HL7 Provider Registry

Design Issues Discussion
May 2010
Simple Notification (PA)

• The simplest dynamic model pattern is a notification with no receiver responsibility. A 'Registry Informer' sends information to one or more 'Registry Tracker' application roles whenever a registry record is added or changed. The current Patient Administration content includes the following simple notification interactions:
  – Registry Record Added
  – Registry Record Revised
  – Registry Record Nullified
  – Registry Duplicates Resolved
Simple Notification (PM)

- Personnel Management does not include all of the above
  - **Provider Registry** does not include Nullified or Duplicates Resolved
  - **Organization Registry** does not include Duplicates Resolved
Simple Notification (EIS)

- The EIS Functional Model includes a single function "Notify Identity Updates" to cover all state transitions.
- The EIS Functional model includes a 'Request Identity Update Notifications' function. This seems similar to the HL7 v2 publish/subscribe process.
Query and Query Response (PA)

• The next most complicated dynamic model pattern is query and query response. A 'Registry Query Placer' sends a set of parameters to a 'Registry Query Fulfiler' application role that is responsible for returning all records that match the parameters. The current Patient Administration content includes the following query interactions:
  – Registry Find Candidates Query
  – Registry Find Candidates Query Response
  – Registry Get Details Query
  – Registry Get Details Query Response
  – Registry Get Identifiers Query
  – Registry Get Identifiers Query Response
Query and Query Response (PM)

- Personnel Management adds an Administrative Report Query Response for returning preformatted reports. The formatted report is returned in a DocumentEvent class either in the text attribute or by reference.


Request and Fulfillment (PA)

• The most complicated class of dynamic behavior is the request and fulfillment pattern. A 'Registry Request Placer' sends a request for action to a 'Registry Request Fulfiller' application role that is responsible for accepting or rejecting the request and informing the requestor of the result or reason for rejection. The current Patient Administration content includes the following request interactions:
  – Registry Add Request
  – Registry Add Request Accepted
  – Registry Add Request Rejected
  – Registry Revise Request
  – Registry Revise Request Accepted
  – Registry Revise Request Rejected
Request and Fulfillment (PM)

- Personnel Management defines a single Response interaction to confirm processing and return results or indicate why the request was not fulfilled. PA defines separate Accepted and Rejected interactions with different Control Act Wrappers.

- Personnel Management returns a small message consisting of identifying information. Patient Administration returns a complete record with each Accepted response.
Request and Fulfillment (EIS)

- The EIS includes more Requests than PA or PM:
  - Update Identity State (from any state to any state)
  - Merge Identities (resolve duplicates)
  - Unmerge Entities
  - Link Entities (within or across jurisdictions)
  - Unlink Entities
  - Remove an Identity Instance from EIS
Update Mode

• Mechanism for indicating record changes at a fine-grained level of detail. The UpdateMode attribute can be enabled for individual attributes, or for a collection of attributes representing a class or a set of objects. The principal purpose of UpdateMode is to allow a sending system to identify to a receiving system:
  
  – Changes that have occurred in an object controlled by the sending system; or
  
  – Changes that the sender desires to be made in an object controlled by the receiving system
Accountability History

- Assumes that all objects have a history that can be described by reference to one or more historic control acts. Some registry systems track trigger event information for individual data objects at a level of granularity not directly supported by the standard RIM classes. The reasons for both tracking and reporting upon changes at this fine level of detail are:
  - Accurately account for all record changes in order to supply audit information
  - Differentiate between sources of data when records are maintained collaboratively i.e., have more than one author and/or custodian
Update Mode & Accountability Hx

• Is this valuable?
• Can existing systems support this?
• Should we define two sets of Application Roles for systems that can and cannot support update mode and accountability history?
HL7 Registry Event Notification
HL7 Registration Request
Registration Act States Supported

- Held
- New
- Aborted
- Suspended
- Active
- Completed
- Nullified
- Obsolete
Registry Record States Supported

Diagram showing the states and transitions: 
- **Normal**
  - **Cancelled**
  - **Suspended**
  - **Active**
  - **Terminated**

Transitions:
- Create
- Revise
- Activate
- Cancel
- Resume
- Suspend
- Terminate
- Reactivate
- Nullify

State transitions are depicted with arrows, showing the flow between states.
HL7 RIM Role State-machine
Interoperability Environment

Community A

Inter-module

Intra-community

Community B

Cross Community

Inter-Community

Intra-Community