INCOSE International Symposium
Philadelphia, PA
June 24-27, 2013

Proposed Attribute Method: A Flexible Method of Change Control for DOORS®

Jerry Huller
Raytheon
cjhuller@raytheon.com, jerry.huller@incose.org

This presentation is intended to give you some background on the PAM method to help with understanding the details provided in the paper provided in the IS2013 *Proceedings*

Introduction (1 of 2)

- Requirements will change over time
  - Sources of change include trade studies, evolving understanding of needed capabilities, engineering change proposals
- Projects need viable means of managing and controlling changes to requirements

Introduction (2 of 2)

- This presentation discusses a flexible approach for change control of requirements maintained in IBM® Rational® Dynamic Object-Oriented Requirements System (DOORS®)
  - Used successfully for over two years with more than 200 Change Requests for DOORS database containing over 14,000 requirements
  - Provides DOORS-native alternative to DOORS Change Proposal System (CPS)

Considerations (1 of 3)

- Objectives for change control method (partial listing)
  - Ensures against unauthorized modifications
  - Does not require additional licenses; low cost or native to DOORS
  - Does not require special administration privileges
  - Can be run and maintained locally
  - Capable of handling hundreds of changes across different DOORS modules

Considerations (2 of 3)

- Objectives (continued)
  - Proposed changes can be identified by a small number of DOORS users and reviewed external to DOORS without the need for additional DOORS licenses
  - Can see all changes at once for a DOORS module and for any object within a DOORS module
  - Provides context within a requirements module for a proposed change

Considerations (3 of 3)

- Objectives (continued)
  - Provides clear and unmistakable indications of whether the project engineer intends to add a new requirement or attribute value, delete (or mark for deletion) an existing requirement or attribute value, or modify an existing requirement or attribute value
  - Allows implemented changes to be undone if an obvious mistake has been made or there is compelling (and approved) justification

Decision

- Selected the Proposed Attribute Method (PAM)
  - A version was in use by another Raytheon project
  - Augmented the method and developed underlying infrastructure, including DOORS eXtension Language (DXL) scripts

PAM Concept (1 of 2)

- There are DOORS attributes under Configuration Management (CM) control
  - E.g., *Object Text*
- Create and use "proposed" attributes as counterpart of controlled attributes
  - E.g., DOORS modules contain *Object Text* and *Proposed Object Text*
- Project engineer is only allowed to fill in values for proposed attributes

PAM Concept (2 of 2)

- Proposed changes (attributes) are reviewed using a defined project process
- Once approval is given, a DXL script is run
  - E.g., the value for *Proposed Object Text* is transferred to the value for *Object Text*, and the value of *Proposed Object Text* is blanked out (reset to its default, usually null, value)

PAM is realization of thinking outside the box

“The devil is in the details”

PAM Avoids Potential Issues
(1 of 2)

- How do I ensure against unauthorized modifications to the controlled attributes?
- How can I remove the controls if the project later decides that a particular attribute no longer requires the rigor of change control?
- How can I limit the changes to only certain objects within a module?

PAM Avoids Potential Issues
(2 of 2)

- How do I entirely remove the value of a controlled attribute, if necessary?
- How do I handle the situation when more than one project engineer wants to make changes to the same or different attributes for one or more objects in the DOORS module?

PAM Implementation (1 of 5)

- PAM consists of seven DXL scripts and some procedural controls
  - Scripts can be categorized as: screening (one), support (four), and implementation (two)
  - Text of all scripts provided in appendix to paper in *Proceedings*

PAM Implementation (2 of 5)

DXL Script	Description
Script #1: Configure Specific Attributes Access-Definition and Type Access_Module.dxl	A support script. Enables a DOORS Manager to lock down the definitions of a subset of the attributes and their types within a module.
Script #2: Configure Specific Attributes Access_Value Module.dxl	A support script. Enables a DOORS Manager to lock down the value of controlled attributes within a module.

