

Hay on FHIR

Thoughts and comments on the emerging HL7 FHIR standard, from a FHIR Evangelist

clinFHIR Resource Builder: version 2

APRIL 24, 2016 [1 COMMENT \(HTTPS://FHIRBLOG.COM/2016/04/24/CLINFHIR-RESOURCE-BUILDER-VERSION-2/#COMMENTS\)](https://fhirblog.com/2016/04/24/clinFHIR-RESOURCE-BUILDER-VERSION-2/#COMMENTS)

In the past few months I've been working on the [clinFHIR \(http://clinfhir.com/\)](http://clinfhir.com/) resource builder that we use at the 'Clinicians on FHIR' events at the Working Group Meetings.

In fact, this really became a complete re-write of the application, as there were a number of issues that I just couldn't fix in the current version – especially around the rendering and creation of extensions. This has resulted in some changes to the User Interface, so the purpose of this post is to describe those changes when creating and viewing resources. It's not quite complete, but ready for people to give it some user testing.

The app itself can be accessed from the main [clinFHIR.com \(http://clinfhir.com/\)](http://clinfhir.com/) site – just click on the 'resource builder' link. (Note that the app is hosted on a different server at the moment- I'll move that later, though the link will always be up to date).

Once invoked, the 'front page' of the app will be displayed – it looks like this:

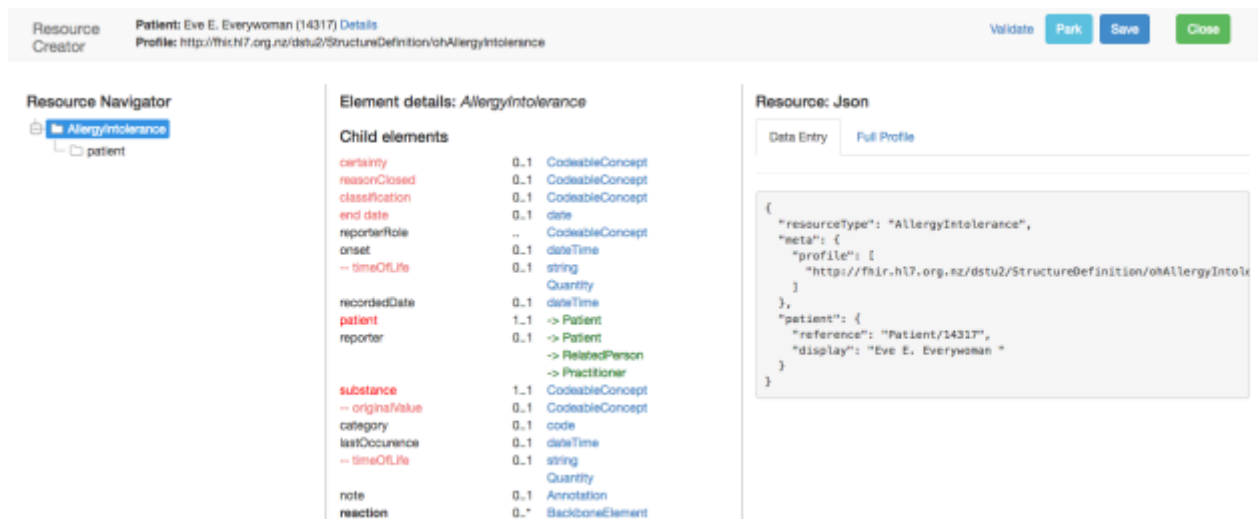
The screenshot shows the 'Resource Creator' interface. At the top, it displays 'Patient: Eve E. Everywoman (14317) Details' and 'Profile: http://hl7.org.au/StructureDefinition/ohAllergyIntolerance'. A 'New Resource' button and a settings icon are on the right. The main area is divided into three panes: 'My recent Patients' with a 'Find Patient' button and a list of patients including 'Eve E. Everywoman'; 'My recent Profiles' with a 'Show Upload' and 'Find Profile' button, and a list of profiles including 'http://hl7.org.au/StructureDefinition/ohAllergyIntolerance'; and 'System Configuration' with a 'Terminology Server' section showing two options: 'http://hl7.org.au/StructureDefinition/' (selected) and 'http://hl7.org.au/StructureDefinition/Condition'.

The idea is that you select the patient in the left pane, the profile you wish to use to build the resource against in the middle pane, and terminology server on the right – though that will generally default to the best choice. It is quite possible to use different servers for patient (plus the resources for that patient) and the profile on different servers, so long as they are on the same STU version. You'll be warned if you select from different versions, though the tool won't prevent you from proceeding. In the example above, the HAPI server is being used for patient and data, the HL7 NZ to access profiles, and Grahames server for terminology.

Both panes work in the same way – you select the server from the drop down (currently there’s a hard coded list) and then a patient or a profile from that server. (There’s also a ‘test’ button that allows you to make sure that the server you select is actually working – these are test servers, so no guarantees). The tool will remember your previous choices, so you can easily re-select them – these are saved in the local browser cache, and are not user specific. (To clear the local cache – which you have to do if the underlying profile is changed – click the ‘gear’ icon to the upper right and select the clear cache option).

