BEL Framework V1.2 System Configuration Guide

Table of Contents

Introduction

The BEL Framework can be configured to work in a variety of modes ranging from single-user access on a desktop or laptop computer through to a set of managed services in a cloud or data center.

This guide will help you understand how to configure the BEL Framework for use and the concepts underlying the configuration of the associated databases and resources used by the BEL Framework.

The components of the BEL Framework include:

- BEL Workbench
- BEL Framework Compiler/Assembler
- BEL Framework Tools
- BEL Framework API
- BEL Framework Web API

These components are documented separately but share the same configuration options and methods described in this document.

Version Changes

Updated for BEL Framework Release 1.2.3.

System Requirements

The BEL Framework is programmed in Java 1.6 and requires a Java runtime environment of 1.6 or better to run. The BEL Framework has been tested and verified on the following systems:

- Windows XP/SP2, 2gb RAM
- Windows 7/SP1, 4gb RAM
- OS/X 10.6, 2gb RAM
- Linux (various), 4gb RAM

Memory Requirements

The above system configurations should be used as guidelines however actual memory and disk space requirements will vary depending on the particular BEL Framework tools that will be used. Compiling and assembling KAMs is generally the most computationally intensive process and requires the most system memory and disk space.

Disk Space Requirements

The BEL Framework installation requires a small amount of disk space for installation (< 100 mb). When running however, some components may create many intermediate files and database tables and can require access to a much larger amount of disk space.

A rule of thumb for estimating the disk space requirements is to ensure that 10 times the size of the input BEL Documents is available.

Configuration Overview

The BEL Framework uses a configuration file to store system configuration preferences. By default, every application in the BEL Framework uses the same configuration file. The default configuration for the BEL Framework uses an internal database to store BEL Documents and KAMs. Modifying the parameters in the configuration file can change these settings.

The current version of the BEL Framework allows the following preferences to be changed:

- Location of the BEL Document Store database
- Location of the KAM Store database
- Location of the work-area for the BEL Compiler
- Location of BEL Framework resources
- Location for storing cached BEL Framework resources

Configuration File

The BEL Framework tools look for a file named **belframework.cfg** in the install directory.

The default location for the configuration file is **[installdir]/config** where **[installdir]** is the folder which contains the downloaded BEL Framework. See Appendix A for a copy of the default configuration.

Specifying a different config file location

The default configuration file can be changed by passing the name of a new configuration file as a command-line argument to any BEL Framework tool. You can use the `--system-config-file` option to specify the name and location of a different configuration file.

Configuration Variables

The BEL Framework configuration system uses several variables to represent logical locations on the install computer that are managed by the operating system. The following variables can be used in the configuration file:

`{home}` – the user's home folder. . On Windows systems this usually defaults to C:\ and on Unix and OS/X based systems to ~

`{tmp}` – a system-provided temp folder. This must be writeable.

Configuring the BEL Framework

This section will help you understand the configuration options for BEL Framework.

Changing the Cache Directory

The BEL Compiler/Assembler can cache resources it needs to compile documents. These resources include:

- Namespaces referenced in BEL Documents,
- Equivalence Files,
- Additional BEL Documents used in Phase III augmentations, and
- Other files used during the compilation/assembly process.

By default the BEL Framework is configured to use **`{home}/.belframework`** as the cache folder where **`{home}`** is the user's home directory as provided by the operating system.

You can change the location where the BEL Framework caches files by changing the **belframework_cache** parameter in the system configuration file. The BEL Framework uses the cache to store files which it uses frequently and which typically change or are updated slowly. For maximum efficiency, this folder should be located somewhere where it will not get removed if the system restarts.

The BEL Framework cache is designed to be managed by the BEL Framework and should not require any modification or alteration by the user. However, if cache maintenance is required, the BEL Framework includes the CacheManager tool that is specifically designed to help you manage the cache. See the *BEL Framework V1.0 Tools User Guide* for more information about the CacheManager.

Changing the Work-Area

The BEL Compiler/Assembler can use a lot of disk space to store intermediate files during the compiler/assembler process. By default the BEL Framework is configured to use **{tmp}/.belframework** as the work-area folder.

You can change the **belframework_workarea** parameter in the system configuration file. Note that the work area must be readable and writable.

Database Support

BELC uses a database to store BEL Documents and compiled KAMs. The BEL Framework has an internal database that is configured to be used by default. The BEL Framework can optionally be configured to use MySQL 5.x and Oracle 11.x database management systems.

Configuring the Internal Database

The BELFramework ships with a copy of Apache Derby 10. This database is a high-performance SQL compliant RDBMS written in Java that uses the host file system to store database tables and data. Derby is configured as the default database for both the DocStore and KamStore databases. For more information about Derby please see the Apache Derby web site at http://db.apache.org/derby

The current version of the BEL Framework supports Apache Derby in embedded mode. In this mode the Derby database runs within the same JVM as the application. This mode is single-user and will not support multiple users accessing the same.

