suite

https://fire.ly/products/

FHIR Data Modeling

- Simplifier.net
- Forge
- Firely Terminal

FHIR Implementation

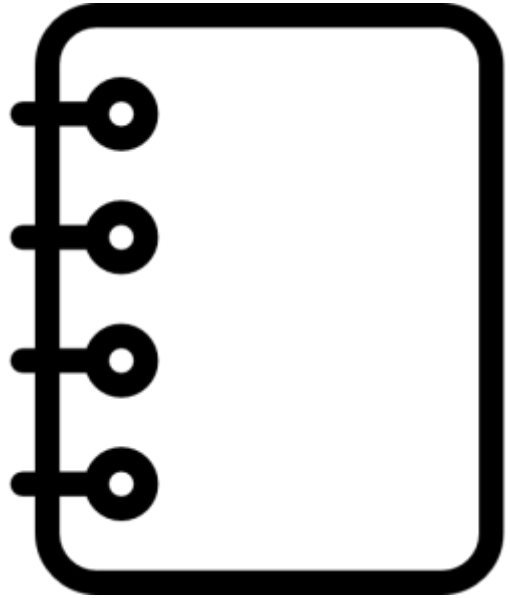
- Firely Server
- Firely Facade

FHIR App Development

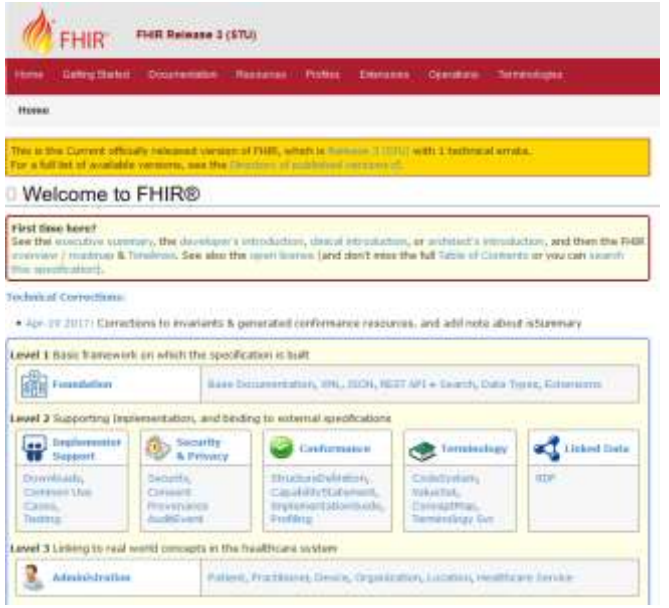
- Firely .NET SDK

Agenda

- What is a FHIR Server?
- Use cases
- Architectures
- Use cases & architectures
- Questions (and maybe answers)



FHIR Server in the specification



The OperationOutcome may be returned with any HTTP 4xx or 5xx response, but this is not required - many of these errors may be generated by generic server frameworks underlying a FHIR server.
(HTTP Status Codes)

When processing create and update interactions, a FHIR server is not obliged to accept the entire resource as it is
(Transactional Integrity)

FHIR does not (yet) define a root document. When defined, it will contain information about what the FHIR server has done (as opposed to a Capability Statement, which describes what it is capable of doing)
(OMG hData RESTful Transport)

From <http://www.hl7.org/implement/standards/fhir/http.html>

FHIR Servers do not have to support versioning, though they are strongly encouraged to do so.

From <http://www.hl7.org/implement/standards/fhir/overview-dev.html>

A FHIR REST server is any software that implements the FHIR APIs and uses FHIR resources to exchange data.

From <http://www.hl7.org/implement/standards/fhir/overview-arch.html>

FHIR Server definition

Any software
that implements
all or several
parts of the
FHIR RESTful API

Instance Level Interactions	
read	Read the current state of the resource
vread	Read the state of a specific version of the resource
update	Update an existing resource by its id (or create it if it is new)
patch	Update an existing resource by posting a set of changes to it
delete	Delete a resource
history	Retrieve the change history for a particular resource
Type Level Interactions	
create	Create a new resource with a server assigned id
search	Search the resource type based on some filter criteria
history	Retrieve the change history for a particular resource type
Whole System Interactions	
capabilities	Get a capability statement for the system
batch/transaction	Update, create or delete a set of resources in a single interaction
history	Retrieve the change history for all resources
search	Search across all resource types based on some filter criteria

FHIR Server purpose

- FHIR Servers should handle the hard parts of FHIR
  so that
- FHIR Clients are easy to create and use.

Functions in the API

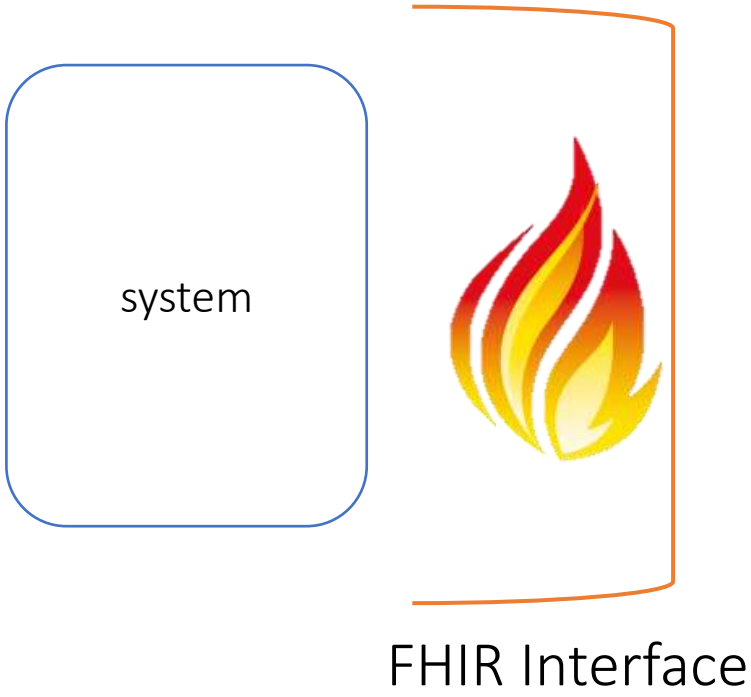
- Capabilities
- Store resources (crud)
- Search
- History / versioning
- Validation
- Format support (xml, json, turtle)
- Transactions / batches
- Custom operations
- Across FHIR versions

Generic FHIR Servers

- Support most of the RESTful API
- For all types of resources
- With storage of their own

