Justin suggested using Spring injection to set the DAO flavor in

org.openmrs.api.context.Context

. RG agreed with this
• TODO: Justin will commit the change
• Tables are in LATIN-1 encoding for now.
  • TODO: Paul will look into best encoding format to use generally
• Messaging
  • TODO: Justin to provide specs for messaging service/table
  • TODO: RG needs to create messaging table(s)
• Transaction and Event
  • TODO: RG to spec out model changes to support "projects" (multiple databases housed in the same repository) and state flow modeling (studies, treatment plans, etc.)
• Some talk about decision support modeling. RG’s utopia would include...
  • decision support rules that could be defined as "complex concepts" or "derived concepts" (i.e., concepts that are calculated from one or more atomic concepts or other derived concepts). For example, "ASTHMATIC" would be a derived concept defined as a rule considering several diagnoses, prior test results, etc. Anywhere within the API, observations could then be queried without knowing/caring whether the reported value was read directly from a table or calculated (derived) from other observations.
  • decision support rules that return a set of "observations" — e.g., a set of 0..n objects with timestamp, value, etc. This way atomic observations and decision rules (derived observations) can return the same structure and, therefore, make the code more flexible. Additionally, each object in the result set could have .getAsBoolean(), .getDate(), and .getValue() methods that returned an appropriate representation so that the same objects could be used in different contexts — e.g., in one rule I might want to know "is the patient asthmatic" as a boolean result and in another rule I might want to know when the patient was diagnosed as asthmatic; both rules could use the derived concept "ASTHMATIC", one looking at the result as a boolean and the other as a date.