Transition to HL7 Version 3.0 - A Path for Technical Committees

Introduction

In January of 1997, the Technical Committees (TCs) of HL7 will be completing their balloting on Version 2.3 and undertaking the transition to development of 3.0. This presents a multi-faceted challenge.

1. The Technical Committees must become comfortable with an begin using the Version 3 process. This process calls for a number of new deliverables including models to be developed. While the TCs have developed information models in the past, Use Case Models, Interaction Models, and Message Design Models are all new artifacts which will need to be defined.

2. It is essential that the intellectual content of HL7 Version 2.3, which has been painstakingly defined, be carried forward into the Version 3.0 standard.

3. Finally, the time available to Technical Committees to work on standard content is extremely limited, and it is essential to make the most effective use of the working group meetings.

This document describes a transition path for the HL7 Technical Committees as they undertake Version 3 development, and includes a strategy by which to move the existing chapter content to Version 3.0. The strategy uses the existing HL7 2.3 specifications to expeditiously define the new models that will be part of the 3.0 specification.

What we have & What we need

The existing HL7 Version 2.3 includes the following elements:

Trigger events which specify the actions or occurrences in the healthcare domain that cause messages to be exchanged;

Messages that specify the information content to be transferred at the occurrence of each trigger event;

Segments which are the repeating structures within each message and which provide for “reuse” of information content specifications between messages; and

Data elements which provide for a standardized representation of data content when it is exchanged.

Data types assigned to the various data elements.

In addition, HL7 has sponsored the creation of a Draft Reference Information Model (Draft RIM) for use by the TCs. The Draft RIM is an information model of the healthcare domain based upon the information content of the Version 2.3 standard. To the extent possible, this model was drawn from information models developed by each TC and submitted for harmonization. Although this model undoubtedly has errors and will need correction, it forms a sound starting point for each TC and provides access within each committee to the intellectual work of the rest of the committees.

What the TCs will need to deliver for the future includes:

A charter for the TC that lays out the primary domain of interest for the TC and its function within HL7;

Scope statement(s) for one or more "projects" that the TC will undertake;

A Use Case Model which provides the rationale for the events and messages that occur, and captures scenarios that document the use of HL7 messaging;

A Domain Information Model (DIM) that defines the specific classes and attributes from which the data elements of the HL7 messages are drawn; and

An Interaction Model that specifies in detail the trigger events and messages;

A Message Definition Model that binds the preceding models together to produce an implementable message specification.

Basic strategy

The HL7 Version 3 Message Development Framework (MDF) describes a complete process for developing message specifications. The MDF process is intended for use by a TC that is either launching a new activity in Version 3 or is maintaining existing Version 3.0 content. Thus, the MDF does not include a strategy for moving from 2.3 to 3.0. The strategy proposed in this document provides that transition by building upon the existing HL7 specifications in the context of the MDF. It includes six major steps:

1. Define and review the TC Charter and Scope Statements

2. Map Version 2.3 Trigger Events to identify candidate Version 3 Subject Classes from the chapter DIM;

3. Review and enhance the DIM based on the Subject Classes and a state analysis of these classes;

4. Build the Use Case Model for the TC's projects;

5. Review and augment the Domain Information Model for the TC;

6. Build the Interaction Model for the chapter; and

7. Develop the Message Information Model and HMD's and validate these against Version 2.3.

These six steps are spelled out in more detail in the following sections.

Step 1 Define and review the TC Charter and Scope Statements

Each Technical Committee should have a **charter or mission statement** that has been reviewed with the Technical Steering Committee (TSC). While these have been required in order to form new SIGs and TCs in the recent past, some of the older TCs may not have such a document.

Regardless, as part of the transition to V3, the TSC has been asked to review the charter for all of the TCs and SIGs during the April 1997 Working Group Meeting. Therefore, each TC should retrieve, dust-off, refine, and/or re-invent their charter before April. Charter statements read like: "Create specifications for messages and events in the area of patient administration, which is defined as …." or

"Drafts and recommends a message development methodology for HL7 and supports the maintenance of a set of reference models upon which the Working Group of HL7 bases ….."

Scope statements are created for individual projects undertaken by a Technical Committee. A project is usually the specification of a particular, coherent set of messages and events based around one (or a few) Subject Classes. Under the recommended procedures for V3, a project will define a ballot package that can be advanced to be a "draft standard for trial use" independently of other HL7 ballot packages. The Scope statement defines what is to be done in the project and is a high level use case that encompasses the entire project.

The MDF says that the generation of a scope statement should be the first order of business for a chapter as it undertakes a project. It also says that the Scope statement should be reviewed and accepted by the TSC before the TC proceeds with its efforts. Therefore, it is incumbent on each TC to develop Scope statements for their project(s) and provide these for discussion at the TSC during the April 1997 Working Group meetings.

Once they have been accepted by the TSC, project scope statements will be published as in the ANSI standards project notification system (PINS).

