

How to build JVoiceXML

Version 1.1

Dirk Schnelle,
`dirk.schnelle@web.de`

Abstract

This documents describes the steps you have to perform, when you want to build JVoiceXML or develop code for the JVoiceXML project. It gives information about the requirements of your development environment and our coding conventions.

Acknowledgment

Thanks to Adrindam Das for writing the TortoiseCVS [15] part and some screen shots.

1 Introduction

JVoiceXML is a free VoiceXML [16] implementation written in the JAVA programming language. It offers a library for easy VoiceXML document creation and a VoiceXML interpreter to process VoiceXML documents using JAVA standard APIs such as JSAPI [12] and JTAPI [12].

VoiceXML is hosted at SourceForge [11] as an open source project. You find everything that is related to this project under <http://sourceforge.net/projects/jvoicexml/>.

This document refers to UNIX and Windows systems. JVoiceXML will work any other operating systems that support Java 5, too.

Nobody is perfect, so you may find some errors or small things to correct. Please let me know if you think you found something that should be written differently.

2 Copyright

JVoiceXML uses the GNU library general public license [9]. This is mentioned in all our source files as a unique header, see section 7. You can find a copy in the file COPYING in the `#{JVOICEXML_HOME}` directory. This means that you are allowed to use JVoiceXML library in your commercial

programs. If you make some nice enhancements it would be great, if you could send us your modifications so that we can make it available to the public.

JVoiceXML is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

JVoiceXML is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

3 Download the source code

3.1 Download the source archive

You can download a the source code for the available releases from <http://jvoicexml.sourceforge.net/downloads.htm>. Create a directory JVoiceXML and unpack the zipped source file into it. In the rest of this document this directory will be referred as `${JVOICEXML_HOME}`.

```
mkdir JVoiceXML
cd JVoiceXML
unzip jvxml-src-VERSION.zip
```

`VERSION` has to be replace by the used version number, i.e. 0.0.1.

3.2 CVS repository

If you want to stay at the current state of development you have to use the CVS repository from Source Forge [11].

3.2.1 Anonymous CVS Access

UNIX The project's SourceForge.net CVS repository can be checked out through anonymous (pserver) CVS with the following instruction set. When prompted for a password for anonymous, simply press the *Enter* key.

```
export CVSROOT=\
:pserver:anonymous@cvs.sourceforge.net:/cvsroot/jvoicexml

cvs login

cvs -z3 co -P JVoiceXML
```

Updates from within the module's directory do not need the `-d` parameter.

Note: UNIX file and directory names are case sensitive. The path to the project CVSROOT must be specified using lowercase characters (i.e. `/cvsroot/jvoicexml`)

Windows If you are using Windows, you can use a GUI client to check out through the anonymous (pserver) CVS. A popular GUI client recommended by SourceForge[11] is *TortoiseCVS*. We have shown its usage here on Windows XP. You can download TortoiseCVS from <http://tortoise cvs.sourceforge.net/> and install it on your computer. After installation, suppose you want you check out the JVoiceXML module of the JVoiceXML project (you can see all the modules of the JVoiceXML project in the website <http://cvs.sourceforge.net/viewcvs.py/jvoicexml/>). Say, you want to check out the module in the folder, *Test*. In the Windows Explorer, right click on *Test* and pick *CVS Checkout...* from the pop-up menu. The Checkout Dialog appears as shown in figure 3.2.1. In the text box beside *CVSROOT*, enter

```
:pserver:anonymous@cvs.sourceforge.net:/cvsroot/jvoicexml.
```

In the text box beside *Module*, enter *JVoiceXML*. Then click *OK*.

If everything goes fine, you should see the folder JVoiceXML under *Test*. For troubleshooting, refer to <http://tortoise cvs.sourceforge.net/faq.shtml>. Another place to look for troubleshooting is <http://tortoise cvs.sourceforge.net/ssh.shtml>.

