LexEVS 5.0 Installation Guide

From Vocab_Wiki

Contents

Overview

This installation guide outlines the supported configurations and technical installation instructions for LexGrid Vocabulary Services for caBIG®, referred to as LexEVS for the remainder of the guide. Directions for configuring administrating the installation are also included in this document.

Examples and screenshots in this guide mix use of Unix and Windows platforms for installation. You will have to adapt for your OS.

Target Audience

The LexEVS Installation and Administrator guide is intended to provide detail instructions for installing and administrating LexEVS software. The details and configuration information contained are written for skill levels of a typical system or database administrator.

Overview of LexEVS Services

The LexEVS package set represents a compressive set of software and services to load, publish, and access vocabulary (and do so in a variety of web-enabled and grid environments.) Cancer Centers can use the LexEVS package set to install NCI Thesaurus and NCI Metathesaurus content queryable via a rich application programming interface (API). LexEVS services can be used in numerous applications wherever vocabulary content is needed.

LexEVS is intended to address the needs of the following groups:

- *Vocabulary service providers.* Describes organizations currently supporting externalized API-level interfaces to vocabulary content for the caBIG® community.
- *Vocabulary integrators.* Describes organizations within the caBIG® community that desire to integrate new vocabulary content to be served to the caBIG® community.
- *Vocabulary users.* Describes the caBIG® community interested in utilizing vocabulary services within the context of other caBIG® projects.

LexEVS Components (Overview)

1. **Service Management** consists of programs to load, index, publish, and manage vocabulary content for the vocabulary server.
2. **Application Programming Interface** (API) is comprised of methods to support Lexical Operations, Graph Operations, and History Operations.
3. **Documentation** consists of detailed JavaDocs and Programmers Guide.
4. **Examples** are provided as sample source code for common vocabulary queries.
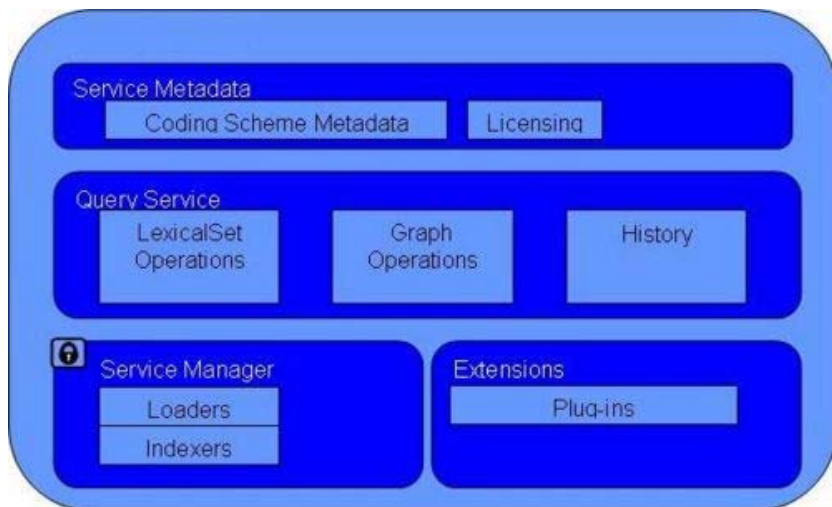5. **Test Suite** is provided to validate the LexEVS installation.

Figure 1 - LexEVS Service Components

NOTE:	Additional information about the LexEVS API is provided in the LexEVS Programmers Guide located at: `{LEXEVS_DIRECTORY}/docs/LexEVS_Programmer_Guide.pdf`

LexEVS Components (Detailed)

The LexEVS installation includes the following components:

- **Service Manager** or Administrative Programs for managing LexEVS server including Loaders, Indexers and Service Metadata Queries.
  - ActivateScheme
  - ClearOrphanedResources
  - CodingSchemeSelectionMenu
  - DeactivateScheme
  - ExportLgXML
  - ExportOBO
  - ExportOWL
  - ListExtensions
  - ListSchemes
  - LoadFMA
  - LoadHL7RIM
  - LoadLgXML
  - LoadMetadata
  - LoadNCIHistory
  - LoadNCIMeta
  - LoadNCIThesOWL
  - LoadOBO
  - LoadOWL
  - LoadRadLex
  - LoadUMLSDatabase
  - LoadUMLSFiles

- LoadUMLSHistory
- LoadUMLSSemnet
- RebuildIndex
- RemoveIndex
- RemoveMetadata
- RemoveScheme
- TagScheme
- TransferScheme
- **Query Service** including Program Examples for common vocabulary functions using sample vocabulary and CodedNodeSet functions
  - **Lexical Set Operations**
    - FindCodesForDescription
    - SoundsLike
    - Union
    - Intersection
    - Difference
  - **Graph Operations**
    - FindPropsAndAssocForCode
    - FindRelatedCodes
    - FindTreeForCodeAndAssoc
- LexEVS Automated Verification Test Suite
- LexEVS Runtime jar (combined archive)
- LexEVS Runtime components (combined archive with 3rd party jars outside of archive)
- LexEVS Uninstaller
- LexEVS License Terms and Conditions
- Configuration files to enable you to customize your installation to meet your specific database, server, and other network needs
  - llbconfig.props
- Documentation
  - JavaDocs
  - Links to:
    - LexEVS Programmer Guide
    - LexEVS Installation and Administration Guide

Deployment alternatives

The LexEVS local Runtime package has flexible database deployment alternatives depending on the underlying dbms. For dbms's like MySQL and PostGres the user can deploy terminologies contained in a single database or in multiple databases. Single database configurations allow the user to manage databases more effectively. Multiple data base configurations provide a little more transparency to the underlying terminology load. Some dbms's, like Oracle, require the single data base configuration.

Contacting Technical Support

LexEVS Application Support	caBIG® Vocabulary Knowledge Center Division of Biomedical Statistics and Informatics Mayo Clinic 200 1st ST SW Rochester, MN 55905 vocabkc@mayo.edu
LexBIG and LexEVS news, forums and Bug/Feature tracking	LexBIG and LexEVS information is maintained at a caBIG® Knowledge Center wiki [LexBIG and LexEVS news] [Discussion forums] [Trackers for feature and bug submissions]

Installing LexEVS with its LexGrid Model/DB (back-end)

Minimal System Requirements

- Internet connection
- 2 GB RAM
- Tested Platforms (Similar Hardware Specification for Operating System)

LexEVS has been tested on the platforms shown in Table 1.

	Linux Server	*Linux Server*	*Windows*
Model	HP Proliant DL 380	Penguin	Dell Latitude
CPU	2 x Intel® Xeon™ Processor 2.80GHz	Dual AMD Opteron 248 processors (64 bit)	1 x Intel® Pentium™ Processor 2.00GHz
Memory	4 GB	16Gb	1.5Gb
Local Disk	System 2 x 36GB (RAID 1) Data = 2 x 146 (RAID 1)	250 GB Raid 1 disk drive(s) 250 GB stand along disk drive	System 1 x 80GB
OS	Red Hat Linux ES 3 (RPM 2.4.21-20.0.1)	Fedora Core 3 (64 bit) OS	Windows XP Professional

Table 1 - Platform Testing Environment

Software Requirements

Required Software - Not Included in LexEVS

You must download and install the required software that is not included with LexEVS (listed in Table 2). The software name, version, description, and URL hyperlinks (for download) are indicated in the table.

Software Name	*Version*	*Description*	*URL*
Java Software Development Kit (SDK):Java 2 Standard Edition (J2SE)	j2sdk1.5.0_04 or higher	The J2SE Software Development Kit (SDK) supports creating J2SE applications	http://java.sun.com/javaee/downloads/
MySQL Database*	MySQL (5.0.45) or higher	MySQL 5.0 Community Edition	http://downloads.mysql.com/archives.php?p=mysql-5.0&v=5.0.45
PostgreSQL*	8.x or higher	Open source relational database management system	http://www.postgresql.org/
*MySQL or PostgreSQL installation is required.			

Table 2 - Required software and technology for LexEVS

Optional Software

Optional software to use with LexEVS is listed in Table 3. The included (**Incl**.) column indicates (with a **Yes**) if the software is packaged with the SDK. **No** indicates that you must supply the software. A hyperlink is included for your reference to appropriate sources.

Software Name	*Version*	*Description*	*URL*	*Incl.*
Eclipse IDE	3.4.x	An open platform for tool integration which provides tool developers with flexibility and control over their software technology used for product development. This tool can be optionally used to review Java source code.	http://www.eclipse.org/downloads/	No

Table 3 - Optional software and technology for LexEVS

|	
NOTE:	Some database drivers not included with the LexEVS installer. Downloaded drivers are placed in the

	{LEXEVS_DIRECTORY}/runtime/sqlDrivers and the {LEXEVS_DIRECTORY}/runtime-components/sqlDrivers directories .

Documentation

The following documentation is part of the LexEVS install package.

