FHIR® Webinar

FHIR Implementation Guides – Structuring and Reusing

June 15, 2022

Agenda & Structure

- House rules, intro, agenda (5’)
- Standards and FHIR recap
- FHIR Implementation Guide inner structure
  - Knowledge Artifact Levels
  - Architecture Layers
  - Examples: IHE, WHO
- Defining multi-layered specifications with FHIR IGs
- 3: Hands-on: Create a derived profile
  - Goals & approach
  - Constraints
  - Structure
  - Requirements
- 4: Q&A
  - Your context / Challenges / ideas?

Remarks and disclaimers

- FHIR® is the registered trademark of Health Level Seven® (HL7®) International.
- The use of the FHIR® trademark does not constitute endorsement of this course/product/service by HL7 ®.
- This is not an official HL7 training. For such training opportunities, you are encouraged to
  - http://www.hl7.org/training
- This presentation is a collection of freely available materials.
  - All diagrams, except those that state otherwise, are original materials or taken from the FHIR website and support materials; all screenshots are from the FHIR website
- This presentation is shared under a Creative Commons Attribution 4.0 (CC BY 4.0) license - (ok to share and adapt if credits are given)

Goals

1. See how FHIR specifications are structured
2. Use the structured content of a FHIR IG to allow inheritance
3. Hand-on exercise

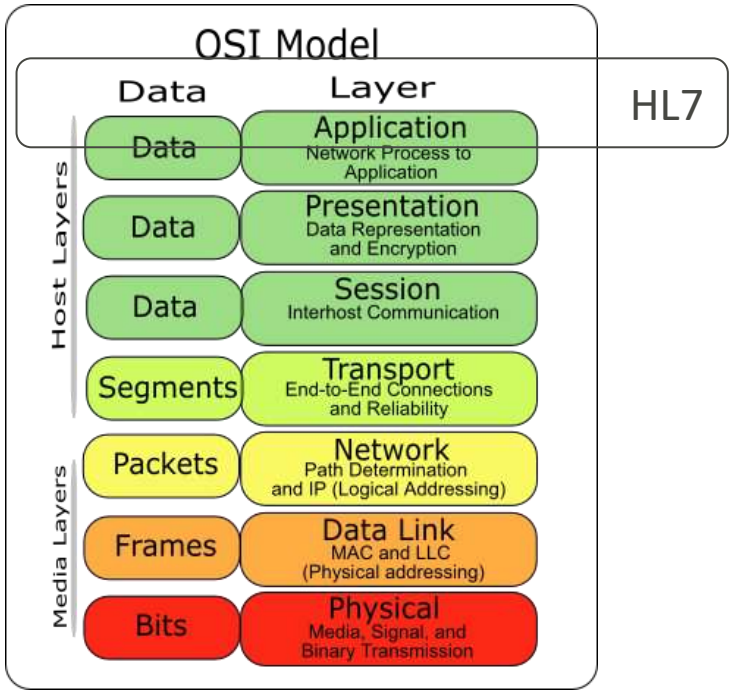
Reusing standard specifications

digital square

Motivations for Standardization

- Be conformant to standard X
- Support internationalization
- Be compatible with solution Y
- Standardize the data
- Reduce complexity while supporting variety

