

Qualify

Qualify

The purpose of this tools is to allow batch changes to be applied to data within a jena [model](#).

Reason for Use

When there are sections of data that need to be removed or changed then the qualify tool is a way of performing those changes without the need to know how to form the sparql update queries.

Parameters

wordiness - (optional) sets the lowest level of log messages to be displayed to the console. The lower the log level, the more detailed the messages.

- Possible Values:
 - OFF - Results in no messages being displayed.
 - ERROR - Results in only messages from the ERROR level to be displayed. Error messages detail when the tool has experienced an error preventing it from completing its task
 - WARN - Results in only messages above and including WARN level messages to be displayed. Match does not produce any WARN level messages.
 - INFO - (Default) Results in all messages above and including INFO level messages to be displayed. INFO level messages detail when the tool has started and ended and when it begins/ends a phase ('Finding matches' and 'Beginning Rename of matches') and how many matches have been found.
 - DEBUG - Results in all messages above and including DEBUG level messages to be displayed. DEBUG level messages detail each matching input URI to its VIVO URI as they are processed. Additionally, it will display stacktrace information if an error occurs.
 - ALL or TRACE - Results in all messages above and including TRACE level messages to be displayed, since trace is the lowest level it is the same as ALL in practice. TRACE level messages details every matching set as it is processed in each phase along with SPARQL queries and start and stop for their execution.
- **modelsource** - Provides the information needed for the connection to the source data model, which is the model that will be searched and modified.
- model.conf.xml
- **predicate** - When making changing the data related to a predicate this is the value which will be used as that predicate.
- predicate
- **regexMatch** - When using regular expressions from a matching string, this is the field which will hold that string with regex characters.
- Regex string
- **textMatch** - When matching a text string, this is the field which will hold that string.
- match string
- **value** - This field gives the value that is used to replace the selected strings.
- replacevalue
- **remove-namespace** - This namespace is to be removed during the run of quality when paired with **predicate-clean** or **clean-resources**. Any resources within this namespace are removed from the model. It is used when there is a namespace used for part of the harvest but is not part of the data that is due for harvest.
- namespace
- **predicate-clean** - A flag to signify that the triples which have a predicate within the given **remove-namespace** are removed from the source. This is useful when removing the data can be specified by a predicate namespace.
- true
- **clean-resources** - This flag signifies that triples which have either a subject or object within the given **remove-namespace** is to be removed.
- true

Qualification will execute specific user defined sparql queries against a model in order to clean and qualify the data contained inside the model before storing inside of VIVO. Qualification queries will be site specific. As such the default configuration for the harvester doesn't currently invoke Qualification.

Overview

Short Option	Long Option	Parameter Value Map	Description	Required
d	datatype	RDF_PREDICATE	data type (rdf predicate)	false
i	jenaConfig	CONFIG_FILE	config file for jena model	false
l	jenaOverride	JENA_PARAM = VALUE	override the JENA_PARAM of jena model config using VALUE	false
r	regexMatch	REGEX	Match this regex expression	false
t	textMatch	MATCH_STRING	Match this exact text string	false
v	value	REPLACE_VALUE	Replace matching record data with this value	false
n	remove-namespace	RDF_NAMESPACE	Specify namespace for p/predicate-clean and -o/-clean-resources flag	false
p	predicate-clean		remove all statements where the predicate is from the given n/-remove-namespace	false
c	clean-resources		remove all statements where the subject or object is from the given n/-remove-namespace	false

Usage

```
preparation:
Qualify=" java $OPTS -Dprocess-task=Qualify org.vivoweb.harvester.qualify.Qualify"
MATCHEDINPUT="-i $H2MODEL -ImodelName=$MATCHEDNAME -IdbUrl=$MATCHEDDBURL -IcheckEmpty=$CHECKEMPTY"
```

```
Call:
$Qualify $MATCHEDINPUT -n http://vivoweb.org/ontology/score -p
```

Methods

strReplace

1. Get statements of the specified *dataType* with the *oldValue*
2. iterate through those statements
 - a. replace *oldValue* with *newValue*

regexReplace

1. get statements with the given *predicate* and who's object matches the *regexMatch* filter
2. assemble insert and delete sparql statements to delete the old and insert the new.

cleanResources

1. construct and call the query to remove subjects and objects in the given *namespace*

cleanPredicates

1. construct and call the query to remove predicates in the given *namespace*