Useful tools included in LexEVS

- *Javadoc* – The full API is described using Javadocs. Your JavaDocs will be generated to the {LEXEVS_DIRECTORY} \doc\javadoc directory. Use a web browser to open the index.html file to start browsing documentation. For more information on Javadoc see http://java.sun.com/j2se/javadoc/.
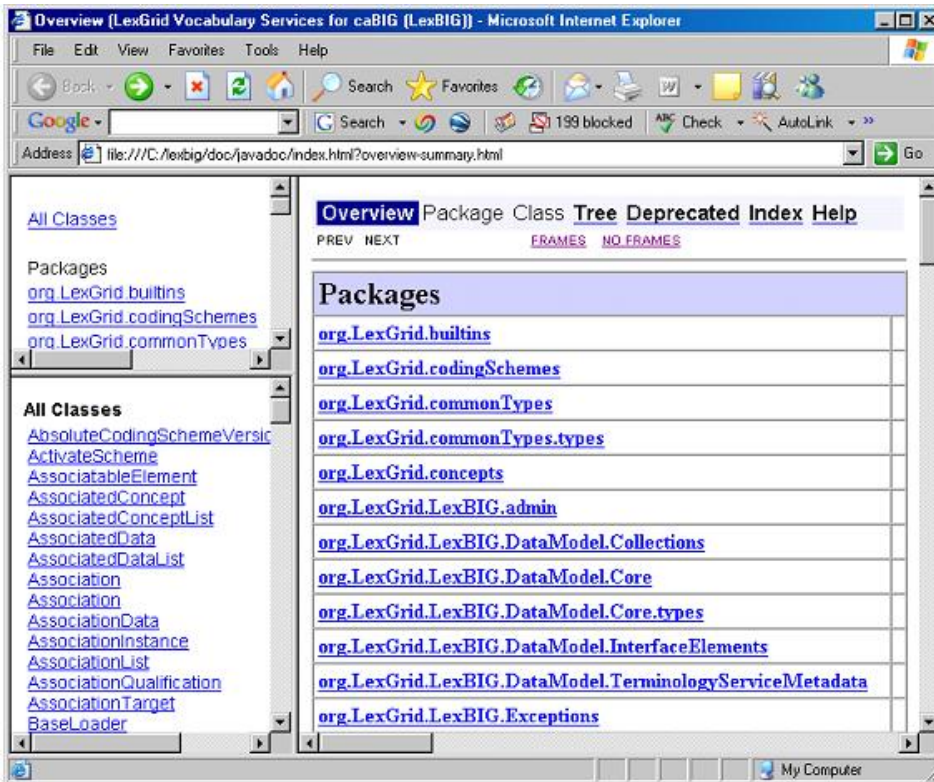
NOTE:	Main LexEVS Components listed above give an api summary. Full LexEVS javadocs are avalable here: LexEVS Javadocs zip file Javadocs can also be browsed at these locations: Local Runtime LexEVS Web-endabled LexEVS LexEVS Analytical Grid Services LexEVS Data Grid Services

What's Inside

This section describes the location and organization of installed materials. Following installation, many of the following hierarchy of files and directories will be available (some features are optionally installable):

<As located in the LexEVS installation root directory>

Directory	Description of content
	Installed by default. This directory provides a centralized point for command line scripts that can be

`/admin`	*executed to perform administrative functions such as the loading, activation/deactivation, and removal of vocabulary resources.* *Object code used to carry out these functions is included directly in the LexEVS runtime components. Source code is included in the* `/source` *directory in the* `lbAdmin-src.jar` *(described below).*
`/doc`	*Optionally installed. This directory provides documentation related to LexEVS services, configuration, and execution. This guide is distributed in the* `/doc` *top-level directory.*
`/doc/javadoc`	*This directory provides documentation for model classes and public interfaces available to LexEVS programmers. Also included with each object representation is a UML-based model diagram that shows the object, its attributes and operations, and immediately linked objects. The diagrams work to provide clickable navigation through the javadoc materials.*
`/examples`	*Optionally installed. This directory provides a small number of example programs.* *Refer to the* `README.txt` *file in this directory for instructions used to configure and run the example programs. The examples are intended to provide a limited interactive demonstration of LexEVS capabilities.* *Source and object code for the example programs is provided under the* `/examples/org` *subdirectory. Source materials are also centrally archived under the* `/source` *directory in the file* `lbExamples-src.jar`.

`/examples`
 	
`/resources`
	Contains sample vocabulary content for reference by the example programs; use the `/examples/LoadSampleData` *command-line script to load.*

`/gui`	*Optionally installed. This folder contains programs and supporting files to launch the LexEVS Graphical User Interface (GUI). The GUI provides convenient centralized access to administrative functions as well as support to test and exercise most of the LexEVS API.* *The GUI is launched using a platform-specific script file in the* `/gui` *directory. The name of the platform (e.g. Windows, OSX, etc) is included in the file name.* *Program source and related materials are centrally archived under the* `/source` *directory in the file* `lbGUI-src.jar`.
`/logs`	*Default location for log files, which can be modified by the LOG_FILE_LOCATION entry in the* `lbconfig.props` *file (see next section).*
`/resources`	*Installed by default. This directory contains resources referenced and written directly by the LexEVS runtime. It should, in general, be considered off-limits to modify or remove the content of this directory without specific guidance and reason to do so. Files typically stored to this location include the vocabulary registry (tracking certain metadata for installed content) and indexes used to facilitate query over the installed content.* *One file of particular interest in this directory is the* `/resources/config/lbconfig.props` *file. This file controls access to the database repository and other settings used to tune the LexEVS runtime behavior. Contents of this file should be set according to instructions provided by the* LexEVS Administrator's Guide.
`/runtime`	*Installed by default. This directory contains a Java archive (.jar) file containing the combined object code of the LexEVS runtime, LexEVS administrative interfaces, and any additional code they are dependent on. All required code for execution of LexEVS administrative and runtime services is installed to this directory.* ■ `/runtime/lbPatch.jar` *In the course of the product lifecycle, it is possible that smaller fixes will be introduced as a patch to the initially distributed runtime. Including this file in the classpath ensures automatic accessibility to the calling program without requiring adjustment. All patches are cumulative (there is at most one patch file introduced per release; all patch-level fixes are cumulative).* ■ `/runtime/lbRuntime.jar` *This is the standard runtime file, including all LexEVS and dependency code required for program execution except for SQL drivers (see next).*

/runtime /sqldrivers	*The JDBC drivers used to connect to database repositories are not included in the* `lbRuntime.jar`. *Instead, the runtime scans this directory for the drivers to include. This can be overridden by path settings in the* `lbconfig.props` *file.* ***Note:*** *while the LexEVS software package ships with JDBC drivers to certain open source databases such as mySQL and PostgreSQL, this folder provides a mechanism to introduce updated drivers or to add drivers for additional supported database systems.* *For example, the Oracle database is supported by the runtime environment. However, the drivers are not redistributed with the LexEVS software. To run against Oracle, an administrator would add a jar with the appropriate JDBC driver to this directory and then reference it in the* `lbconfig.props` *settings.*
/runtime- components	*Optionally installed. Due to license considerations for additional materials (as described by the* `license.pdf` *and* `license.txt` *files in the install directory), the cumulative runtime provided in the* `lbRuntime.jar` *is not redistributable.* *This directory contains a finer grain breakdown of object code into logical components and 3rd party inclusions. All components are redistributable under their own license agreements, which are provided along with each archive.* *The top-level of the* /runtime-components *directory contains all code produced for the LexEVS project in the* `lexbig.jar`. ***Note:*** *These files are included as an alternative to the* `lbRuntime.jar` *for code execution and redistribution. There is no need to include any of these files in the Java classpath if you are already including the* `lbPatch.jar` *and* `lbRuntime.jar` *described above.*
/runtime- components/ext Lib	*This subdirectory includes all 3rd party code redistributed with the LexEVS runtime, along with respective license agreements.*
/source	*Optionally installed. This directory provides central accessibility to Java source for all code developed for the LexEVS project.*
/test	*Optionally installed. This directory provides an automated test bucket that can be used by System Administrators to verify node installation. Note that the* /runtime/config/lbconfig.props *file must still be configured for database access prior to invoking the test bucket.* *Testcases are launched via the TestRunner command-line script. Several reporting options are provided and are further described in the* LexEVS Administrator's Guide.
/uninstaller	*Contains an executable jar that can be invoked by an administrator to uninstall files originally introduced by the LexEVS installation.*

Installation

Preliminary Considerations

BEFORE YOU BEGIN	LexEVS has been tested with the operating systems and hardware specified earlier in this guide. While LexEVS is expected to run on many variations of hardware and software similar to the test platforms, results cannot be guaranteed.

LexBIG Object Model

To describe LexEVS, the LexEVS service, CodeNodeGraph and CodeNodeSet interfaces are included. The model, as shown in Figure 2, contains the core query service from the `org.LexBIG.LexBIGService` domain package. The full and most recent version of the object model is described and illustrated as part of the JavaDocs.

Figure 2- LexBIGService Model

| NOTE:	Figure 2 is a UML class diagram. For more information about UML, see the LexEVS Programmers Guide at https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexEVS_5.0_Programmer%27s_Guide
	For more information on the LexEVS architecture see the architecture and design guide here: https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexEVS_5.0_Design_and_Architecture_Guide

After successfully installing LexEVS and running the verification test suite, as described in this guide, you should be ready to start programming using the API to meet the needs of application needs. If you have the required software installed on your system (see the previous section), then installing and running the test should not take more than 60 minutes.

Downloading and Using the LexEVS local runtime Install Wizard

	To best understand the installation and testing procedures for LexEVS, it is recommended that you follow the procedures described in this section with minimal deviation.
SUGGESTION	

Complete the following steps to download LexEVS:

