HRE – FILTERS – OVERVIEW

Revision history

2018-06-03	Robin Lamacraft	Original draft
2018-06-12	Robin Lamacraft	Updated Substitution table references
2023-07-15	Don Ferguson	Updated (not yet implemented)

SCOPE

A filter in HRE is a chain of steps that start with a SINGLE SUBSET and can have two kinds of outcome which need to be kept logically separated because one does not modify user data while the other can:

- SELECTION FILTER: Creates one or more new SINGLE subsets or combines SINGLE subsets to form other SINGLE subsets
- ACTION FILTER: This is an extension of the SELECTION FILTER. It performs actions on other objects while executing the filter. This type of filter acts very much like an HRE internal equivalent of John Cardinal's TMG Utility that uses rules to modify user data. Because of its potential to corrupt user data if incorrectly defined, it needs extra controls over its operation. On running the action filter it should be able to:
    - Make no changes and produce a log of intended changes
    - Make changes and produce a log of changes made
    - Make changes and don't produce a log of changes made.

  In all cases of running a filter, the GUI needs to have an option to proceed at one step at a time with summary of that last step.

FILTER ANALOGY

A filter in HRE can be considered like a set of carriages on a railway network heading for a destination (the carriages are the equivalent of HRE objects). Some don't make it, some are pushed into sidings and reassessed; some new carriages are introduced from other routes on the way. In an ACTION filter on some routes the carriages change their cargo or change their properties and then again join the railway traffic.

FILTER COMPONENTS

- Each Filter must start with an initial Subset of a nominated object type
- Usual steps in all filters have the form of a verb followed by a list of parameters, some of which are default and can be omitted. An indicative list of filter elements is:
    - <***Load Subset>*** (Parameter list>)
    - <***Comparison Property***>(<parameter list>)
    - <***Combine Subsets***> (<extra subset PID, <parameter list>)
      requires 2 subsets to be in the subset stack
    - <***Save Subset***> (<parameter list>)
      NOTE: Subset names with "_" as the first character of their name are automatically deleted on completion of execution of the filter
    - <***Delete Subset***> (<parameter list>).
- Extra steps available in ACTION filters are:
    - <***Object Property Action***> (<parameter list>) –set flags, edit values, etc
    - <***Object Action***> (<parameter list>) –print output, etc.
- These actions will be represented on the GUI as an indented, editable tree. Each indented list is the sequence of operations of a single Subset. Once that branch is completed the start

of the next branch is evaluated. Often this is where the contents of 2 existing subsets are compared to create a new subset

- The action of the script should be available as a step by step view, if requested for debugging or as a summary report (if desired by the user)
- Note that the Object Property Action, Object Action and the Object Property access all use the Substitution codes. This means that these steps can request complex field selections
- Comparison actions must be value type compliant
- Comparison actions must be context compliant
- A filter can be many branches and can edit different object properties depending on which path an object PID passes.