Specific FHIR Servers

- Implement some parts of the API
- Often bound to a backend system
  - e.g. an EHR
- Can often be seen as a ‘FHIR Facade’ to a system
- Domain specific operations
  - e.g. ‘get appointment slots’



Some examples

- External data reporting
- Portal support
- App platform
- Clinical Data Repository
- Systems integration

Factors

1

Use cases
Read or also write?
Requests: # vs size
Resource types and parameters
Special operations
Specific IG

2

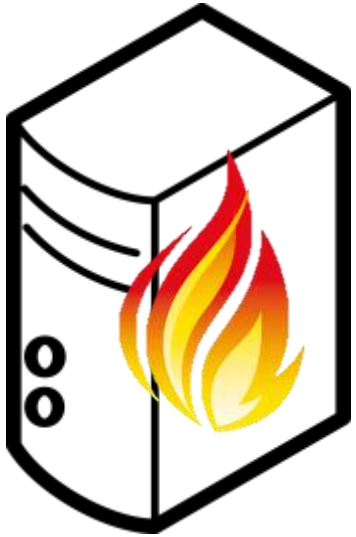
Authz
SMART on FHIR (v1/v2)
Custom
Integration

3

Existing data
Load / sync
Timeliness
Conversion
Size

4

Hosting
Cloud / on-premise
Single party / network
Constant / peak loads



Architecture

source — source of data other systems are interested in

consumer/target of data from 1 or more sources — consumer

query transform — thin arrow: data request

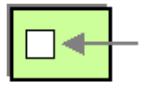
data response / data push — data transform

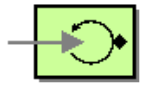
FHIR Facade

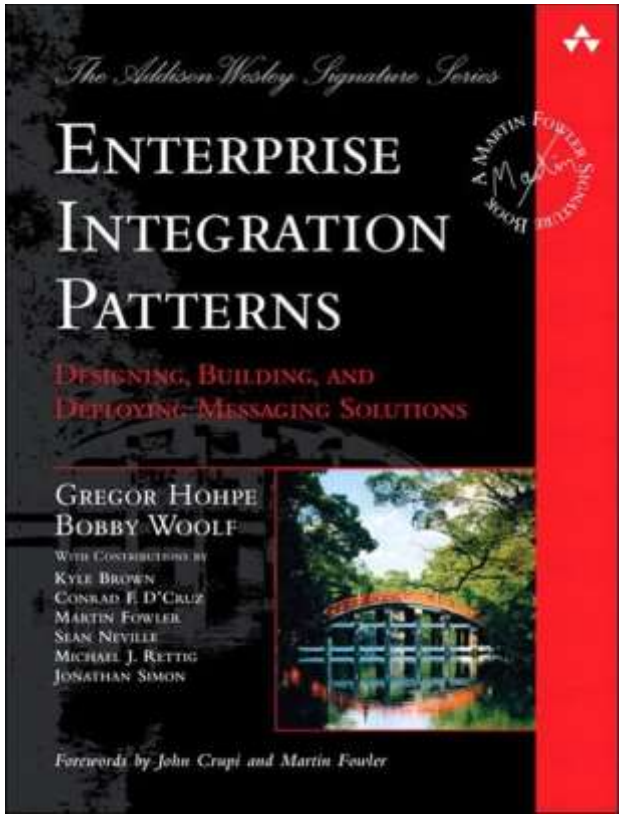
FHIR Server

Integration patterns

- Canonical Data Model
- Scatter-gather, Aggregator
- Event Driven Consumer
- Polling consumer
- Channel Adapter
- Message Translator
- Messaging Mapper
- Messaging Gateway
- Publish Subscribe

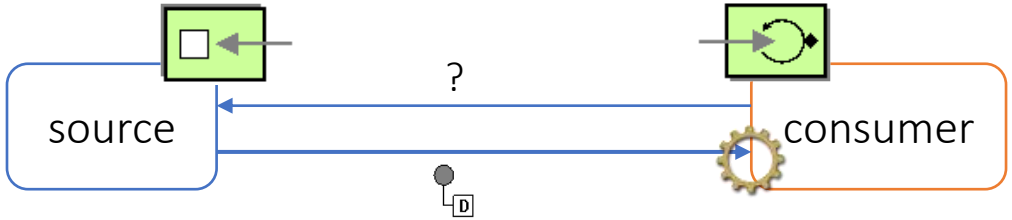






Start simple

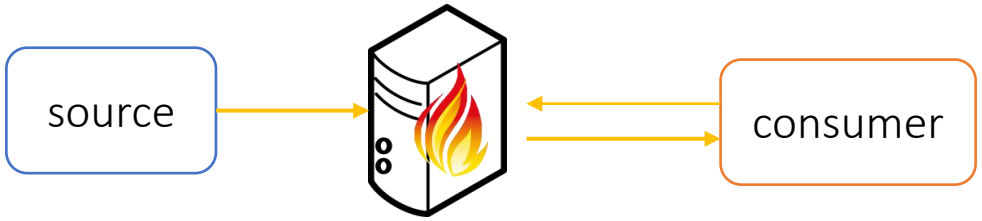
- Single consumer, needs that data
  - Channel Adapter
  - Polling consumer
  - Message Mapper
- Single source of data
  - Message Endpoint
- You can do without FHIR
- But it works as the simplest example...



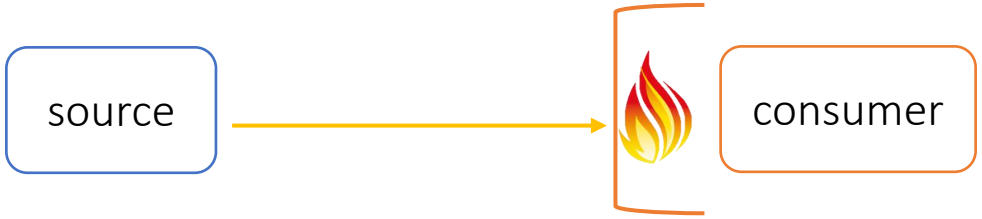
Where to put the FHIR Server?