Step	Action
1	Download the LexEVS local runtime install package here: https://gforge.nci.nih.gov/docman/view.php/491/17562/LexEVS_50_localRuntimeAndGUI_installer.jar
2	Select the most recent version of the LexEVS Software Package LexEVSxx-install.jar. Save this file to your computer. This location will be referred to as the SAVE_DIRECTORY. You may have to disable Pop-up blockers to allow save the install package to your local computer.
	Using Microsoft Windows Environment Command Prompt change directory to the SAVE_DIRECTORY of the LexEVS software package you saved in step 2. At the command prompt enter the following command to begin the installation wizard. Enter ***java -jar LexEVSxx-install.jar*** As an alternative to the command line instruction you can navigate to the SAVE_DIRECTORY with the File Explorer. Double Click on the LexEVSxx-install.jar file. This will launch the install wizard with a typical java installation
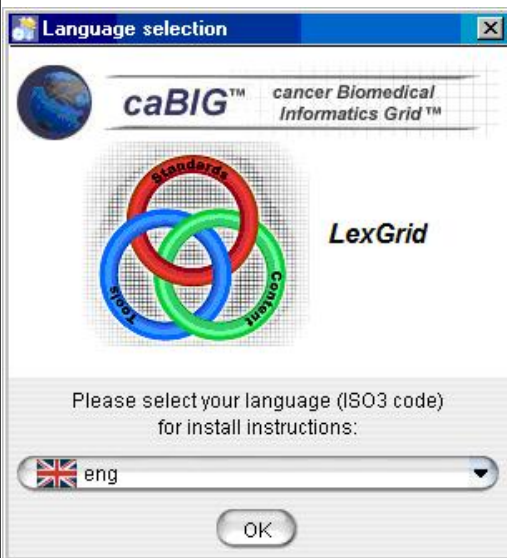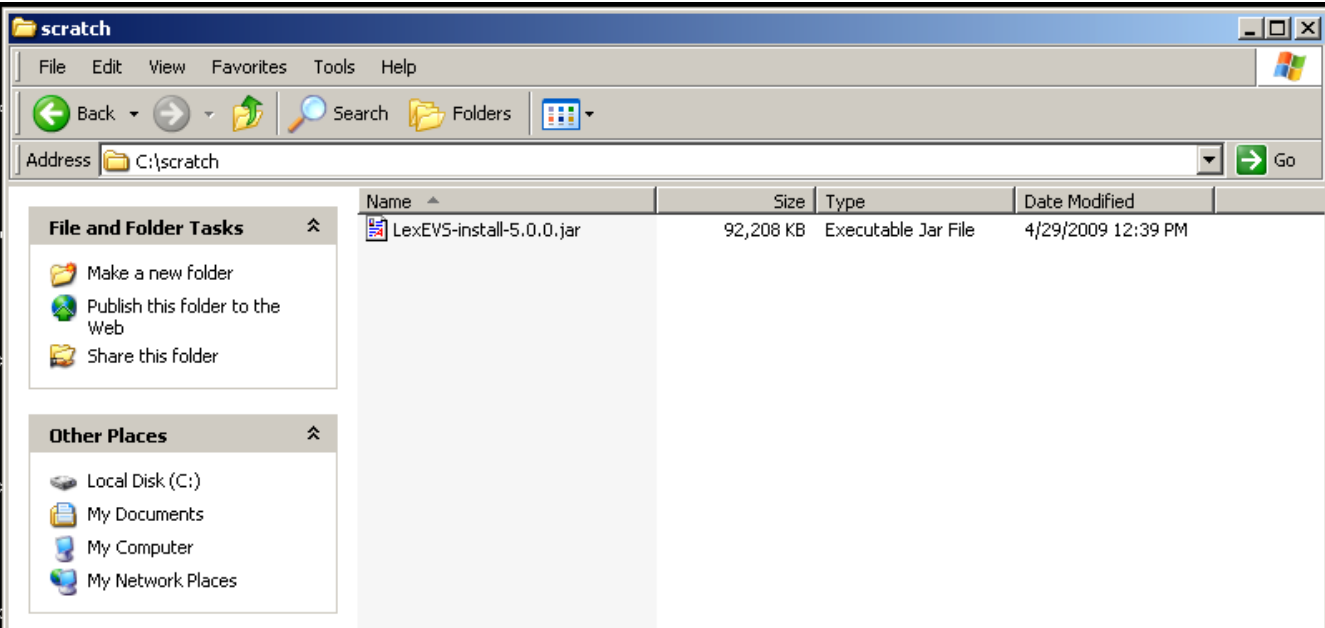
3	 Click **OK** button to begin the installation. **Note:** The only language currently supported is English.
4	After the initial welcome screen the release notes for the LexEVS distribution are displayed. Once you have read through the release notes click the **Next** button to continue.

| 5 | The next step is to review the license agreement of the LexEVS software and accept the terms of the agreement. Click **Next** button to continue with installation. |

Enter the path where you would like the LexEVS software installed. Click the **Next** button to continue installation. This will be referred to as the LEXEVS_DIRECTORY throughout the remaining instructions.

Note: If the directory does not exist, the program will prompt to proceed with creating the new directory. If the directory does exist, the program will prompt to overwrite the directory and files in the installation path.

Select the components to be installed for LexEVS. Two of the components LexEVS Runtime and Admin Toolkit and LexEVS Info are required and cannot be unchecked. The remaining components are optional. Once components have been

7	decided click the **Next** button to continue with installation.
8	Once all the components have been installed a "Finished" prompt will be displayed. Click the **Next** button to continue installation.
9	The last step of the installation wizard provides the ability to generate an automatic installation script that can be used on other machines. This installation script can be used to install LexBIG without graphic wizard. Click **Done** to complete the installation process.
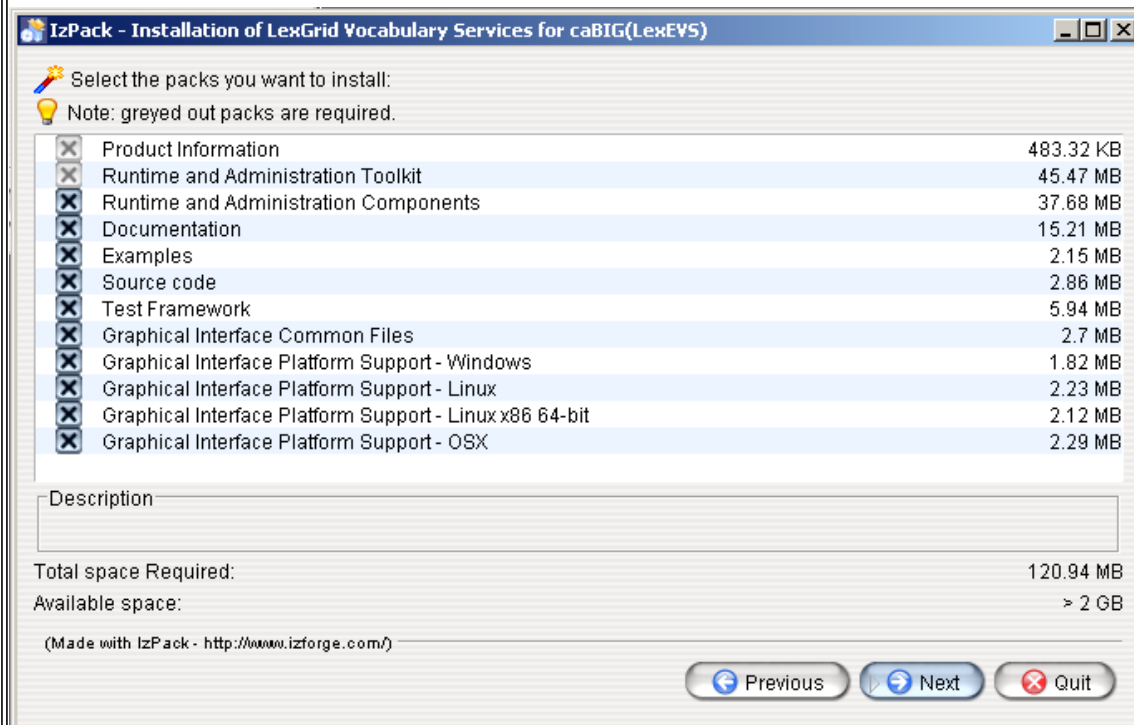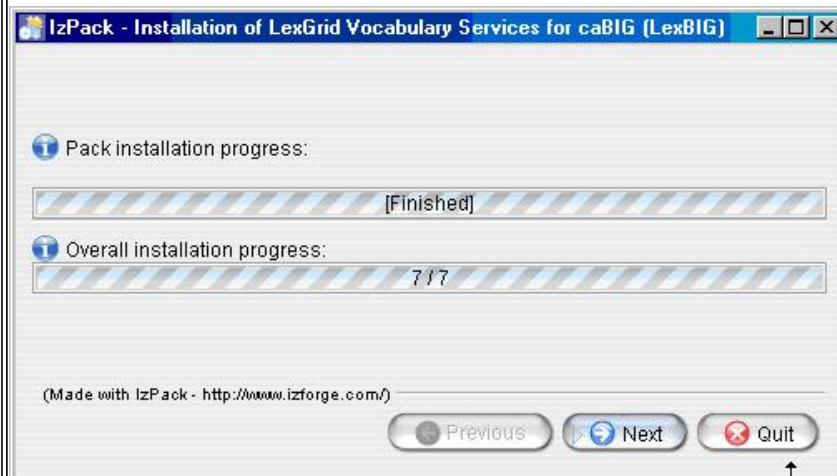
Installing LexEVS Using Command Line

SUGGESTION	To best understand the installation and testing procedures for LexEVS, it is recommended that you follow the procedures described in this section with minimal deviation.

Complete the following steps to download and install LexEVS using command line option:

Step	Action
1	Download the LexEVS local runtime install package here: https://gforge.nci.nih.gov/docman/view.php/491/17562/LexEVS_50_localRuntimeAndGUI_installer.jar

