Using an identity broker to augmenting existing login solution to be SMART on FHIR compatible

Joshua Varner - Lyniate

HL7 FHIR DevDays International 2022 | Hybrid Edition, Cleveland, OH | June 6–9, 2022 | @HL7 | @FirelyTeam | #fhirdevdays | www.devdays.com

ORGANIZED BY firely | HL7 International

Who am I?

- Joshua Varner
- Software Architect at Lyniate
- Me ->

Scope

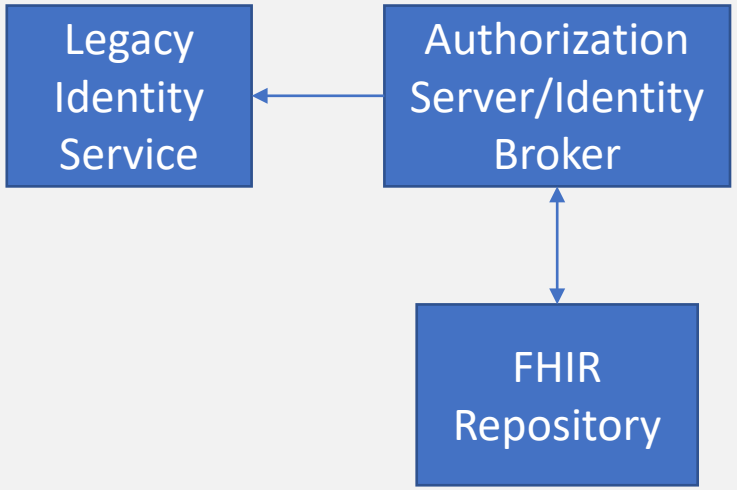
- Authentication and Authorization are complicated – I'll cover as little background as possible
- Concentrate on discovery and requirements process – what do you need to consider in this area
- High level technical design / light on details – mostly nitpicky and tedious

The Project

- Provide a SMART on FHIR compatible repository with initially one client application. - Simple
- Ingest data from existing systems to populate the repository and keep it up to date – Simple
- Leverage existing login service owned by another department and shared across multiple, without making changes to it
  - Complicated - Eliminates off the shelf products

Linking to an existing provider

- Organization invest heavily into identity management solutions
- Standard solution to this problem: Identity broker/provider
- Brings Authorization server back under same team as repository



Adding SMART on FHIR semantics to brokered identity

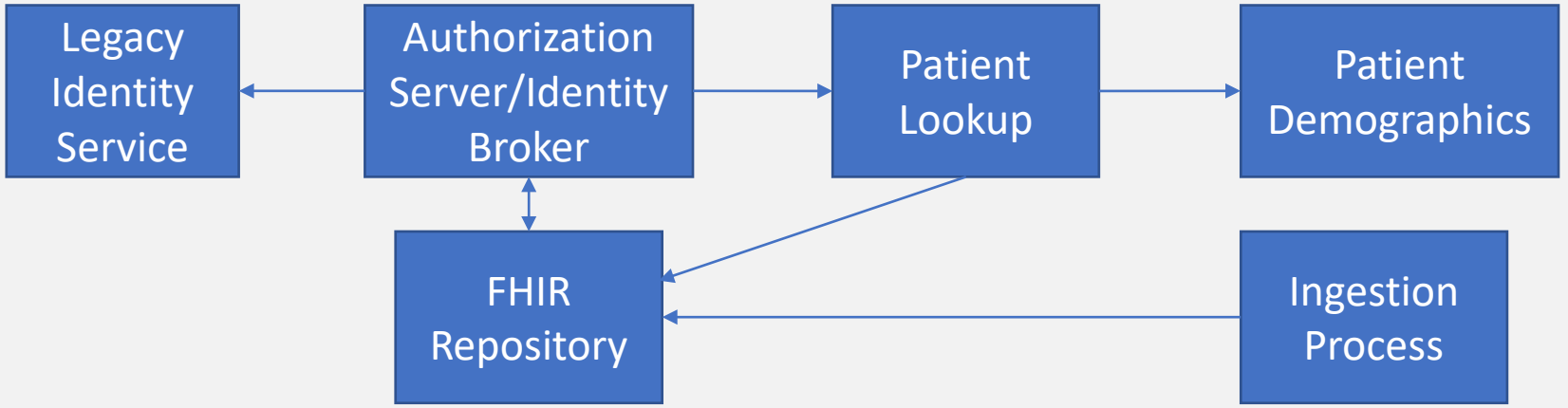
- Authorization is complex – leverage existing technology
- We chose Keycloak as the authorization server
- Leveraging Alvearie keycloak extensions for fhir
  - Standalone app launch
  - Audience validator
  - Patient context picker