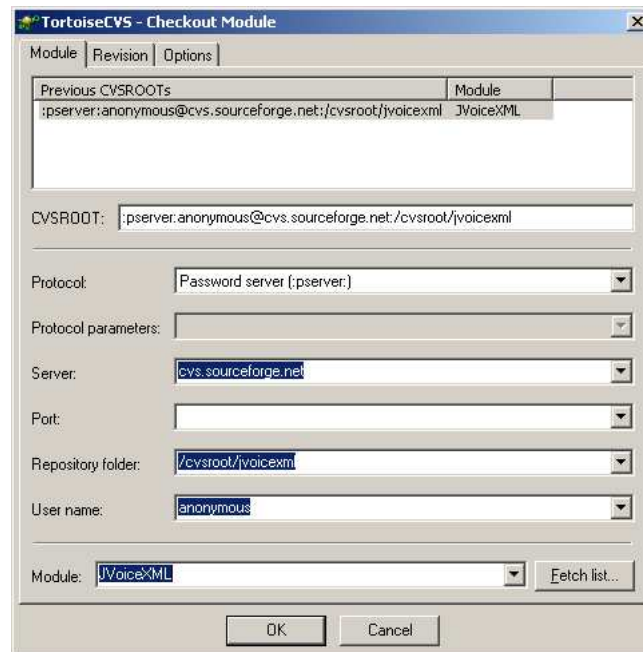
Note: Windows users may also use the command line to access the CVS repository. A cvs implementation for Windows is i.e. included in cygwin [5].

3.2.2 Developer CVS Access via SSH

This section describes methods which assume that you have a valid SourceForge.net account and you are registered as a developer in the JVoiceXML project, with the rights to access the CVS repository. Project administrators may limit CVS access to this repository using *cvs_acls* or the CVS access member permissions option. Developers may verify their CVS access status on the admin page for this project.

UNIX A SSH client must be installed on your client machine. Substitute your developer name with the proper value. Enter your site password when prompted. Developers may also make use of shared SSH keys for authentication.

Figure 1: Anonymous CVS access using TortoiseCVS



```

export CVS_RSH=ssh
export CVSROOT=\
    :pserver:anonymous@cvs.sourceforge.net:/cvsroot/jvoicexml
5 cvs login
cvs -z3 co -P JVoiceXML

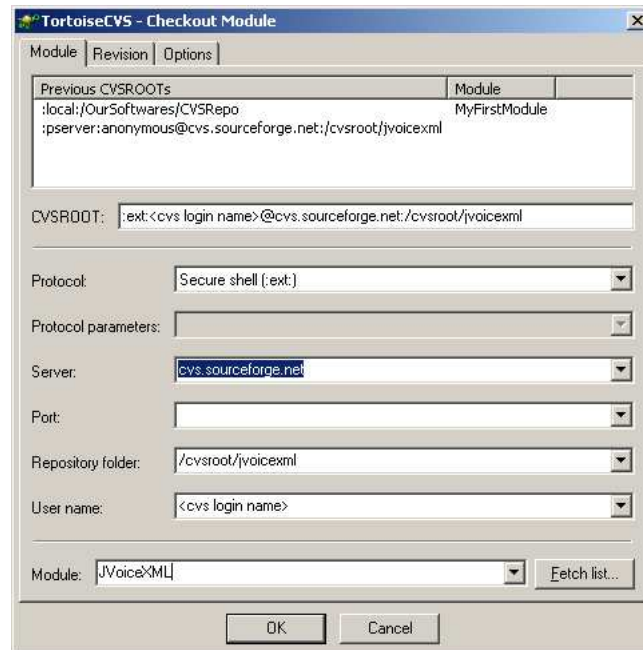
```

Note: UNIX file and directory names are case sensitive. The path to the project CVSROOT must be specified using lowercase characters (i.e. /cvsroot/jvoicexml)

Windows Do the same steps as mentioned in section 3.2.1. Only data to be filled in the Checkout Dialog is different. Figure 3.2.2 shows the Checkout Dialog for Developer CVS access.

Say, you want to checkout the module in the folder, Test. In the Windows Explorer, right click on Test and pick *CVS Checkout...* from the pop-up menu. The Checkout Dialog appears as shown in figure 3.2.2. In the text box beside *CVSROOT*, enter the following string replacing `<cvs login name>` by your CVS Login Name (the id that you use to login your SourceForge.net account):

Figure 2: Developer CVS access using TortoiseCVS



:ext:<cvs login name>@cvs.sourceforge.net:/cvsroot/jvoicexml

In the text box beside Module, enter *JVoiceXML*. Then click *OK*. Depending on the speed of your Internet connection, after a while (may be few seconds) you will be prompted with a dialog asking for a password. Enter the password of your SourceForge.net account. Click *OK*. If everything goes fine, you should see the folder *JVoiceXML* and its content under *Test*. For troubleshooting, refer to <http://tortoisecvs.sourceforge.net/faq.shtml>. Another place to look for troubleshooting is <http://tortoisecvs.sourceforge.net/ssh.shtml>.

