Clinicians on ClinFHIR

**ClinFHIR Hands-on Tutorial**

Russ Leftwich, MD
James McClay, MD
Dave Carlson
This Hands-On Session

• Goals
  • Learn ClinFHIR tool
  • Explore FHIR resources and their relationships
  • Introduce scenario modeling

• Webex use
  • www.webex.com Join by number
  • Allows for “close up” viewing of detailed screens
  • Be sure to use the tools such as zooming etc within webex as you need to

• Set-up
  • Recommend Chrome Browser
  • If you have numeric keypad CTRL-+/- zooms in and out
    • Use this if some items are off the screen or hard to read.
    • (On some laptops you can use CTRL-FCN to access the numeric keypad overlay)
  • Keep a notepad handy to jot down information

• Open ClinFHIR.com
Main modules (open in new tab)  Experimental modules (open in new tab)

**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.

**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.

**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

**Patient resources are stored on the Data Server. The server should support the Patient/$everything operation.**

**Current servers**

<table>
<thead>
<tr>
<th>Data Server</th>
<th>HSPC-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conformance Server</td>
<td>Grahames R4 server</td>
</tr>
<tr>
<td>Terminology Server</td>
<td>OntoserverR4 (terminology)</td>
</tr>
</tbody>
</table>

**Add Server**
Set all the same as the Data Server

**FHIR Links (open in new tab)**

- R4 Specification (current)
- STU-3 Specification
- STU-2 Specification
- FHIR wiki
- Hay on FHIR
- FHIR Chat
- FHIR.org
- Clinicians Workshop
# User Settings

**Main modules (open in new tab)**

<table>
<thead>
<tr>
<th>Module</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient Viewer</td>
<td>Display resources for a specific patient, using a number of different views such as a list by</td>
<td>Patient resources are stored on the Data Server. The server should support the</td>
</tr>
<tr>
<td></td>
<td>resource type, json &amp; tree views, encounters by condition, numeric Observation charting and</td>
<td>Patient/$everything operation.</td>
</tr>
<tr>
<td></td>
<td>graphical relationship views.</td>
<td></td>
</tr>
<tr>
<td>Server Query</td>
<td>Supports ad hoc queries against any FHIR server. Includes a simple query builder. The response</td>
<td>Can access any compliant FHIR server (must expose a Capability Statement).</td>
</tr>
<tr>
<td></td>
<td>can be displayed as Json or a Tree view, and FHIRPath is supported.</td>
<td></td>
</tr>
<tr>
<td>Scenario Builder</td>
<td>The Scenario Builder is used to join together the resources needed to represent a specific</td>
<td>Note that the builder still has issues with more complex resource types - this is a work in</td>
</tr>
<tr>
<td></td>
<td>clinical scenario. It can use Core Resource types, Profiles and Logical models as it does this.</td>
<td>progress.</td>
</tr>
<tr>
<td></td>
<td>The intention is to help people understand how resources can tell a clinical story, and to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>validate that the resource types available (including profiles) are sufficient.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Note that the builder still has issues with more complex resource types - this is a work in</td>
<td></td>
</tr>
<tr>
<td></td>
<td>progress.</td>
<td></td>
</tr>
<tr>
<td>Logical Modeler</td>
<td>The Logical modeler allows the creation of a model that represents a particular interoperability</td>
<td>Models are saved on the Conformance Server. Models can reference ValueSets from the</td>
</tr>
<tr>
<td></td>
<td>requirement in a format that is easy to use. It uses FHIR datatypes, and can be based on an</td>
<td>Conformance Server.</td>
</tr>
<tr>
<td></td>
<td>existing resource type or completely ‘ad hoc’. It is intended to act as a ‘bridge’ between</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Modeller and User, and can act as the basis for the generation of the profiling components</td>
<td></td>
</tr>
<tr>
<td></td>
<td>required by FHIR.</td>
<td></td>
</tr>
<tr>
<td>Implementation Guide</td>
<td>Display the contents of an Implementation Guide, and the relationships between the contents of</td>
<td>The Implementation guide, profiles and Extension Definitions are on the Conformance Server,</td>
</tr>
<tr>
<td>Browser</td>
<td>the Guide.</td>
<td>the terminology resources (e.g. ValueSet) are on the Terminology Server.</td>
</tr>
<tr>
<td>Bundle Visualizer</td>
<td>Various displays for the contents of a bundle. Bundles can be paged into the viewer and</td>
<td></td>
</tr>
<tr>
<td></td>
<td>optionally saved in the data server.</td>
<td></td>
</tr>
<tr>
<td>Extension Definition</td>
<td>Views and builds extension definitions. These can be defined and applied to the Logical Modeler,</td>
<td>Extension definitions are saved on the Conformance Server.</td>
</tr>
<tr>
<td>builder</td>
<td>which will allow them to be included in the generated Profile.</td>
<td></td>
</tr>
<tr>
<td>CodeSystem builder</td>
<td>The CodeSystem deffines a set of Concepts from which a ValueSet provides possible values for</td>
<td>CodeSystems are saved on the Terminology Server.</td>
</tr>
<tr>
<td></td>
<td>a resource element. The actual <code>binding</code> between CodeSystem and element is done by the ValueSet.</td>
<td></td>
</tr>
<tr>
<td>ValueSet explorer</td>
<td>Lets you view existing ValueSets. The builder works best with SNOMED (at the moment).</td>
<td>ValueSets are stored on the Terminology Server.</td>
</tr>
</tbody>
</table>

