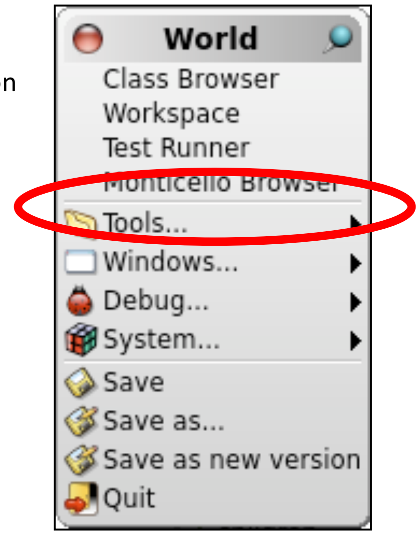
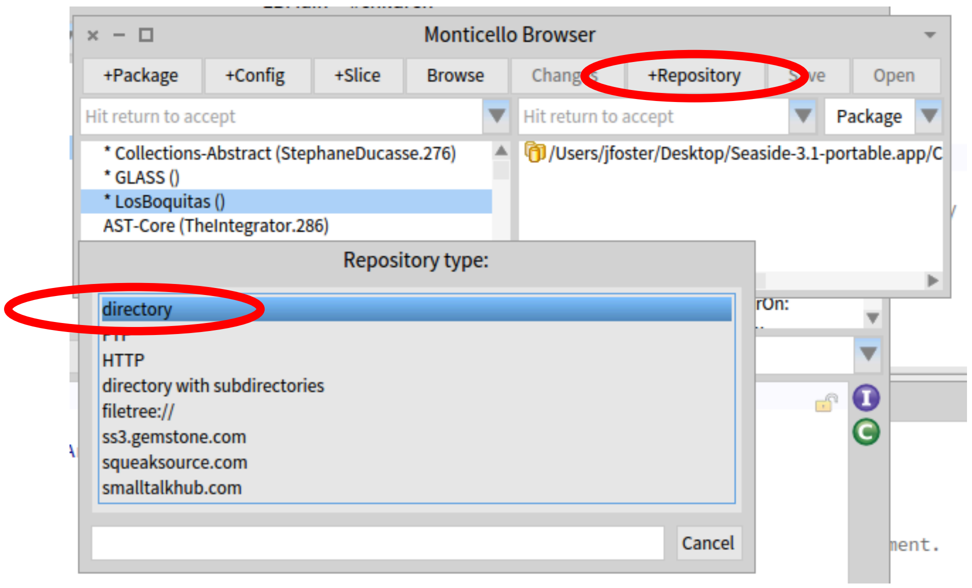
Monticello (pronounced either män-tə-`sel-ō or män-tə-`chel-ō) “is a distributed optimistic concurrent versioning system for Pharo code source. It was written by Avi Bryant and Colin Putney, with contributions from many members of the Pharo community” (see http://www.wiresong.ca/Monticello/). So far we have been saving our code by saving the Pharo image. Monticello gives us much more powerful tools for managing source code.

1. Create a Monticello package.

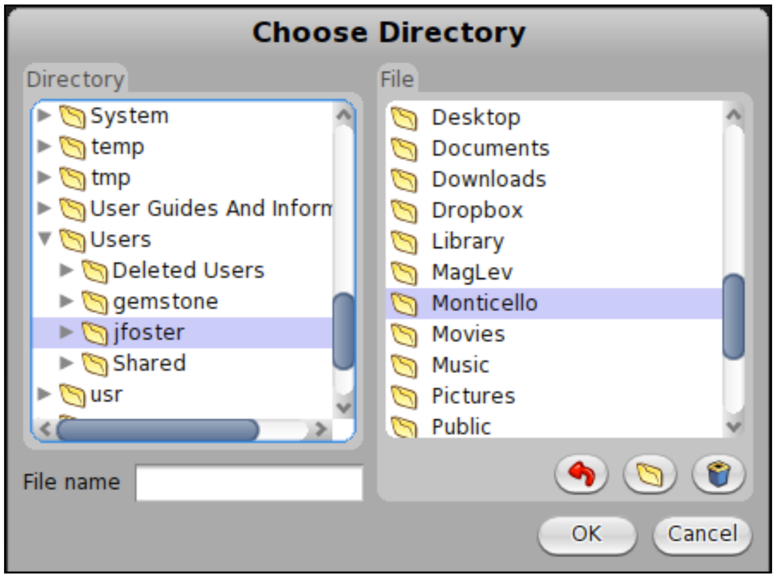
   Open your Pharo image that has the Los Boquitas application we have developed in earlier chapters. Open a Monticello Browser by left-clicking on the desktop and selecting ‘Monticello Browser.’



