We've now had a chance to review your first set of artefacts, and unfortunately there are quite a few issues with all three templates.

There's a fairly detailed description of our findings below which hopefully will be clear, but we are obviously more than happy to provide further clarification either by email or via a phone call / GoToMeeting if required.

So here goes:

**General points:**

1. If you want to use create your own archetypes, do not increase the version number on an existing (especially published) archetype, but change the name to your own, e.g. body weight ET. Changing the version will cause clashes with the international and potentially local repositories where the same named but differently versioned archetype is published.
2. Creating your own archetypes unnecessarily (see below for specific examples) means that you lose interoperability, so you should think very carefully whether it’s required or not. Often existing archetypes can be easily constrained and adapted in the template.
3. Code mappings can be done in the template, so there is no need to build local archetypes just for this purpose.
4. If you are using existing archetypes as the basis for creating new ones, remember to change references and authorship accordingly.

**General template:**

1. Use international archetypes for weight and height and constrain the upper limit at template level. Using your own archetypes in this situation means that despite effectively identical data models, the data is completely non-interoperable with anyone using the published models.

**Virology template:**

1. Laboratory test:
   1. Lab test panel specialisations are redundant, because the data points are unchanged - there is just a constraint on the result value to limit to coded text only.
   2. It would be much more straightforward to create a single specialisation to constrain the result value to coded text and then clone that cluster in the template and rename it, adding analyse names as terminology codes.
   3. Whilst using specialisations protects interoperability, the maintenance overhead for having a separate cluster for every single test become unmanageable.
2. Immunisation Summary:
   1. The Immunisation Summary evaluation isn’t suitable for this. We’d suggest you create a local archetype to just capture the very basic data you need for this. Not quite sure why you need Hep B immunised Yes / No and also the status of ‘not commenced’ - what’s the difference between ‘No’ and ‘not commenced’?

**Abdominal template:**

1. **Each of the different body parts should be a separate observation, rather than a single observation with embedded clusters.**
2. **Body part examination**: This should not be a local archetype, but use the imaging result observation. We have just submitted a change request to align the imaging result pattern with the laboratory test pattern, i.e. have a slot for result(s) rather than internal clusters.
3. **Body examination panel:** This is redundant, the individual exam result clusters can be plugged directly into the result details cluster.
4. **Examination details cluster**: This is redundant - the date and examiner details come directly from the reference model.
5. The ‘examination of the abdomen specialisation’ is redundant, as it contains ONLY the single data point for ‘device’ which is actually the imaging modality which needs to be captured for each of the body part exams.
6. **Status** data point in all the clusters: This is not the right way to capture the fact that a particular ‘something’ was not investigated. I understand from the text in the ‘Use’ section of some of the archetypes that there is some idea of using this to drive the user interface logic, but that’s really not the role of the data model. There are two ways of managing ‘not investigated’ correctly (our preference would be the first):
   1. add the ‘not investigated’ as another value to the corresponding clinical valueset, e.g. Intra-hepatic bile duct values = normal, dilated, not assessable, not investigated. That would still allow you to explicitly capture that something was not investigated, and enable a separate UI rules engine to use the values in order to display (or not display) other data fields.
   2. use a null value flavour to indicate ‘not investigated’. The UI could display ‘not investigated’ as a choice, but in the data you just have a null value and not mix clinical values with a status value.
7. The ‘Lesion type' in the ‘space occupying lesion’ cluster should just be a single datapoint with a choice of text and coded text rather than two separate data points.