Situation: Levels / types of standards

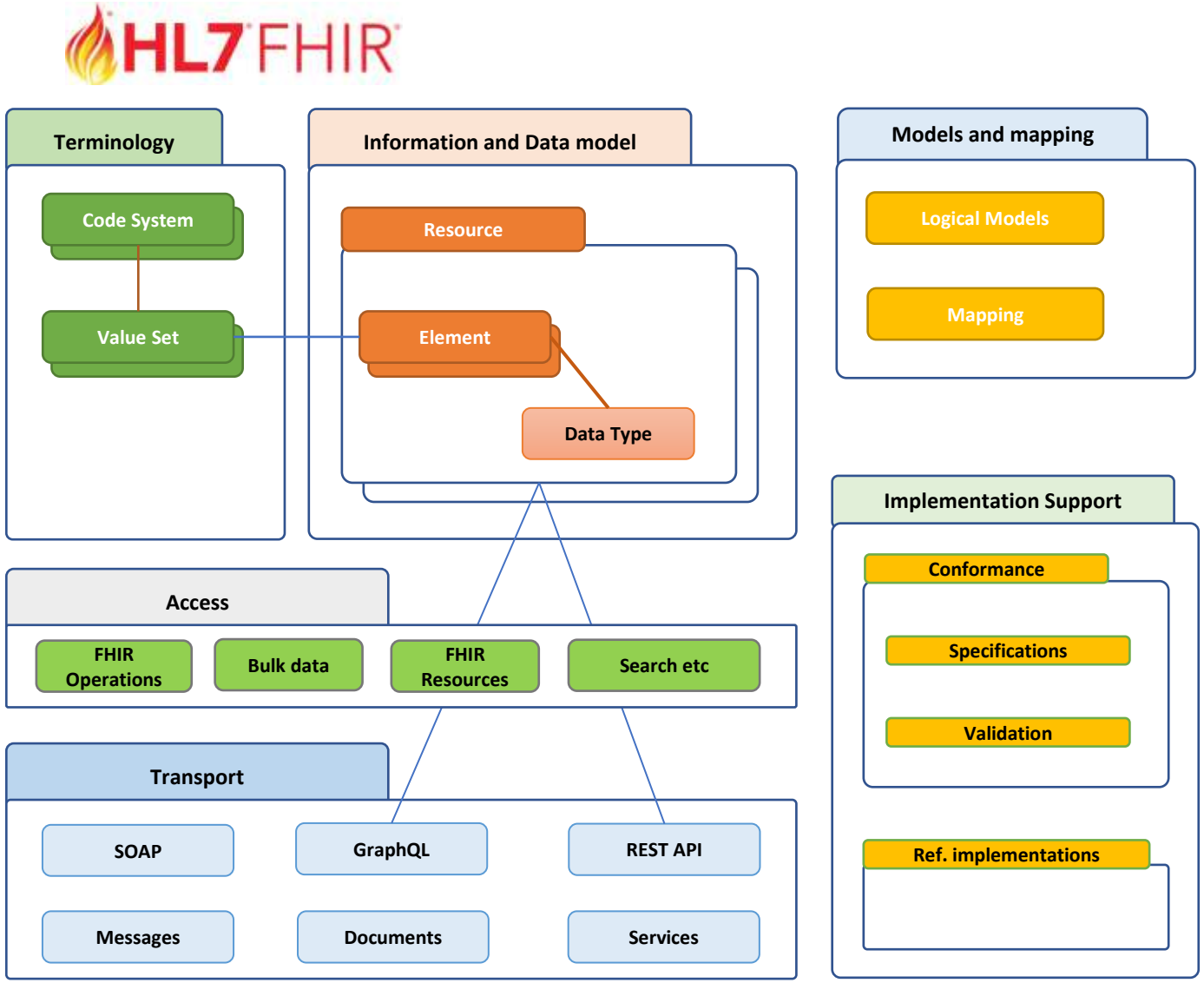


https://commons.wikimedia.org/wiki/File:Osi-model-jb.svg

	Why	How	What	Who	Where	When
Contextual	Goal List	Process List	Material List	Organisational Unit & Role List	Geographical Locations List	Event List
Conceptual	Goal Relationship	Process Model	Entity Relationship Model	Organisational Unit & Role Relationship Model	Locations Model	Event Model
Logical	Rules Diagram	Process Diagram	Data Model Diagram	Role Relationship Diagram	Locations Diagram	Event Diagram
Physical	Rules Specification	Process Function Specification	Data Entity Specification	Role Specification	Location Specification	Event Specification
Detailed	Rules Details	Process Details	Data Details	Role Details	Location Details	Event Details

https://commons.wikimedia.org/wiki/File:The_Zachman_Framework_of_Enterprise_Architecture.jpg

