

---

# Getting Started

The purpose of this chapter is that in five minutes from now you can be ready to use the environment you will use in this book. In this chapter you will learn how to install the environment, understand the different parts of the environment and how to interact with the robots that live in this environment. You shall program these robots to accomplish challenging tasks by sending them messages.

Right now we just want to get you started and get ready for the rest of the book. Note that if your environment is already installed then simply jump directly to the subsequent sections that explain the environment. Chapter ?? will explain in more detail the environment.

## 1 Installing the Environment

The environment used in this book is developed on top of Squeak. Squeak is a rich and powerful open-source multimedia environment entirely written in Smalltalk and freely available for most of the computer and operating systems at <http://www.squeak.org>. Note however that you will not use directly a default Squeak distribution but use instead a distribution I prepared for you.

Squeak runs exactly the same on all the platforms, however to ease your start we prepared some platform dependent compressed files. The principle is exactly the same on Mac, PC, or any other platforms. Only the decompressing tools and the way to invoke Squeak may differ. Once you have a file named `ReadyToUse.zip`, you decompress it then drag the file named `Ready.image` on the Squeak application, and this is it! The file `Ready` contains the complete environment used in this book

Now let us look at each of the steps one by one. Note that you may get files with slightly different names but it should work exactly the same.

**On Macintosh.** You should have a file named `readyToUseForMac.zip` a zipped archive. Normally double-clicking on the file should invoke the right decompressor such as StuffIt Expander. When you decompressed this file, you should obtain 4 files as shown by Figure 1.1. You should identify two files: the file named `Ready.image` and the *Squeak application* file (the one without extension in Figure 1.1 it is named Squeak).

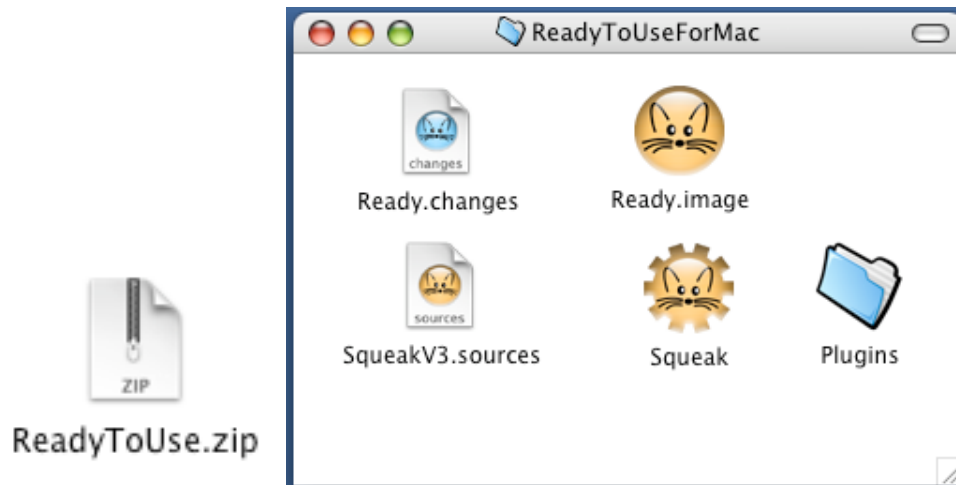


Figure 1.1: The files for the ready to use environment for Macintosh. Left: the zipped file. Right: the files.

**On Windows.** You should have a file named `readyToUse.zip`, an archive. When you decompressed this file using Winzip, you should obtain 4 files as shown by Figure 1.2. You should identify two files: the file named `Ready.image` and the *Squeak application* file (the one without extension in Figure 1.2 it is named `Squeak`).

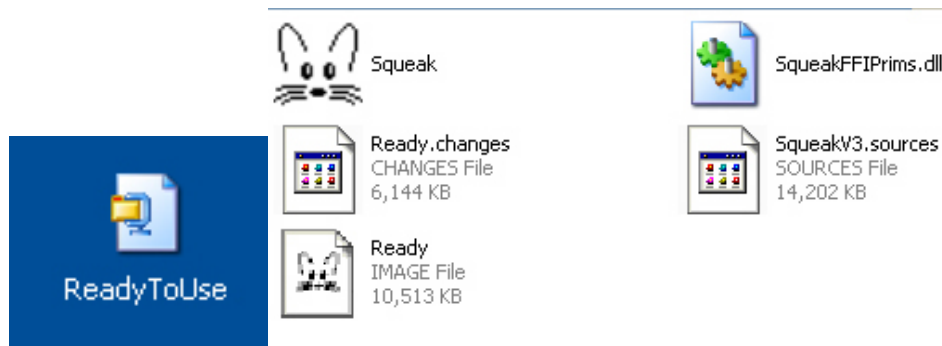


Figure 1.2: The files for the ready to use environment on PC. Left: the zipped file. Right: the files.