Facade on source

Generic FHIR Server

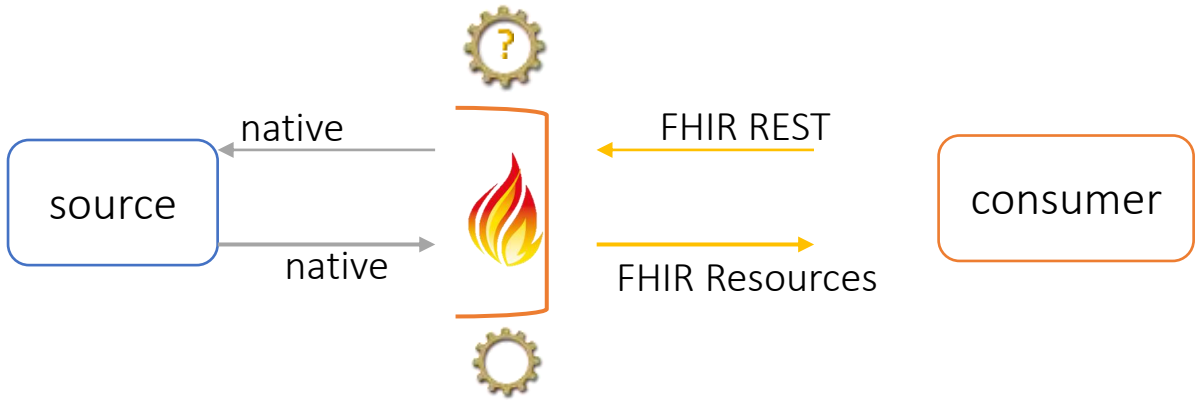


Facade on consumer



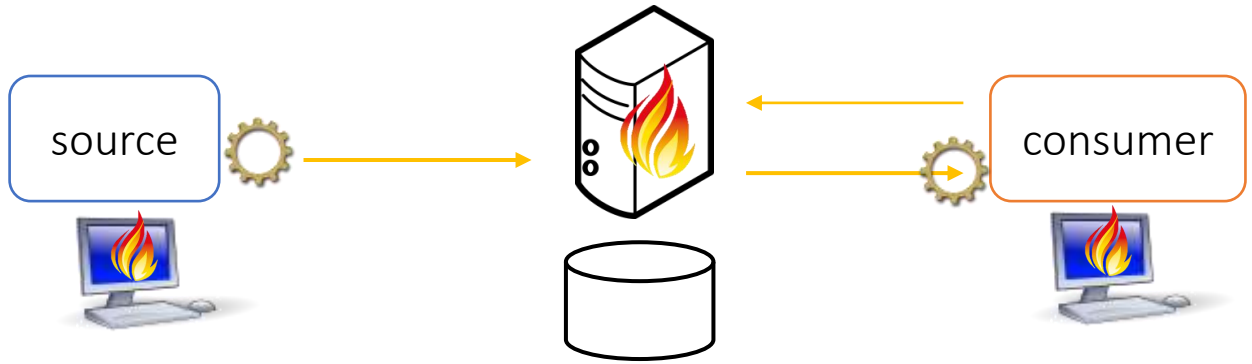
Facade on source

- put the API directly on the source
- read natively from the source
- map read / search inbound (!)
- map data outbound
- source becomes a FHIR Server
  - supporting read / search



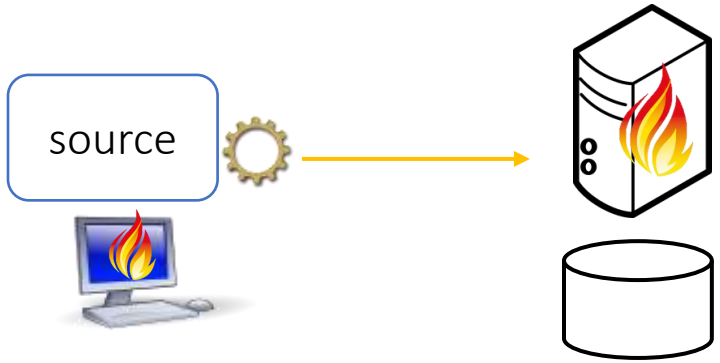
Generic FHIR Server

- pre-aggregate in a FHIR Server
  - results in a copy of data
- map data outbound
- scheduled or event driven (delay)
- defer load
- FHIR Server can be COTS
  - with all associated features