**Experimental modules (open in new tab)**

<table>
<thead>
<tr>
<th>Module</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient Viewer</td>
<td>Display resources for a specific patient, using a number of different views such as a list by</td>
<td>Patient resources are stored on the Data Server. The server should support the</td>
</tr>
<tr>
<td></td>
<td>resource type, json &amp; tree views, encounters by condition, numeric Observation charting and</td>
<td>Patient/$everything operation.</td>
</tr>
<tr>
<td></td>
<td>graphical relationship views.</td>
<td></td>
</tr>
<tr>
<td>Server Query</td>
<td>Supports ad hoc queries against any FHIR server. Includes a simple query builder. The response</td>
<td>Can access any compliant FHIR server (must expose a Capability Statement).</td>
</tr>
<tr>
<td></td>
<td>can be displayed as Json or a Tree view, and FHIRPath is supported.</td>
<td></td>
</tr>
<tr>
<td>Scenario Builder</td>
<td>The Scenario Builder is used to join together the resources needed to represent a specific</td>
<td>Note that the builder still has issues with more complex resource types - this is a work in</td>
</tr>
<tr>
<td></td>
<td>clinical scenario. It can use Core Resource types, Profiles and Logical models as it does this.</td>
<td>progress.</td>
</tr>
<tr>
<td></td>
<td>The intention is to help people understand how resources can tell a clinical story, and to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>validate that the resource types available (including profiles) are sufficient.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Note that the builder still has issues with more complex resource types - this is a work in</td>
<td></td>
</tr>
<tr>
<td></td>
<td>progress.</td>
<td></td>
</tr>
<tr>
<td>Logical Modeler</td>
<td>The Logical modeler allows the creation of a model that represents a particular interoperability</td>
<td>Models are saved on the Conformance Server. Models can reference ValueSets from the</td>
</tr>
<tr>
<td></td>
<td>requirement in a format that is easy to use. It uses FHIR datatypes, and can be based on an</td>
<td>Conformance Server.</td>
</tr>
<tr>
<td></td>
<td>existing resource type or completely ‘ad hoc’. It is intended to act as a ‘bridge’ between</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Modeller and User, and can act as the basis for the generation of the profiling components</td>
<td></td>
</tr>
<tr>
<td></td>
<td>required by FHIR.</td>
<td></td>
</tr>
<tr>
<td>Implementation Guide</td>
<td>Display the contents of an Implementation Guide, and the relationships between the contents of</td>
<td>The Implementation guide, profiles and Extension Definitions are on the Conformance Server,</td>
</tr>
<tr>
<td>Browser</td>
<td>the Guide.</td>
<td>the terminology resources (e.g. ValueSet) are on the Terminology Server.</td>
</tr>
<tr>
<td>Bundle Visualizer</td>
<td>Various displays for the contents of a bundle. Bundles can be paged into the viewer and</td>
<td></td>
</tr>
<tr>
<td></td>
<td>optionally saved in the data server.</td>
<td></td>
</tr>
<tr>
<td>Extension Definition</td>
<td>Views and builds extension definitions. These can be defined and applied to the Logical Modeler,</td>
<td>Extension definitions are saved on the Conformance Server.</td>
</tr>
<tr>
<td>builder</td>
<td>which will allow them to be included in the generated Profile.</td>
<td></td>
</tr>
<tr>
<td>CodeSystem builder</td>
<td>The CodeSystem deffines a set of Concepts from which a ValueSet provides possible values for</td>
<td>CodeSystems are saved on the Terminology Server.</td>
</tr>
<tr>
<td></td>
<td>a resource element. The actual <code>binding</code> between CodeSystem and element is done by the ValueSet.</td>
<td></td>
</tr>
<tr>
<td>ValueSet explorer</td>
<td>Lets you view existing ValueSets. The builder works best with SNOMED (at the moment).</td>
<td>ValueSets are stored on the Terminology Server.</td>
</tr>
</tbody>
</table>

