ETL User Guide

Contents

Overview

The ETL program *Extracts* patient information from existing systems, *Transforms* the results into standard HIPAA safe medical vocabularies, and *Loads* the results into a locally controlled Peer node.

The ETL is an upgrade to the original SPIN Submission Suite with four notable improvements:

(1) ETL is highly modular and configurable to new types of clinical data,

(2) ETL allows easy deployment with interfaces to JDBC databases, XML exports, and SOAP

(3) ETL supports a Two Phase Commit protocol to ensure data integrity during synchronization.

(4) ETL can be run on demand or as a scheduled task synchronizing cases within a date range.

Definitions

ETL: (1) Extract **(2) T**ransform **(3) L**oad

Extract

Extract from existing database, XML file, or custom interface

Transform

Transform results of the extraction (autocoding and de-identification)

Load

Load transformed results into the SPIN peer, DB, or XML file

Required Software:

Java 1.5 or later

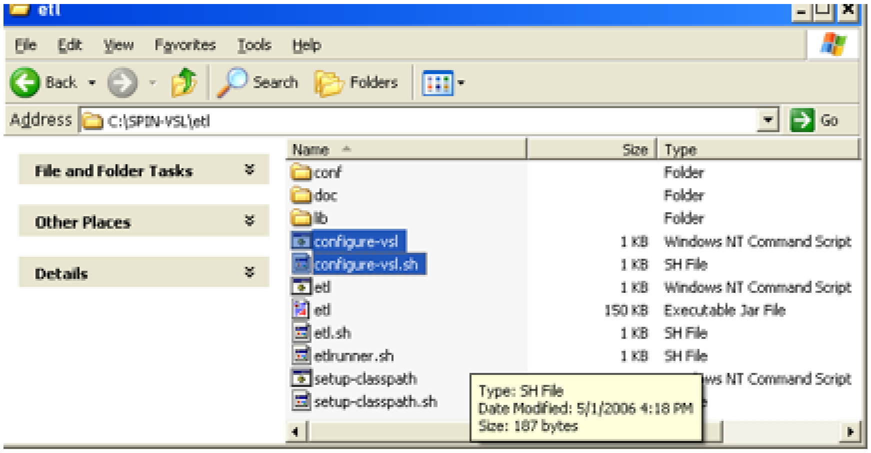
MySQL 4.1 or later

Installation

(1) Unzip the ETL zip archive to a directory of your choice

(2) Keystore Setup

(3) ETL Configuration Wizard



Address C:\SPIN-VSL\etl

Name	Size	Type
conf		Folder
doc		Folder
lib		Folder
configure-vsl	1 KB	Windows NT Command Script
configure-vsl.sh	1 KB	SH File
etl	1 KB	Windows NT Command Script
etl	150 KB	Executable Jar File
etl.sh	1 KB	SH File
etlrunner.sh	1 KB	SH File
setup-classpath		ws NT Command Script
setup-classpath.sh		

Type: SH File
Date Modified: 5/1/2006 4:18 PM
Size: 187 bytes

Windows users, select "configure-vsl.cmd"

Mac / Linux users, select "configure-vsl.sh"

The configuration wizard will launch a series of command line choice prompts.

We now provide an example using the most common use case: *we will* ***Extract*** from a standard JDBC database and ***Load*** into the SPIN Peer and Codebook.

```
# Extractor for the VSL pipeline #

                    [DB]          Database Connection

                    [XML]         XML Formatted File

                    [CUSTOM]    Custom

Extractor for the VSL pipeline=DB
```

The wizard will then ask which type of DB extractor to use:

```
# Database Extractors #

             [STANDARD]    Standard Pathology LIMS

             [MGH]               MGH Frozen Tissue Repository

             [BWH]               BWH Frozen Tissue Repository

             [CRIMSON]       Crimson Biomaterials Collection Core

Database Extractors=STANDARD
```