4 Directory structure

Having the source code in your `${JVOICEXML_HOME}` directory you find the following directory structure:

<code>\${JVOICEXML_HOME}</code>	
— build.xml	ANT build file
— COPYING	copy of the LGPL
— AUTHORS	Developers who contributed to JVoiceXML
— ChangeLog	Changes in the history of development
— INSTALL	A short installation instruction
— README	Short start guide
— src	JAVA source files
— classes	compiled source files
— lib	jar files for distribution
— doc	documentation
└─ api	JAVADOC documentation
— 3rdparty	third party libraries
— demo	demo programs
— test	Unit tests for the sources

Not all of the files are present at start but are created by the ANT build file or through procedures described in this document, see section 6.

5 Required Software

Since JVoiceXML is written in JAVA you will at least need a JAVA compiler 5.1, an editor or preferably a JAVA IDE 5.1 and ANT 5.3 to build the binaries. All used third party libraries can be downloaded from the CVS repository or from their home pages.

5.1 IDE

You can use the IDE of your choice to edit the sources and compile the binaries. You can even use a simple text editor to perform this job. Nevertheless there are some restriction that you cannot work around.

Your IDE must support

- J2SE 1.5
- ANT

5.2 JAVA

Parts of the code of JVoiceXML are using features from the JAVA 5 API, so that you will need at least J2SE 1.5 to compile the code. You can download it for free from <http://java.sun.com/j2se/1.5.0/download.jsp>.

5.3 ANT

JVoiceXML is being built by an ANT build file. It is recommended that you use at least ANT 1.6.2. If you don't have ANT installed, you can download the current release from <http://ant.apache.org>.

The build file allows you to override the settings by using a custom properties file `jvoicexml.properties` in the `${JVOICEXML_HOME}` directory.

5.4 Third party libraries

JVoiceXML uses some third part libraries. This section names them and tell you where you can get them. All of them are at least freeware so that it was possible to store them in the CVS repository at SourceForge. You can download them either from that location or from their home pages.

You can also use your local copy by adjusting the settings in your custom properties file.

All third party libraries are located in the directory `${JVOICEXML_HOME}/3rdparty`.

5.4.1 Log4j

JVoiceXML uses log4j [1] for logging. We think that log4j has some advantages over `java.util.logging` and appears to be more mature and reliable.

The log4j library and API documentation are located in the `${JVOICEXML_HOME}/3rdparty/log4j1.2.8` directory. If you already have log4j installed, you can use your copy of the library by modifying your `jvoicexml.properties` file. For log4 the properties named in table 1 are relevant.

Table 1: Properties for the log4j library configuration

Property	Comment	Default
<code>log4j.dir</code>	Base directory of your log4 installation.	<code>\${3rdparty.dir}/log4j1.2.8</code>
<code>log4j.lib.dir</code>	Directory containing the log4 jar file	<code>\${log4j.dir}/lib</code>

5.4.2 JSAPI

JVoiceXML uses the Java Speech API v1.0 (JSAPI) [12] to address speech recognition and speech synthesis issues.

Note that the JSAPI specification is undergoing changes. The official work being done on JSAPI is now for JSAPI 2.0 via the Java Community Process (JCP) [7] under JSR-113 [8].

The JSAPI API documentation is located in the `${JVOICEXML_HOME}/3rdparty/jsapi1.0` directory. If you already have JSAPI installed, you can use your copy of the library by modifying your `jvoicexml.properties` file. For JSAPI the properties named in table 2 are relevant.

Table 2: Properties for the JSAPI library configuration

Property	Comment	Default
<code>jsapi.dir</code>	Base directory of your JSAPI distribution. It is assumed that this directory has a subdirectory named <i>lib</i> containing the <code>jsapi.jar</code> .	<code>\${3rdparty.dir}/jsapi1.0</code>
<code>jsapi.lib.dir</code>	Directory containing the <code>jsapi.jar</code> file	<code>\${jsapi.dir}/lib</code>

If you don't have JSAPI installed, you have to perform the following steps to obtain the `jsapi.jar`.

Set Up Your JSAPI Environment The sources to the JSAPI 1.0 specification implementation (i.e., the `javax.speech.*` classes) are not available under a BSD-style license. Instead, we make the JSAPI binary available under a separate binary code license (BCL).