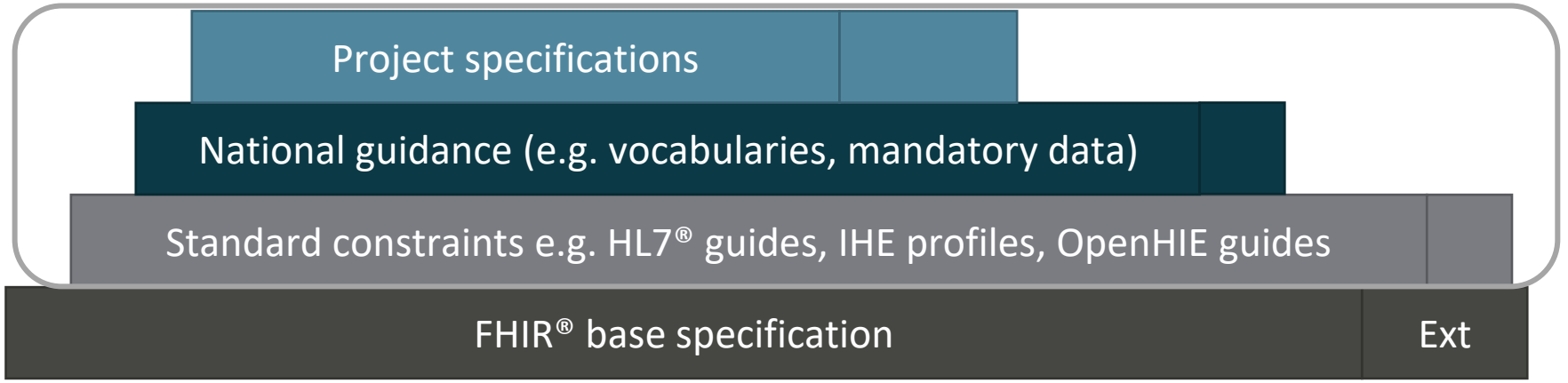
The HL7® FHIR® standard



Source: HL7 Belgium

Profiling in layers

- Profiling FHIR means constraining a FHIR specification
  - Fixing or binding some aspects of the specification
  - Defining which expansions to use
- This allows a layered specification – Today we'll discuss how.



Reminder –

- FHIR profiling consists of creating resource instances
- FHIR artifacts are structured, computable specs
  - Which supports inheritance from known artifacts
- FHIR Profiling goes in constraints
  - We should not overconstrain on reusable specifications

FHIR IG Structure

Standardization and Layers

digital square

How are FHIR ImplementationGuides Structured?

Knowledge Levels

Knowledge Level	Description	Example
L1	Narrative	Guideline for a specific disease that is written in the format of a peer-reviewed journal article
L2	Semi-structured	Flow diagram, decision tree, or other similar format that describes recommendations for implementation **(HUMAN READABLE)**
L3	Structured	Standards-compliant specification encoding logic with data model(s), terminology/code sets, value sets that is ready to be implemented **(COMPUTER/MACHINE READABLE)**
L4	Executable	CDS implemented and used in a local execution environment (e.g., CDS that is live in an electronic health record (EHR) production system) or available via web services

http://hl7.org/fhir/uv/cpg/2019Sep/documentation-approach.html1

Narrative

Level 1 (L1) is the clinical guideline, generally a narrative, often in Portable Document Format (PDF), and includes:

- Clinical Focus
- Scope
- Setting(s)
- Recommendations
- Who / what is included (inclusions)
- Who / what is excluded (exclusions)
- Policy considerations (e.g. privacy, access, regulations)

For example, one such guideline is the Guideline for Prescribing Opioids for Chronic Pain (2016) published by the Centers for Disease Control (CDC).

Semi-structured

- Level 2 (L2) is semi-structured, and generally consists of process and workflow diagrams, user stories, and personas that provide contextual and descriptive information about the steps involved in each recommendation of the guideline. This level generally includes:

Structured

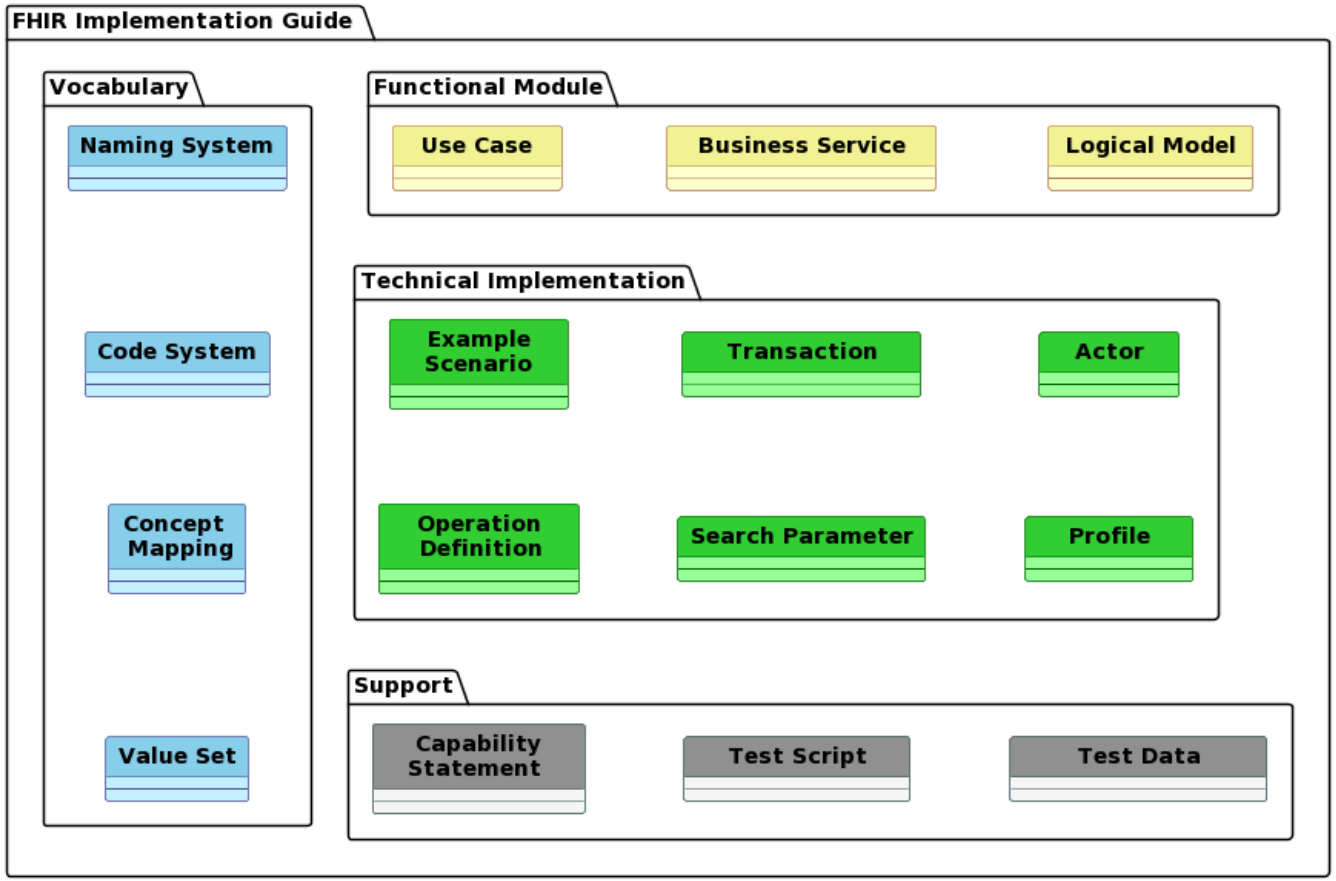
- Level 3 (L3) is a computable representation of the content in the guideline and can be shared across health systems and implementation settings. This level is the primary focus of this implementation guide.

Executable

- Level 4 (L4) is executable software running in a clinical system. This is the local implementation.

Architecture layers

Example of IG structure



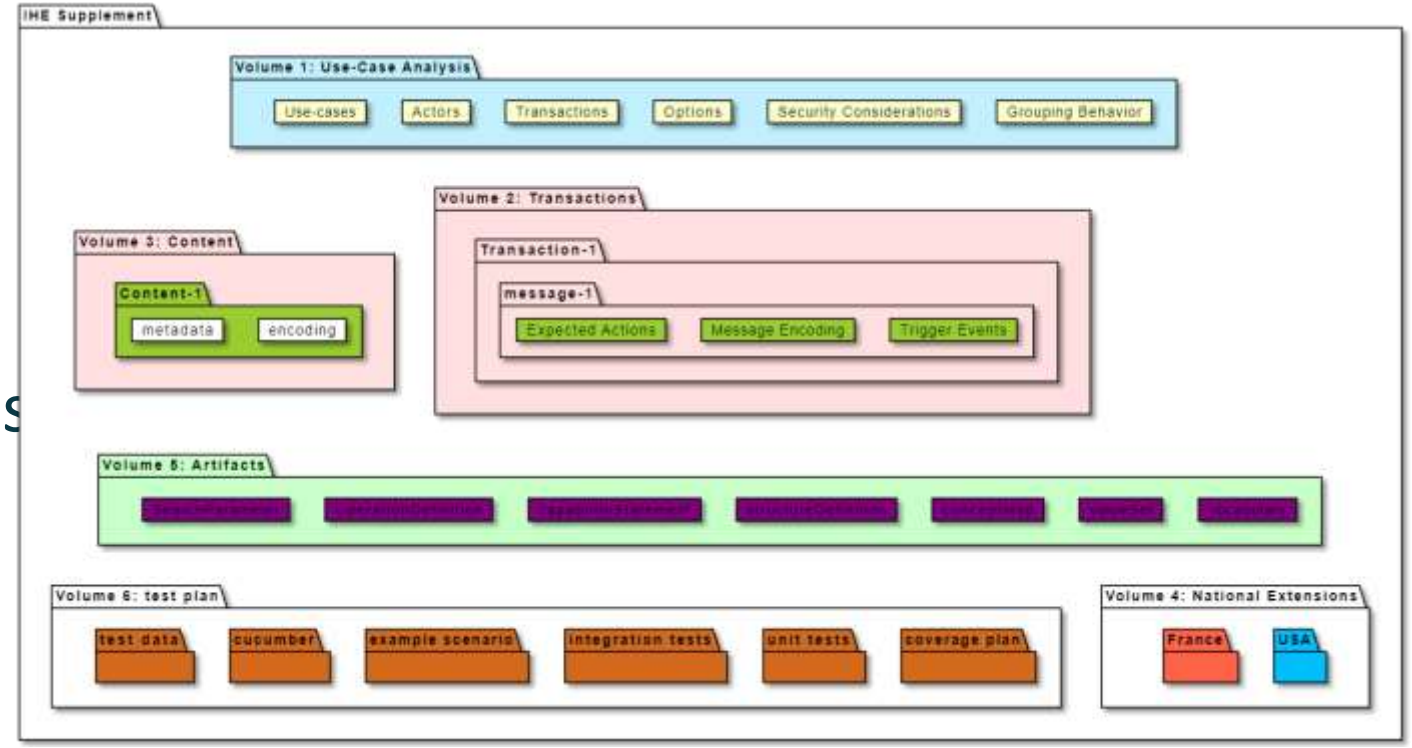
Source: HL7 Belgium (2020)

