HRE – CONDITIONAL TEMPLATE SUBSTITUTIONS - OVERVIEW

Revision history

2018-06-18	Robin Lamacraft	Original draft

SCOPE

HRE uses a markup syntax in a template to identify what should be substituted for that markup symbol. There are many types of action that are available within the markup system. This document only describes the evaluation of a conditional markup symbol which, when evaluated as TRUE or FALSE, executes the associated TRUE or FALSE clause. NOTE: the conditional actions can be nested within other conditional clauses. Each conditional clause has one obligatory component (the comparison test), then immediately following there must be at least one TRUE or FALSE clause or both TRUE and FALSE clauses. The clauses must specify output content for this substitution.

The main examples of this feature are Tag Sentences and retrieval of Name Style Template values.

CONDITIONAL CLAUSE STRUCTURES

There are 3 distinct types of conditional template substitutions:

- VALUE IS SET or NOT –value available or not
  e.g. PERSON BIRTH LOCATION is not set
  [<^^prior text^^[=PERSON.BIRTH.LOCATION=]>^^post text^^|*not set commands* >]
  If birth location:
    - **Is set**, then this will output the **prior text** the **birth location** then the **post text**
    - **Is not set,** then **not set commands** are executed**.**
- VALUE FIXED COMPARISON – compare retrieved value with a fixed constant
  e.g. PERSON BIRTH COUNTRY is equal to “USA”
  [?EQUAL([=PERSON.BIRTH.COUNTRY=], ^^USA^^)?]
  [+*if TRUE commands*+] [-*if FALSE commands*-]
- VALUE CHANGING COMPARISON – compare retrieved value with another changing retrieved value
  e.g. PERSON BIRTH COUNTRY=] is equal to PERSON DEATH COUNTRY
  [?EQUAL([=PERSON.BIRTH.COUNTRY=], [=PERSON.DEATH.COUNTRY=])?]
  [+*if TRUE commands*+] [-*if FALSE commands*-]

CHOICE OF COMPARISON FUNCTIONS

The list of comparison functions will include all normally expected functions, some of which will be data type dependent like comparing Historical Dates. Some function may have more than 2 parameters like “check whether a date is within a range of values”.

NOT-SET, IF-TRUE and IF-FALSE COMMANDS

These commands are normal commands and can include further conditional comparison template substitutions.

TABLES USED

- Table **878 (SUBSTN_OTHER_ALIASES)** is where a record defines each comparison function
- Table **879 (SUBSTN_STEPS)** stores a collection of records used to define the conditional structure
- Table **880 (SUBSTN_PARAM_LISTS)** has a record for each parameter of each function
- Table **881 (SUBSTN_PARAM_SET_DEFNS)** defines the parameter set for each function

- Table **882 (SUBSTN_PARAM_NAMES)** holds a list of parameter names for each function (for use where only some parameters are to be set and others are maintained at their default value)
- Table **883 (SUBSTN_PARAM_VALUES)** has records to hold the current value for each parameter of each function. A set of these records are identified as the default parameter values for this function
- Table **884 (SUBSTN_COMMON_DEFNS)** has records to define Substitution Elements that are used in many types of Substitution
- Table **885 (SUBSTN_COMMONS)** has a record for each Common Element that uses the same set of parameter values
- Table **886 (SUBSTN_TEMPLATE_DEFNS)** has a record for each template definition
- Table **887 (SUBSTN_TEMPLATES)** has one record for each defined template.

CONDITION SUBSTITUTION STORAGE (TABLE 879 (SUBSTN_STEPS)

1. Typically the initial step is the selection of the function, including references to the first step in the IF_TRUE path and the start of the IF-FALSE path
2. The next set of steps are the individual ordered function parameters
3. The next step is the start of the IF-TRUE commands which ends with a reference to a continuation step
4. After that the next step is the start of the IF-FALSE commands which ends with a reference to a continuation step.

GUI

When shown by the GUI these will be shown as indented blocks.