Obtaining the JSAPI binary is as simple as unpacking the `jsapi.jar` file in the `${JVOICEXML_HOME}/3rdparty/jsapi1.0/lib` directory. Open a shell in the `${JVOICEXML_HOME}` directory and follow these steps:

```
cd 3rdparty/jsapi1.0/lib
chmod u+x ./jsapi.sh
sh ./jsapi.sh
```

Read the BCL. If the BCL is acceptable, accept it by typing *y*. The `jsapi.jar` file will be unpacked and deposited into the current directory.

5.4.3 FreeTTS

JVoiceXML uses FreeTTS [6] for TTS output. The FreeTTS library is located in the `${JVOICEXML_HOME}/3rdparty/freetts1.2`

directory. If you already have FreeTTS installed, you can use your copy of the library by modifying your `jvoicexml.properties` file. For FreeTTS the properties named in table 3 are relevant.

Table 3: Properties for the FreeTTS library configuration

Property	Comment	Default
<code>freetts.dir</code>	Base directory of your FreeTTS installation. It is assumed that this directory has a subdirectory named <i>lib</i> containing the jars.	<code>\${3rdparty.dir}/freetts1.2</code>
<code>freetts.lib.dir</code>	Directory containing the FreeTTS jar files.	<code>\${freetts.dir}/lib</code>

FreeTTS requires a proper JSAPI installation, refer to section 5.4.2.

5.4.4 CMU sphinx

JVoiceXML uses sphinx 4 [4] from Carnegie Mellon University for speech recognition. The sphinx library is located in the `${JVOICEXML_HOME}/3rdparty/sphinx4` directory. If you already have sphinx4 installed, you can use your copy of the library by modifying your `jvoicexml.properties` file. For sphinx the properties named in table 4 are relevant.

Table 4: Properties for the sphinx library configuration

Property	Comment	Default
<code>sphinx.dir</code>	Base directory of your sphinx installation. It is assumed that this directory has a subdirectory named <i>lib</i> containing the jars.	<code>\${3rdparty.dir}/sphinx4</code>
<code>sphinx.lib.dir</code>	Directory containing the sphinx jar files.	<code>\${sphinx.dir}/lib</code>

Sphinx requires a proper JSAPI installation, refer to section 5.4.2. You will also need an acoustic model to run the demos. This is not necessary for compiling. All our demos use the `WSJ_8gau_13dCep_16k_40mel_130Hz_6800Hz`

model which is shipped with sphinx. Other models or languages can be supported by your custom models, generated by the sphinx trainer.

5.4.5 Rhino

JVoiceXML uses rhino [10] from Mozilla to enable scripting. The rhino library is located in the

`${JVOICEXML_HOME}/3rdparty/rhino1_6R1`

directory. If you already have rhino installed, you can use your copy of the library by modifying your `jvoicexml.properties` file. For rhino the properties named in table 5 are relevant.

Table 5: Properties for the rhino library configuration

Property	Comment	Default
<code>rhino.dir</code>	Base directory of your rhino installation. It is assumed that this directory has a subdirectory named <i>lib</i> containing the jars.	<code>\${3rdparty.dir}/rhino1_6R1</code>
<code>rhino.lib.dir</code>	Directory containing the rhino jar files.	<code>\${rhino.dir}/lib</code>

6 The ANT build file

JVoiceXML uses ANT to build the binaries and the documentation. This section explains the most important targets of the build file. Just start `ant` without any target specified if you want to build everything. Call

```
ant -projecthelp
```

to get an overview of the tasks and their purpose.

clean Delete all compiled class files in the directory *classes* and the jars that are created by the *jar* target in the directory *lib*.

compile Compile all JAVA files in the directory *src* into the directory *classes*.

jar This target depends on the target *compile* and creates the jar files of your distribution in the directory *lib*. If successful, you will find the following jar archives:

- `jvxml.jar` This jar file contains the core of JVoiceXML.
- `jvxml-xml.jar` This jar contains all files that are required to create and parse VoiceXML documents.
- `jvxml-impl.jar` This jar contains a demo implementation platform.

docs Create JAVADOC documentation from the JAVA files in the directory `doc/api`.

distribution Create the zip files for release.

checkstyle Perform a check of the JVoiceXML coding standard as specified in section 7.

all This is the default target. Build the jars and the JAVADOC documentation.