Download the command line configuration xml file here: https://gforge.nci.nih.gov/docman/view.php/491/17609/LexEVS-install-config-5.0.0.xml |
2	Select the most recent version of the LexEVS Software Package LexEVSxx-install.jar (e.g. 1.0.1, lexbig-install-1.0.1.jar). Save this file to your computer. This location will be referred to as the SAVE_DIRECTORY. You may have to disable Pop-up blockers to allow save the install package to your local computer.
3	Select the install-config.xml file. Save this file to your SAVE_DIRECTORY.
4	Edit the install-config.xml file to configure the components to be installed. The install path can be modified to the location of choice. Components LexEVS Runtime and Admin Toolkit and LexEVS Info are required. Remove lines in install-config.xml file for those components you do not want to be installed. By default, the file is configured to install all packages.

```
19
20  -->
21  <AutomatedInstallation langpack="eng">
22      <com.izforge.izpack.panels.HelloPanel/>
23      <com.izforge.izpack.panels.InfoPanel/>
24      <com.izforge.izpack.panels.LicencePanel/>
25      <com.izforge.izpack.panels.TargetPanel>
26          <installpath>\lexbig\</installpath>
27      </com.izforge.izpack.panels.TargetPanel>
28      <com.izforge.izpack.panels.PacksPanel>
29          <selected>
30              <!-- #0: LexBIG Runtime and Admin Toolkit - REQUIRED.
31                   Installs a single Java archive containing all code required for invocation of the LexBIG Java API, and r
32              <pack index="0"/>
33              <!-- #1: LexBIG Runtime - Components - OPTIONAL.
34                   Installs redistributable components and associated terms and conditions for the LexBIG Java API.  -->
35              <pack index="1"/>
36              <!-- #2:  LexBIG Examples - OPTIONAL.
37                   Installs example code for running common tasks against the LexBIG API. -->
38              <pack index="2"/>
39              <!-- #3: LexBIG Source code - OPTIONAL.
40                   Installs an archive containing source code for LexBIG Java classes. -->
41              <pack index="3"/>
42              <!-- #4: LexBIG Test Framework - OPTIONAL.
43                   Installs programs for automated validation of the LexBIG runtime. -->
44              <pack index="4"/>
45              <!-- #5: LexBIG Documentation - OPTIONAL.
46                   Installs administration and programmer guides, including JavaDoc. -->
47              <pack index="5"/>
48              <!-- #6: LexBIG Info - REQUIRED.
49                   Informational materials, including readme and license terms. -->
50              <pack index="6"/>
51          </selected>
52      </com.izforge.izpack.panels.PacksPanel>
53      <com.izforge.izpack.panels.InstallPanel/>
54      <com.izforge.izpack.panels.FinishPanel/>
55  </AutomatedInstallation>
```

At command prompt in the SAVE_DIRECTORY enter the command:

java -jar LexEVSxx-install.jar install-config.xml

```
C:\scratch>java -jar lexbig-install-1.0.0.jar install-config.xml
[ Starting automated installation ]
[ Starting to unpack ]
[ Processing package: LexBIG Runtime and Admin Toolkit (1/7) ]
[ Processing package: LexBIG Runtime - Components (2/7) ]
[ Processing package: LexBIG Examples (3/7) ]
[ Processing package: LexBIG Source code (4/7) ]
[ Processing package: LexBIG Test Framework (5/7) ]
[ Processing package: LexBIG Documentation (6/7) ]
[ Processing package: LexBIG Info (7/7) ]
[ Unpacking finished. ]
[ Writing the uninstaller data ... ]
[ Automated installation done ]

C:\scratch>
```

Configuring the LexEVS environment

The LexEVS install provides a `lbconfig.props` file to configure options for the LexEVS service and database settings. The LexEVS Service can be configured to work with many different databases – but the recommended databases are MySQL 5.0.45 (or higher) or PostgreSQL 8.x. Following installation, the Administrator should examine the lbconfig.props file and make any changes required to match the target database and runtime environment.

Server Properties

Modifying the

- The file `{LEXEVS_DIRECTORY}/resources/config/lbconfig.props` contains properties

lbconfig.props file for LexEVS	controlling the behavior of the LexEVS runtime.

- This guide has an overview of the options in this file – however the file also has documentation embedded inside of it. The documentation inside the `lbconfig.props` file should be considered authoritative if there is a conflict between the documentation written here and that contained in lbconfig.props.
- Table 4a contains the variables that you must modify so that LexEVS can properly use your database.
- Table 4b contains the variables that you can change for performance reasons or alternative deployment scenarios, but you probably don't need to change in a standard LexEVS installation.
- When constructing file paths, you must use either '/' or '\\'. '\' is not valid within the lbconfig.props file for file paths (it is ok for JDBC connection strings).

Table 4a Variables that must be set prior to use of LexEVS	
Property Name	**Description**
`SINGLE_DB_MODE`	LexEVS can be configured to run within a single database (and it will use a numbering scheme on its tables) – or it can be configured to use multiple databases on a single server (and it will use a numbering scheme on its databases). It is completely up to the administrator which way will work better in their database environment. The default value is '`false`' – which will cause it to use multiple databases on a server.

`DB_URL`	The address of your database server. The value that you put here will be dependant on the `SINGLE_DB_MODE` variable.
	
	If `SINGLE_DB_MODE` is '`true`' then this value should be a complete path that includes the DB name. For example:
	
	`DB_URL=jdbc:mysql://hostname/LexEVSDB`
	
	If `SINGLE_DB_MODE` is '`false`' then this value should be a path that does NOT include the DB name. For example:
	
	`DB_URL=jdbc:mysql://hostname/`

`DB_PREFIX`	The prefix to use on all tables or databases that LexEVS creates. If `SINGLE_DB_MODE` is '`true`' then this prefix will be used on tables. If `SINGLE_DB_MODE` is '`false`' then this value will be used on databases. **Note:** If you wish to run multiple LexEVS installations on the same database server, give them each a unique prefix. **Note:** Do not use dashes '-' in the db_prefix value. Recommended characters are alphanumeric (a-z, 0-98) and underscore '_'. **Note:** If your database is Oracle, you may not use this feature. Leave the value blank.
`DB_PARAM`	Optional variable for passing extra database parameters. These will be appended to the end of the database connection string.
`DB_DRIVER`	The Java class name that represents the driver that you wish to use with your database.

`DB_USER`
 	
`DB_PASSWORD`
	The database username and password.
	
	If `SINGLE_DB_MODE` is '`true`' this account must have permission to add and remove tables, indexes, etc inside of this database.
	
	If `SINGLE_DB_MODE` is '`false`' this account must have permission to create and drop new databases.

Table 4a - LexEVS configuration parameters

Table 4b Variables that must be set prior to use of LexEVS	
Property Name	**Description**
	This is not actually a variable that you would set within this file. It is documented here for

LG_CONFIG_FILE	clarity of other variables that depend on this variable. Normally, this variable is automatically set (at runtime) to the location of the lbconfig.props file that is being used by the runtime. Alternatively, you can set the java system variable `LG_CONFIG_FILE` at system startup to point to the `lbconfig.props` file that you want LexEVS to use. Refer to additional documentation in the `lbconfig.props` file if you need to use this feature.
LG_BASE_PATH	This variable is the path that will be used to resolve any other relative (or unqualified) paths in the `lbconfig.props` file. This variable is optional, and should usually be left blank. If this variable is left blank, it will automatically be set (at runtime) to the location of the folder which contains the `lbconfig.props` file that the system was started with. This variable can also be overridden by setting the java system variable `LG_BASE_PATH`.
JAR_FILE_LOCATION	The path of the folder that contains your SQL drivers and LexEVS extensions (if you have any). This value can be relative to the `LG_BASE_PATH` or absolute.
REGISTRY_FILE	The location of the file that will store information about all loaded terminologies. This value can be relative to the `LG_BASE_PATH` or absolute.
INDEX_LOCATION	The folder where all LexEVS generated indexes will be stored. This folder can potentially be large (several GB) – depending on the terminologies loaded. This value can be relative to the `LG_BASE_PATH` or absolute.
MAX_CONNECTIONS_PER_DB	LexEVS maintains a pool of connections to each database that it connects to. This sets the limit on the number of connections that will be opened. If SINGLE_DB_MODE is `true` you may want to set this to a higher value – 20 or so (depending on expected user load) If SINGLE_DB_MODE is `false` you should keep this value smaller – the default is 8.
CACHE_SIZE	LexEVS maintains an internal cache of some information that it needs to query from the database to resolve queries. This controls the size of that cache. This cache does not hold entire user queries. The default size is 500.
ITERATOR_IDLE_TIME	The length of time to allow an unused (and unclosed) iterator to stay valid before it is closed (and its resources freed) by the system.
MAX_RESULT_SIZE	This controls the maximum number of results that a user can resolve at one time for the CodedNodeSets and CodedNodeGraphs. Iterators are not limited by this value.
LOG_FILE_LOCATION	The path where LexEVS log files will be written. This value can be relative to the `LG_BASE_PATH` or absolute.
DEBUG_ENABLE	Setting debug to `true` will give you more verbose logging information to debug problems. The default setting is `false`. This should normally be set to `false` since debug logging causes a negative performance impact.
LOG_CHANGE	Indicates when a new log file should be started. This can be set to `monthly`, `weekly` or `daily`. This can also be set to a number – which will cause it to start a new log file whenever it reaches X MB in size.
	If 'LOG_CHANGE' is set to 'daily', 'weekly', or 'monthly', this variable instructs the service to remove log files that have not been written to in X days.

ERASE_LOGS_AFTER	**Note:** The unit is treated as days regardless of the LOG_CHANGE value. Cleanup will only occur on restart of the JVM. If 'LOG_CHANGE' is set to a number, this is the number of old log files that will be kept.
EMAIL_ERRORS	Used to enable or disable e-mail notification of system errors and warnings. Default is 'false'. If you set this to 'true', you must set the next two variables.
SMTP_SERVER	The SMTP server to use to send errors over e-mail. Only applicable when EMAIL_ERRORS is set to 'true'.
EMAIL_TO	A comma separated list of e-mail address to set failure and warning notifications to. Only applicable when EMAIL_ERRORS is set to 'true'.

Table 4b - LexEVS configuration parameters

  Attention	It is considered beyond the scope of this manual to address database (e.g. MySQL or PostgreSQL) setup and administration. However, proper database configuration is critical to the performance and long-term health of the LexEVS environment. System administrators should consult the MySQL or PostgreSQL documentation to determine settings that are appropriate to the host machine and environment. Tuning should be performed prior to loading vocabularies. The following tables provide settings that have been modified in database environments used during LexEVS development and adoption, and are provided for consideration by database administrators.

MySQL properties

Modifying the my.ini file for MySQL
- The file {MYSQL_HOME_DIRECTORY}/my.ini contains properties controlling the behavior of the MySQL database server.