**Current servers**

<table>
<thead>
<tr>
<th>Server</th>
<th>Address</th>
<th>Edit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Server</td>
<td>HSPC-4</td>
<td></td>
</tr>
<tr>
<td>Conformance Server</td>
<td>Grahames R4 server</td>
<td></td>
</tr>
<tr>
<td>Terminology Server</td>
<td>OntoserverR4 (terminology)</td>
<td></td>
</tr>
</tbody>
</table>

**Add Server**

- Set the same as the Data Server

**FHIR Links (open in new tab)**

- R4 Specification (current)
- STU-3 Specification
- STU-2 Specification
- FHIR Chat
- FHIR Log
- Clinicians Workshop

**FHIR Videos (open in new tab)**

- Scenario Builder
- Adding structured data
- Logical Modeler
- Logical Modeler and Scenario Builder
- RESTful query tool

**Notes:**

- Some of these videos may describe earlier versions, so they may not completely match the current functionality.
Modules

ClinFHIR Modules
- Patient Viewer
- Server Query
- Scenario Builder
- Logical Modeler
- Implementation Guide Browser
- Extension Definition Builder
- Code System Builder
- Value Set Explorer

Each opens in new tab
Server Selection
Server Selection

Setting FHIR Servers

- **Data Server**
  *Where your information is stored*

- **Conformance Server**
  *Enforces conformance to FHIR spec*

- **Terminology Server**
  *Vocabulary source*

Follow recommendations during tutorial
## Useful Links

<table>
<thead>
<tr>
<th>Useful FHIR Links</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>R4 Specification</strong></td>
</tr>
<tr>
<td><strong>STU-2 Specification</strong></td>
</tr>
<tr>
<td>FHIR wiki</td>
</tr>
<tr>
<td>Hay on FHIR</td>
</tr>
<tr>
<td>FHIR Chat</td>
</tr>
<tr>
<td>FHIR.org</td>
</tr>
<tr>
<td>Clinicians Workshop</td>
</tr>
</tbody>
</table>

Subject to change
ClinFHIR Video Demos

ClinFHIR videos

Scenario Builder
Adding structured data
Logical Modeller
Logical Modeller and Scenario Builder
RESTful query tool

Subject to version variations
Terminology Links

Other Links
- SNOMED browser
- SHRIMP (Terminology browser)

Subject to version variations
Task – Recap

• Open clinfhir.com
• Set up an account – DO NOT REUSE AN OLD PW
• Set up your servers –
  • Data Server – HSPC-4
  • Conformance – Grahames R4 Server
  • Terminology – OntoserverR4 (Terminology)
Step 1 - Create A User Account

- **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.

- **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.

- **Logical Modeler**
  - The Logical modeler allows the creation of a model that represents a particular interoperability requirement in a format that is easy to use. It uses FHIR data types, and can be based on an existing resource type or completely 'ad hoc'. It is intended to act as a bridge between Modeler and User, and can act as the basis for the generation of the profiling components required by FHIR.

- **Implementation Guide Browser**
  - Display the contents of an Implementation Guide, and the relationships between the contents of the Guides.

- **Bundle Visualizer**
  - Various displays for the contents of a bundle. Bundle can be parsed into the viewer and optionally saved in the data server.

- **Extension Definition builder**
  - Views and builds extension definitions. These can be defined and applied to the Logical Model, which will allow them to be included in the generated Profile.

- **CodeSystem builder**
  - The CodeSystem defines a set of Concepts from which a ValueSet provides possible values for a resource element. The actual 'binding' between CodeSystem and element is done by the ValueSet. This component allows you to build (and edit) a CodeSystem, and optionally builds the ValueSet as well.

