Plan for Adaption of Component Architecture
9/23/10
Michael D. Morris

1) Create Single Schema Solution for all current standards
   a) Modify top level schema and sector libraries so that they contain no duplicate type names to core-main by adding a sector library identifier to the name
   b) Run xlst conversion stylesheet against each released schema set to produce a single schema. Set the PESC defined version attribute on each component (type, group or element) to the namespace of the sector library and core from which the component was fetched.
   c) Provide the single file schema on the PESC web page for each standard
   d) Update the implementation guide diagrams so that they do not included the namespace prefix.
2) Provide single file schema solution for new development
   a) Development team puts all new elements and types into the top level schema
   b) Import the sector and core libraries as usual
   c) When the schema is ready for approval, run the conversion xslt so that there is only one schema file
   d) Create the implementation guide so that the diagrams do not include the library prefixes
   e) Elements that are considered by the CCB for incorporation into core or sector libraries may be inserted in the revision cycle.
3) Modify the Registry and Repository so that it provides web services that allow creation and maintenance of schemas from components stored in the R&R
   a) One Web service would accept an XML component names and versions and return a schema file that included the requested components and all the components on which the requested components are dependent. This functionality is available in the current R&R through the User Interface; however, instead of the component being returned with a comment for the version, the R&R would set a version attribute of the component.
   b) A second web service would parse a set of XML components that are not in the repository and store them in the repository as core or sector components so they could be accessed by new schemas.
   c) A third web service would take a complete schema file and determine if there are more recent components and create a new schema file with the updated components. A component attribute called update may be used to indicate which schema components are eligible for update (a revision indication scheme such as used by Ivy may be appropriate)