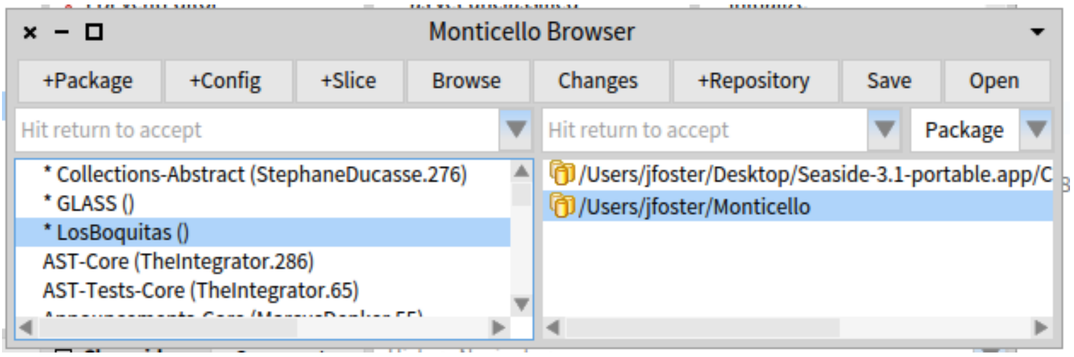
2. Create a Monticello Repository to hold the code.
   a. In your operating system create a directory to be used for Monticello packages. For now, consider something like ‘C:\Monticello’ or ‘/Monticello’ since it will be easy to find. (If you don’t want to clutter your root directory then you probably know how to pick a better spot!)
   b. With the LosBoquitas package selected, click on the ‘+Repository’ button (center). This will open a menu with a list of repository types. For now we will use a directory repository. Click on ‘directory.’



c. This will open a folder selection dialog in which you can select the directory you created above. In this example, I'm on a Macintosh and have selected '/Users/jfoster/Monticello' as my folder.



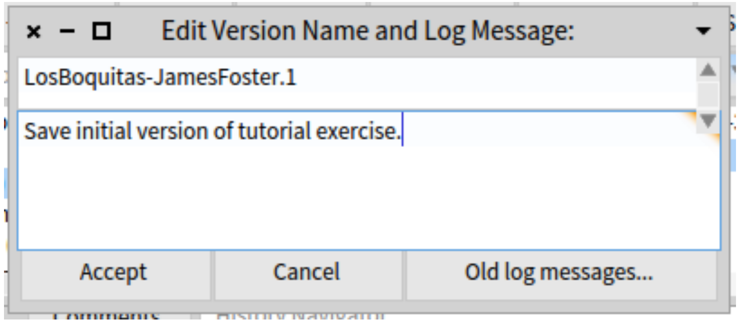
d. At this point the Monticello Browser should show the LosBoquitas package on the left and the directory repository on the right.



```
* Collections-Abstract (StephaneDucasse.276)
* GLASS ()
* LosBoquitas ()
AST-Core (TheIntegrator.286)
AST-Tests-Core (TheIntegrator.65)
```

```
/Users/jfoster/Desktop/Seaside-3.1-portable.app/C
/Users/jfoster/Monticello
```

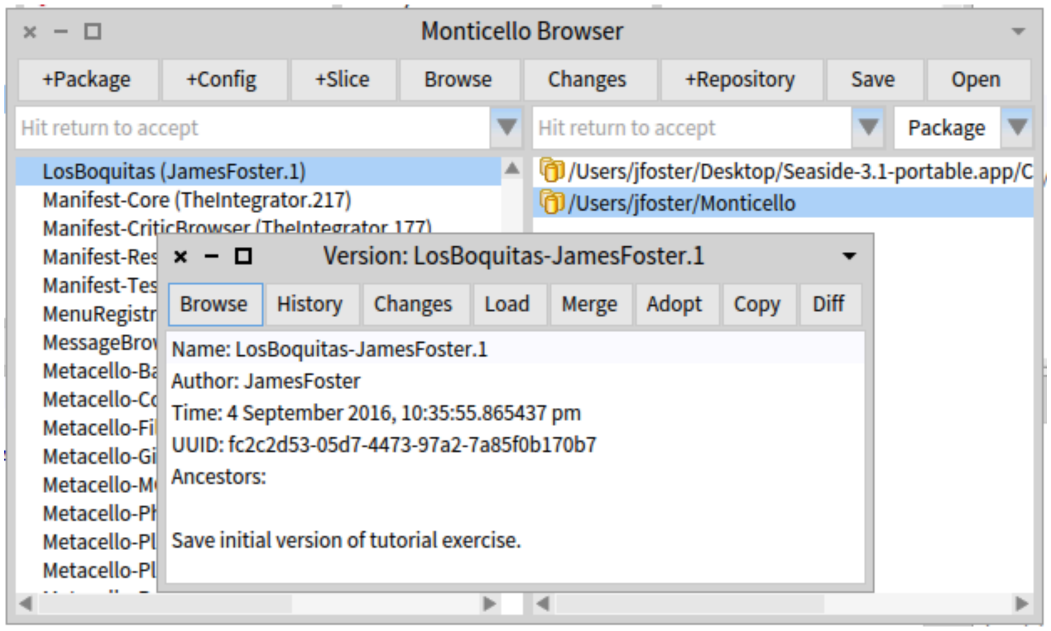
3. Save your code to the new repository.
   a. In the Monticello Browser, click the 'Save' button (the one in the center). This will bring up a dialog where you can edit the version name and a log message.



```
LosBoquitas-JamesFoster.1
Save initial version of tutorial exercise.
```