## 2 Opening the Environment

To open the environment, drag the file `Ready.image` on the *Squeak application* that is the file named `Squeak` as shown by Figure 1.3. You should get the environment shown in Figure 1.4. If you do not get this environment then read Section 6.

**Hints.** The environment can be opened by simply double-clicking on the *image* file. However, such a practice has several disadvantages: you may have to identify the *Squeak application* and sometimes another application may by accident try to use the image. Moreover it can lead you to trouble when you have multiple installations of different versions of Squeak. So we suggest to always open the environment by dragging and dropping the *image file* on the *Squeak application* file or alias to it.

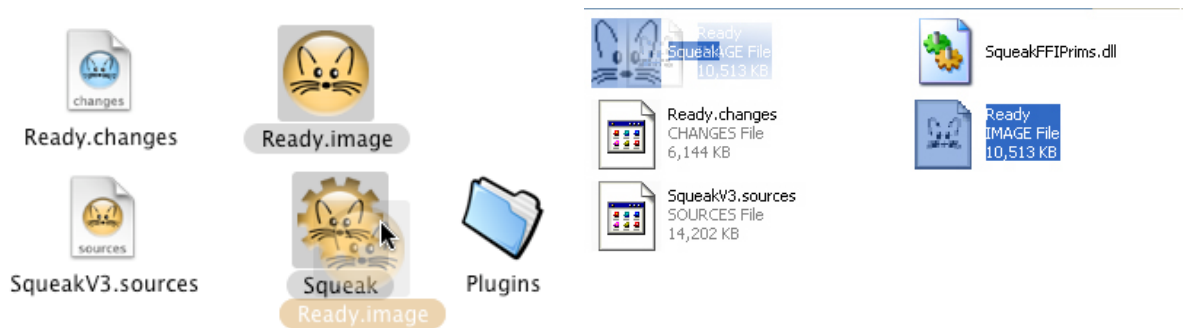


Figure 1.3: Dragging and dropping the *image* file on the *Squeak application* to open on Mac (left) on PC (right).

Note that if you have space problem you can use an alias to the SqueakV3.sources file as this file can be shared between multiple installations.

To start the environment. Drag and drop the file terminating with the .image extension into the squeak application.

### 3 First Interactions with a Robot

Once you open the environment by dragging the file named **Ready.image** on the Squeak executable as explained previously, you should obtain an environment similar to the one presented by Figure 1.4.

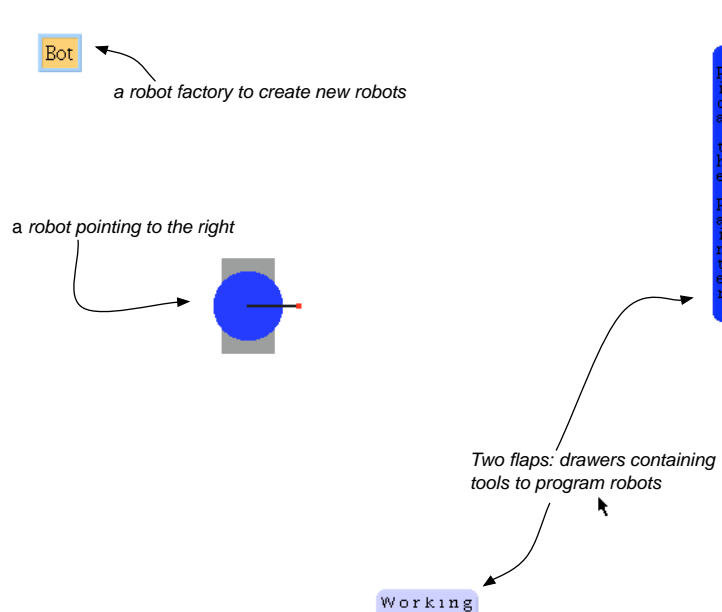


Figure 1.4: The environment ready to use.

The environment is composed of robot factories and two flaps. A flap is a drawer containing

programming tools that you do not need to use now and we will describe later in Chapter ???. Now you should see a small blue robot in the middle of the screen. Of course this is not a physical robot, but a software robot seen from above pointing on the right of the screen. A robot is a round blue circle; it has two caterpillars and a small red head indicating its current direction. In this book you shall send orders to robots, we say that we send them *messages* and robots execute these messages.

