**MARK’S SUGGESTIONS**

1) decide on a consistent capitalization of the predicates (or... use some kind of numerical identifier and make the predicate the label... that's best-practice, but I know what a pain in the ass it is to work with!!... still...)

2) has\_phenotypic\_process ---> has\_phenotypic\_consequence (there is no such thing as a "state process")

*We could change the property name, yes.*

3) I am a bit unsure about using rdfs:comment to carry "real information"... it might be OK, but it feels wrong to use annotation properties for real data... I’d be tempted to use http://purl.org/dc/terms/description instead...??

*So, creating a property called Comment/description and inserting the data there??*

4) Gene has\_function.... can you send me a couple of examples of this? Do they describe gene function using something other than the biological\_function ontology?

*Function: Trichodiene synthase*

*Function: G alpha protein subunit*

*Function: Pectate lyase*

5) Gene name --> please reuse the more widely-used http://purl.org/dc/terms/identifier... or if there are multiple identifiers (as there almost always are!) please model it by using a combination of SIO and EDAM - the identifier is a typed-attribute (type "EMBL\_Identifier" or type "UniProt Identifier" taken from the EDAM ontology) and has\_value "ABC123".

*Gene\_name are not identifiers (i.e. PELD, CPGB-1, GPA4, TOXF)*

6) typo? db\_xref instead of dbx\_ref? (you could re-use the corresponding predicate from the openLifeData vocabulary)

7) LocusID --> becomes redundant if you model (5) as an object rather than a string...

8) AA\_Sequence/NT\_Sequence predicates --> rather than inventing a predicate, it might make more sense to model this using the UniProt Core ontology... however that only works for the AminoAcid sequence (unfortunately, it looks like Jerven over-specified the range of his 'sequence' predicate so that it could ONLY be an AminoAcid Sequence! ...moreover, the Sequence itself is modeled as a class, not a string... which according to the rules of SIO is a better way to do it (and I agree! Everything should be a class until you have no choice... so the Sequence is an object, which is typed (again, using EDAM or SIO or SO or all three) and that class has\_value "AACGTTGATG..."). I have sent a message to Robert Hoendorf asking what predicate he would suggest we use here.

9) Accession --> becomes redundant if you model (5) as an object rather than a string

10) DB\_type --> this should also not be a string, but rather a Resource typed according to EDAM's list of databases (if you need to add a database to that list, just ask them!).... Identifiers.org also carries a list of databases that is synchronozed with EDAM's list, as far as I know...

11) w.r.t. all of the GMP stuff - please be sure to look carefully at the Sequence Ontology (SO) and its various extensions... I think much of what you want is in there.

12) Experimental evidence should be using classes from the Ontology of Biomedical Investigation or Experimental Factor Ontology (EFO)

13) I believe Strain (or something equivalent) is also within the EFO... and should NOT be a string! (there's a pattern here... NOTHING is a string! LOL! Strings are just values for some ontologically-typed resource... always!)

14) the taxonomy stuff should be re-thought following some taxonomic database (look at how the GBIF project is doing this...)