

## Chapter 2

# Setting your Classpath

The classpath is a collection of locations. A typical location is a directory on your computer's hard drive, but other kinds of locations are possible. When you compile or run a Java program, the computer looks in these locations for some files that it needs. (Depending on what you're trying to do, the computer may look for `.java` files, `.class` files, or both in these locations.)

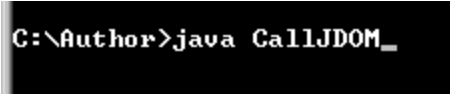
For instance, the `CLASSPATH` string

```
.;C:\JavaPrograms;C:\JARS\xerces.jar
```

lists three locations.

- \* **The first location, denoted by a dot, points to a user's current working directory.**

If you're using a command prompt window, and you see the `C:\Author>` prompt shown in Figure 2-1, then the classpath's dot stands for the `C:\Author` directory. When you issue the `java` command, the computer looks in the `C:\Author` directory for any `.class` files that it needs.



```
C:\Author>java CallJDOM_
```

Figure 2-1: Working in the `C:\Author` directory.

- \* **The second location is the directory `C:\JavaPrograms`.**

Let's say that, to run `CallJDOM`, the computer needs an additional `.class` file named `MyContentHandler.class`. If the computer can't find this `.class` file in the `C:\Author` directory, then the computer looks next in the `C:\JavaPrograms` directory.

- \* **The third location is a Java Archive file (a `.jar` file).**

A *Java Archive (JAR)* file contains `.class` files and other interesting things. With a JAR file in your classpath, Java programs can look inside the JAR file to find classes that they need.

<Tip>

A JAR file is really a collection of files, compressed in ZIP format. If you copy a file named `xerces.jar`, and change the name of the copy to `xerces.zip`, then you can examine the file with your favorite unzipping program. You can also use the `jar` command that comes with the Java Platform.

### <TechnicalStuff>

Starting with Java 2, you don't need to include files that come with the Java download in your classpath. These files are included in your classpath automatically.

Remember, the classpath is a collection of locations.

- \* In Windows, you separate the locations from one another with semicolons.

```
.;C:\JavaPrograms;C:\JARS\xerces.jar
```

- \* If you run `sh`, `ksh`, or `bash` in Unix or Linux, then you separate the locations from one another with colons.

```
./home/bburd/JavaPrograms:/usr/share/xerces.jar
```

- \* If you run `csh`, then you separate the locations from one another with spaces.

```
(. /home/bburd/programs /usr/share/xerces.jar)
```

## ***Classpath examples***

Before I tell you how to set your classpath, I want to go over a few classpath examples. Just look at each example, and say "That's nice." Don't do anything on your computer until you reach the next section.

### ***One location: The working directory***

You just downloaded the latest Java Platform Standard Edition from `java.sun.com`. Your system has no `CLASSPATH` variable.

In this case, you should put the working directory in your classpath. You'll make the value of the variable be a single dot. For an illustration, see Figure 2-2. (I hope you can see the tiny dot in the figure.)

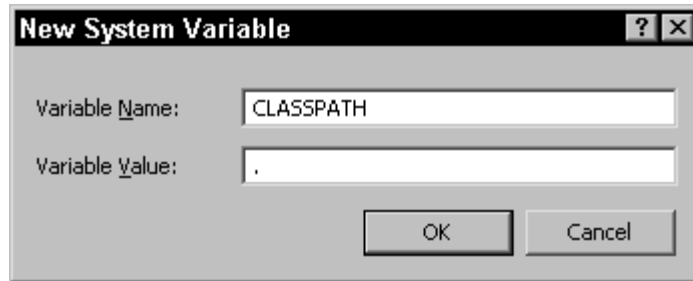


Figure 2-2: Putting the working directory in the CLASSPATH.

***Several locations, including the working directory***

You just downloaded a new version of the Java Platform Standard Edition. When you check for a lone dot among the entries in the classpath, you don't find one. You add the dot to the beginning of the classpath. (See Figure 2-3.)



Figure 2-3: Adding the working directory to the CLASSPATH.

***Several locations, including a JAR file***

You just downloaded the JDOM API. Since JDOM isn't part of Standard Java 2, you have to add the JDOM `.class` files to your Java environment. Among other things, you add `jdom.jar` to your classpath. (See Figure 2-4.)

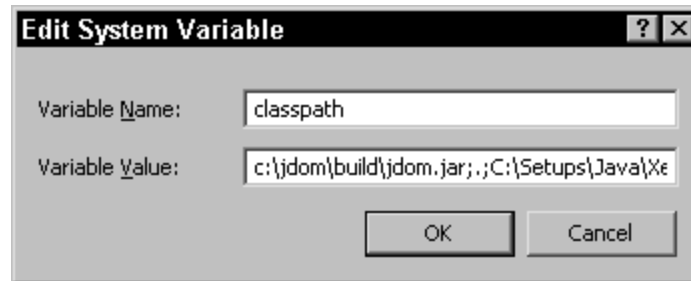


Figure 2-4: Adding jdom.jar to the CLASSPATH.

## *Setting the classpath*

The way you set the classpath depends on a few different things. Mainly it varies from one operating system to another. If you're running Unix, it varies with your choice of shell.