Place the mouse over the robot and wait a second there. A balloon pops up with some information about the robot such as its current location and its direction as shown in Figure 1.5. As your screen is of different size than the one used to produce this book, you may have other values.

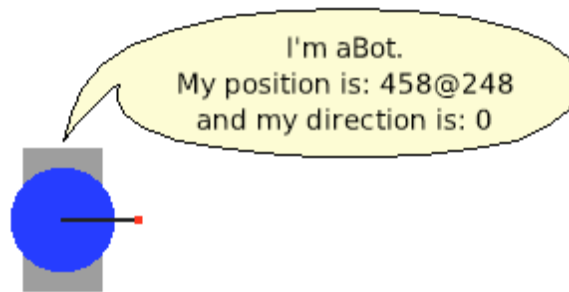


Figure 1.5: Place the mouse over a robot to get a balloon showing some information related to the robot in question.

**Sending Messages to a Robot.** To interact directly with a robot by left-clicking on the robot with the mouse. A messaging balloon pops up as shown by the left picture of Figure 1.6. You can type some messages that are sent to the robot. Once you typed these messages, hitting the **return** key actually sends them to the robot which will then execute them.

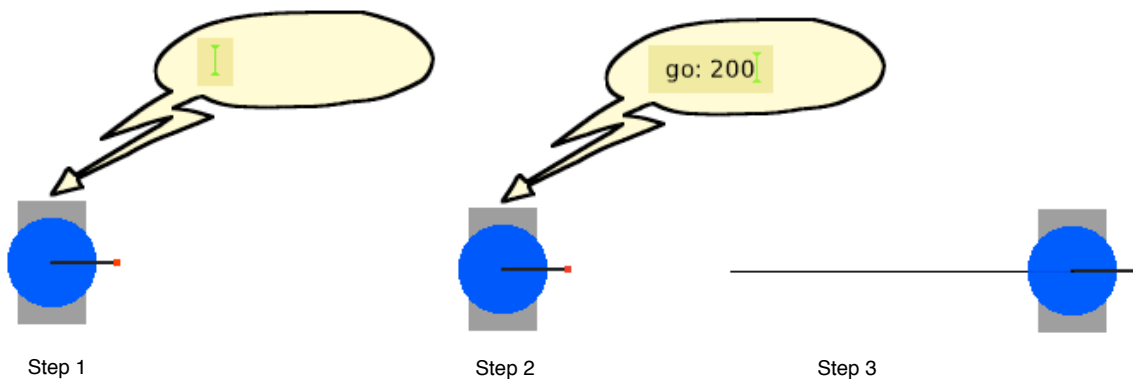


Figure 1.6: Step 1: left-clicking on a robot produces a balloon to send a message to a robot. Step2: Typing a message to ask a robot to move forward. Step 3: the robot moved and left a trace on the ground.

For example by typing the message `go: 200` and hitting the return key, we ask the robot to move forward 200 pixels in its current direction, or typing the message `turnLeft: 20 + 70` we ask the robot

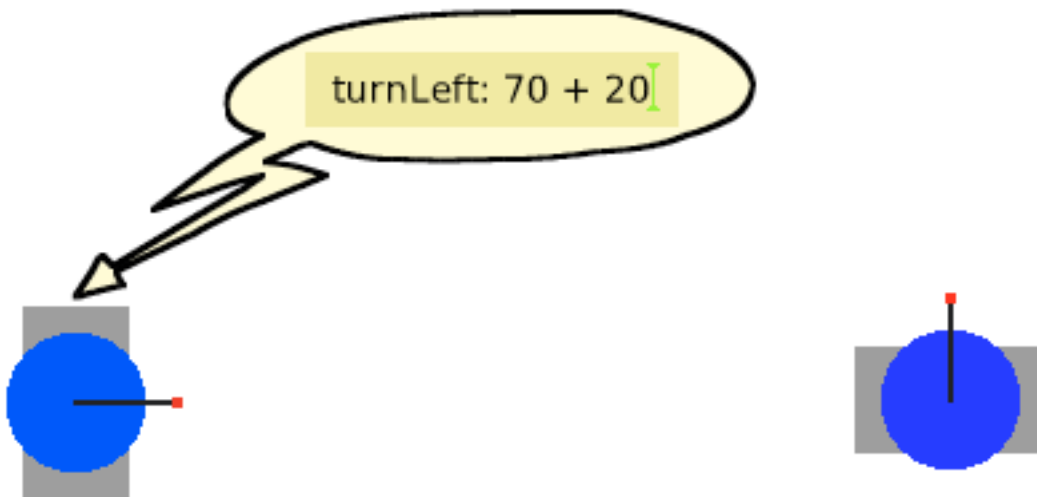


Figure 1.7: Left: Sending a complex message. Right: Its effect.

to turn on its left of 90 degrees. The message color: Color green changes the color of the robot as shown by the Figure 1.8.

