Representing TADDM enumerations in a database table

In TADDM, many attributes values are stored as enumeration values, so that the attribute value is a numeric digit rather than a descriptive text.

When creating reports from the TADDM data, it is – currently – only possible to present the user with meaningful information by hardcoding enumeration translations in the report, or the queries that generate the data for the report.

To make such enumeration translation more flexible, the enums_view script has been developed.

The enums_view script reads all TADDM enumerations from the platform-model.jar file, and creates and executes sql scripts to maintain the enumeration specifications in a database table and a view. This allows you to retrieve the actual enumeration value from report queries by simple joins between the enumerated value and the table or view created by the enums_view script.

Usage example

The CIRole attribute of a ComputerSystem is an example of one such enumerated attributes.

Without translation, the CIRole is represented as a numeric value – as demonstrated below:

```
select displayName_C, CIRole_C from BB_COMPUTERSYSTEM40_V
where cirole_c is not null fetch first 3 rows only

DISPLAYNAME_C                   CIROLE_C
----------------------------- --------
NC041130.kraklab.pl.ibm.com          3
NC041043.kraklab.pl.ibm.com          3
ibmsccd751demo.tivoli.edu            3
```

By joining the results with the view created by the enums_view script, the real value of the enumerated attribute is returned:

```
select displayName_C, FIELD_NAME_C from BB_COMPUTERSYSTEM40_V as c
join BB_ENUMERATIONS_V as e on e.FIELD_VALUE_C = c.CIROLE_C and
e.enumeration_name_c = 'CIRoleEnum'
where cirole_c is not null fetch first 3 rows only

DISPLAYNAME_C                   FIELD_NAME_C
----------------------------- ------------
NC041130.kraklab.pl.ibm.com TEST
NC041043.kraklab.pl.ibm.com TEST
ibmsccd751demo.tivoli.edu    TEST
```

Tables and views

The enums_view script creates following objects in the TADDM database:

Table: ENUMERATIONS

Column name	type	Description
ENUMERATION_NAME_X	varchar(192)	Name of the enumeration
FIELD_NAME_X	varchar(192)	Name of the enumeration field
FIELD_TYPE_X	varchar(16)	Type of enumeration value
FIELD_VALUE_X	integer	Enumeration value

View: BB_ENUMERATIONS_V

Column name	Description
ENUMERATION_NAME_C	Name of the enumeration
FIELD_NAME_C	Name of the enumeration field
FIELD_TYPE_C	Type of enumeration value
FIELD_VALUE_C	Enumeration value

Installation

To minimize the risk of removing the script inadvertently as part of an upgrade, copy the associated files, enums_view.py and enums_view.bat to a location on your TADDM storage server which is not part of the standard TADDM directory structure. It is recommended that you use a two-level subdirectory structure – for example $COLLATION_HOME/custom/bin – to use a directory structure that is similar to the directory structure TADDM uses for support and sdk tools.

All directories and files you create should be owned by the TADDM instance user.

Depending on the chosen location for the scripts, you may have to modify the first line to correctly launch the TADDM environment from the script.

The first line in each eunms_view script should point to the jython_coll_253 script in the $COLLATION_HOME/bin directory. Use the relative location of the directory to initialize the environment.

If for example, the enums_view scripts are located in $COLLATION_HOME/cust/bin use the following path to launch the jython interpreter and initiate the TADDM environment:

      Linux: ../../dist/bin/jython_coll_253

      Windows: ..\..\dist\bin\jython_coll_253.bat

If the script enums_view scripts are located in $COLLATION_HOME/bin use the following path to launch the jython interpreter and initiate the TADDM environment:

Linux: ./jython_coll_253

Windows: .\jython_coll_253.bat

Invocation

To launch the enums_view script simply execute the script. If no arguments are provided, the help information is displayed.

The invocation syntax is:

enums_view.py <action> [-u|-user <username>] [-p|--password <password>] [-q|--quiet] [-h|--help]

<action>	action to perform. Valid actions are:	
	scripts	Creates sql scripts to create the database objects.
	create	Executes the create_enums_view_table.sql script
	remove	Executes the drop_enums_view_table.sql script
	help	Show this help informaiton
-u, --user <user>		User to perform the action as. Default value is 'administrator'
-p, --password <password>		Password that authenticates this user. Default value is 'collation'
-q, --quiet		Suppress output to the console
-h, --help		Displays this help message.

The sql sripts created by using the 'scripts' action are stored as create_enumeration_table.sql and drop_enumeration_table.sql in $COLLATION_HOME/etc/views.

Messages from this script are logged in $COLLATION_HOME/log/enums_view.log.

The enums_view script leverages the TADDM dbupdate script to perform database operations. For that reason, the TADDM storage server must be operational when the enums_view script is invoked with the 'create' or 'remove' actions.

Examples

To create sql scripts that can be used to create the enumeration database objects, use the 'scripts' option

```
enums_view.py scripts
```

To create the enumeration database objects, use the 'create' action after the scripts have been created

```
enums_view.py create
```

To drop the enumeration database objects, use the 'remove' action after the scripts have been created

```
enums_view.py remove
```

To use specific set of credentials to log on to TADDM use the -u and -p options

```
enums_view.py <action> -u <your-user> -p <your-password>
```