- **ValueSet explorer**
  - Lets you view existing ValueSets. The builder works best with SNOMED (at the moment).

**FHIR Links**
- R4 Specification (current)
- STU-3 Specification
- SNOMED
- FHIR Chat
- FHIR.org
- FHIR wiki
- Clinicians Workshop

**Other links**
- SNOMED browser
- SNOMEDmp (Terminology browser)
Create User Account

WARNING!!
This is an UNSECURED server! Use a dummy password!

When you login the first time this becomes your user name and password. You may need to pick a different user name if its already in use.
Step 2 - Server Selection
Open Patient Viewer module
<table>
<thead>
<tr>
<th>Main modules (open in new tab)</th>
<th>Experimental modules (open in new tab)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Patient Viewer</strong></td>
<td>Display resources for a specific patient, using a number of different views such as a list by resource type, json &amp; 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.</td>
</tr>
<tr>
<td><strong>Server Query</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Scenario Builder</strong></td>
<td>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.</td>
</tr>
</tbody>
</table>
Patient Viewer

Please select a patient using the 'Select Patient' button at the upper right.

If you want to add a new patient, then click the 'Select Patient' button, and in the modal dialog that appears, there's a link to add a new patient.
Selecting a Patient

Search for Patient

1. Enter the patient's name (e.g., betsy)
2. Search for patient using the entered name or identifier
3. View the patient details (e.g., Betsy Johnson, female, 1957-11-07)
Search for Patient

betsy

Enter id of patient on this server

Enter identifier of patient

Betsy Johnson female 1957-11-07

Add new patient
Task – Select and View Patient

• Go to Patient Viewer Module
• Enter “Betsy” in Patient Search
• Select “Betsy Johnson” patient
• Explore patient’s FHIR resources
Patient Viewer – Resource Explorer

1. Resource Types
2. Condition resources
3. Type 2 diabetes
4. Versions
5. Condition

- resourceType: Condition
- id: co-cond-betsy-diabetes
- meta:
  - lastUpdated: 2019-08-06T18:52:16Z
- category:
  - text: Condition
- category:
  - text: Condition
- category:
  - text: Condition
- category:
  - text: Condition
- category:
  - text: Condition
Patient Viewer – Resource Reference Graph

Scroll to zoom graph
Click and drag to move

Note that the Patient is not displayed in this graph.
Click on a resource in the diagram. See the details on the right.
Task – Select and View Patient

• Go to Patient Viewer Module
• Enter “Betsy” in Patient Search
• Select “Betsy Johnson” patient
• Explore patient’s FHIR resources
• Create a sample patient and view the data
Patient Viewer

Please select a patient using the ‘Select Patient’ button at the upper right.

If you want to add a new patient, then click the ‘Select Patient’ button, and in the modal dialog that appears, there's a link to add a new patient.
Task – Add new patient

Search for Patient

- Search
- Load
- Search

Betsy Johnson female 1957-11-07

Add new patient
Create Basic Set of Resources

Add new Patient

Identifier: Identifier should be unique
First Name: Hanley
Last Name: Strappman
Date of Birth: 2019-08-01
Age: 2 weeks
Gender: Male

Generate samples

Add patient

Find existing patient
Add new Patient

Progress...
Adding Hanley Strappman
Added patient with the id: 144872
Checking that the required reference resources exist
adding Conditions...
Added Conditions List
adding Encounters...
added encounters Added 10 Encounters
Added 30 Observations
Added Medications List
Added 2 Appointments
Added Allergies List

All resources have been created. Click the close button to return to the front page
You can review the resource instances that were created using the 'Details' link at the upper left on the screen.

Close
Add new Patient

Progress...
Adding Hanley Strappman
Added patient with the id: 144716
Checking that the required reference resources exist
adding Conditions...
Added Conditions List
adding Encounters...
added encounters Added 10 Encounters
Added 30 Observations
Added Medications List
Added 2 Appointments
Added Allergies List

All resources have been created. Click the close button to return.
You can review the resource instances that were created using

Patient Viewer

Hanley Strappman male 1955-08-22 (144872)