fhirUser and coupling between auth and repository

- Background: fhirUser scope contains the url of the resource representing this set of credentials (Patient, Practitioner, RelatedPerson, Person)
- Mapping from the existing identity platform to the resources in the repository is highly context sensitive
  - Id token will contain a scope containing a medical identifier
- Implementation – custom extension to retrieve the identifier scope value, perform a lookup on repository, then populate the fhirUser scope

Missing Patient Resources

- What happens if the look up fails?
  - If there's no resource for this set of credentials
- Ingestion is driven by the exporting system, most likely independent of the identity provider
- Introduce a secondary on demand import

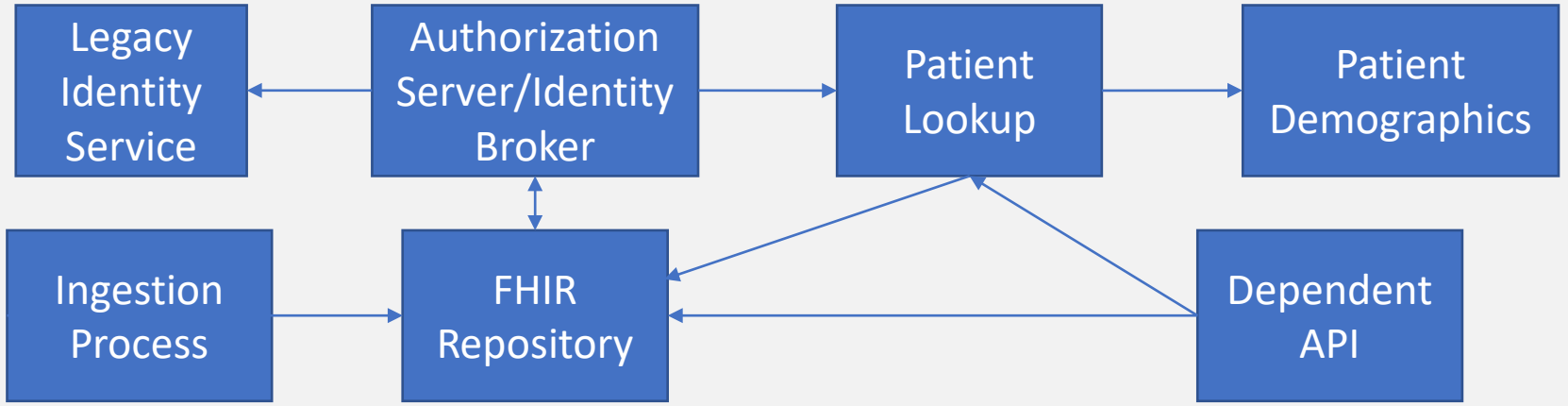


Dependent workflows needs

- Patient facing application – how do you handle dependents?
- Existing identity solution does not store relationships, but provides sufficient information to enable verification (phone number)
- Where do you store the relationships?
  - If the app stores them, then it has to access the repository with system level credentials
  - If the repository stores them, the verification process needs a secondary API to push that information to the repository

Dependent workflows solution

- App implements the verifications process
- Secured API to allow dependents to be linked
  - Persisted as Consent records
  - Leverages the Patient lookup functionality due to the missing record



Contact

- During DevDays, you can find / reach me here:
  - Via Whova App – Speaker's Gallery
  - Email – josh.varner@lyniate.com

Q&A

ORGANIZED BY