FHIR/RDF Content Negotiation

Version 2

Issues

1. Existing tools such as Protege need to retrieve:
    - FHIR Resources/queries (as OWL)
    - FHIR Code Systems/value sets (as OWL)
2. FHIR RDF services should be able to:
    - Support multiple RDF formats
    - Differentiate requests for FHIR/RDF from OWL
        - ~~Not terribly important on for resources/queries unless someone decides to care deeply…~~
        - V*ery* important for code systems and (?) value sets

Recommendation 1

We don’t need to separate OWL Resource instances from non-OWL instances

- There will be a single file w/ OWL ontology declaration.

Goals

1. Enhance FHIR RDF specification to define what an “RDF compliant” FHIR service must/may/must not do — so that everyone knows what to do

2. Create a set of tests that validate service conformance — so that everyone can know whether they’ve done it right

Mime Types (according to FHIR spec)

- JSON - “application/fhir+json”
- XML - “application/fhir+xml”
- RDF*
  - “text/turtle” - for RDF Turtle format
  - “application/json-ld” - for RDF json-ld format (does this exist?)
- ShEx*
  - “text/shex” for ShEx schema

* **http://build.fhir.org/rdf.html**

URL fragment _format=[mime-type]

http://build.fhir.org/http.html

- XML
    - “xml”, “text/xml”, “application/xml”, “application/fhir+xml”
- JSON
    - “json”, “application/json”, “application/fhir+json”
- RDF (Turtle format)
    - “ttl”, “text/ttl” — (interestingly, only “text/turtle” works on Graham’s server)
        - http://test.fhir.org/r3/Patient/f201?_format=text/turtle
- HTML
    - “html”, “text+html”

Gaps

“***Note: the _format parameter does not override the content-type header***”
(from web page)

- Not sure whether this is strictly content-type or? but…
- It seems fairly obvious that _format should override, as, otherwise the only thing you would ever get from a browser would be html

Recommendation: _format in the URL overrides the accept header. “?_format=xyz” == Accept: xyz; q=1.0

“_format=” in spec doesn’t match service behavior

Recommendation: Make “must” / “may” and “must not” clear in spec and add conformance testing tool.

Recommendation 2

_**format** in the URL overrides the accept header.

- Documentation on http://build.fhir.org/http.html will be updated and clarified

- “must” / “may” and “must not” will be made clear in the spec wrt. *_format=ttl, format=text/ttl, …*

- rules will be added to a conformance testing tool.

Gaps (continued)

Redirects

- Current HL7 server strips “_format=”
    - http://hl7.org/fhir/Patient/foo1?_format=text/turtle —> http://hl7.org/fhir/patient-example-f001.pieter.html
        - (Means that URLs w/ _format don’t work in Protege)
- Current HL7 server behaves oddly on accept header
    - Accept: text/turtle;q=1.0 (**works on HL7, not on test.fhir.org)**
    - Accept: text/turtle;q=0.9,text/html,application/xhtml+xml,application/xml;q=0.8,*/*;q=0.8 (**does not work)**

Recommendation: Document how this should work (I’m not sure myself…). Make “must” / “may” and “must not” clear in spec and add conformance testing tool.

Recommendation 3

_format redirect rules will be made explicit

- _format must be included in URL rewrite
- rules will be added to a conformance testing tool.

Gaps (continued)

Recommendation: Add requirement that servers recognize application/rdf+xml as a request for FHIR/OWL (can't do the other two (application/xml and text/xml) because those are used for FHIR/XML format

Gaps (continued)

RDF - Many different RDF formats possible:

- Turtle: text/turtle
- XML: application/rdf+xml
- XML “Pretty”: ???
- ntriples: ??? not registered,but probably text/ntriples
- n3: text/n3

Recommendation: All FHIR (RDF) servers must support Turtle, but here is an (open) list of the known alternatives (because they aren’t obvious) that servers may support.

Recommendations

Semantics vs. format

1. FHIR Resource as RDF — w/o ontology header if needed
    - text/turtle, text/n3 (Maybe can’t get xml but so what?)
2. FHIR Resource as OWL — w/ ontology header
    - application/rdf+xml or any of the OWL 2 mime types
    - If w/o header not needed, then merge
3. FHIR *Terminology* Resource as RDF - w/o ontology header
    - (same as 1)
    - http://hl7.org/fhir/CodeSystem/AccountStatus?_format=text/turtle
4. FHIR *Terminology* Resource as OWL - w/ ontology header
    - (same as 2
5. FHIR OWL Terminology (as OWL)
    - application/rdf+xml or any of the OWL 2 mime types
    - What format must all servers support? (Prefer owlF)
    - http://hl7.org/fhir/CodeSystem/AccountStatus/owl?_format=application/owl+functional

Recommendation 4

Where there *is* a difference between FHIR format and OWL format (i.e. code system and (?) value set):

- http://<server>/CodeSystem/account-status?_format=text/turtle — returns FHIR representation (fhir:concept [fhir:code [fhir:value “active”]]
    - http://<server>/CodeSystem/account-status/version/1.0
- http://<server>/CodeSystem/account-status/owl?_format=text/turtle — returns OWL representation
    - OWL and RDF mime type support will be described (Turtle “must”, rest “may”)
    - http://<server>/CodeSystem/account-status/owl/version/1.0

One more thing…

Ontology header today:

<http://test.fhir.org/r3/Patient/f201.ttl> a owl:ontology ;
owl:imports fhir:fhir.ttl ;
owl:versionIRI <http://test.fhir.org/r3/Patient/f201.ttl> .

Recommendation:

<http://test.fhir.org/r3/Patient/f201/owl> a owl:ontology ;
owl:imports fhir:fhir.ttl ;
owl:versionIRI <http://test.fhir.org/r3/Patient/f201/owl/version/1> .

URI should be *logical* URI — same name as in the FHIR resources

versionIRI should match FHIR versioning spec, and be present *only* when version is present.

Gaps (continued)

Existing OWL clients use OWL 1.0 Mime Recommendation

OWL Mime Types (Owl 1)

“The Web Ontology Working Group has not requested a separate MIME type for OWL documents. Instead, we recommend to use the MIME type requested by the RDF Core Working Group, namely `application/rdf+xml` [*RDF Concepts*], or alternatively the XML MIME type `application/xml`.”

https://www.w3.org/TR/2004/REC-owl-ref-20040210/#MIMEType

Protege uses: *application/rdf+xml, application/xml; q=0.5, text/xml; q=0.3, */*; q=0.2*

This seems to be a default header for “OWL in any form”

OWL Mime Types (OWL 2)

- Manchester Syntax: *text/owl-manchester*
- Functional Syntax: text/owl-functional
- OWL XML Syntax: application/owl+xml
- OWL RDF Syntax: application/rdf+xml (???)

Recommendation 5

Short term:

When an Accept header is formatted *exactly* as:

"application/rdf+xml, application/xml; q=0.5, text/xml; q=0.3, */*; q=0.2”

…and the request is for a FHIR Resource instance…

FHIR server should

a) opt for “*/*” and return “text/turtle”

b) Return rdf/xml representation of the resource, (that’ll teach them)

(This is the one that “breaks the internet”, as a FHIR server always has application/xml available, but it is in FHIR format…)

NOT RESOLVED — may want to use user-agent instead

Recommendation 5

Long term:

The correct header will be defined for OWL requests — Eric P is working on this.

- It will either be an Accept: (long list of possible OWL formats) and/or Accept-Profile: (URL for OWL)
- Protege and other reasoners will be updated to reflect this
- FHIR spec will be updated (if necessary) to support this