Property Name	*Description*
innodb_flush_log_at_trx_commit	Flush the transaction logs at each commit. Value: It is **highly recommended** that this option be set to '0' in Windows installations to improve load performance.
innodb_additional_mem_pool_size	Additional memory pool that is used by InnoDB to store metadata information. Value: 16M
innodb_buffer_pool_size	Buffer pool used to cache both indexes and row data. Value: 1G (consider going higher based on physical RAM available)
tmp_table_size	Maximum size for internal (in-memory) temporary tables. Value: 256M
query_cache_size	Query cache is used to cache SELECT results and later return them without actually executing the same query once again. Value: 64M
sort_buffer_size	This buffer is allocated when MySQL needs to rebuild the index in REPAIR, OPTIMZE, ALTER table statements as well as in LOAD DATA INFILE into an empty table. Value: 16M

Table 5 – MySQL configuration parameters

PostgreSQL properties

Modifying the posgresql.conf File for PostgreSQL
- The file {PostgreSQL_HOME_DIRECTORY}/posgresql.conf contains properties controlling the behavior of the PostgreSQL database server.

Property Name	Description
shared_buffers	Number of shared buffers. Value: 1000.
work_mem	The amount of memory in kilobytes allocated to working memory Value: 51200.
maintenance_work_mem	The amount of memory in kilobytes allocated to maintenance working memory. Value: 512000.
enable_seqscan	We set the 'enable_seqscan' to false to use always use an index versus a table scan.

Table 6 – PostgreSQL configuration parameters

Note: MySQL can be passed a jdbc option for the Windows local environment that may improve perfomance 30 to 50%.

Try the following values in the lbconfig.props file for the DB_URL:

```
SINGLE_DB_MODE=true DB_URL=jdbc:mysql:///<dbname>?socketFactory=com.mysql.jdbc.NamedPipeSocketFactory
```

This uses Windows Named Pipe function and avoids use of the TCP/IP protocol. It only works when connecting with a local iteration of the MySQL database on Windows.

Testing the LexEVS Configuration

This LexEVS installation provides a test suite to verify and test the environment.

Note	The LexEVS runtime and database environments must still be configured prior to invoking the test suite, as described above.
Step	**Action**
1	In a **Command Prompt** window, enter *cd {LEXEVS_DIRECTORY}/test* to go to the test directory
2	Run the TestRunner utility to start the test process. For Windows Environment enter *> TestRunner.bat -h* For Linux Environment enter *> TestRunner.sh -h*
	Use file explorer to navigate to the directory that contains the test report. The report is placed in the {LexEVS_DIRECTORY}/test.
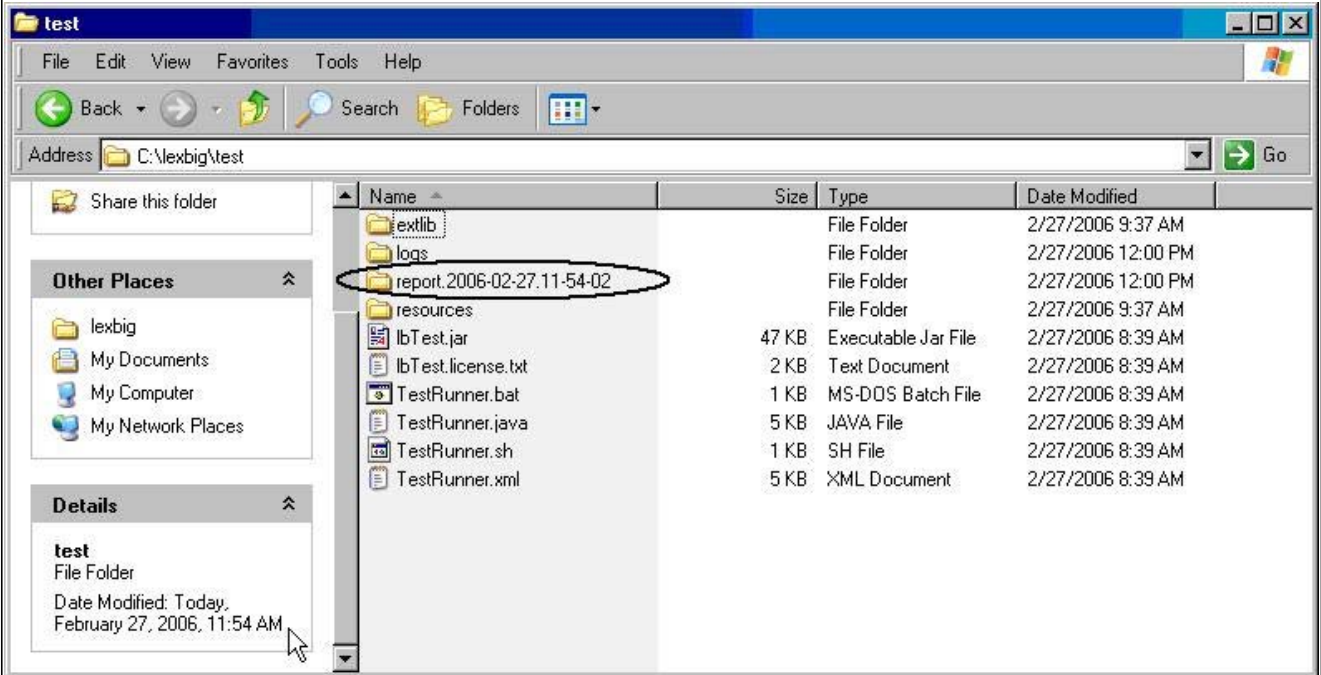
3   Navigate to the report that represents the date and time you executed the test.
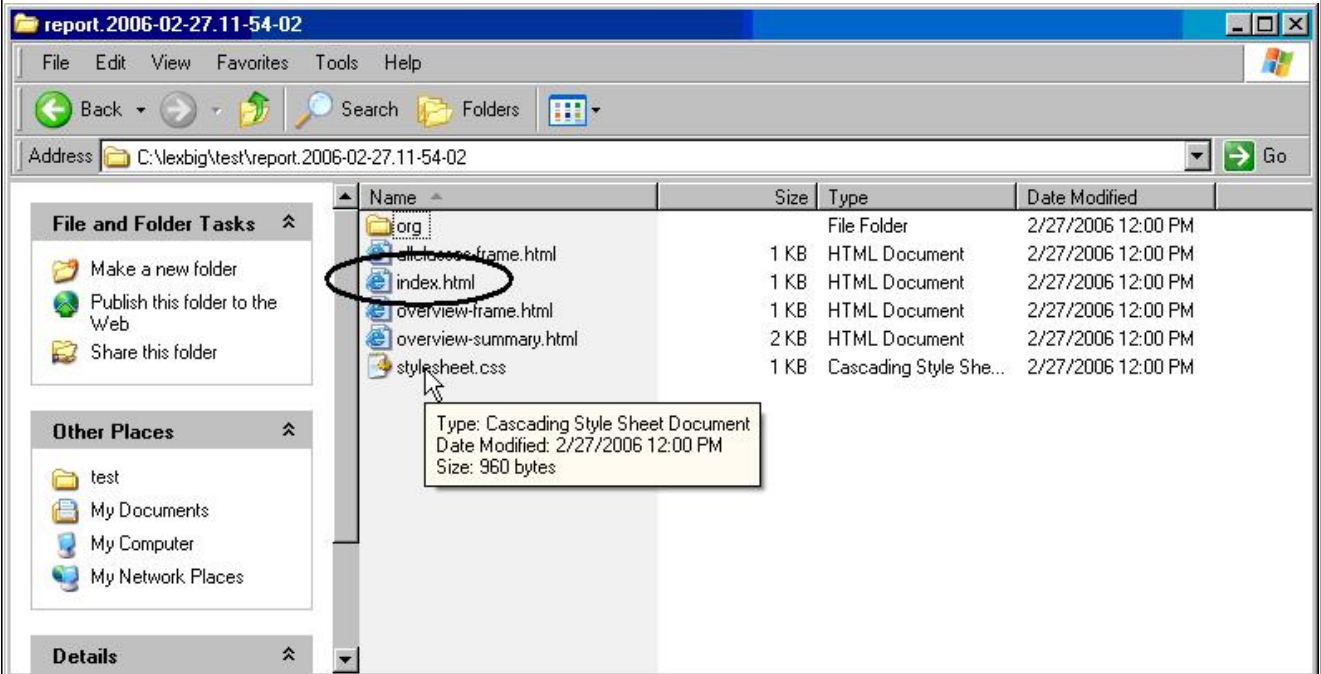
Review the test results opening index.html file using a web browser.

Name	Tests	Errors	Failures	Time(s)
AllTests	6	0	0	362.310

Tests

Name	Status	Type	Time(s)
testT1_ADM_06_1a	Success		16.764
testT1_ADM_06_1b	Success		16.163
testT1_ADM_08_3a	Success		329.023
testT1_ADM_03_1a	Success		0.020
testT1_ADM_03_1b	Success		0.020
testT1_ADM_03_1c	Success		0.020

Congratulations! If the test passes all tests, you have successfully installed the LexEVS software.

Troubleshooting

Warning	**Tips and pitfalls**

Upgrading LexEVS
Upgrading LexEVS may require reloading content. Be sure to read the release notes for each LexEVS release before installing the latest version. Preserve configuration files and indexes unless instructed to reload or do otherwise in the release notes. These files include config.props, registry.xml and the entire lbIndex directory in the resources directory.

Single and Multiple Database Configuration Changes
Do not attempt to change database configurations from single database mode to multi-database mode and vice versa after loading in one mode or the other. LexEVS does not support both configurations at once.

Failed and Interrupted Loads
LexEVS loads of content are generally handled in a robust manner and failed loads clean up after themselves relatively well when dbms' are properly configured to allow database drop functions by LexEVS. However, exiting the application in the middle of a load may cause unpredictable consequences with databases, indexes and lock files left in a state that will cause subsequent loads of the same terminology to fail.

- Often these can be remedied by deleting the lock.xml file followed by using the cleanup function.
- Other steps may need to be taken if this doesn't work, including dropping databases as a dba, deleting the index file for the offending terminology, and editing the registry.xml and metadata.xml files by hand.
- A quick, dirty solution is to drop all databases and delete all configuration files except config.props.