*Note: The **config** folder in the install directory contains a basic configuration file for Apache Derby support called **belframework.embedded.cfg**. We suggest that you modify a copy of this file and rename it to belframework.cfg in order to use the embedded database.*

Database Files

In embedded mode Apache Derby uses the host file system to store the files that are used to contain the tables, indexes, data, and other artifacts for each database. By default, the BEL Framework stores these files in a folder pointed to by the variable

```
{home}/.belframework/database
```

You can change this setting by adjusting the JDBC URLs for the `docstore_url` and `kamstore_url` parameters.

Database Users and Passwords

As the Derby database runs in single-user embedded mode, there is no need for passwords or user accounts. The default configuration for Apache Derby has the docstore and kamstore user and password information disabled.

Schema Names
The database schema names for the docstore and kamstore databases are controlled by the **`docstore_schema`** and **`kamstore_schema_prefix`** parameters.

Configuring MySQL 5.x

The BEL Framework can optionally use a MySQL 5.x database to store BEL Documents and KAMs. To configure a MySQL database you must change the settings in the **belframework.cfg** system configuration file.

*Note: The **config** folder in the install directory contains a basic configuration file for MySQL 5 support called **belframework.mysql.cfg**. We suggest that you modify a copy of this file and rename it to belframework.cfg in order to use the MySQL database.*

The BEL Framework uses a Type-4 JDBC driver to access MySQL databases. The database instance can be set up locally or on a remote server. The BEL Framework uses a JDBC URL to access the instance.

The actual JDBC MySQL URL may vary depending on your installation but generally are of the form:

```
jdbc:mysql://[host]:[port]/[database][?propertyName1][=propertyValue1][&propertyName2]
[=propertyValue2]..
```

where

`[host]` is the name of the database server

`[port]` is the port the MySQL database is configured to listen at. This is usually 3306.

`[database]` is the name of a schema to connect to. This could be set to `doc_store` or `kam_store_catalog` if they already exist.

`[propertyName]` and `[propertyValue]` are pairs of optional parameters to be passed to the mysql JDBC connector.

A typical JDBC URL for the DocStore database on a MySQL 5 installation might look like:

```
jdbc:mysql://192.168.100.130:3306/doc_store?keepalive=true
```

Managing Schemas
By default, this version of the BEL Framework will attempt to manage the MySQL instance, creating new and removing old schemas as needed. The user that is configured to access the database must have permission to create new databases, create tables and views, create indexes, and delete the same.

If you prefer to manage the MySQL instance yourself you must change the system configuration parameter that controls this function and create the schemas manually.

Parameter	Default	Description
system_managed_schemas	1	When set to one the BEL Framework will attempt to create and manage the KamCatalog and Kam schemas. Setting this to 0 will turn off this feature.

The following schemas are required in the MySQL instance:

- doc_store
- kam_catalog
- kam{?} where {?} is a number identifying a specific KAM schema.

The BEL Framework provides the following set of SQL DDL files located in the **`[installdir]/setup/mysql`** folder to help configure the MySQL instance for use with the BEL Framework.

Schema	SQL File
doc_store	doc_store.sql
kam_catalog	kam_catalog.sql
kam{?}	kam.sql

The BEL Framework stores each KAM in a separate schema that is referenced by the `kam` table in the `kam_catalog` schema. In order to allow multiple KAMs to be stored in the KamStore, the DBA must create a set of KAM schemas and pre-populate the `schema_name` column for the `kam` table in the `kam_catalog` schema.

DocStore Settings
The following parameters must be set to configure the BEL Workbench to use a MySQL database for storing BEL Documents:

Parameter	Default	Description
docstore_schema	doc_store	Name of the schema to use to create the Document Store.
docstore_url		JDBC URL for accessing the MySQL database.
docstore_username		Name of the database user with permission to create databases, create tables, perform queries, inserts, and deletes in the MySQL database.
docstore_password		Password for the docstore_username user.

KamStore Settings
The following parameters must be set to configure the BEL Workbench to use a MySQL database for storing KAMs:

Parameter	Default	Description
kam_catalog_schema	kam_catalog	Name of the schema to use to create the KAM catalog.
kamstore_schema_prefix	kam	Prefix for schema containing Kams generated by the compiler. Each kam schema will be created with a unique name using this prefix. For example kam1, kam2. The actual scheme depends on the

	database.
kamstore_url	JDBC URL for accessing the MySQL database.
kamstore_username	Name of the database user with permission to create databases, create tables, perform queries, inserts, and deletes in the MySQL database.
kamstore_password	Password for the kamstore_username user.

Configuring Oracle 11.x

The BEL Framework can optionally connect to an Oracle 11.x database and use it to store BEL Documents and KAMs. To configure an Oracle database you must first setup the Oracle instance with the required schemas and the change the settings in the **belframework.cfg** system configuration file.