Resource explorer

Resource Types

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllergyIntolerance</td>
<td>2</td>
</tr>
<tr>
<td>Appointment</td>
<td>2</td>
</tr>
<tr>
<td>Condition</td>
<td>11</td>
</tr>
<tr>
<td>Encounter</td>
<td>10</td>
</tr>
<tr>
<td>List</td>
<td>3</td>
</tr>
<tr>
<td>MedicationStatement</td>
<td>19</td>
</tr>
<tr>
<td>Observation</td>
<td>30</td>
</tr>
<tr>
<td>Patient</td>
<td>1</td>
</tr>
<tr>
<td>Practitioner</td>
<td>2</td>
</tr>
</tbody>
</table>

Show Patient Json
### Condition resources

<table>
<thead>
<tr>
<th>Condition</th>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allergy/Intolerance</td>
<td></td>
</tr>
<tr>
<td>Appointment</td>
<td></td>
</tr>
<tr>
<td>Condition</td>
<td>rheumatoid arthritis - both hands</td>
</tr>
<tr>
<td></td>
<td>depression</td>
</tr>
<tr>
<td></td>
<td>neuropathic pain</td>
</tr>
<tr>
<td></td>
<td>high cholesterol</td>
</tr>
<tr>
<td></td>
<td>onychomycosis</td>
</tr>
<tr>
<td></td>
<td>rheumatoid arthritis - left elbow</td>
</tr>
<tr>
<td></td>
<td>GERD</td>
</tr>
<tr>
<td></td>
<td>diabetes</td>
</tr>
<tr>
<td></td>
<td>hypertension</td>
</tr>
<tr>
<td></td>
<td>angina</td>
</tr>
<tr>
<td></td>
<td>asthma</td>
</tr>
</tbody>
</table>

### Outward references

- Condition subject: Patient/144872
- Hanley Strappman

### Inward references

- List/144873. item Problem list
- Encounter/144889. condition Aug 20th 2019 for sore throat
- Encounter/144893. condition Jul 2nd 2019 for sore throat
<Condition xmlns="http://hl7.org/fhir">
  <id value="144876"/>
  <meta>
    <versionId value="1"/>
    <lastUpdated value="2019-08-22T06:08:30.406+00:00"/>
    <source value="#72d0283b05a59907"/>
  </meta>
  <text>
    <status value="generated"/>
    <div xmlns="http://www.w3.org/1999/xhtml">Hypertension</div>
  </text>
  <category>
    <coding>
      <system value="http://snomed.info/sct"/>
      <code value="14657089"/>
      <display value="Diagnosis"/>
    </coding>
    <text value="Diagnosis"/>
  </category>
  <severity>
    <coding>
      <system value="http://snomed.info/ct"/>
      <code value="255084002"/>
      <display value="Mild"/>
    </coding>
    <text value="Mild"/>
  </severity>
  <code>
    <coding>
      <system value="http://snomed.info/ct"/>
      <code value="38341003"/>
      <display value="hypertension"/>
    </coding>
  </code>
</Condition>
That was the Patient Viewer Module...

Now let’s look at the Scenario Builder Module
Scenario Builder Module

Main modules (open in new tab)

- 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.
- 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. FHIRPath is supported.
- Logical Modeler: The Logical Modeler allows the creation of a model that represents a particular Interoperability requirement in a manner that is easy to use. It uses FHIR datatypes, and can be based on an existing resource type or completely custom. It is intended to act as a 'bridge' between Modeler and User, and can act as the basis for the generation of the profiling components required by FHIR.
- Implementation Guide Browser: Various displays for the contents of a bundle. Bundles can be pasted into the viewer and optionally saved in the data server.
- Bundle Visualizer: Various displays for the contents of a bundle. Bundles can be pasted into the viewer.
- Extension Definition Builder: Views and builds extension definitions. These can be defined and applied to the Logical Model, which will allow them to be included in the generated Profile.
- CodeSystem Builder: The CodeSystem defines a set of Concepts from which a ValueSet provides possible values for a resource element. The actual 'binding' between CodeSystem and element is done by the ValueSet. This component allows you to build (and edit) a CodeSystem, and optionally builds the ValueSet as well.
- ValueSet Explorer: Lets you view existing ValueSets. The builder works best with SNOMED (at the moment).

Experimental modules (open in new tab)

- Patient Viewer: Patient resources are stored on the Data Server. The server should support the Patient/$everything operation.
- Server Query: Can access any compliant FHIR server (must expose a Capability Statement).
- Implementation Guide Browser: The Implementation guide, profiles and Extension Definitions are on the Conformance Server, the terminology resources (eg ValueSet) are on the Terminology Server.
- Bundle Visualizer: Various displays for the contents of a bundle. Bundles can be pasted into the viewer and optionally saved in the data server.
- Extension Definition Builder: Extension definitions are saved on the Conformance Server.
- ValueSet Explorer: ValueSets are stored on the Terminology Server.