The wizard will then ask for the DB connection parameters and SQL script

```
 # Pathology Database #

driver (default 'com.mysql.jdbc.Driver')

=

URL (default 'jdbc:mysql://localhost:3306/spinDB')

= 'jdbc:mysql://myServer:3306/myTissueBank'

username (default 'spin')

= share

password (default 'lcsspin')

= specimens
```

The wizard will then test the DB connection

```
Test Connection successfull !

driver=com.mysql.jdbc.Driver
```

```
URL=jdbc:mysql://myServer:3306/myTissueBank
username=share
password=specimens
```

The wizard will also ask you for the location of the SQL script

```
# File containing PathologyCase query #
location= c:\mydatabase\extract.sql
Found c:\mydatabase\extract.sql
```

The wizard will then ask what you want to do with the extracted results. You can either load them into the SPIN PEER & codebook or simply load the results into XML files.

```
# done configuring the database extractor #
Do you want load into the Peer and Codebook?  (default 'Y/n')
= y
# Submission Tool Keystore #
location= C:\keystore\myspinpeer.keystore
password= myspinpeer
private key alias= myspinpeer_spinkey
Keystore settings are valid!
```

After testing the keystore, the wizard will then prompt and test your codebook settings.

```
driver (default 'com.mysql.jdbc.Driver')
=
URL (default 'jdbc:mysql://localhost:3306/codebookTPC')
=
username (default 'spin')
=
password (default 'lcsspin')
=
Test Connection successfull !
driver=com.mysql.jdbc.Driver
URL=jdbc:mysql://localhost:3306/codebookTPC
username=spin
password=lcsspin
```

The wizard will then ask for the webservice address of the SPIN peer for submissions

```
# Submission Handler URL #
URL= https://vsl-taskbox.tch.harvard.edu:8080/jboss-net/services/SubmissionFacade
URL is well formed.
Writing configuration to 'C:\SPIN-VSL\ETL\ETLConfiguration.xml'
```

Done.

Running the ETL Program

Running ETL on Demand

Syntax:

```
ETL configurationFile

ETL configurationFile startDate

ETL configurationFile startDate endDate
```

Examples:

```
ETL myConfig.xml

ETL myConfig.xml "2005-01-10 5:00:00"

ETL myConfig.xml "2005-01-10 5:00:00" "2005-01-10 12:00:00"
```

Notes:

IF startDate and endDate are specified,   THEN ETL will process records in that date range.

IF startDate or endDate are NOT specified, THEN ETL will process all records.

IF endDate is NOT specified,   THEN ETL will process all records after startDate.

Dates must be in the format 'yyyy-MM-dd HH:mm:ss'

(See http://java.sun.com/j2se/1.5.0/docs/api/java/text/SimpleDateFormat.html)

Running ETL as a scheduled Task

The ETLRunner can be run on a routine basis using the scheduling abilities built into your operating system:

Windows (Schtasks):
http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/schtasks.mspx?mfr=true

Macintosh, Linux, or Unix (CRON)

http://uis.georgetown.edu/software/documentation/macosx1/macosx1.cron.html

http://www.scrounge.org/linux/cron.html
http://www.unixgeeks.org/security/newbie/unix/cron-1.html

Syntax:

```
ETLRunner configurationFile <interval unit> <interval duration> <optional date format>
```

Where <interval unit> is one of --seconds (-s), --minutes (-m), --hours (-h), or --days (-d)

Examples: (all use an interval of one hour.)

```
ETLRunner myConfig.xml --seconds 3600

ETLRunner myConfig.xml -m 60

ETLRunner myConfig.xml hours 1

ETLRunner myConfig.xml hours 1 yyyy-MM-dd HH:mm:ss

ETLRunner myConfig.xml hours 1 MM-DD-YYYY HH:mm:ss
```

SCRUBBER

The **SCRUBBER** can be setup to de-identify text "on the fly" before it is loaded into the peer database.
We recommend using the 2.8 scrubber because it has numerous upgrades. For more information, contact us.