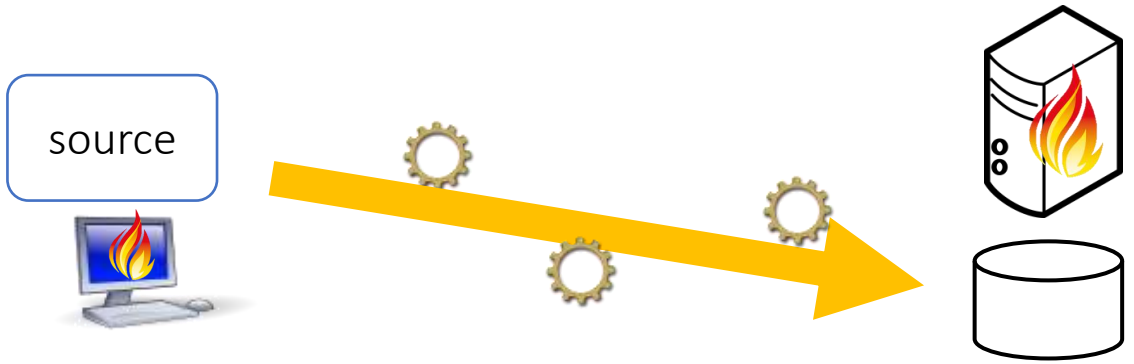
Loading data

REST API



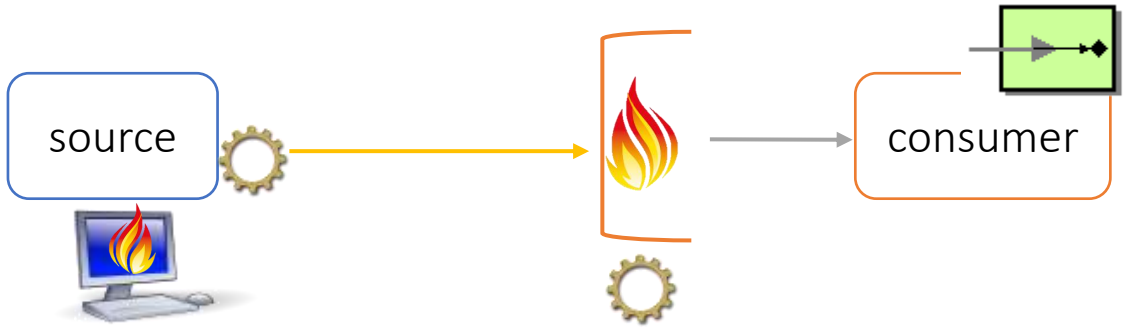
Specialized

- API
- CLI
- Data pipeline



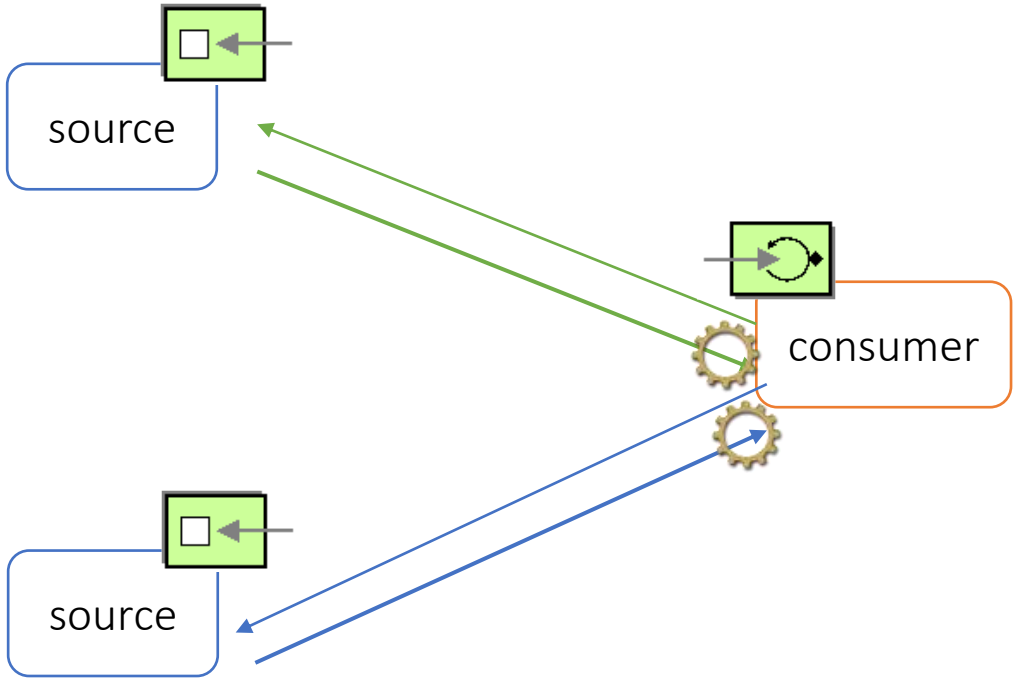
Facade on consumer

- Event-Driven Consumer
  - no longer Polling Consumer
- map create / update inbound
- map data inbound
- consumer becomes a FHIR Server
  - supporting create/update
- source becomes a FHIR Client



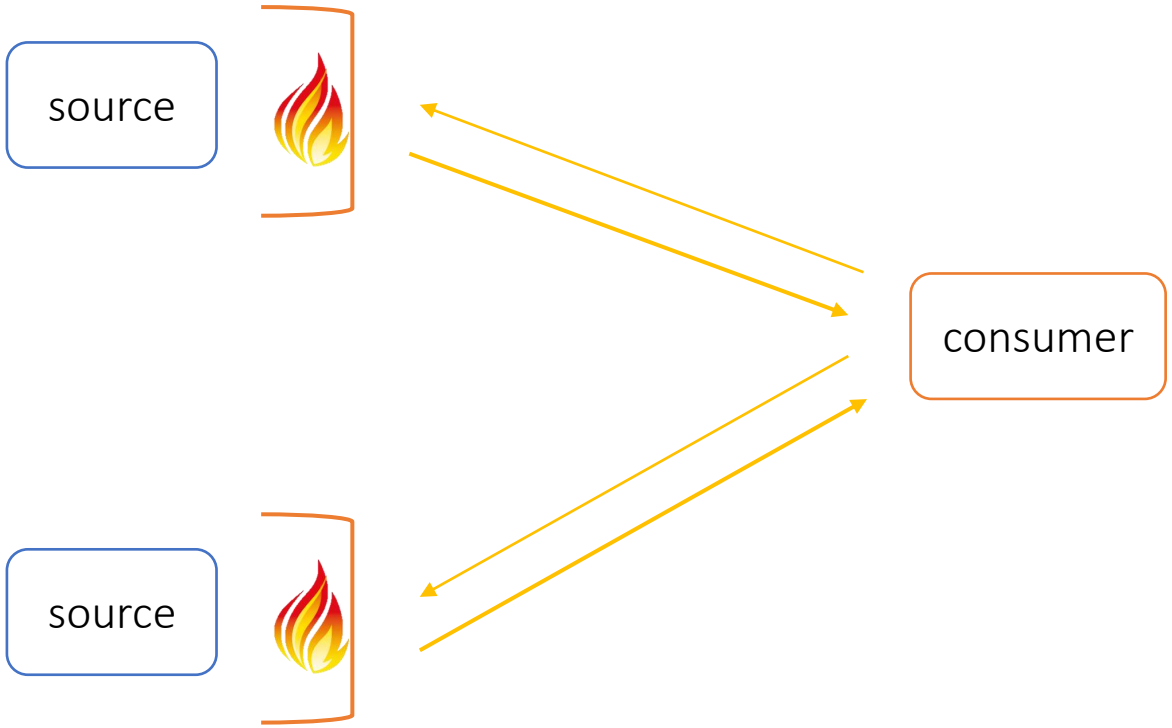
Multiple sources

- Consumer is the integration point
- Has the initiative
  - Polling Consumer
- Has a mapping for each source
  - Message Mapper
- May need to ask all sources
  - Scatter-Gather
- Has to combine all responses
  - Aggregator