IHE structure

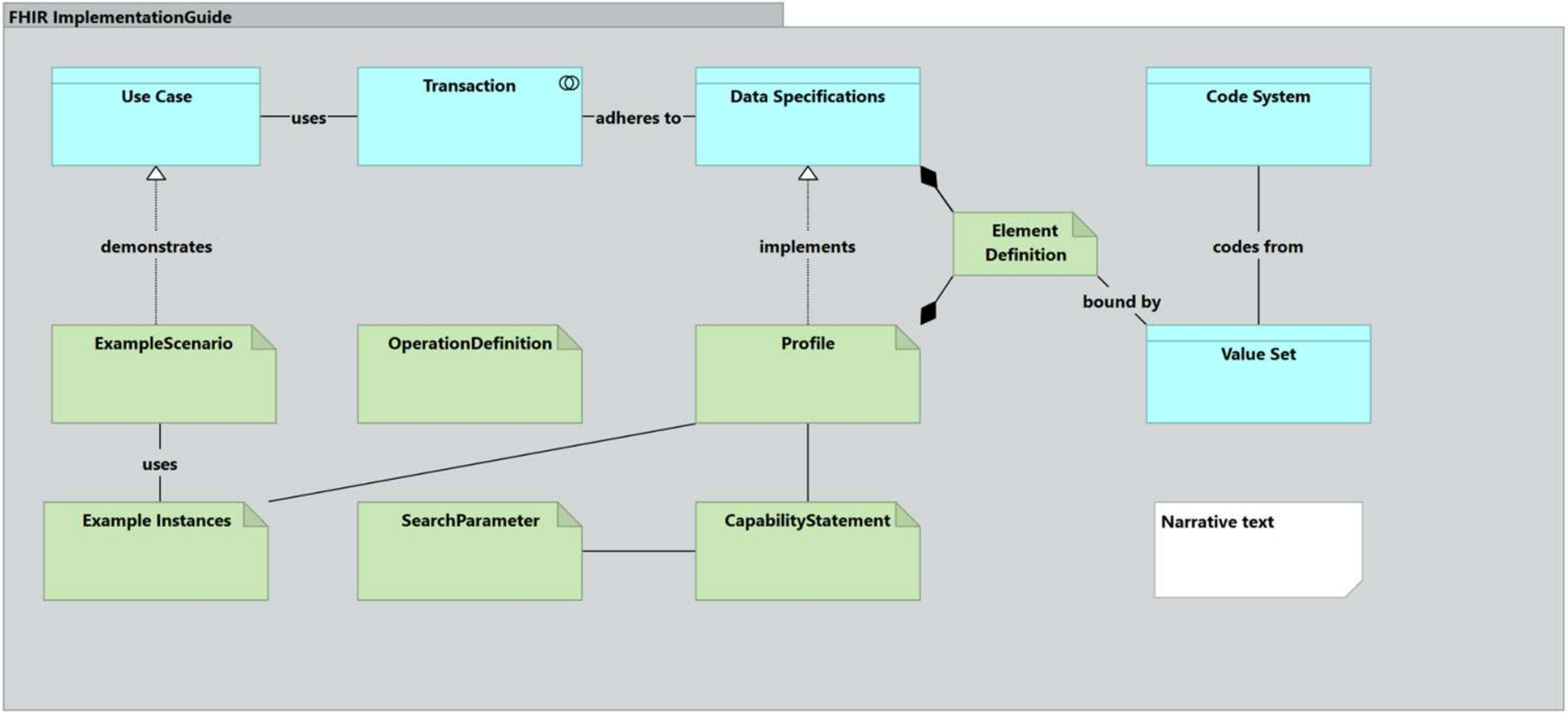
Organization of This Guide

This guide is organized into five main sections:

1. Volume 1: Integration Profile
    1. Introduction
    2. Actors and Transactions
    3. Actor Options
    4. Actor Required Groupings
    5. Overview
    6. Security Considerations
    7. Cross Profile Considerations
2. Volume 2: Transaction Detail
    1. Store Multimedia Report [RAD-141]
    2. Display Multimedia Report [RAD-142]
    3. Find Multimedia Report [RAD-143]
    4. Retrieve Rendered Multimedia Report [RAD-144]
3. Volume 3: Metadata
    1. N/A for IMR
4. Volume 4: National Extensions
    1. N/A for IMR
5. Appendix
    1. Test Plan



https://github.com/IHE/supplement-template/wiki/Getting-Started


FHIR ImplementationGuide
Use Case
uses
Transaction
adheres to
Data Specifications
Code System
demonstrates
implements
Element Definition
codes from
bound by
ExampleScenario
OperationDefinition
Profile
Value Set
uses
Example Instances
SearchParameter
CapabilityStatement
Narrative text

Conclusion - organize your IGs for layering

- Each IG is a package
  - Choose the right size

- With an IG we get the packages as they are
  - Find the right balance for constraining in the right level

Hands-on

Let’s make our own Food Allergy Specification

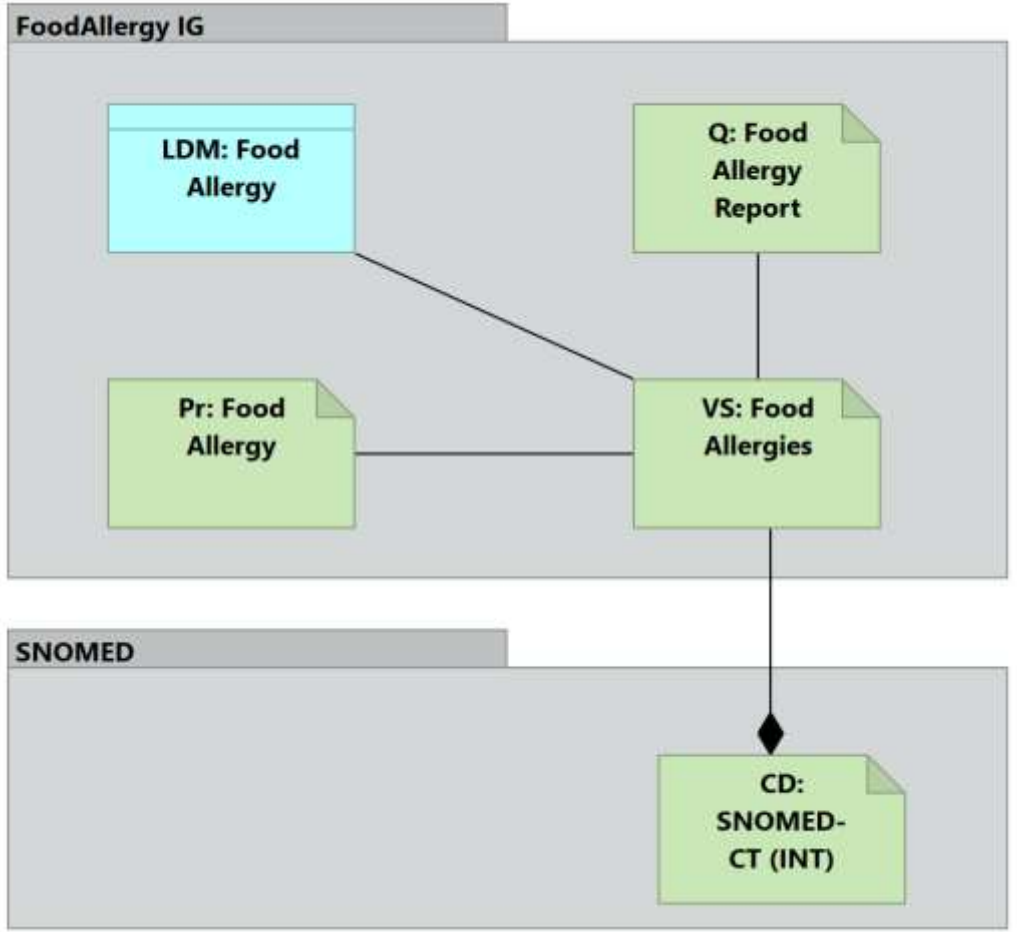
- Goal: Add allergy to wheat - Reusing the IG for Food Allergy

- Reuse or inherit
  - Logical data model?
  - Profile
  - Terminology?