As the messages show, we can write complex messages and we will explain how you can write such messages later. For now simply type what we show you. Note that if you want to repeat some expressions, you do not have to retype them but simply use the up and down arrows to navigate over the previous messages you sent to the robot. In the subsequent chapters, you shall learn step by step all the messages that a robot understands and more important you shall learn how to define new behavior for your robots.



Figure 1.8: Left: Changing the color of a robot to another color. Right: its effect

To interact with a robot. Click on it, type the message and hit the **return** key.

## 4 Creating a New Robot

The environment already contains a robot but now we show you how to create new robots. To create a new robot we ask a robot *factory* to create a new one. A robot factory is graphically represented by an orange box surrounded by a blue light box in the middle of which the word Bot is written as shown in Figure 1.9. A robot factory is called in Squeak jargon a *class*. Classes (object factories) have a name starting with an uppercase letter. Hence this is the class **Bot** and not **bot**.

As for robots, we interact with a robot factory by sending it messages. The message to create a new robot is the message **new** as shown in Figure 1.10. Note that newly created robots point also to the right of the screen. Now you can send messages to the two robots individually, each of them living its own life.

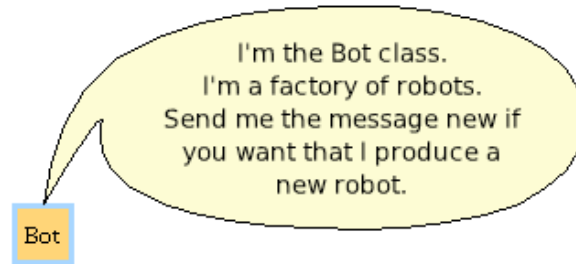


Figure 1.9: A robot factory called in Squeak jargon a *class*. It produces new robots.

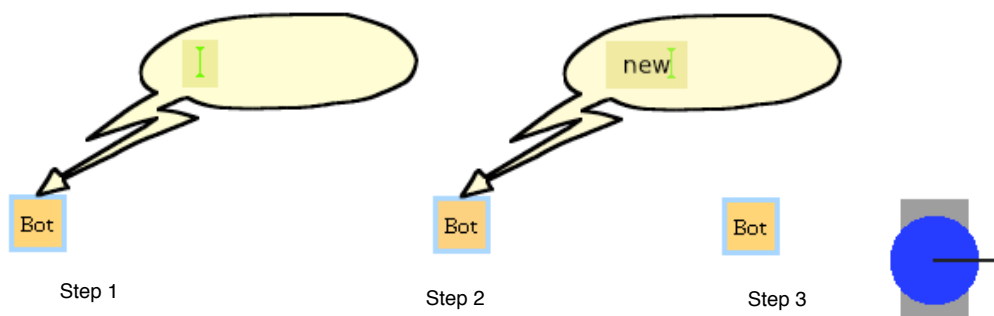


Figure 1.10: Step 1: Start to type a message. Step 2: sent the message **new** to the factory to get a new robot. Step3: The factory creates a robot and gives it to you.

To create new robots. Send the message **new** to the robot factory, a class.

When a robot is created, it is always pointing to the east or the right of the screen.

## 5 Quitting and Saving

The background of the Squeak window application is called the World. The World has a menu offering a lot of functionality. To display the World menu just click on the background. You should get a menu similar to the one shown in Figure 1.11. The last group of elements are all the actions you can do to quit or save your work.

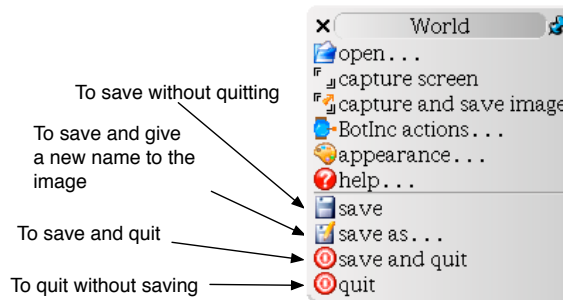


Figure 1.11: The World menu.

Selecting the item 'quit' simply quit without saving what you did. Therefore the next time that you will restart the environment it will be exactly in the same state as when you started this time. Selecting the item 'save', save the complete environment. The next time you till start it, it will be exactly in the same time that at the moment you saved it. Finally when selecting the item 'save as...', the environment asks you to give a new name and it will create two new files with that name: one with the extension `.image` and one with the extension `.changes`. This is this way that we created the files `Ready.image` and `Ready.changes`. To open the environment you saved with a new name, drag and drop the file with the new name which has the extension `.image` on the squeak executable as you did to start the environment by dragging and dropping the file `Ready.image`.