Other links

- SNOMED browser
- SHRIMP (Terminology browser)

FHIR Links (open in new tab)

- R4 Specification (current)
- STU-3 Specification
- STU-2 Specification
- FHIR wiki

clinFHIR Videos (open in new tab)

- Scenario Builder
- Adding structured data
- Logical Modeler
- 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.
The Scenario Builder

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.

Logical Modeller

The Logical modeller allows the creation of a model that represents a particular aspect of the clinical data. It can use patterns and rules that can be used to create models. Patient information is on the Data Server. Profiles on the Conformance server. ValueSets on the Terminology server.

Statement)

Create a simple scenario
Adding structured data to a scenario
Create a Document

Helpful Blogs by the developer
Links to the blogs direct from this deck.....

Creating FHIR Resource INSTANCES

https://fhirblog.com/creating-a-simple-scenario/

Adding Structured Data

https://fhirblog.com/adding-structured-data-to-a-scenario/

Time Check – set working time....
Create New Scenario

To edit a scenario, you can either:
- Click the 'New Scenario' link to the left to create a new Set
- Select the Library of scenarios ('View Library' link to the upper right) and download one to edit or view

TRY THIS!
Create new scenario

Scenario Details  Patient  Implementation Guide

Name
Clinicians on FHIR Tutorial

Description

Category
Default
Note: Scenarios are saved locally and need to be posted to a server in order to be shared. Stay tuned!
Task – Create a Patient (15 min)

• Use Scenario Builder module to create your own patient
  • Include a name, gender and birthdate
  • Record patient name and id on your note to find it later
• Validate your resource instance
• Update (POST) the resource to the data server
• Confirm that your patient is on the data server using the Patient Viewer module and the patient id
Clinicians on FHIR Tutorial

Type: Patient
Text: Hanley Strappman

Structure & Reference

- **Patient**
  - **identifier** *
  - **active** *
  - **name** *
    - **telecom** *
    - **gender** *
    - **birthDate** *
    - **deceased**
    - **address** *
    - **maritalStatus** *
    - **multipleBirth**
    - **photo** *
    - **contact** *
    - **communication** *
    - **generalPractitioner** *
    - **managingOrganization** *
    - **link** *

Current resource views

- Hanley Strappman

Changes

- **Patient.name**
  - **DataType(s) (click to add data)**
    - **HumanName**
      - A name associated with the individual.
Add HumanName property to Patient.name

Use: Usual

First Name: Hanley
Middle: Middle
Last Name: Strappman
Suffix: 

Result: Hanley Strappman
1. Select data element
2. Look at value set
3. Validate resource
Validate Resource

1. Confirm valid resource
2. Validate all resources & Post resource to server
3. Resource id of resource on the server
Create FHIR Condition Instance
Task – Create a Condition (15 min)

- Use Scenario Builder module to create your own condition
  - SNOMED Code and link to your patient
  - E.g. Diabetes mellitus type 2 (SCTID 44054006)
- Validate your resource instance
- Update (POST) the resource to the data server
- Confirm that your patient is on the data server using the Patient Viewer module and the patient id
- Add additional conditions if you are done early
Add CodeableConcept property to Condition.category

- [ ] Encounter Diagnosis
- [x] Problem List Item
<table>
<thead>
<tr>
<th>Type</th>
<th>Text</th>
<th>Valid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition</td>
<td>Hypertension</td>
<td></td>
</tr>
<tr>
<td>Patient</td>
<td>Hanley Strappman</td>
<td></td>
</tr>
</tbody>
</table>

**Structure & Reference**

```
<table>
<thead>
<tr>
<th>code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

**Condition.code**

```

DataTypes (click to add data):
- CodeableConcept

ValueSet Binding (example)

Identification of the condition, problem or diagnosis.
## SNOMED Coded Condition Concept