When to Scale a Dbms for a Large Number of Connections

If LexEVS is configured for multi-database loads and has multiple users connecting to all terminologies, then the administrator will need to scale database configurations to adapt to this. If you have large number of terminologies loaded and a large user base connecting to the service using the lbGUI, then you will need the database configuration for number of connections scaled appropriately or users may not be able to connect. *Note: Loading in single database mode can eliminate this problem.*

MySQL Driver Issues

LexEVS is no longer distributed with a Java MySQL driver due to licensing concerns. If LexEVS reports an error concerning establishing a connection to the MySQL server yet the MySQL CLI is able to connect, an adjustment in the version of Connector/J may be required.

The latest version of Connector/J is available from MySQL.org. The new jar should be placed in the LexGrid/LexBIG/2.0.0/runtime/sqlDrivers/ directory. If you are changing drivers remove the existing driver jar file to ensure that the class loader does not incorrectly load the older driver file.

MySQL Perfomance Problems

If the user experiences slow performance when loading or accessing terminologies a review of the suggested configuration parameters for the dbms is recommended. Pay particular attention to the MySQL values.

Installing LexEVS in a Web based environment

Minimal System Requirements

- Internet connection
- 2 GB RAM
- Tested Platforms (Similar Hardware Specification for Operating System)

LexEVS has been tested on the platforms shown in Table 1.

	Linux Server	*Linux Server*	*Windows*
Model	HP Proliant DL 380	Penguin	Dell Latitude
CPU	2 x Intel® Xeon™ Processor 2.80GHz	Dual AMD Opteron 248 processors (64 bit)	1 x Intel® Pentium™ Processor 2.00GHz
Memory	4 GB	16Gb	1.5Gb
Local Disk	System 2 x 36GB (RAID 1) Data = 2 x 146 (RAID 1)	250 GB Raid 1 disk drive(s) 250 GB stand along disk drive	System 1 x 80GB
OS	Red Hat Linux ES 3 (RPM 2.4.21-20.0.1)	Fedora Core 3 (64 bit) OS	Windows XP Professional

Table 1 - Platform Testing Environment

Software Requirements

Required Software - Not Included in LexEVS

You must download and install the required software that is not included with LexEVS (listed in Table 2). The software name, version, description, and URL hyperlinks (for download) are indicated in the table.

(Required software that *is* included with the LexEVS is listed in Appendix I)

Software Name	*Version*	*Description*	*URL*

Java Software Development Kit (SDK):Java 2 Standard Edition (J2SE)	j2sdk1.5.0_04 or higher	The J2SE Software Development Kit (SDK) supports creating J2SE applications	http://java.sun.com/javaee/downloads/
MySQL Database*	MySQL (5.0.45) or higher	MySQL 5.0 Community Edition	http://downloads.mysql.com/archives.php?p=mysql-5.0&v=5.0.45
PostgreSQL*	8.x or higher	Open source relational database management system	http://www.postgresql.org/
*MySQL or PostgreSQL installation is required.			

Table 2 - Required software and technology for the LexEVS

NOTE:	Some JDBC drivers are included with LexEVS installer others will have to be downloaded by the user due to licensing considerations. These drivers are placed in the `{LEXEVS_DIRECTORY}/runtime-components/extLib` directory.

Installation

Since LexEVS is server oriented software, a common deployment scenario is to package LexEVS into a WAR file and deploy it into JBoss or another Application Server. The following are recommendations for packaging and configuring LexEVS for JBoss usage.

If not deploying into an Application Server such as JBoss, this chapter can be skipped. While these instructions are tailored to JBoss, they should serve as a reasonable guide in packaging for other Application Servers.

Complete a standalone installation of LexEVS before starting this section. These files will be copied to create your JBoss package.

Creating your WAR file

To add LexEVS functionality to a WAR file, add the `lexbig.jar` file to the lib folder of the WAR file. Adding the `lbRuntime.jar` file to the lib folder may result in the duplication of many common libraries that LexEVS and JBoss share. In particular errors will occur due to a collision between the log4j libraries that LexEVS and JBoss both use.

The `lexbig.jar` can be found in the `'runtime-components'` folder of the LexEVS installation.

The `lexbig.jar` file adds LexEVS runtime functionality to JBoss; however, 3rd party dependencies must be added to the WAR file lib folder as well. These jar files are located in the `'runtime-components/extLib'` subfolder shown below.
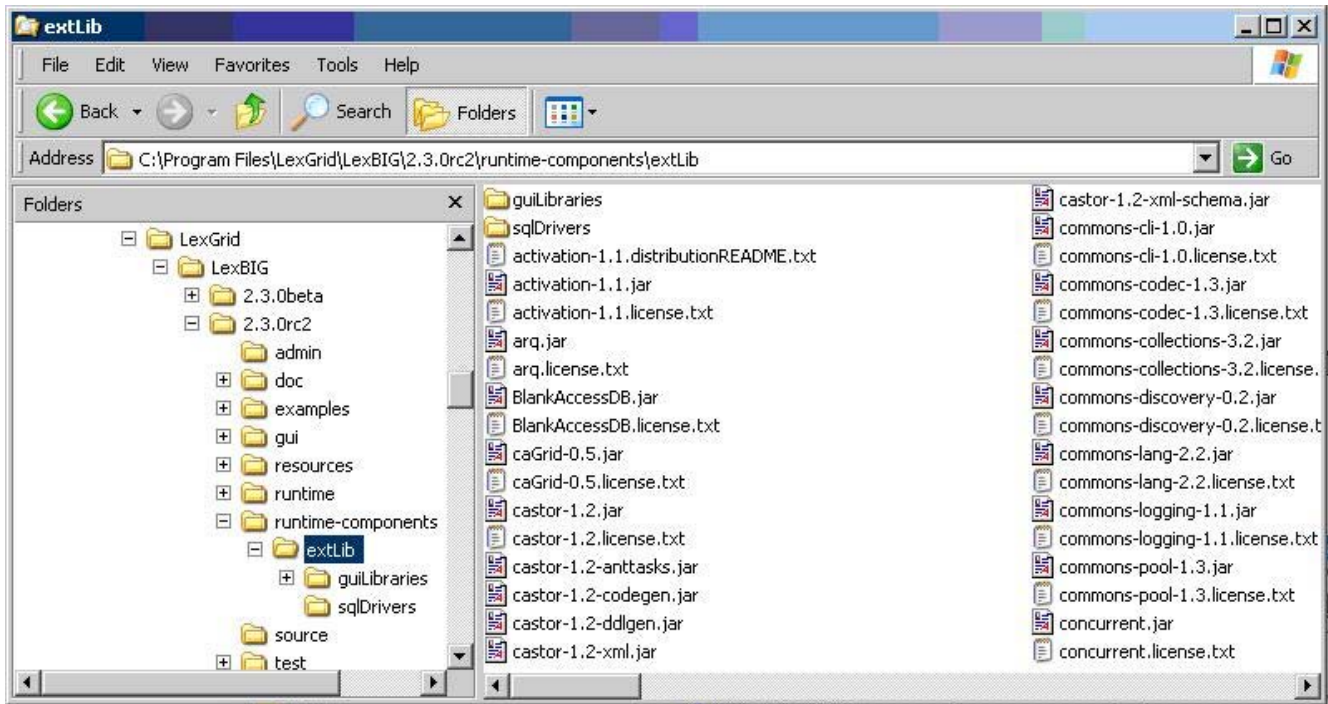
Avoid duplicating jars already contained in the WAR package or jar files that JBoss already provides. For example, JBoss already provides a log4j `x.x.x.jar'` – this file should NOT BE COPIED into the WAR package. **Note:** this jar is singled out because it is known to cause errors in JBoss if you deploy a WAR file that contains a log4j library.

Do not copy files from `'runtime-components/extLib/guiLibraries'` or any of the sql drivers in the `'sqlDrivers'` folder. The gui libraries are not needed and the sql drivers will be dealt with later.

Creating the file storage location for the LexEVS runtime

LexEVS runtime needs to have the ability to read and write from a local file system in order to function properly. The files that it writes also need to persist through a stop and start of the JBoss server. These files cannot be placed inside of the WAR file, since JBoss extracts WAR files into a differently named temporary folder every time it starts up.

Create a single folder to store this information, and configure the LexEVS JBoss instance to use this folder while it is running for all of its file system access needs.

Below is a screenshot of the file structure to create:
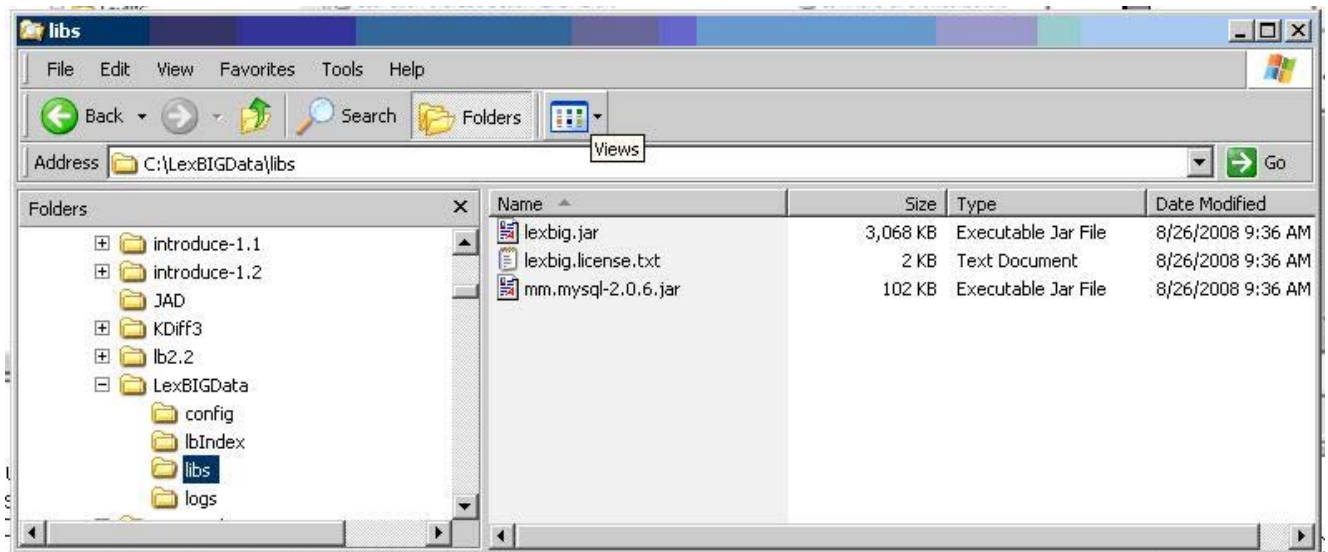
Create a top level folder named 'LexBIGData'. Create sub-folders in this directory named 'config', 'lbIndex', 'libs', and 'logs'.

