

Self-directed learning activity BMDS7001-03

w/c 2 May 2022

Exploring information graphs in the ClinFHIR modelling tool

1. Introduction

ClinFHIR is an educational and testing tool for FHIR, created by Dr David Hay, a retired GP from New Zealand. David writes a blog called Hay on FHIR: <https://fhirblog.com/>

In this practical exercise, you will start to learn how to use ClinFHIR to introduce you to the concepts of FHIR Resources and how they are linked to build complex information structures that are interoperable. Your work will be saved within your current browser session, but if you want to save your work for future reference, create a ClinFHIR account by clicking on the login button at the top right of the landing page (see <http://clinfhir.com/>):



In this exercise, text in **This Font Colour** means a link in the ClinFHIR application.

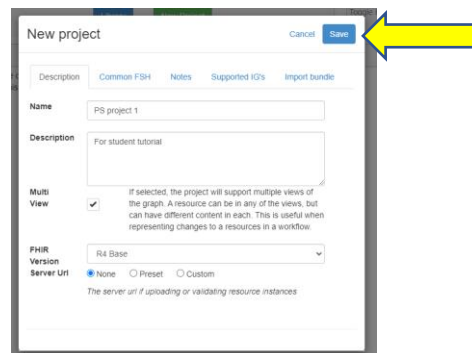
For reference about which data elements are in which Resource, you can check the FHIR specification at: <https://hl7.org/fhir/>

2. Open ClinFHIR and create a project and basic Resources

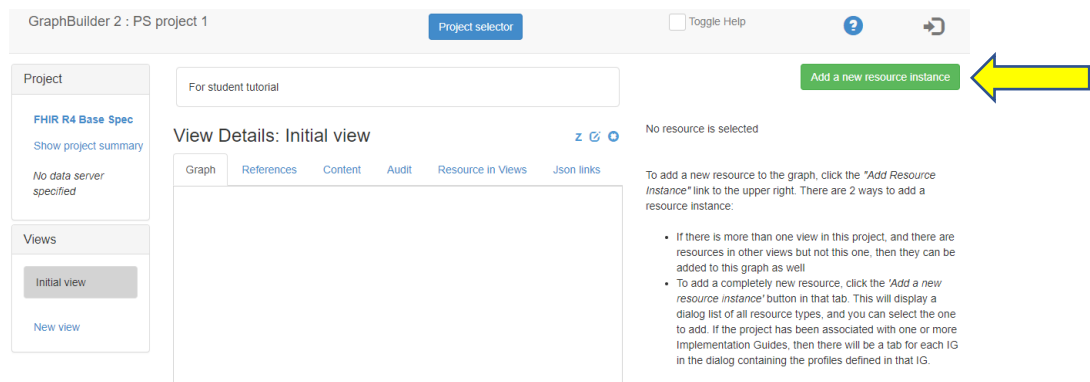
Click on **Graph Builder 2**:

clinFHIR Launcher		
For Implementers For designers Older modules		
Patient Viewer	Display resources for a specific patient, using a number of different views such as a list by resource type, json & tree views, encounters by condition, numeric Observation charting and graphical relationship views. There is also the option to add a new patient, and to create sample data for that patient.	Patient resources are stored on the Data Server. The server should support the Patient/\$everything operation.
Graph Builder 2	Build graphs of inter-connected FHIR resources. These can be created through the User Interface or downloaded from a FHIR server and further manipulated. The app uses the FHIR shorthand language to represent resource contents. This is the successor to Scenario Builder	Doesn't use the servers specified in this launcher. Conformance resources (R4) are built in to the app, and the data server specified 'in app' as well. Better support for Implementation Guides and Profiles.
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.	Can access any compliant FHIR server (must expose a Capability Statement)
Implementation Guide Browser	Display the contents of an Implementation Guide, and the relationships between the contents of the Guide.	The Implementation guide, profiles and Extension Definitions are on the Conformance Server, the terminology resources (eg ValueSet) are on the Terminology Server
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	Patient information is on the Data Server. Profiles on the Conformance server. ValueSets on the Terminology server. Create a simple scenario Adding structured data to a scenario Create a Document

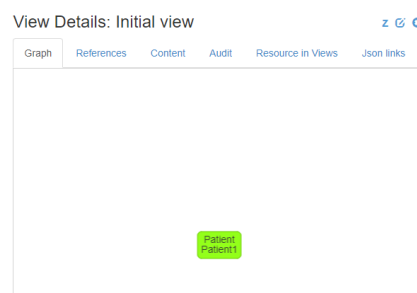
Click on **New Project** then give it a name and description (whatever you want) and click **Save**.



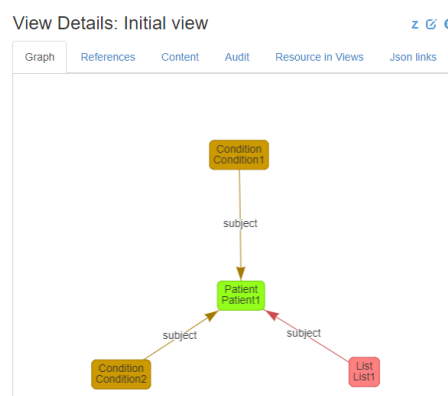
Click on the newly created project, then click **Add a new resource instance**



Add a Patient and click ok (accepting all the defaults) until you can see the Resource has been added:



Next add a List and two Conditions. You should then see:



ClinFHIR has automatically selected Patient1 as the “subject” of the List and the Conditions. Click on the **References** tab to see the default links that have been created:

View Details: Initial view z ↺ ⚙

References Content Audit Resource in Views Json links

Resource	Text	Type	References
Patient1		Patient	i
List1	PS test list	List	i subject -> Patient1
Condition1	T2DM	Condition	i subject -> Patient1
Condition2	Latex allergy	Condition	i subject -> Patient1

3. Adding links between Resources

All we have defined about the List so far is that it is about Patient1. Now you want to make the List a “container” for the two Conditions – a problem list for Patient1. Go back to the **Graph** view and click on **List1** in the diagram. Then click the **Add** button next to “entry”:

View Details: Initial view z ↺ ⚙

Graph References Content Audit Resource in Views Json links

Details in view Resource definition

Resource content

Display	List1
Type	List i
Id	cfsb1651594032787
Text	PS test list ✎

[Remove node from view](#)

Potential references in this view

BackBone elements **1**

[entry](#) **Add**

Clicking the 'Add' button will display a dialog allowing all child elements of the Backbone to be selected / set

Then click the link to **Reference**:

BBE: entry[0] Close Insert

Path	Mult.	Datatype/s	Value
flag	0..1	CodeableConcept	
deleted	0..1	boolean	
date	0..1	dateTime	
item	1..1	Reference	

and then select **Condition1**:

entry.item (Reference) Cancel Insert

Current resources

Display	Type	Text	Select
Patient1	Patient		Select
List1	List	PS test list	Select
Condition1	Condition	T2DM	Select
Condition2	Condition	Latex allergy	Select

Can be any resource type

Create new resource

Type Display

This will create a new resource and reference when the FSH Editor is saved.

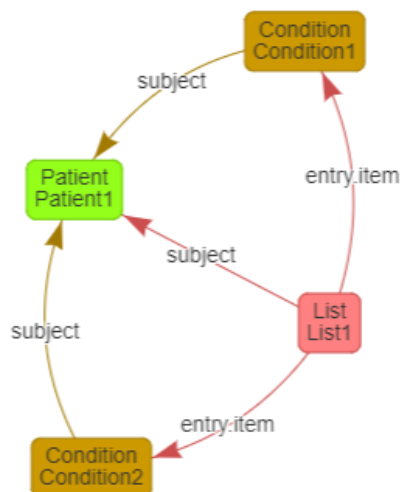
[Datatype details in spec](#)

This will then show you the link that will be created (in the Value column), so you can click **Insert**:

BBE: entry[0] Close Insert

Path	Mult.	Datatype/s	Value
flag	0..1	CodeableConcept	
deleted	0..1	boolean	
date	0..1	dateTime	
item	1..1	Reference	// Reference to Condition: Condition1 * entry[0].item = Reference(cfsb1651594048127)

Repeat that process to add Condition2 to the List. You'll then have something like this:

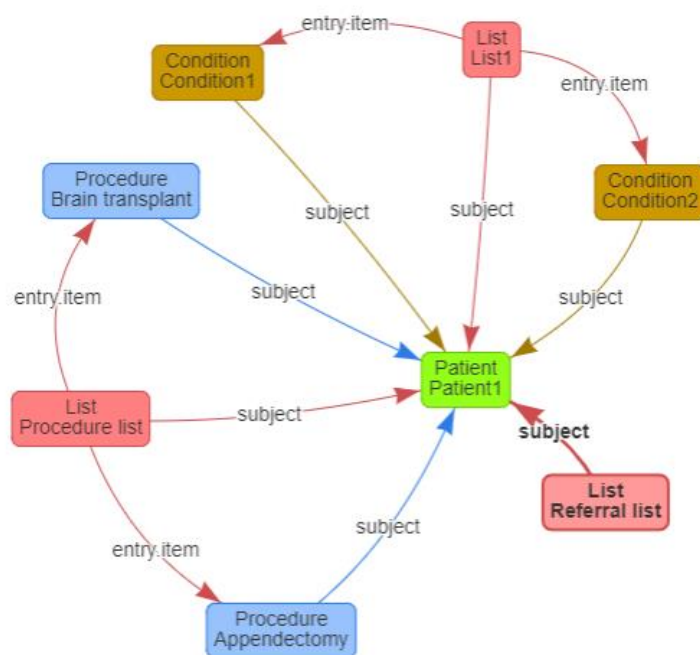


4. Adding further Resources and connections

You can now start to add other Resources (health and care concepts). Add another List for Procedures (one List and however many Procedures you want to insert) and one for a List of referrals (*which FHIR Resource would you use for a referral? Look around in Level 4 of the FHIR specification...*)

If the ClinFHIR diagram starts getting messy with links crossing over, you can drag the objects around to tidy up the layout.

You should end up with something like this (excluding the referrals, as you need to find out which Resource type to use for them):



For additional guidance, see <https://fhirblog.com/2021/10/04/creating-sample-data-with-graphbuilder/>

5. Supplementary extra work

To experiment further, try the following:

1. Add a medications List, with MedicationStatements using dm+d codes to identify the drugs (*Where do you put the codes?*).
2. Add SNOMED CT codes into the Condition objects.
3. Create a DiagnosticReport with links to several Observations (*Can you add identifier codes to the Observations to say which tests they are and what the results are?*).
4. Create a Composition (document) with sections containing text for “Recipient address”, “Salutation”, “Subject line” and “Body”.
5. Use a scenario from your own work to build a simple information model using FHIR concepts and SNOMED CT, ICD-10 or dm+d codes.