Patient selection is by name (I’ll add the ability to create a new patient shortly) and Profile selection uses the same dialog as previously – you select the resource type you want, and then either that type directly or a profile on that type.

Having selected Patient and Profile, a button labeled ‘New Resource’ will appear in the top right of the screen. Clicking on that button will load the actual resource builder UI. Here’s a screen shot:



This screen also has 3 panes.

- To the left side is a tree view that acts as a navigator to the resource. As you add elements to the resource, they will be added to the tree.
- In the middle pane is a display that will show the value of the currently selected element, and all of the ‘child’ elements of that element. When the resource is first loaded, the root of the resource is selected in the navigator, and the middle pane will show the direct children of the root.
- In the right hand side is the pane where the json view of the resource is displayed, and where the data entry screen for an element is displayed when an element datatype is selected in the middle pane. This is the same as the previous version.

To add an element to the resource, first select the ‘parent’ you want to add it to in the navigator, and then select the element itself in the middle pane. The display in the middle is very similar to the previous version, with one big exception. If the datatype is ‘BackboneElement’ (like the reaction in the example above) then it is a node that can have children, rather than a value. Clicking on one of these will add a node to the navigator and then display the possible children in the middle. Multiple child elements can be added in this way (if the underlying profile permits).

Any other datatype will result in a data entry form being displayed in the upper right hand side – exactly the same as in the previous version.

Note that extensions are displayed 'in-line' with the other elements (extensions are normal in FHIR remember) with only a different colour to distinguish them. 'Certainty' is an example above. If the extension has a couple of 'minus' signs (-) to the left, then it is an extension against the element directly above it rather than against the resource root. Note that at the moment only 'simple' extensions are supported – not complex ones containing 'child' extensions.

So you build your resource in this way:

1. Select parent node in the navigator
2. Select element from the middle pane
3. Enter data in right pane
4. And repeat

You can also now delete an element by selecting it in the navigator, then clicking the 'delete' button in the middle pane.

The parking function still also works – though you do need to be in the 'build' screen with the patient selected before you invoke it.

To save your resource (which will go to the same server as you selected the patient from), there is a 'Save' button to the upper right of the navbar. Alongside the save button is a link labeled 'validate'. This is handy, as it will tell you if the resource is valid – ie if it will be accepted by the server (saves disappointment later 😊). This link actually used the validate operation (<http://hl7.org/fhir/resource-operations.html#validate>) on the server – it sends a request like:

```
POST [server]/[resource type]/$validate
```

Once saved, the resource can be viewed in the Resource Viewer – which you access by clicking the 'details' link next to the Patients name on the left side of the navbar. This is similar to the screen in the previous version – the only real difference being that the inward and outwards links are in a column to the right of the display rather than either side (there is less room to play with). Here's a screen shot:

The screenshot shows the FHIR Resource Viewer interface. At the top, the patient's name 'Eve E. Everywoman (14317)' is displayed with a 'Details' link. Below this, the 'Resource Types' section lists 'AllergyIntolerance', 'CarePlan', 'Encounter', 'Patient', and 'Procedure'. The 'Procedure resources' section shows 'Asthma management Asthma management [406162001]' and 'Appendectomy Appendectomy [80146002]'. The 'Procedure/21635' section displays the JSON representation of the resource. The 'Outward references' section shows the 'Procedure subject' as 'Patient/14317' with details for Eve E. EVERYWOMAN. The 'Inward references' section is currently empty.

```
{
  "resourceType": "Procedure",
  "id": "21635",
  "meta": {
    "versionId": "1",
    "lastUpdated": "2016-04-11T22:02:55.939-04:00",
    "profile": [
      "http://fhir.hl7.org.nz/dstu2/StructureDefinition/ohProc"
    ]
  },
  "subject": {
    "reference": "Patient/14317",
    "display": "Eve E. Everywoman "
  },
  "status": "in-progress",
  "code": {
    "coding": [
      {
        "system": "http://snomed.info/sct",
        "code": "80146002",
        "display": "Appendectomy"
      }
    ],
    "text": "Appendectomy"
  }
}
```

To return to the builder screen click the 'New Resource' button. The 'Front' link returns to the front page, though you'll lose any new resource you may be building.

So that's a quick overview of the new builder – it still has some rough edges and bugs that I'll address between now and the Working Group Meeting, but do have a play with it and let me know of any errors you find. You can email me directly – david.hay25 at gmail.com – but a better choice might be the stream in FHIR Chat (<https://chat.fhir.org/#narrow/stream/clinFHIR>) (where all the cool kids hang out 😊).

There are some other capabilities of the new tool that I'll describe in some upcoming posts – notably viewing a profile and ad-hoc querying of a FHIR server.

I'm also planning a post describing how to create and upload profiles using the community tools – forge (<http://fhir.furore.com/Forge>) and clinFHIR extension builder.

cheers...

FILED UNDER CLINFHIR

About David Hay

I'm a Product Strategist at Orion Health, Chair of HL7 New Zealand and co-Chair of the FHIR Management Group. I have a keen interest in health IT, especially health interoperability with HL7 and the new FHIR standard.

One Response to *clinFHIR Resource Builder: version 2*

Pingback: [Using clinFHIR for profiling | Hay on FHIR](#)

Blog at WordPress.com.