The folder 'config' is used for storing the LexEVS configuration, storing the terminology registry, and storing lock files. Copy 'LexEVS <version>\resources\config\config.props' into this folder now. This file will be customized later.

The folder 'lbIndex' is where LexEVS will store the indexes that it creates. Leave this folder empty.

In the folder 'libs' copy the SQL drivers. You may have to download drivers compatable with the database verion your LexEVS application connects to. Any LexEVS extensions can also be placed here. Copy the jar 'LexEVS 5.0\runtime-components\lexbig.jar' into this folder. A duplicate of this jar file needs to be here to assist with some custom classloader work that bypasses the normal classpaths.

The folder 'logs' is where LexEVS will write its log messages. Leave this folder empty.

Your directory structure should now look like the screen shot above.

Configuring lbconfig.props for JBoss deployment

There are a few variables in the lbconfig.props file in your LexBIGData folder that will need to be changed for the JBoss deployment. There are a number of ways that this file can be configured – but these instructions set up directories relative paths – allowing the WAR file to be easily moved from one server to another.

The lbconfig.props file should already be configured for standalone mode at this point, as described earlier in the manual. This section will only describe the changes that should be made for JBoss deployment.

There following five variables need to be set to the following values in the lbconfig.props file.

LG_BASE_PATH= This variable is being left blank. This will cause LexEVS to automatically set this variable to the location of the lbconfig.props file when it starts up. All other paths in the lbconfig.props file can then be set relative to the folder that contains the lbconfig.props file that was found when the system started up – and it should use the lbconfig.props file in your LexBIGData folder.

JAR_FILE_LOCATION=../libs/ This variable is being set to point to the 'libs' folder that we created in the LexBIGData folder. The two periods tells it to go up one folder from the place where it started (the config folder) and then down into the libs folder.

REGISTRY_FILE=registry.xml By only putting a file name here, a file of that name will be created in the same folder where it started (the config folder in the LexBIGData folder).

INDEX_LOCATION=../lbIndex Sets the index storage location to the lbIndex folder in the LexBIGData folder.

LOG_FILE_LOCATION=../logs Sets the log location to the logs folder in the LexBIGData folder.

Locating the file storage location for the LexEVS runtime

Now that you have created your 'LexBIGData' folder, and configured the config.props file, you need to decide where to place it on the system where JBoss and your WAR file will be deployed. Location is up to the user – so long as it is permanent. A couple of possibilities include keeping it with JBoss -- in the JBoss data foler,, '\…\jboss-4.0.4.GA\server\default\data\LexBIGData,' or it could go into a typical Linux application folder such as /apps/LexEVS/LexBIGData/.

Locating the lbconfig.props file when JBoss starts

The only step remaining now is to help LexEVS find the proper lbconfig.props file when it initializes. There are a number of ways that this could be done – but the easiest way is to set the Java System Variable LG_CONFIG_FILE to the location of the lbconfig.props file.

This can be done by modifying the script that launches JBoss – adding the following (modify as appropriate for the place that you picked for your LexBIGData folder)

-DLG_CONFIG_FILE=C:\Program Files\jboss-4.0.4.GA\server\default\data\LexBIGData\config\lbconfig.props

Alternatively, you could programmatically set this system variable from your application when it starts up and before it makes its first LexEVS call.

Read the documentation in the lbconfig.props file for other `LG_CONFIG_FILE` configuration possibilties.

NOTE: JBoss tends to run out of memory while running with the default settings of JBoss. To fix the problem, modify the run.conf file under JBOSS_home/bin to change the memory setting and how frequently the rmi garbage collector runs, by editing the JAVA_OPTS line as follows:

```
if [ "x$JAVA_OPTS" = "x" ]; then
   JAVA_OPTS="-server -XX:MaxPermSize=128m -Xms256m -Xmx2000m -Dsun.rmi.dgc.client.gcInterval=120000 -
Dsun.rmi.dgc.server.gcInterval=120000"
fi
```

Moving a terminology to another SQL server

In some deployment scenarios, it may be necessary to move a very large terminology onto a different SQL server than the standard LexEVS server configured above for scalability reasons.

This is supported, but there are some limitations and cautions:

- Manually edit the `registry.xml` file (It may have an alternate name as denoted in the `REGISTRY_FILE` variable of `lbconfig.props`) with a text editor.
- This terminology cannot be removed using the LexEVS API. To remove the terminology, edit the `registry.xml` file again, and manually drop the database from the server. This leaves an orphaned index -- running the orphaned resources clean up tool wipes the index and associated entries.
- This "extra" SQL server must use the same driver and same username and password as the default LexEVS server since the configuration file only supports a single password and username set.

Steps to move a terminology to a new server:

1. Manually move the proper database (or proper subset of tables in `SINGLE_DB_MODE`) to the new server. The LexEVS `TransferScheme` tool (in the admin scripts folder) can help you determine the proper SQL databases, tables, and commands that you will need to use to accomplish this. Alternatively, you can look at the `registry.xml` file to determine what database and/or tables you need to move.
2. Manually edit the registry.xml file. You will need to change the value of the dbURL parameter on the proper line for this terminology that you are using.
3. Restart LexEVS, and ensure that the terminology is still available.

Installing Web-enabled LexEVS (Remote API, Web Access and Data Service)

Minimal System Requirements

- Internet connection
- 2 GB RAM
- Tested Platforms (Similar Hardware Specification for Operating System)

LexBIG has been tested on the platforms shown in Table 1.

	Linux Server	*Linux Server*	*Windows*
Model	HP Proliant DL 380	Penguin	Dell Latitude
CPU	2 x Intel® Xeon™ Processor 2.80GHz	Dual AMD Opteron 248 processors (64 bit)	1 x Intel® Pentium™ Processor 2.00GHz
Memory	4 GB	16Gb	1.5Gb
Local Disk	System 2 x 36GB (RAID 1) Data = 2 x 146 (RAID 1)	250 GB Raid 1 disk drive(s) 250 GB stand along disk drive	System 1 x 80GB
OS	Red Hat Linux ES 3 (RPM 2.4.21-20.0.1)	Fedora Core 3 (64 bit) OS	Windows XP Professional

Table 1 - Platform Testing Environment

Software Requirements

Required Software - Not Included in LexEVS

You must download and install the required software that is not included with LexEVS (listed in Table 2). The software name, version, description, and URL hyperlinks (for download) are indicated in the table.

(Required software that *is* included with the LexEVS is listed in Appendix I)

Software Name	Version	Description	URL
Java Software Development Kit (SDK):Java 2 Standard Edition (J2SE)	j2sdk1.5.0_04 or higher	The J2SE Software Development Kit (SDK) supports creating J2SE applications	http://java.sun.com/javaee/downloads/
MySQL Database*	MySQL (5.0.45) or higher	MySQL 5 Community Edition	http://downloads.mysql.com/archives.php?p=mysql-5.0&v=5.0.45
PostgreSQL*	8.x or higher	Open source relational database management system	http://www.postgresql.org/
JBoss	4.0.5 or higher	application server	http://www.jboss.org/
Apache Tomcat	5.x or higher	servlet container	http://tomcat.apache.org/
*MySQL or PostgreSQL installation is required.			

Table 2 - Required software and technology for the LexEVS

NOTE:	Some database drivers are not included with the LexEVS installer. Downloaded drivers are placed in the `{LEXEVS_DIRECTORY}/runtime/sqlDrivers` and the `{LEXEVS_DIRECTORY}/runtime-components/sqlDrivers` directories

Installation

NOTE:	Web-enabled LexEVS requires that LexBIG (soon to be called LexEVS local runtime), be installed and an appropriate database connection configured. Since database loads do not occur in LexEVS, they must be done from the LexEVS local runtime iteration.

Downloading and Installing the web-enabled LexEVS api

	To best understand the installation and testing procedures for LexEVS, it is recommended that you follow the procedures described in this section with minimal deviation.
SUGGESTION	

Complete the following steps to download and install the LexEVS api:

Step	Action
1	To download the LexEVS API war file go to the NCI GForge web site https://gforge.nci.nih.gov/frs/?group_id=491
2	Select the most recent version of the LexEVS API war file, lexevsapi<xx>.war. You may have a choice between war files tailored for jboss or tomcat installations. Save this file to your computer. You may have to disable Pop-up blockers to allow save the install package to your local computer.