*Note: The **config** folder in the install directory contains a basic configuration file Oracle 11 support called **belframework.oracle.cfg**. We suggest that you modify a copy of this file and rename it to belframework.cfg in order to use the Oracle database.*

Managing Schemas

This version of the BEL Framework will NOT attempt to manage the Oracle instance but assumes that every schema and user needed by the BEL Framework has been previously created by a DBA.

The following schemas are required:

- doc_store
- kam_catalog
- kam{?} where {?} is a number identifying a specific KAM schema.

The BEL Framework provides the following set of SQL DDL files located in the **`[installdir]/setup/oracle`** folder to help a DBA configure an Oracle instance for use with the BEL Framework.

Schema	SQL File
doc_store	doc_store.sql
kam_catalog	kam_catalog.sql
kam{?}	kam.sql

The BEL Framework stores each KAM in a separate schema that is referenced by the `kam` table in the `kam_catalog` schema. In order to allow multiple KAMs to be stored in the KamStore, the DBA must create a set of KAM schemas and pre-populate the `schema_name` column for the `kam` table in the `kam_catalog` schema.

Database Users

The following database users are suggested for use within the BEL Framework.

Suggested User	Schema	Required Access
doc_store	doc_store	Select from tables

		Update tables Delete from tables Insert into tables
kam	kam_catalog	Select from tables Update tables
kam	kam{?}	Select from tables Delete from tables Insert into tables Disable / Enable table constraints Truncate a table

Note the `kam` database user is used to access the `kam_catalog` and `kam{?}` schemas and therefore must have access to the all the kam-related schemas.

DocStore Settings
The following parameters must be set to configure the BEL Workbench to use an Oracle database for storing BEL Documents:

Parameter	Default	Description
docstore_schema	doc_store	Name of the Document Store schema
docstore_url		JDBC URL for the Oracle database.
docstore_username		Name of the database user with permission to create session, create tables, create sequences, and create triggers in the Document Store database
docstore_password		Password for the docstore_username user.

KamStore Settings
The following parameters must be set to configure the BEL Workbench to use an Oracle database for storing KAMs:

Parameter	Default	Description
kam_catalog_schema	kam_catalog	Name of the KAM catalog schema
kamstore_schema_prefix	kam	Prefix for schema containing Kams generated by the compiler. Each kam schema will be created with a unique name using this prefix. For example kam1, kam2. The actual scheme depends on the database.
kamstore_url		JDBC URL for the Oracle database
kamstore_username		Name of the database user with permission to create session, create tables, create sequences, and create triggers in the KAM catalog schema. Each KAM schema must also be accessible and modifiable by the KAM catalog schema
kamstore_password		Password for the kamstore_username user.

Additional Information

This section provides additional information that might be helpful to you.

Obtaining Technical Support

Technical support is available by phone or email during normal business hours (8am to 5pm EST).

Email Support

Send an email to support@selventa.com. Please make sure to include your customer account number, user name, a phone number where you can be reached and details about the issue.

Phone Support

Please call Selventa's technical support line at (617) 851-5273 during normal support hours.

Learning More About Selventa's Software and Services

For all sales and other inquires, please contact:

Louis Latino
EVP Sales and Marketing
One Alewife Center,
Cambridge MA 02140

Phone: (617) 547-5421 x237
Email: llatino@selventa.com

Appendix A: System Configuration File

This is the default system configuration file for the BEL Framework. The configuration uses the internal database management system which is not password protected.

```
# BEL framework system configuration using local Derby database connections.

# Pathway for local caching of BELFramework resources
belframework_cache = {home}/.belframework/cache

# Pathway to work area for creating and storing intermediate
# files. Must be writable
belframework_work = {tmp}/belframework

# Document store schema name; defaults to 'doc_store'.
# This configuration setting should match the ;user=doc_store; connection string in
docstore_url.
docstore_schema = doc_store

# Document store database URL (Derby)
docstore_url =
jdbc:derby:{home}/.belframework/database/DocStore;create=true;user=doc_store;

# Optional Document store database username
# docstore_user = <username>

# Optional Document store database password
# docstore_password = <password>

# KAM store catalog schema name; defaults to 'kam_catalog'.
kamstore_catalog_schema = kam_catalog

# KAM store prefix for each KAM schema; defaults to 'kam'.
kamstore_schema_prefix = kam

# KAM store database URL (Derby)
kamstore_url = jdbc:derby:{home}/.belframework/database/KAMStore;create=true;

# Optional KAM store database username
# kamstore_user = <username>

# Optional KAM store database password
# kamstore_password = <password>

# Application log path
# application_log_path = {tmp}

# Path for storage of BEL Document Templates
beltemplate_path = {home}/.belframework/templates

# URL to the BELFramework resource index XML file
resource_index_url = http://resource.belframework.org/belframework/1.0/index.xml

# KAM store encryption passphrase
kamstore_encryption_passphrase = secret passphrase 1!
```