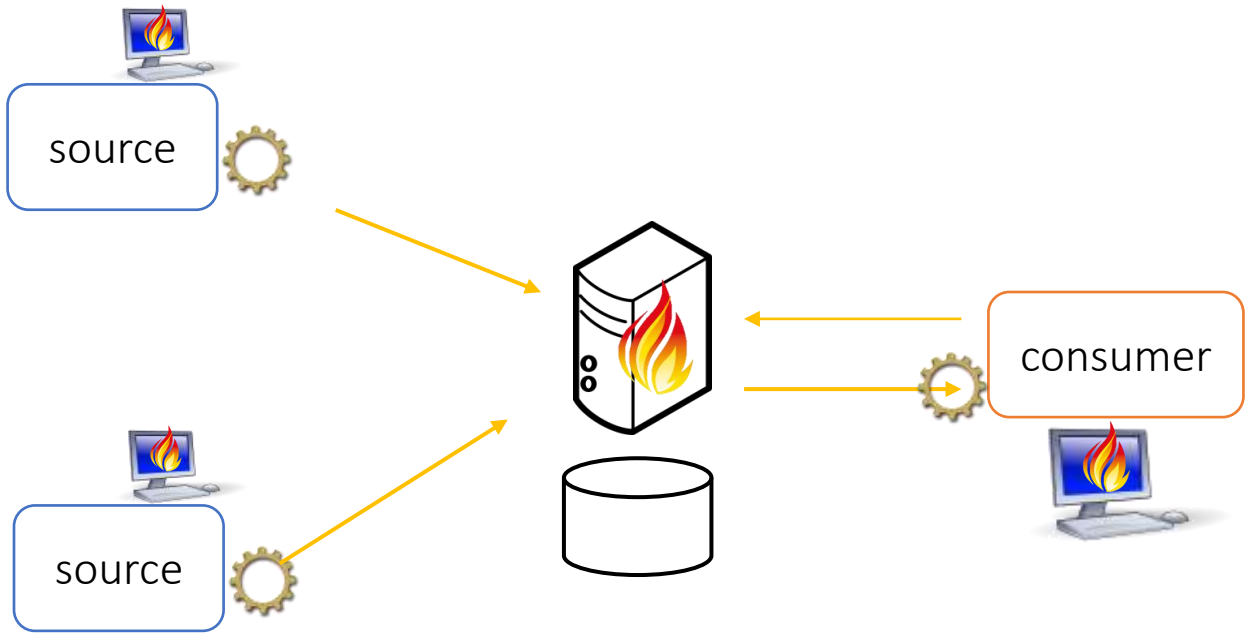
Facade on source

- Consumer
  - is the integration point
  - Polling Consumer
  - Has only one mapper
  - Scatter-Gather
  - Aggregator
- Source
  - become FHIR Servers
  - mapping of REST and data



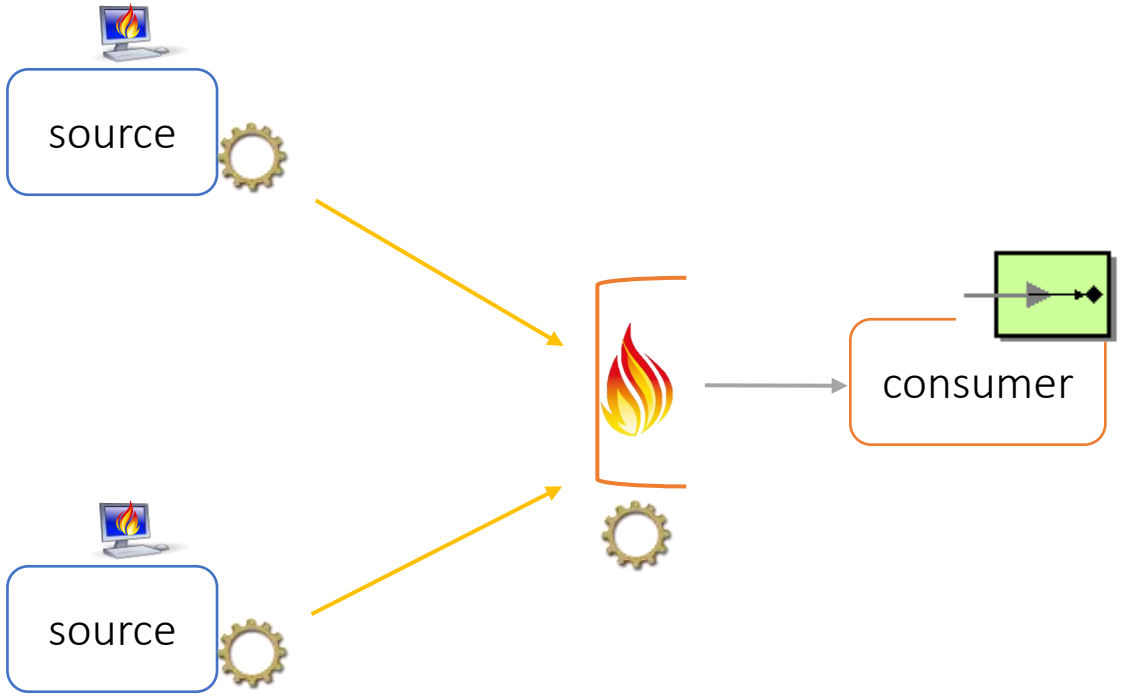
Use a Generic FHIR Server

- Consumer
  - is a FHIR Client
  - Message Mapper (FHIR – native)
- Sources
  - are FHIR Clients
  - Message Mapper (native – FHIR)
- FHIR Server
  - Aggregator
  - Copy of data (with delay)



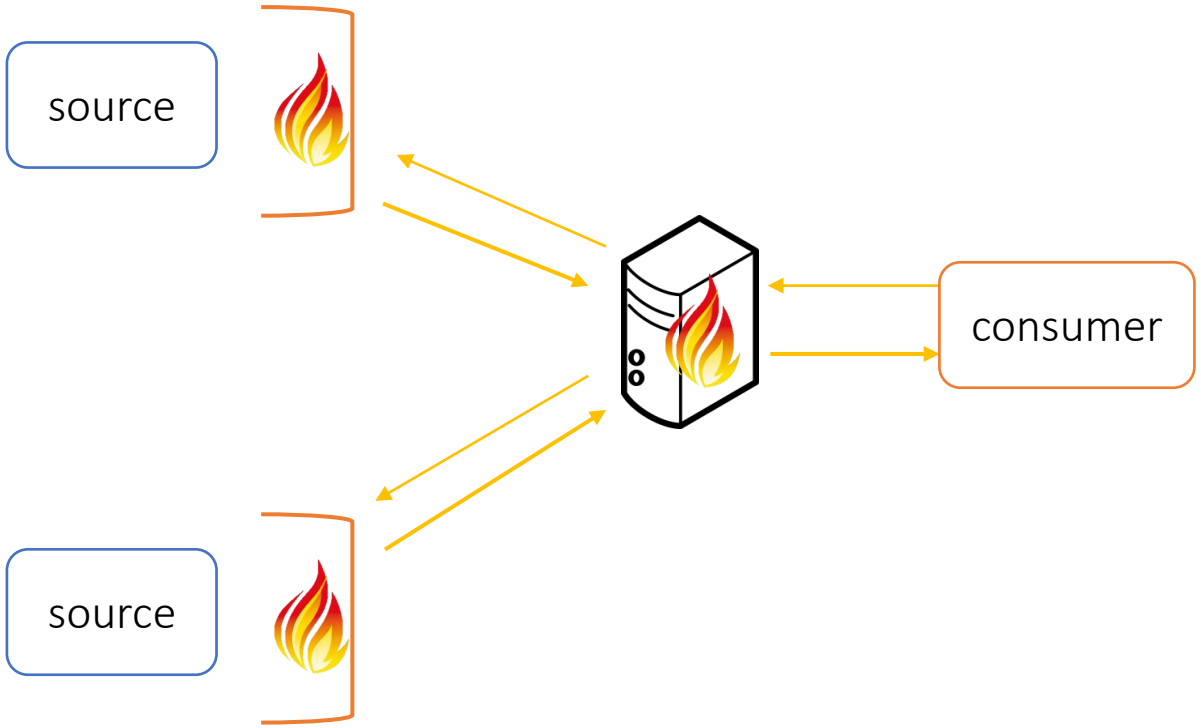
Facade on consumer

- Event-Driven Consumer
  - Message Mapper (FHIR – native)
  - consumer becomes a FHIR Server
  - supporting create/update
  - Aggregator
- Source
  - becomes FHIR Client
  - Message Mapper (native – FHIR)