PAM Implementation (3 of 5)

DXL Script	Description
Script #3: Set All Specific Attributes Access-Value to Inherit_Module.dxl	A support script. Enables a DOORS Manager to reset the lock down state of attribute values to their original state (in DOORS vocabulary, to “inherit [their access] from module”).
Script #4: Set All Specific Attributes Access-Definition and Type Access to Inherit_Module.dxl	A support script. Enables a DOORS Manager to reset the lock down of attribute definitions and types to their original state (in DOORS vocabulary, to “inherit [their access] from module”).

PAM Implementation (4 of 5)

DXL Script	Description
Script #5: Evaluate_NewInternalAuth for PAM.dxl	An implementation script. Checks for proper format of the Change Request number and identifies *Object Identifier* values containing that number.
Script #6: ProposedAttribute.dxl	An implementation script. Moves the proposed attribute values to the actual attribute values. Blanks out attribute values if so marked.

PAM Implementation (5 of 5)

DXL Script	Description
Script #6: (cont.)	Blanks out *Reason for Change* attribute. Provides count of number of objects with changes, and total number of changes made.
Script #7: Screen Non-Blank PAM Attr.dxl	A screening script. Provides list of proposed attribute changes by name and *Object Identifier* value.

Other Considerations

- Processing of deletions
- Blanking out an attribute value
- PAM conflicts

Analysis of PAM (1 of 6)

- Disadvantages (partial listing)
  - Can delay the implementation of proposed changes when more than one team on a project needs to update the same or different attributes for an object in a module (procedural approaches for resolving PAM conflicts lead to the delay)
  - Requires familiarization training for unique aspects such as usage of the *Blank Out-Chg* and *New Internal Authority (NIA)* attributes, and running of the NIA script

Analysis of PAM (2 of 6)

- Disadvantages (continued)
  - Providing write access to modules allows the inadvertent creation of unneeded new objects, which must be then marked for deletion
- Advantages (partial listing)
  - Provides a method of orderly and traceable change while ensuring against unauthorized modifications
  - Capable of handling one to hundreds of changes to attributes within a module

Advantages of PAM outweigh the disadvantages

Analysis of PAM (3 of 6)

- Advantages (continued)
  - Not really sensitive to number of changes within a module or to number of modules affected by a Change Request
  - DOORS-native, does not require additional licenses to implement
  - Can be implemented locally by DOORS Managers with RMCDA privileges

Analysis of PAM (4 of 6)

- Advantages (continued)
  - Requires minimal knowledge of DXL scripting to tailor the scripts for new projects (assuming only a certain format for identifying the change control authority)
  - Has intuitive appeal to project engineers accustomed to working with “was-is” change views

Analysis of PAM (5 of 6)

- Advantages (continued)
  - Provides instant visibility into all proposed changes in work
    - Can see all changes at once for a DOORS module and for any object within a DOORS module
  - Proposed changes are developed by a small group of project engineers with write access to requirements modules, while being reviewed by a larger group without the need to view the changes in DOORS
  - Allows attribute values to be removed (blanked out)

Analysis of PAM (6 of 6)

- Time required to run main DXL script varies between a few seconds and a few minutes, depending on number of changes
- Largest approved CR affected 2,620 objects—all requirements—in single DOORS module and contained 10,480 attribute changes
- PAM concepts are not unique to DOORS
  - Should be extendable to other requirements management tools

Conclusion (1 of 2)

- PAM provides a flexible, understandable, and inexpensive method of change control for requirements
- PAM is easily trained to new project engineers
- PAM enables project engineers to work directly in DOORS and harness the capabilities of DOORS to explore the impacts of proposed changes on parent, child, and sibling requirements

Conclusion (2 of 2)

- DOORS ingest of approved, proposed changes occurs rapidly, usually within a few seconds
  - Only the processing of deletions (*Object Text*, *Object Heading*) takes additional time
  - Metrics data on number of changes is provided by module
- PAM can be quickly updated to address additional DOORS attributes