Env

The environment file contains setup variables and information which is important for the scripts to execute properly. There are some options of note which should be reviewed before execution. The file itself is located in scripts/env relative to your harvester base directory.

Memory

Configure your JVM heap size

```
#update memory to match your hardware -- set both to be the same, in general the more memory the better, but
too much can cause errors as well.
#8G-12G on large vivo's seems to work well
MIN_MEM=2048m
MAX_MEM=2048m
```

Backups

By default, the example scripts make use of the backup and restore functionality defined in the env file. Should you wish to change where these backups are located, change these variables accordingly.

```
BACKUPPATH="$HARVESTERDIR/backups"
LATESTBACKUPPATH="$BACKUPPATH/latest"
```

Empty Model

When set to "true", check every model to see if it contains any statements – useful for debugging, but is slower when loading models. Whenever a model is empty, this will print a warning to the console and log. Set to "false" for enhanced performance.

```
CHECKEMPTY=true
```

Optimizations

These are some optimizations that can be enabled to potentially improve your performance. Each one is briefly explained. By default, none of these are set, besides the heap size.

colspan = "2"	Variable for optimizations to the Java virtual machine.
-server	Run in server mode, which takes longer to start but runs faster
-d64	Use 64-bit JVM
-XX:+UseConcMarkSweepGC	Use concurrent (low pause time) garbage collector
-XX:+DisableExplicitGC	Prevent direct calls to garbage collection in the code
-XX:+UseAdaptiveGCBoundary	Allow young/old boundary to move
Target maximum for garbage collection time	
-XX:-UseGCOverheadLimit	Limit the amount of time that Java will stay in Garbage Collection before throwing an out of memory exception
Shrink eden slightly (Normal is 25)	
-Xnoclassgc	Disable collection of class objects
Use SSE3 Processor extensions	
Maximum number of Parallel garbage collection tasks	

Aliases

To shorten script lines aliases are declared for each tool.

```
RenameResources="java $OPTS -Dprocess-task=RenameResources org.vivoweb.harvester.qualify.RenameResources"
```

in the general form:

Alias = java $OPTS -Dprocess-task=*taskname package.path.name.for.class*

Server Information

The server information is stored in the vivo.xml file and various parts of it are parsed out with the XPathTool

```
SERVER=`$XPathTool -e "/Model/Param[@name='dbUrl']" -x $VIVOCONFIG | sed 's|^.*://\([^:^/]*\)[:/].*$|\1|g'`
DBNAME=`$XPathTool -e "/Model/Param[@name='dbUrl']" -x $VIVOCONFIG | sed 's|^.*/\(.*\)$|\1|g'`
USERNAME=`$XPathTool -e "/Model/Param[@name='dbUser']" -x $VIVOCONFIG`
PASSWORD=`$XPathTool -e "/Model/Param[@name='dbPass']" -x $VIVOCONFIG`
NAMESPACE=`$XPathTool -e "/Model/Param[@name='namespace']" -x $VIVOCONFIG`
```

Functions

Some functions have been created for series of calls which are repeated.

prep

Preparation for scripts

1. Check for namespace and end if not present
2. Creation of directories

backup/restore-path

usage: backup-path <Directory> <BackupBaseFileName>

Creates a link to the latest file.

Compresses a directory into a "tar.gz" file.

usage: restore-path <Directory> <BackupBaseFileName>

Uncompresses a directory from a "tar.gz" file.

backup/restore-file

usage: backup-file <FileName> <BackupBaseFileName>

Copies a file into the backup directory with a time stamped name.

Creates a link to the latest file.

usage: restore-file <FileName> <BackupBaseFileName>

Copies a latest file from the backup directory.

backup/restore-mysqldb

usage: backup-mysqldb <BackupBaseFileName>

Dumps database into a file.

usage: restore-mysqldb <BackupBaseFileName>

Restores the database from the dumpfile