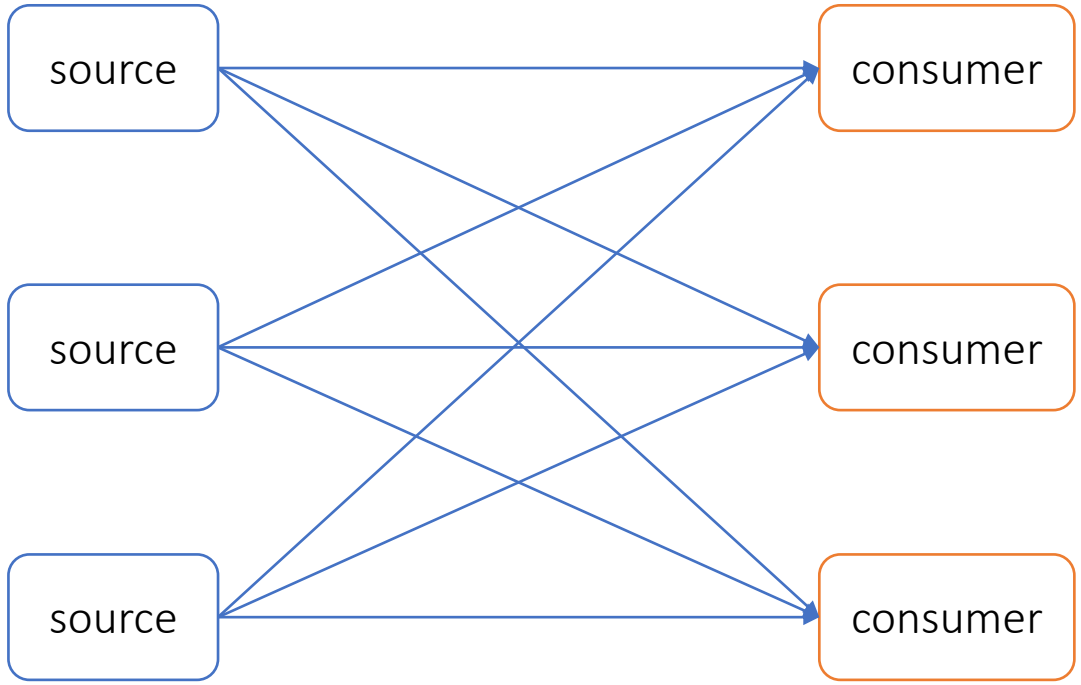
Facades + Generic Server

- Sources
  - Become FHIR Servers
  - Message Mapper
- FHIR Server
  - Message Router or
  - Scatter-Gather
  - Aggregator
  - No storage
- Consumer
  - FHIR Client



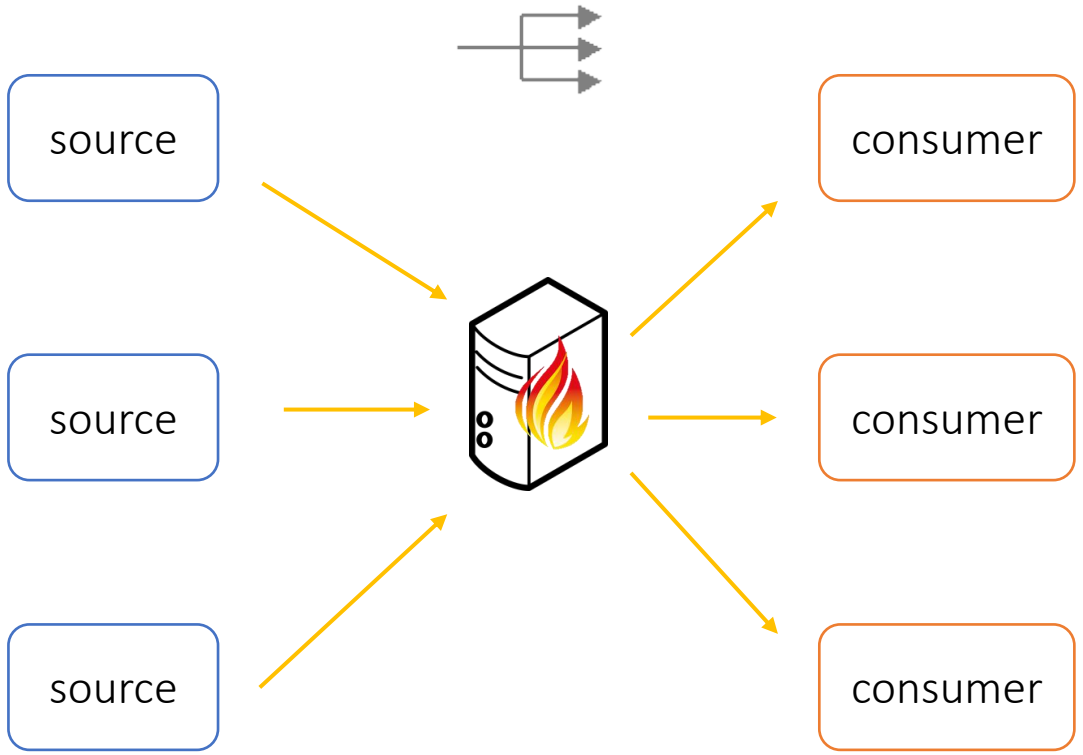
More sources and consumers

- From 2x2 up: mapping explodes
  - FHIR Resources as CDM useful
  - Encapsulate mappings
- Scatter-gather becomes hard
  - Put an Aggregator in the middle
- Not all systems can run a Facade
  - make them polling consumers
  - or event-driven providers