Step 2 Map Version 2.3 Trigger Events to Subject Classes

In Version 3.0, each Trigger Event is tied directly to a state transition in a "Subject Class" of the information model. The first step of this transition process is to identify the candidate Subject Classes for the chapter and to begin to assign Trigger Events to the state transitions in those Subject Classes.

This step is carried out simply by clustering the V2.3 Trigger Events that seem to be related to a single concept or information class in the chapter DIM. For each such cluster, identify (or define) a Subject Class to be the focal point of that cluster.

If there are any trigger events from the chapter that seem to not fall into a cluster and do not have an obvious Subject Class in the information model, these may be set aside for consideration later. It is likely that subsequent analysis will reveal the appropriate linkage for these events.

Step 3 Review and enhance the DIM based on the Subject Classes

Within the Domain Information Model for the TC, review the definition of each of the Subject Classes for completeness, correctness, and attribute contents. Similarly, review each of the classes related to the Subject Class by an instance connection, a whole part connection, or a gen-spec structure. Make changes to the DIM as needed.

Once the DIM has been reviewed, perform a complete life-cycle analysis of each of the Subject Classes. This involves defining the state variable, the possible state values for the Subject Class, the transition arcs that move from state to state, and the labels for those arcs.

For each arc in the state diagram, determine whether an existing trigger event is directly associated with this arc or whether the arc represents a potentially new trigger event. If the latter case, provide a tentative name for that trigger event.

Remember that each Trigger Event in Version 2.3 is potentially a set of related Trigger Events in 3.0. That is, a 2.3 Trigger Event might seem to point to multiple arcs in a state transition diagram. The ambiguity of existing Trigger Events will be revealed as a result of the Subject Class state transition analysis or as a result of the Use Case analysis.

It is one of the goals of 3.0 to reduce the optionality present in the 2.3 standard. This will be possible because additional trigger events can be defined to more precisely define circumstances for which 2.3 provided only a single trigger event. . For example, if *Admit-Patient* was a Trigger Event in 2.3, one might need to define separate Trigger Events for *Admit-Inpatient*, *Admit-Outpatient*, and *Switch-Outpatient-To-Inpatient* in order to have messages with no optional elements in 3.0.

Step 4 Build the Use Case Model for the chapter

For the most part, this step in the strategy is identical to the equivalent steps in the Message Development Framework. Here, the TC will define the Use Cases that motivate the chapter. The only distinction that should be made is that, as the Use Cases are refined down to their leaf-level, it should be borne in mind that each of the Trigger Events identified earlier represents the action in one leaf-level Use Case. Thus, once the Use Cases have been defined to their leaf level, the TC should identify where the Trigger Events fit into this structure.

Step 5 Review and augment Domain Information Model

The initial Domain Information Model (DIM) provided to each TC was drawn from the Draft HL7 RIM. The initial selection of DIM classes is based on a preliminary analysis performed by the Methodology and Modeling committee (ex-QADM) in preparation for the January 1997 Working Group meeting. Where possible, this DIM was drawn based upon existing chapter models. Nonetheless, the DIM needs to be reviewed critically to be sure it is in fact the correct DIM for this chapter and that it is complete in its coverage of concepts.

Among the steps that the TC should perform are to:

1. Review all of the classes in the DIM to determine if they are pertinent to the chapter. If extraneous classes are found, remove these from the DIM.
2. Validate the references that link the attributes in the DIM to Version 2.3. This will make it easier for TC members to relate the DIM to the prior version and be make it possible to trace 3.0 content to Version 2.3 concepts;
3. Assign an HL7 data type to each attribute in the DIM.
4. Augment the DIM with new classes or attributes if a need for these was identified as part of a requirements analysis in the Use Case Model defined above. If additional classes are needed, make certain that an equivalent class is not already part of the Draft RIM before defining a new class.
5. If not yet done, develop state transition diagrams for any new Subject Classes that been identified.

Step 6 Perform the Interaction Model process

This step in the transition strategy is identical to that in the Message Development Framework. It involves the definition of the Trigger Events, the Messages, and the Application Roles that are part of the Interaction Model. As this process is carried out, the TC should be mindful of the objective that Version 3.0 include all of the prior 2.3 messages.

Step 7 Develop the Message Design Model

[**Note** This section is still VERY rough – as of 11/29/96. gwb] The final step also follows the steps in the Message Development Framework for Version 3.0. As that process is carried out, however, it will be necessary to cross-check the new message specifications with the old 2.3 messages. The steps for doing this cross check can be outlined as follows:

1. Identify the Interactions that relate to each of the “old” Version 2.3 messages;
2. Carry out the Message Information Model and Hierarchical Message Descriptor process for each of these “legacy” Interactions.
3. Compare the HMD’s defined for these Interactions with the prior 2.3 messages. In this process, ask:
   a) If any element of the 2.3 message is missing and whether or not this is reasonable. (Given the loose optionality of Version of 2.3, there may well be reasonable omissions.)
   b) If there are any new elements in the messages and, if so, why did those elements arise? (If they do arise, one may need to question whether this is truly an Interaction that existed in 2.3)
4. Document any differences between the Version 3.0 HMD’s and 2.3 messages.