This section has instructions for setting the classpath in several different operating systems. (If you've done this kind of thing before, and you're comfortable setting environment variables, then just do it without reading the rest of this section.)

### *Windows XP, Windows Vista, or Windows 7*

- 1. Choose Start-->Control Panel-->Performance & Maintenance-->System.**
- 2. In the System Properties window, select the Advanced tab. Then click the Environment Variables button.**
- 3. In the Environment Variables window, check to see if there's a variable named CLASSPATH.**

The name is not case-sensitive. It could be `classpath`, `cLaSSPaTh`, or anything spelled that way.

- a. If there's a variable named `CLASSPATH`, then select the `CLASSPATH` line in the Environment Variables window. Click the Edit button that's below the `CLASSPATH` line.

An Edit Variable dialog box opens. (See Figure 2-5.) First, make sure that the little Variable Name box has the word `CLASSPATH` in it. Then in the Variable Value box, type whatever you need to add to the classpath. To separate your new addition from what's already in the Variable Value box, type a semicolon.

For instance, in Figure 2-5, I need to add a dot (meaning the current working directory) to the classpath. So, in the Variable Value box, I type a dot, followed by a semicolon.

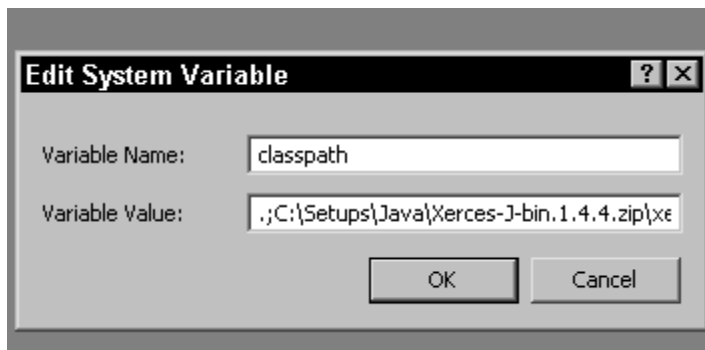


Figure 2-5: Adding a dot to the classpath.

- b. If the Environment Variable window doesn't already have a variable named `CLASSPATH`, then click either of the New buttons. (One New button creates a variable for you alone. The other New button, if it's available, creates a variable for all users on the system.)

A New Variable dialog box is opened. For the variable name, type **CLASSPATH**. For the value, type whatever you need in the classpath. (To find out what you need, read the previous section.)

In Figure 2-6, I put a dot in the window's Variable Value box.

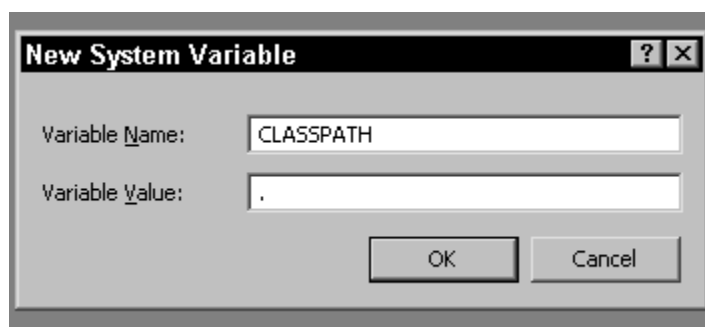


Figure 2-6: Creating a CLASSPATH variable.

#### 4. Click OK in every window that's been opened.

Click OK in the Edit Variable or New Variable window, in the Environment Variables window, and in the System Properties window.

### ***Unix and Linux***

What you do to set the classpath in a Unix-like operating system depends on several things, including the kind of command shell that you're using. For instance, if you're using `sh` or `ksh`, then add a line of the following kind to your `~/ .profile` file:

```
CLASSPATH=.; export CLASSPATH
```

If you're using `bash`, then add a line of that kind to either your `~/ .bashrc`, `~/ .bash_profile`, `~/ .bash_login`, or `~/ .profile` file. Your choice of file depends on the way in which you use the command shell, and on which of these files already exist. (For more details, check the `bash` man pages.)

If you're running `csh`, then add a line like the following to the `~/ .cshrc` file:

```
set CLASSPATH=.
```

On some versions of `csh`, you use a `setenv` command in the `~/ .cshrc` file:

```
setenv CLASSPATH .
```

### ***Setting your classpath on the fly***

There must be dozens of ways to set your classpath. For instance, you can set the classpath for one invocation of the `javac` or `java` command. To do this, you type something like

```
java -classpath .;C:\JavaPrograms CallJDOM
```

This command runs the `CallJDOM` class's main method using classpath `.;C:\JavaPrograms`.

If you're running Windows, you can type a line like

```
set CLASSPATH=.;C:\JavaPrograms
```

in the command prompt window. This modifies the classpath for any commands that you issue in that particular command prompt window.

#### **<Warning>**

The `set` command holds for only one copy of the command prompt window. When you open a command prompt window, and type this `set` command, then none of your other open command prompt windows will use that modified classpath value. When you

close the window in which you typed the `set` command, then that modified classpath value goes away. If you want the modified classpath again, you have to type the `set` command in another command prompt window.