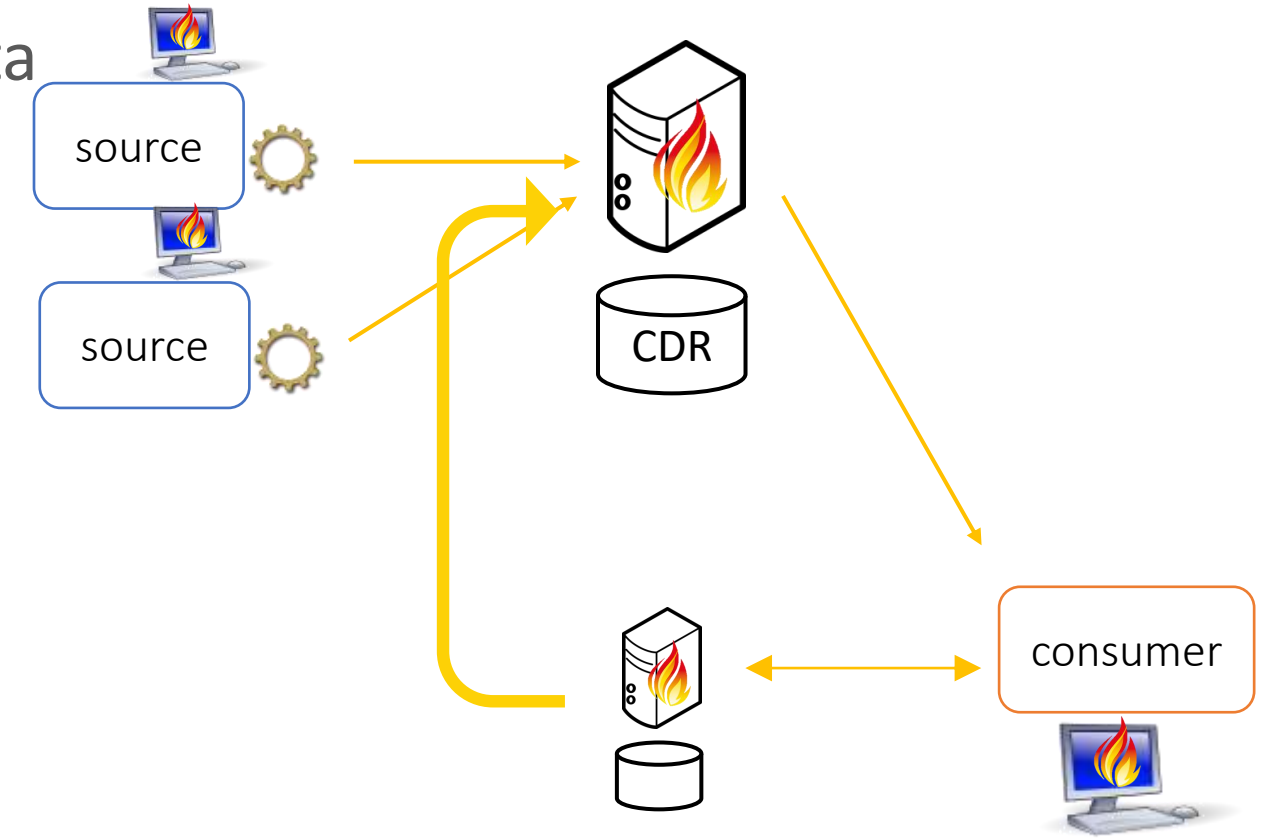
Pub/sub

- Subscriptions in FHIR R3/4
- Revamped in R4B/5
- Needs change tracking
- Hard with Facade



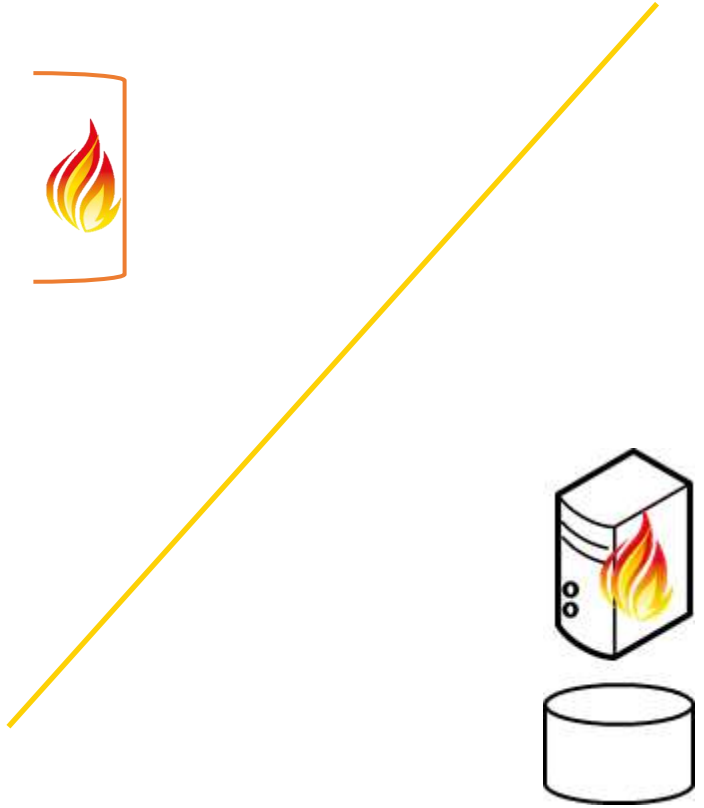
CDR & writing

- Most consumers read existing data
  - high volume (e.g. EHR data)
  - integrated (department systems)
  - read only
- Some (also) want to write data
  - survey app
  - personal devices
- Solution
  - separate writeable server
  - acts as an extra source to the CDR



