

# CESE Build System

by Sergey Missan

## Table of contents

1 Prerequisites.....	2
2 2-stage build process of CESE.....	2
2.1 Stage 1: building the environment.....	2
2.2 Stage 2: building models.....	2
3 CESE CVS build.....	3

This document gives you a short overview of the CESE build system. You can build system to compile CESE from the source code obtained from CESE [CVS repository](#) or to build prepackaged or your own models.

## 1. Prerequisites

CESE uses [Apache Ant](#) to control the build system. Download the binary distribution of Ant and unpack it. Set variable `$ANT_HOME` to point to the root of Ant installation directory.

Download the Java Development Kit (J2SDK) from [java.sun.com](http://java.sun.com), version 1.5.0 (5.0). Install it and set variable `$JAVA_HOME` to point to the root of J2SDK directory.

## 2. 2-stage build process of CESE

CESE build process comprises of 2 stages: building of the environment and building of models.

Both steps can be completed in one run: enter the directory where you installed CESE (`$CESE_HOME`) and execute command

```
$ANT_HOME/bin/ant
```

You will see a list of messages from Ant followed by BUILD SUCCESSFUL message. This means that both steps of the build process have completed and you obtained a binary CESE distribution ready to run and execute models.

### 2.1. Stage 1: building the environment

This step is controlled by `$CESE_HOME/build.xml` script.

The important targets in this script are:

- **all** (default): performs a complete build of CESE including the environment and all models. Use it if you want to rebuild the whole system from sources.
- **archive**: compiles CESE (the environment only) and packages it to `$CESE_HOME/bin/lib/cese.jar`.
- **javadoc**: generates API documentation and saves it to `$CESE_HOME/doc/`.
- **models**: compiles and packages models only. This target invokes stage 2 of the build process. It is useful if you made modifications to the model source code and need to repackage it without recompiling the whole environment. Use `$ANT_HOME/bin/ant models` in the `$CESE_HOME` directory.

### 2.2. Stage 2: building models

This step is controlled by `$(CESE_HOME)/bin/models/build.xml` script. This script controls building and packaging of the model source code and can be used on its own or called from stage 1 script using target **models**.

If you created a new mode and want to compile and package it in order to run in CESE, you should add it to this script.

### 3. CESE CVS build

If you are a developer and would like to build the most recent *unstable* CESE code, you can use special targets to facilitate the CESE CVS update.

Before you start, make sure that command `cvs` is in your path. Issue the `cvs login` command by typing

```
cvs -d:pserver:anonymous@cese.cvs.sourceforge.net:/cvsroot/cese login
```

in the command line. Simply type <Return> when asked for a password. You have now primed the `cvs` password file that will be used for the future access to CESE CVS repository.

Use target **checkout** to perform the initial download of the source code from CESE CVS repository. This target assumes that CESE is installed in directory `cese`. The full rebuild using target **all** (above) is performed after the source code checkout.

After the initial code checkout it is no longer necessary to download the whole source tree. You can reduce the network traffic and download only the files that have changed since the last download using target **update**. Only the files modified in CVS repository will be downloaded. This target assumes that CESE is installed in directory `cese`. The full rebuild using target **all** (above) is performed after the source code update.