7 Code conventions

We follow the JAVA code conventions [14] for our code. All methods and member variables must be commented using JAVADOC [13].

In addition we use a custom `@todo` JAVADOC tag to mark sections that need further work.

Example:

```
/** @todo Implement the untreated case XYZ */
```

Parts of our code are checked using checkstyle [2]. Unfortunately checkstyle is currently not capable of handling JAVA 5 source files, see also RFE 925070 [3].

We use a snapshot from their CVS repository, that is able to handle JAVA 5 sources. It works most of the time, but is still a bit buggy.

All source files must contain the following header about the copyright, see section 2, that we use for JVoiceXML.

```
/*
 * $Id: howtobuild.tex,v 1.7 2005/04/26 07:25:53 schnelle Exp $
 *
 * JVoiceXML – A free VoiceXML implementation.
 *
 * Copyright (C) 2005 JVoiceXML group –
 *   http://jvoicexml.sourceforge.net
 *
 * This library is free software; you can redistribute it
 * and/or modify it under the terms of the GNU Library
 * General Public License as published by the Free Software
```

```

* Foundation; either version 2 of the License, or (at your
* option) any later version.
*
15 * This library is distributed in the hope that it will be
* useful, but WITHOUT ANY WARRANTY; without even the
* implied warranty of MERCHANTABILITY or FITNESS FOR A
* PARTICULAR PURPOSE. See the GNU Library General Public
* License for more details.
20 *
* You should have received a copy of the GNU Library
* General Public License along with this library; if
* not, write to the Free Software Foundation, Inc.,
* 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
25 *
*/

```

The keyword `Id:`, embarrassed by `$` is to be expanded by CVS. This means that for any `cvs update` the option `-kkv` has to be used.

The class comment has to contain the following information:

```

/**
* Comment about the purpose of this class.
*
5 * @author Name of the author.
*
* @version $Revision: 1.7 $
*
* <p>
10 * Copyright ©copy; 2005 JVoiceXML group –
* <a href="http://jvoicexml.sourceforge.net">
* http://jvoicexml.sourceforge.net/</a>
* </p>
*/

```

Again the keyword expansion from `cvs` is used to expand the keyword, `$Revision:` `$` in this case. The name of the author and the purpose have to be replaced by their proper values.

Document history

Version	Comment	Author	Date
0.1	Initial Release	Dirk Schnelle	01/20/2005
0.1	Review	Steven Doyle	01/20/2005
1.0	Release	Dirk Schnelle	01/20/2005
1.1	Added copyright notice in Java source files. Added TortoiseCVS part for Windows. Added third party libraries.	Dirk Schnelle	05/02/2005
1.1	Review	Steven Doyle	05/12/2005
1.1	Release	Dirk Schnelle	05/12/2005

References

- [1] Apache. log4j. <http://logging.apache.org/log4j>.
- [2] checkstyle. <http://checkstyle.sourceforge.net>.
- [3] Checkstyle JDK 1.5 support (generics, foreach, ...). http://sourceforge.net/tracker/index.php?func=detail&aid=925070&group_id=29721&atid=397081.
- [4] CMU Shinx 4. <http://cmusphinx.sourceforge.net>.
- [5] cygwin. <http://www.cygwin.com>.
- [6] FreeTTS. <http://freetts.sourceforge.net>.
- [7] JCP. Java Community Process. <http://www.jcp.org>.
- [8] JCP. JSR 113 Java Speech API 2.0. <http://www.jcp.org/en/jsr/detail?id=113>.
- [9] GNU library general public license. <http://www.opensource.org/licenses/lgpl-license.php>, author = GNU.
- [10] Rhino: JavaScript for Java. <http://www.mozilla.org/rhino/>.
- [11] SourceForge.net. <http://sourceforge.net>.
- [12] SUN. Java Speech API 1.0 (JSAPI). <http://java.sun.com/products/java-media/speech/forDevelopers/jsapi-doc/%index.html>.
- [13] SUN. Javadoc guidelines. <http://java.sun.com/j2se/javadoc/writingdoccomments/index.html>.

- [14] SUN. SUN Code Conventions. <http://java.sun.com/docs/codeconv/>.
- [15] TortoiseCVS. <http://tortoisecvs.sourceforge.net>.
- [16] W3C. Voice Extensible Markup Language (VoiceXML) Version 2.0. <http://www.w3.org/TR/voicexml20/>, March 2004.