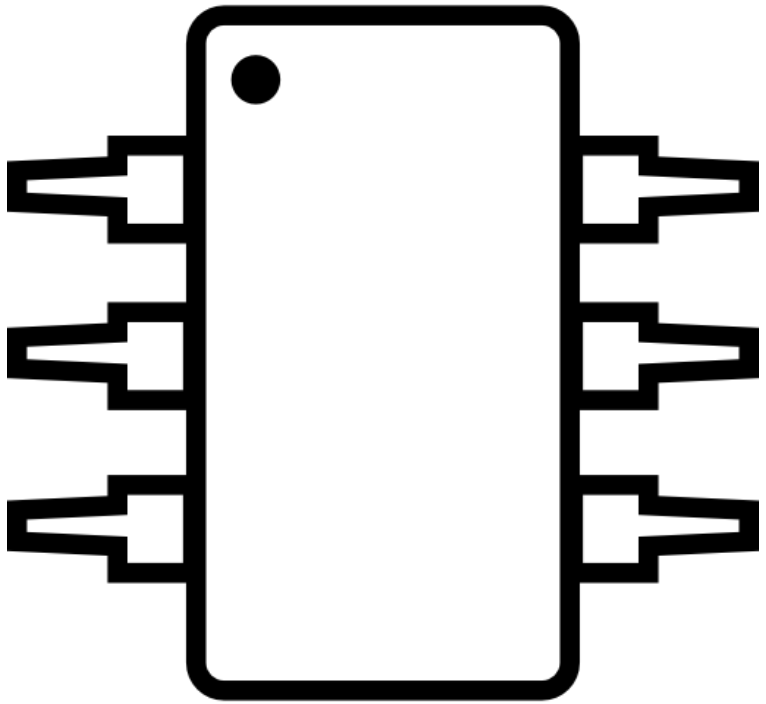
Experience

- Facade?
  - < ~ 10 resource types
  - < ~ 3-5 searchparameters each
  - read only
  - expose small part of big dataset
- Beyond that, a full server is more manageable
  - more loosely coupled
  - full search out of the box
  - data sync can be complex

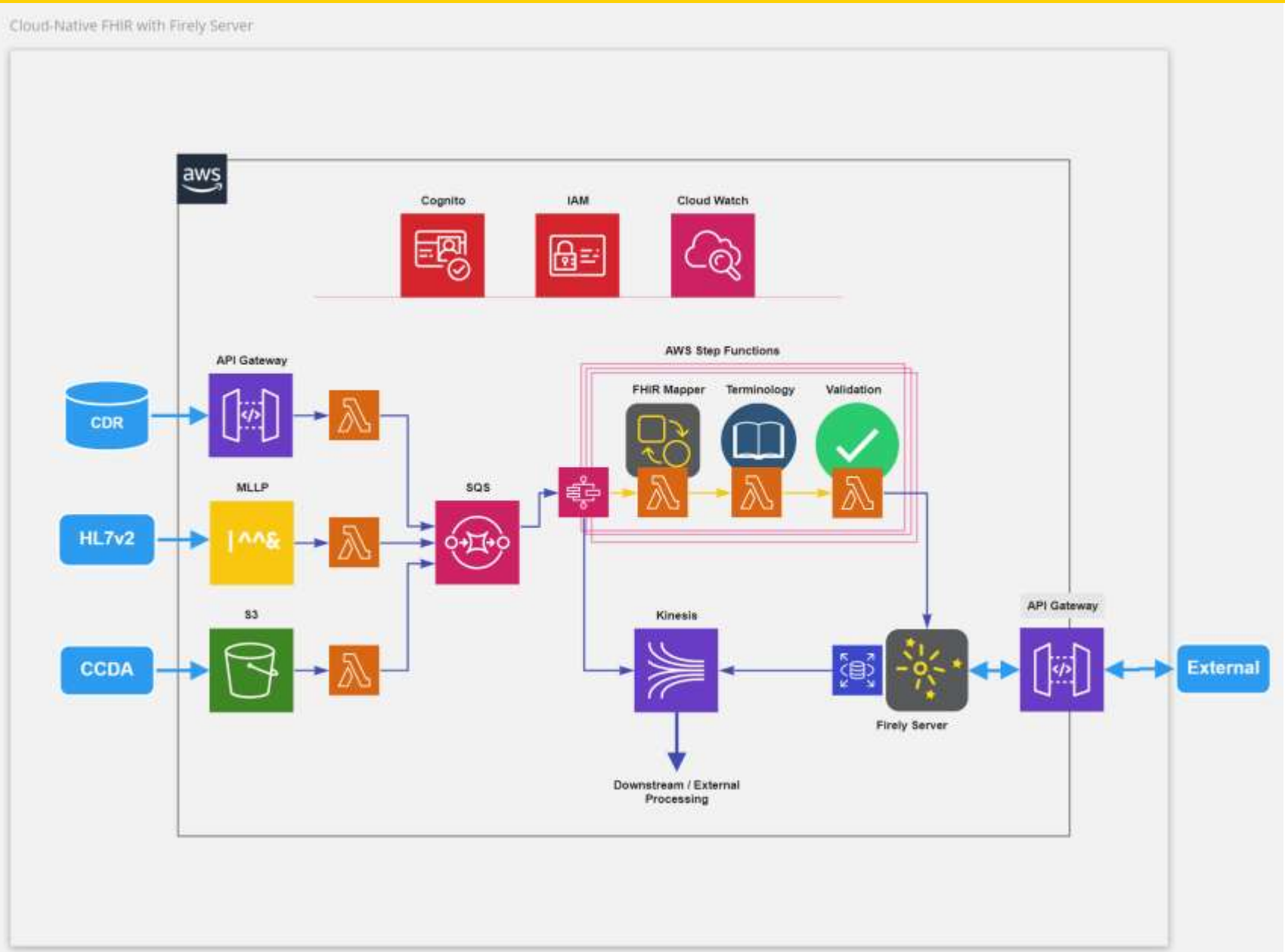


Common Integration Engine?

- Can provide scheduling
- Can do mapping, to/from FHIR or otherwise
- May be a router
- Probably better fit for a message broker solution
  - with or without FHIR Resources as CDM



Cloud data pipeline



Related IG's

SMART on FHIR
- FHIR Server = resource server (evaluate SMART scopes)
- needs additional Authz server (assign SMART scopes)

SMART Web Messaging
- App to browser communication (HTML5)
- No FHIR Server involved

FHIRCast
- Application context synchronization
- No FHIR Server involved

Bulk Data Export
- $export: custom op. on REST API
  - asynchronous
  - data export + file download
  - combined with SMART on FHIR

CQL
- Clinical Quality Language
- Not tied to FHIR Server, not expressed in FHIR REST
- But FS may support data retrieval

FAST
- FHIR at Scale Taskforce
- Routing FHIR REST through intermediaries
- FHIR Server can be an endpoint or an intermediary / proxy

CDS Hooks
- Gets a context from the EHR
- Can query additional data from FS
- No special requirements for FS

V2 to FHIR / C-CDA on FHIR
- standardize / aid in mapping older standards
- FHIR Server may support transforms: custom functionality

Considerations

- Do you allow incoming requests?
- Do you allow (delayed) copies?
- Do you control all participants?
- Who has the initiative?
- Who is the primary source of truth?
- What are the development capabilities of your team?

The answer

It depends ☺

Icons made by several authors from www.flaticon.com
- Freepik
- Pixel perfect