### Value Set Browser:

<table>
<thead>
<tr>
<th>Code</th>
<th>System</th>
<th>Display</th>
</tr>
</thead>
<tbody>
<tr>
<td>38341003</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Hypertension</td>
</tr>
<tr>
<td>10725009</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Benign hypertension</td>
</tr>
<tr>
<td>48146000</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Diastolic hypertension</td>
</tr>
<tr>
<td>59997006</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Endocrine hypertension</td>
</tr>
<tr>
<td>59621000</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Essential hypertension</td>
</tr>
<tr>
<td>429198000</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Exertional hypertension</td>
</tr>
<tr>
<td>123800009</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Goldblatt hypertension</td>
</tr>
<tr>
<td>162659009</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Hypertension resolved</td>
</tr>
<tr>
<td>697929007</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Intermittent hypertension</td>
</tr>
<tr>
<td>70272006</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Malignant hypertension</td>
</tr>
<tr>
<td>288250001</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Maternal hypertension</td>
</tr>
<tr>
<td>204650001</td>
<td><a href="http://snomed.info/sct">http://snomed.info/sct</a></td>
<td>Neonatal hypertension</td>
</tr>
</tbody>
</table>
Enter search term

Add CodeableConcept property to Condition.code

- Ben
- Bends
- Does bend
- Penis bent
- Cutaneous bends
- Able to bend
- Benoze syndrome
- Bend arm position
- Benign haematuria
- **Benign hypertension**
  - Benign meningioma
  - Benign paraproteinaemia
  - Benign pemphigus
  - Benign phaeochromocytoma
  - Benign pneumoconiosis
  - Bent bone dysplasia
  - Congenital bent clavicle
  - Congenital bent humerus
  - Congenital bent ilium
  - Congenital bent ischium
  - Congenital bent nose
  - Congenital bent pubis

http://hl7.org/fhir/ValueSet/condition-code
Add CodeableConcept property to Condition.code

Benign hypertension

http://snomed.info/sct  10725009  Benign hypertension

Text value if not coded

http://hl7.org/fhir/ValueSet/condition-code
1. Select **subject[x]** data element
2. Appropriate data types
3. Available datatypes in this scenario
Note: ClinFHIR automatically links new resources to existing patient resource when appropriate
View Resource Graph of Scenario
Task – Add conditions to Problem List

• Use Scenario Builder module to create another condition
  • SNOMED Code and link to your patient for
  • Asthma
  • Diabetes
Bonus Round – Even if you don't get here, you can still play the game at home.
Task - Create other resources

- Add an observation (or two)
  - HgA1c
  - Blood Pressure
- Add a Medication** *(see next slide)*
- Add a Practitioner resource
- Add an Encounter resource
- Add a List resource
Medications According to FHIR....

- **Medication**
  - Primarily used for the identification and definition of a medication for the purposes of prescribing, dispensing, and administering a medication as well as for making statements about medication use.

- **MedicationAdministration**
  - Describes the event of a patient consuming or otherwise being administered a medication.

- **MedicationDispense**
  - Indicates that a medication product is to be or has been dispensed for a named person/patient. The medication dispense is the result of a pharmacy system responding to a medication order.

- **MedicationKnowledge**
  - Information about a medication that is used to support knowledge.

- **MedicationRequest**
  - An order or request for both supply of the medication and the instructions for administration of the medication to a patient.

- **MedicationUsage**
  - A record of a medication that is being consumed by a patient.
Task – Create an Observation (15min)

• Use Scenario Builder module to create your own Observation
  • Try Fasting Blood Glucose, Height, Weight and BMI
• Use FHIR Spec, Google or LOINC Browser to find LOINC concept code
• Validate your resource instance
• Update (POST) the resource to the data server
• Confirm that your patient is on the data server using the Patient Viewer module and the patient id
• Additional task – add additional observations
Add HgbA1c Observation
Use Value Set Expand to find code
Add CodeableConcept property to Observation.code

- Hemoglobin A1c in Blood
- http://loinc.org
- 55454-3
- Hemoglobin A1c in Blood

Text value if not coded
<table>
<thead>
<tr>
<th>Type</th>
<th>Text</th>
<th>Valid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition</td>
<td>Hypertension</td>
<td></td>
</tr>
<tr>
<td>Observation</td>
<td>Hanley Strappman</td>
<td></td>
</tr>
</tbody>
</table>

Current Value
- Hemoglobin A1c in Blood

Data type: code

Describes what was observed. Sometimes this is called the observation "name".
Add a value to the observation
Add Quantity property to Observation.valueQuantity

Value
7.0

System
Enter units as string (e.g., mmHg)
Validate and POST Observation
Did you make it this far?

Congratulations!

MISSION COMPLETE