Let’s make our own Food Allergy Specification

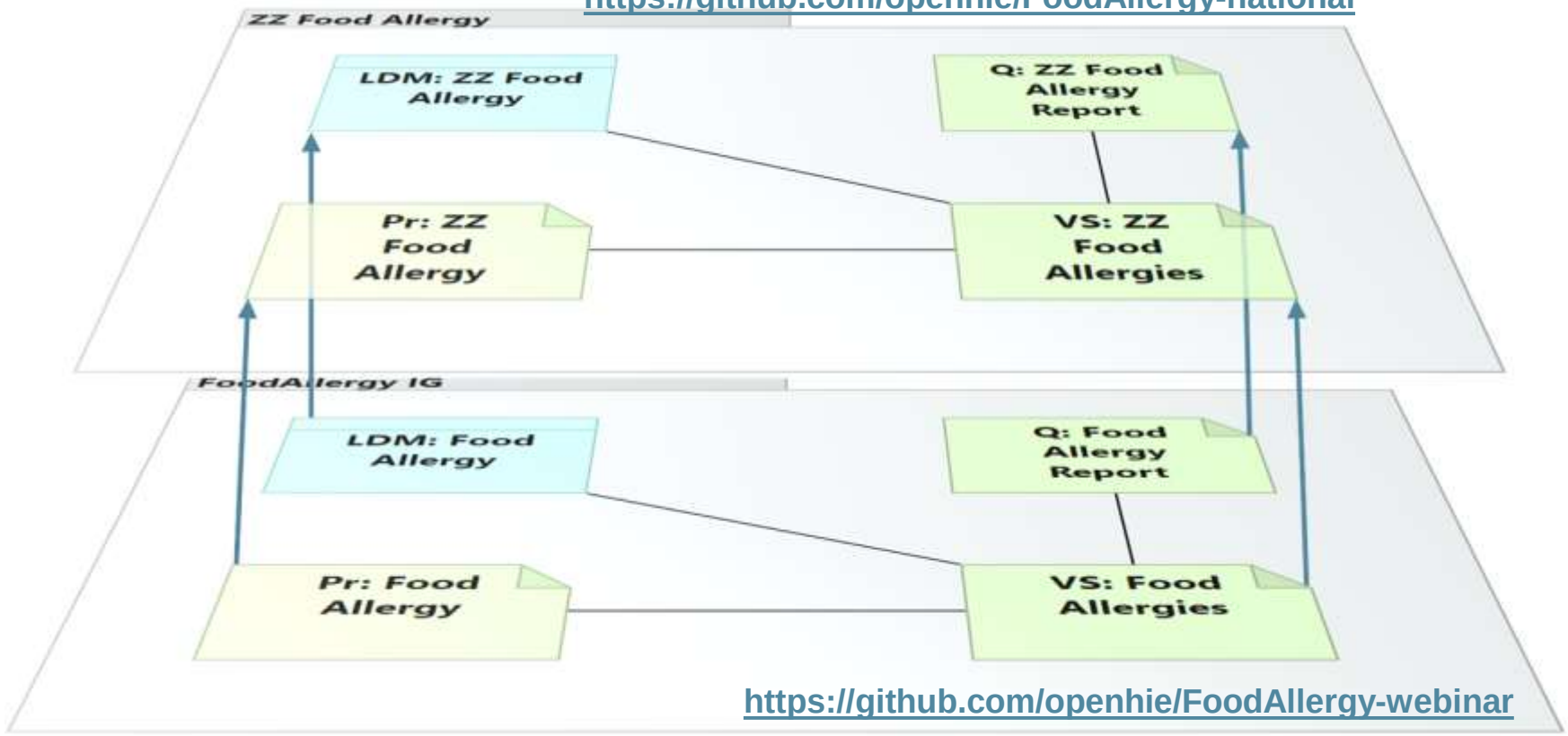
- Goal: Add allergy to wheat

- Reuse or inherit
  - Logical data model?
  - Profile
  - Terminology?



Create a FoodAllergy IG that depends on the existing IG

https://github.com/openhie/FoodAllergy-national



https://github.com/openhie/FoodAllergy-webinar

1. Create the new IG

E.g. by using template in

https://github.com/openhie/empty-fhir-ig-custom

And adapting the content - name of the ImplementationGuide, title, description, etc.

