# 2013.05.28 HL7 Templates Workgroup Conference Call Minutes

## Chair: Mark Shafarman

##  Scribe: Mark Shafarman

## List of Attendees:

|  |  |  |
| --- | --- | --- |
| Name | Affiliation | email |
| Mark Shafarman | Shafarman Consulting Inc. | Mark.shafarman@earthlink.net |
| Kai Heitmann | HL7 Germany | kai@heitmann.de |
| Andy Stechishin | CANA Software & Services Ltd. | andy.stechishin@gmail.com |
| Sean McIlvenna | Lantana Systems | sean.mcilvenna@lantanagroup.com |
| Jane Curry | HL7 Canada | janecurry@healthinfostrategies.com |
| Lisa Nelson | Cox Systems | lisarnelson@cox.net |

Per last week’s conference call, we had a discussion on the versioning of templates. Kai presented (and we used) his present PowerPoint on templates versioning for discussion.

Note: in what follows, \*\* denotes follow-ups from today’s discussions.

Issues pointed out included the versioning requirements of various implementations, such as:

* Vocabulary bindings (static vs. dynamic, and if static: as of what date?)
* Publication dates vs. design dates
* Effective dates (from v3 perspective, the effective date supports an interval of dates from start to finish; from implementation point of view these may or may not be equivalent to “publication date”, or to “approval of this version of the template” (static) or “dynamic” (assume the latest current version)or “historical equivalent (i.e. some early implementation of IHE were “version unaware”)
* The relation of a set of versions (a family of versions of a template) with the status of the conceptual template, i.e. sometimes a template is never implemented (no version is implemented, although the template id was created), or the last version has one of several statuses determining whether the group of versions can be used for ‘historical’ validation, or ‘historical and computational (in other contexts) use’ or ‘should never have been used’ etc.
* The use of the v3 act.status values and the act.moodCode values of (DEF and EVN) to add information to the version of a template (details to still be worked out).
* In the IHE formalisms, one needs to know whether a particular version is in (or before or after) the public comment period, and/or whether it has been used at a particular Connectathon
* The question of whether a ‘validation package’ (of programs) should be able to validate a single version, or multiple versions of a template
* How to create/support “groups” of templates that can all have similar (or the same) status and implementation date.

We discussed several ways of conceptualizing and organizing these issues.

* In terms of the HL7 SAIF framework, the template (family and versions) can be divided into 3 conceptual points of view: governance, design and implementation. Each of the above bullet points may have relevance at one or more of these 3 levels.
* In ‘real-world’ development and implementation requirements, some of the above need to be available at a ‘template registry’ level, and some need to be available at the template repository level.
* From an implementer point of view, the implementer either must have (or be able to derive) all the information necessary to correctly implement a given template, or be able to query (a registry) to obtain that information.

For the next call, the discussion should continue to work on sorting out these issues. The group suggested that we get requirements from IHE (from Lisa), DECOR (from Kai), and Lantana (from Sean) to start a discussion of versioning use cases. Kai has agreed to integrate them into a document for discussion. We will also need to find out how these issues are being handled by MDHT, in Australia/New Zealand and by the UK/NHS, to complete this analysis.