3	Shut down your application or servlet container

```
jboss_lex@bmidev:/usr/local/jboss_lex/bin                      _ □ X
[jboss_lex@bmidev ~]$ cd /usr/local/jboss_lex/bin
[jboss_lex@bmidev bin]$ ./shutdown.sh -S
Shutdown message has been posted to the server.
Server shutdown may take a while - check logfiles for completion
[jboss_lex@bmidev bin]$ █
```

4	Copy the war file to the <server root>/server/default/deploy directory or other appropriate webapp directory

```
jboss_lex@bmidev:/usr/local/jboss_lex/server/default/deploy       _ □ X
[jboss_lex@bmidev jboss_lex]$ cd server/default/deploy/
[jboss_lex@bmidev deploy]$ ls
bsh-deployer.xml                jms
cache-invalidation-service.xml  jmx-console.war
client-deployer-service.xml     jmx-invoker-service.xml
ear-deployer.xml                jsr88-service.xml
ejb-deployer.xml                lexevsapi50.war
hsqldb-ds.xml                   mail-ra.rar
http-invoker.sar                mail-service.xml
jboss-aop.deployer              management
jboss-bean.deployer             monitoring-service.xml
jboss-ha-local-jdbc.rar         properties-service.xml
jboss-ha-xa-jdbc.rar            properties-service.xml.bak
jbossjca-service.xml            schedule-manager-service.xml
jboss-local-jdbc.rar            scheduler-service.xml
jbossweb-tomcat55.sar           sqlexception-service.xml
jbossws14.sar                   uuid-key-generator.sar
jboss-xa-jdbc.rar
[jboss_lex@bmidev deploy]$ █
```

5	Create a text file named "lexevs.properties" and populate it as a properties file as follows: (REQUIRED) Set the LG_CONFIG_FILE value to the path of the lbconfig.props file of your LexEVS installation: LG_CONFIG_FILE=/home/LexEVS/resources/config/lbconfig.props (OPTIONAL) Enable Security for a given Coding Scheme The first value is the CodingScheme name/uri, and the second is the Security Implementation class name. This class MUST implement the interface gov.nih.nci.system.dao.security.DAOSecurity MedDRA=gov.nih.nci.system.dao.security.MedDRASecurity NOTE: Security must be set up for all ways off accessing a Coding Scheme, for example, by URI, Name, Formal Name, etc... Example: LG_CONFIG_FILE=/home/LargeStorage/services/lexbig/5.0.alpha_MySql_MultiDB/resources/config/lbconfig.props MedDRA=gov.nih.nci.system.dao.security.MedDRASecurity urn\:oid\:2\.16\.840\.1\.113883\.6\.163=gov.nih.nci.system.dao.security.MedDRASecurity Medical\ Dictionary\ for\ Regulatory\ Activities\ Terminology\ \(MedDRA\) =gov.nih.nci.system.dao.security.MedDRASecurity
6	Adjust the properties-services.xml file shown above in jboss or otherwise place the lexevs.properties file in the container's classpath. Jboss' properties-service.xml should have something similar the following value added: <attribute name="Properties">org.LexGrid.LexBIG.caCore.Properties=<path to properties>/lexevs.properties</attribute>. In Tomcat placing the file in the conf dir or the /webapps/lexevsapi50/WEB-INF/classes directory should suffice.
7	Restart the container

| 8 | Verify the installation by navigating to <your server's address>/lexevsapi50 |

Troubleshooting

If the LexEVS local runtime is properly installed and tested then web-enabled LexEVS should have few problems. If you recieve deployment errors in JBoss and are unable to see the EVS home page then you may not have configured the properties attribute in JBoss's properties-service.xml. You may also need to the path in lexevs.properties where lbconfig.props location is defined.

Installing LexEVS Grid Services

Minimal System Requirements

- Internet connection
- 2 GB RAM
- Tested Platforms (Similar Hardware Specification for Operating System)

LexBIG has been tested on the platforms shown in Table 1.

	Linux Server	*Linux Server*	*Windows*
Model	HP Proliant DL 380	Penguin	Dell Latitude
CPU	2 x Intel® Xeon™ Processor 2.80GHz	Dual AMD Opteron 248 processors (64 bit)	1 x Intel® Pentium™ Processor 2.00GHz
Memory	4 GB	16Gb	1.5Gb
Local Disk	System 2 x 36GB (RAID 1) Data = 2 x 146 (RAID 1)	250 GB Raid 1 disk drive(s) 250 GB stand along disk drive	System 1 x 80GB
OS	Red Hat Linux ES 3 (RPM 2.4.21-20.0.1)	Fedora Core 3 (64 bit) OS	Windows XP Professional

Table 1 - Platform Testing Environment

Software Requirements

Required Software - Not Included in LexEVS

You must download and install the required software that is not included with LexEVS (listed in Table 2). The software name, version, description, and URL hyperlinks (for download) are indicated in the table.

Software Name	*Version*	*Description*	*URL*
Java Software Development Kit (SDK):Java 2 Standard Edition (J2SE)	j2sdk1.5.0_04 or higher	The J2SE Software Development Kit (SDK) supports creating J2SE applications	http://java.sun.com/javaee/downloads/
MySQL Database*	MySQL (5.0.45) or higher	MySQL 5 Community Edition	http://downloads.mysql.com/archives.php? p=mysql-5.0&v=5.0.45
PostgreSQL*	8.x or higher	Open source relational database management system	http://www.postgresql.org/
JBoss	4.0.5 or higher	application server	http://www.jboss.org/
Apache Tomcat	5.x or higher	servlet container	http://tomcat.apache.org/
*MySQL or PostgreSQL installation is required.			

Table 2 - Required software and technology for the LexEVS

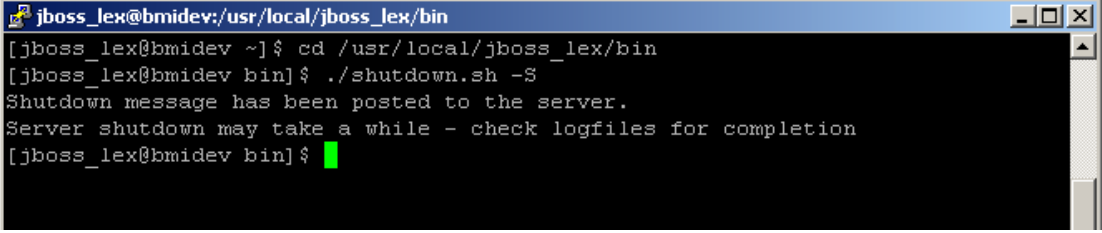| **NOTE:**	Some database drivers are not included with the LexEVS installer. Downloaded drivers are placed in the `{LEXEVS_DIRECTORY}/runtime/sqlDrivers` and the `{LEXEVS_DIRECTORY}/runtime-components/sqlDrivers` directories

Installation

NOTE:	LexEVS Grid Services requires that LexBIG (soon to be called LexEVS local runtime), be installed and an appropriate database connection configured. Since database loads do not occur in LexEVS Grid Services, they must be done from the local LexEVS local runtime iteration. The Grid services access LexEVS via a web interface, therefore the web-enabled LexEVS must be installed as well.

WARNING	Do not attempt to deploy both data and analytical services in the same jboss container.

Downloading and Installing the LexEVS Grid Services

Complete the following steps to download and install the LexEVS api:

Step	Action
1	To download the LexEVS Grid Services API war file go to the NCI GForge web site https://gforge.nci.nih.gov/frs/?group_id=491
2	Select the most recent version, and type of the LexEVS Grid Services API war file. Examples: LexEVS_50_caGridServices_tomcat.war, LexEVS_50_caGRIDServices_data_jboss.zip You may have a choice between war files tailored for jboss or tomcat installations. Save this file to your computer. You may have to disable Pop-up blockers to allow save the install package to your local computer. Select either the Analytical or Data Grid service zip.
3	Shut down your application or servlet container

4	If you are running Jboss unzip the file download to the directory inside named wsrf.war and do so to the \<Jboss root directory\>/server/default/deploy directory. Navigate to the WEB-INF/classes folder and edit the application-config-client.xml file changing all references to the application service url for your web-enabled LexEVS as follows:
	
	  - replace @SERVER_URL@ with \<your lexevsapi server\>/lexevsapi\<XX\>
	  - \<entry key="APPLICATION_SERVICE_URL" value="@SERVER_URL@/http/applicationService"/\>
	  - \<property name="serviceUrl" value="@SERVER_URL@/http/applicationService"/\>
	  - (optional)
	    - \<property name="serviceUrl" value="@SERVER_URL@/http/remoteAuthenticationManager"/\>
	
	If you are running Tomcat just unzip the LexEVSGridService.zip to the webapps directory. After Tomcat expands the directory you may make the same changes to the application-config-client.xml. The Grid Data Services war will have a seperate application-config-client.xml file that will be similarly edited and placed in the WEB-INF/classes folder of the exploded war file.

```
jboss_lex@bmidev:/usr/local/jboss_dat/server/default/deploy

jboss-xa-jdbc.rar
[jboss_lex@bmidev deploy]$ cd ../../../../jboss_dat/
[jboss_lex@bmidev jboss_dat]$ ls
bin         copyright.txt   lgpl.html   readme.html         server
client      docs            lib         readme_j2ee.html
[jboss_lex@bmidev jboss_dat]$ cd server/default/deploy/
[jboss_lex@bmidev deploy]$ ls
bsh-deployer.xml                  jms
cache-invalidation-service.xml    jmx-console.war
client-deployer-service.xml       jmx-invoker-service.xml
ear-deployer.xml                  jsr88-service.xml
ejb-deployer.xml                  mail-ra.rar
hsqldb-ds.xml                     mail-service.xml
http-invoker.sar                  management
jboss-aop.deployer                monitoring-service.xml
jboss-bean.deployer               properties-service.xml
jboss-ha-local-jdbc.rar           properties-service.xml.bak
jboss-ha-xa-jdbc.rar              schedule-manager-service.xml
jbossjca-service.xml              scheduler-service.xml
jboss-local-jdbc.rar             sqlexception-service.xml
jbossweb-tomcat55.sar            uuid-key-generator.sar
jbossws14.sar                    wsrf.war
jboss-xa-jdbc.rar
[jboss_lex@bmidev deploy]$
```

7	Restart the container


```
jboss_lex@bmidev:/usr/local/jboss_lex/bin

[jboss_lex@bmidev bin]$ ./run.sh
```

8	Verify the installation by navigating to <your server's address>/wsrf/services/cagrid/LexEVSGridService or <your server's address>/wsrf/services/cagrid/LexEVSDataService. The user should see a typical Axis service message.

cagrid/LexEVSGridService

Hi there, this is an AXIS service!

Perhaps there will be a form for invoking the service here...

Troubleshooting

Deployment errors may occur if all instances of references in the application-config-client.xml are not correctly configured with the web address of a web-enabled LexEVS instance.

Retrieved from "https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexEVS_5.0_Installation_Guide"