Commit: https://github.com/openhie/FoodAllergy-national/commit/7d5b587fb6d9649ebb3dda57ddb01cf9702ebbf3

2. Add an intro page

This is a new IG so we can and should define some narrative. We add some text to explain the context of the IG.

Note: Narrative is never expected to be inherited. Some parts can be copied, but it's not a managed artifact, so it cannot be inherited as other parts of the specification.

commit:

https://github.com/openhie/FoodAllergy-national/commit/b3dec192b823b8a1525162190ea7a0d5b6b40a1e

3. Add dependency from the existing IG

- The publisher needs to find the package
  - This means it must be published somewhere
  - OR existing in your local cache (which only happens if you built it locally)
  - When starting,
    - Make sure the dependency IG uses the FHIR autobuilder as is available: https://build.fhir.org/ig/openhie/FoodAllergy-webinar
    - use #dev version if there's no official release yet (as is this case - it's published but no major release yet)

- Add dependency:

In sushi-config.yaml:

```
dependencies:
  ohie.fhir.food-allergy: dev
```

OR in the ig xml:

```
  <dependsOn>
    <uri value="http://ohie.org/fhir/food-allergy/ImplementationGuide/ohie.fhir.food-allergy" />
    <packageId value="ohie.fhir.food-allergy"/>
    <version value="dev"/>
  </dependsOn>
```

4. Add ValueSet

That is our main change.

We could reuse the value set but we're just going to recreate it.

Both old and new value set rely on SNOMED code system.

Commit:

https://github.com/openhie/FoodAllergy-national/commit/8d854fbcf0905f9ba431d54c97a878bb81344098

4. Derived Logical Models

1. We want to add the logical model that actually indicates the value set binding.

1. When publishing a LM, there are some details to keep in mind
    a. Elements cannot (currently) be inherited
    b. May as well redo the entire Logical Model - it's not too hard

So we'll redo the Logical model, not inherit

Commit:

https://github.com/openhie/FoodAllergy-national/commit/8a37e3039c079c0e0725043dda7ff779e57ff7f1

5. Add the inherited profile

Most important - but easiest to do - because profiles are inherited

Just add constraints on the elements we need

- In this case, the element “allergen” simply has a different binding - to another valueset
  - This constraint is possible because the inherited element has a “flexible” finding (extensible) - if it were required, we’d be extending a required binding, which is not a constraint - so we couldn’t inherit if that were the case.

Simply declare the profile, its parent, and the elements that are “touched”

```
Profile: FoodAllergyZZ
Parent: FoodAllergy
Title: "National Food Allergy profile"
Description: "National Food Allergy profile"

* code from FoodAllergyZZVS (extensible)
```

(in this case, we declare a new profile, based on the parent, and update its title, description and the binding of the element `code`

Commit:

https://github.com/openhie/FoodAllergy-national/commit/856a663036bcb22982a90ca4fd333bca51da275a

Result:

Contents: https://openhie.github.io/FoodAllergy-national/toc.html

Dependencies:
https://openhie.github.io/FoodAllergy-national/qa.html

Dependency Checks:

Package	Version	FHIR Release	Canonical	Web Base	Comment
ohie.fhir.food-allergy-national	1.0.0	4.0.1	http://ohie.org/fhir/food-allergy		
hl7.terminology.r4	3.1.0 M	4.0.1	http://terminology.hl7.org	http://terminology.hl7.org/3.1.0	
ohie.fhir.food-allergy	0.1.0 U	4.0.1	http://ohie.org/fhir/food-allergy	https://build.fhir.org/ig/openhie/FoodAllergy-webinar/	Matched to latest patch release
hl7.terminology.r4	3.1.0 M	4.0.1	http://terminology.hl7.org	http://terminology.hl7.org/3.1.0	see above

Templates: myorg.fhir.mytemplate#0.0.1 -> openhie.fhir.template#0.1.0 -> fhir.base.template#0.2.2

Discussion

Get in touch, be active

- Check with others (at chat.fhir.org or community.fhir.org)
- Create (or ask someone to create) a change request
- Join a FHIR event like DevDays (devdays.com), discuss
- Join a FHIR connectathon, test and provide feedback

Digital Square is supported by:

Digital Square is a PATH-led initiative funded and designed by the United States Agency for International Development, the Bill & Melinda Gates Foundation, and a consortium of other donors.

This presentation was made possible by the generous support of the American people through the United States Agency for International Development. The contents are the responsibility of PATH and do not necessarily reflect the views of USAID or the United States Government.