## 6 Possible Installation Troubleshootings

Now we try to help to fix the problems you may encounter during the installation. For that we will explain the role of each important files you got once you decompressed the archive.

To run the environment provided with this book or any Squeak distribution, four files are necessary. We describe them now as we will use their name to help solve the possible problems you may encounter. The four files are:

**Image and changes.** The file named `Ready.image`, called simply the **image** file and the file named `Ready.changes`, called simply the **changes** file contain the information of your current system. These two files are synchronized by Squeak automatically and *should be writable*. Each time you save you environment they are synchronized. You should not edit then with another file editor or changes the name of the file manually. If you want to have different names, just use the **save as ...** menu item of the World menu, Squeak will then create a new pair of files for you.

**Source.** The file named `SqueakV3.sources`, also called **sources**, contains the source of a part of the Squeak environment. You do not need it in this book so do not try to edit manually. However, this file should always be in the directory in which the image is.

**Application.** The application Squeak for mac or Squeak.exe for PC is the Squeak system. This is this application that runs when you program in Squeak. It should then be executable. We refer to this file as the **Squeak application**. In computer scientist jargon, this application is called a virtual machine called or a VM in short.

As we already mentioned it the *image* and *changes* files should be writable. Certain operating systems change the properties of the files to read-only when copied from a CD. In such a case, Squeak warns you with a message as shown in Figure 1.12. If you get this message, simply quit Squeak without saving, change the property of the file to be writable and restart.

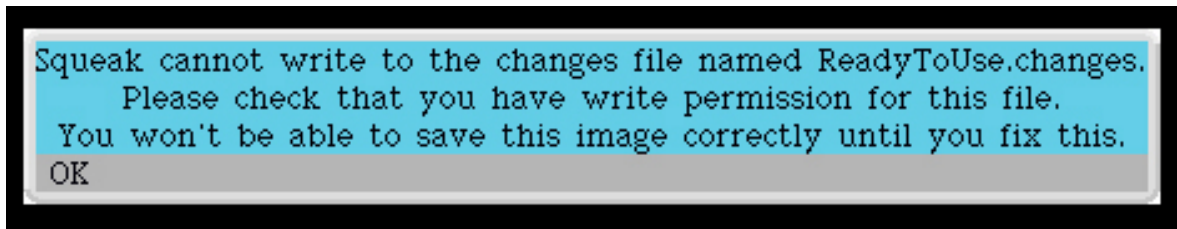


Figure 1.12: The error message showing that the image (Ready.image) or changes (Ready.changes) files are not writable.

Another possible problem you may encounter is related to the sources file named SqueakV3.sources, this file or an alias to this file should be present in the directory where the image file is. When this file is not present you may get on Mac the first message shown in Figure 1.13 (note that this message is not clear and refers to a problem related to aliases that is fixed in the current version of Squeak) or the second message. To cure this problem, create an alias to the source file (SqueakV3.sources) into the directory containing your image or simply copy the sources file to be in the same directory that the image file.

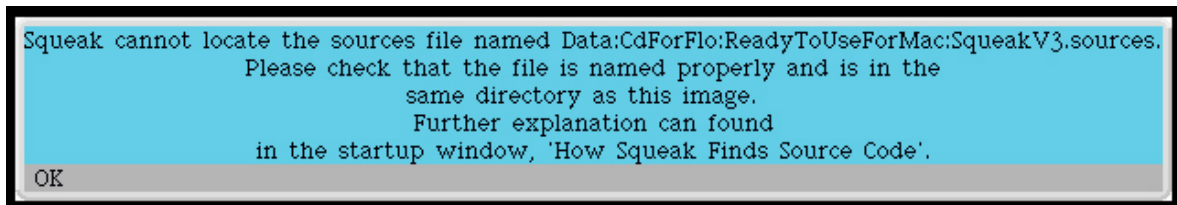


Figure 1.13: Some possible error messages indicating that the *source* file (SqueakV3.sources) is missing in the directory containing the *image* file .

## Summary

- To start the environment, drag and drop the file terminating with the .image extension into the squeak application.
- To send a message to a robot, left-click on it, type the message and hit the **return** key.
- To create new robots. Send the message **new** to the robot factory, a class.
- When a robot is created, it is always pointing to the east or the right of the screen.



- The background of the environment is called the World.
- To get the menu to save the environment, click on the background close to a robot.