**More general semantics and conceptual input**

* A workshop participant (from the French National Archives?) rightfully pointed out that skos:close/exactMatch relations are actually valid only to connect SKOS concepts. But e.g. in GeoNames, a Place is not a subclass of Concept, so that might cause problems further down the line, at the latest when triplestores and reasoning are involved.

This is true – and could be solved once a Linked Pasts Ontology is developed if lpo:Place is a subclass of skos:Concept

* Likewise (a thing that we have in fact been discussing before I think): our inter-gazetteer links are currently represented in JSON-LD in an awkward fashion. Essentially a links resource, with a list of objects, each having a type and an identifier field. As we know, this doesn’t properly expand to the usual (Place)->(closeMatch)->(Place) triple structure. We ultimately might have to revise this.

Everything about JSON-LD is awkward! The current structure [expands as valid RDF according to the JSON-LD playground](http://tinyurl.com/y23ezgdf). The resulting N-Quads contain dozens of blank nodes...that’s what happens when you expand JSON-LD. I went over this in excruciating detail with Graham Klyne back when.

* Since skos relations don’t look ideal anyway (see first point), maybe it makes sense to have our own vocabulary for links. I.e.: i) give a proper name to the link object (perhaps call it analignment); ii) provide a set of possible alignment types, which each can have more detailed definitions; iii) added bonus: extra space for data about how the alignment was made (automated or not, creator information, matching scores, etc.)
* We don’t provide any recommendations on how and where to add license information to a gazetteer record. Another big issue, perhaps, but we probably should have some pattern we can recommend to gazetteer providers. So that not everyone needs to come up with their own flavor.

In other words, rework Linked Pasts format in large measure. I can’t do this and produce WHG. I would welcome someone taking this on, but this means somehow transforming early contributions to WHG to a new format. Truthfully, the prospect of the first several contributed datasets all lost motion is horrifying!

**LP detail questions**

* Does when support a timestamp field, with a single ISO timestamp? We had the frequent case where e.g. a geometry was taken from an external source, known to be valid at one particular time. But with no other information beyond that. A timestamp field would be good to express the semantics “All I know is this was valid at that time”.

Every allowed time syntax represents a commitment on the part of software developers supporting LP format to parse it. The situation described would be handled by having identical start and end.

{"start": {"in":"1600-04"}, "end":{"in":"1600-04"}}.

Using a when object constructed this way was a choice. There are others, particularly now that 8601-2 extensions are near finalization. One can’t make all choices simultaneously.

* What if there’s both a URI identifier, and a URL to a human-readable website/place page that’s different than the identifier. (For those providers who don’t have the resources for the full content negotiation thing.) Should the model allow for a homepage field?

Don’t understand. A URL is a URI. Contributors should use whatever permanent identifier they have. Articulating the difference between a landing page and data record is the sort of detail I thought we wanted to avoid.

* In relations, can relationTo point to an external URI? E.g. can a local community gazetteer just point to e.g. a Wikidata city record, if it doesn’t want to bother with modeling the city itself? Or, in other words, do we require gazetteers to use relationTo only for internal links? (My guess is: no?)

Sure it can. No, not only internal.

**Urban gazetteer-related issues**

(just want to break those out into a separate section, since urban gazetteers are all the rage right now!)

* The relationship examples currently have only getty:broaderPartitive as a relationship type. Carmen pointed out this: <http://observedchange.com/tisc/ns> (You may know this already, I guess. It’s by Tomi Kauppinen. Was new to me though.) It looks like a source of relationships we could recommend as good practice. Mind that this does not cover administrative association (!) but does have a comprehensive set of topological relations – including those needed for the next example…

New to me also. I assumed one can assert any sort of relation from a published vocabulary/ontology they want. The Getty broaderPartitive is what they use mostly for administrative area containment. Ruth has said the world needs a place-to-place relation vocabulary and I agree. This would include things like political alliances, and so on. Someone just needs to do it.

* Maybe my favourite puzzle from the workshop: Vincent has some great examples where a “micro-place”, e.g. a farm or mill, is inside (or part of) a parent place – e.g. a city. The city has a point coordinate. The micro-place does not (really) have its own defined coordinate. We could probably find it out, i.e. it’s not an “unlocated place” per se. But it’s just not that important. No-one might have bothered to identify exact coordinates. But maybe someone will, in the future. The way Vincent handles this now is to repeat the coordinate of the parent place, as the micro-place’s own geometry. But there’s no way to indicate that this coordinate is just a derivative, a shorthand approximation. I pointed out that, in this case, maybe the farm shouldn’t have a coordinate at all, but, instead, just be linked to the city via a relation. That’s already what Vincent does now in addition. I thought that maybe we should just repeat the city coordinate, for convenience, in the relation object instead (just like the label is also repeated). That did feel a bit cleaner, semantics-wise. However, the coordinate would then, obviously, not be renderable through GeoJSON, since it sits at a totally different location than where GeoJSON expects it. I.e. we ended up having no good idea how to best represent this. Basically having two options on the table: the current state (derivative coordinate, but not marked as derivative); or a different solution which feels semantically cleaner, but not achieving proper map rendering. Any ideas?

I have to ponder this. Bandwidth is extremely limited between now and the July beta launch for Pitt. And after that as well, unless I alter the planned deliverables.