b. Verify that the version name begins with 'LosBoquitas-', has your name (not mine!), and ends with '.1' (the version number). Enter a log message in the text area and then click the 'Accept' button.

c. At this point you should see a Version window (shown in front here) and the Monticello Browser should be updated with the current version name in the package list (shown behind the version window in this screen shot).



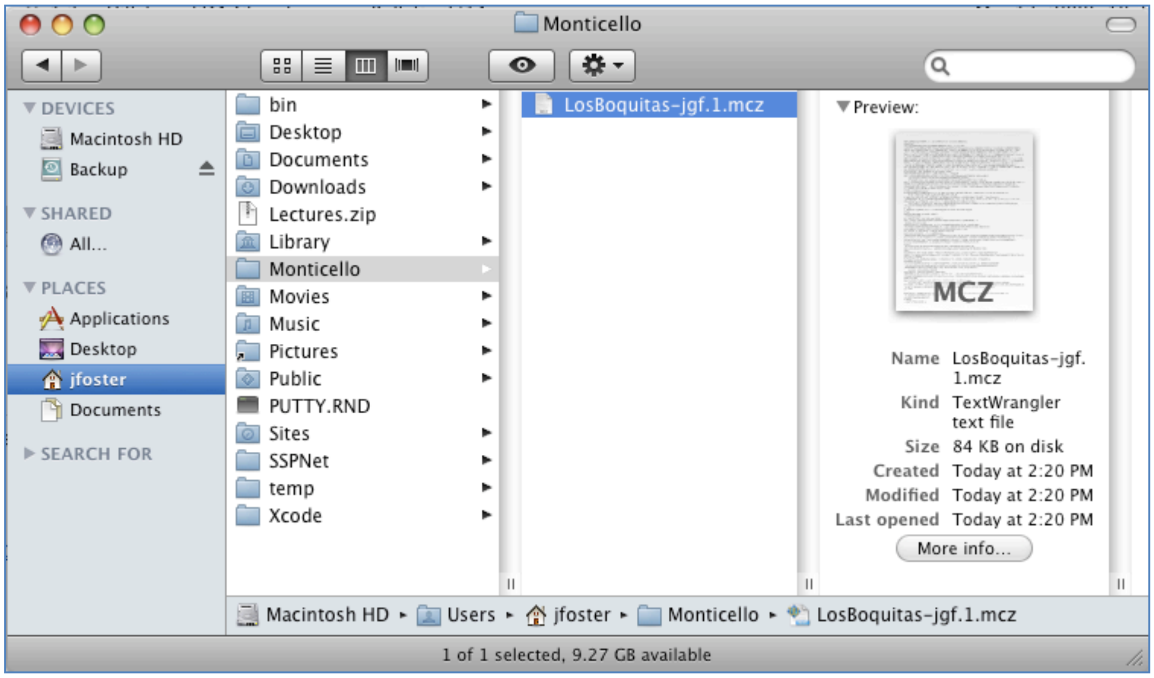
LosBoquitas (JamesFoster.1)
Manifest-Core (TheIntegrator.217)
Manifest-CriticBrowser (TheIntegrator.177)
Manifest-Res
Manifest-Tes
MenuRegistr
MessageBrov
Metacello-Ba
Metacello-Cc
Metacello-Fi
Metacello-Gi
Metacello-M
Metacello-Pl
Metacello-Pl
Metacello-Pl

/Users/jfoster/Desktop/Seaside-3.1-portable.app/C
/Users/jfoster/Monticello

Name: LosBoquitas-JamesFoster.1
Author: JamesFoster
Time: 4 September 2016, 10:35:55.865437 pm
UUID: fc2c2d53-05d7-4473-97a2-7a85f0b170b7
Ancestors:

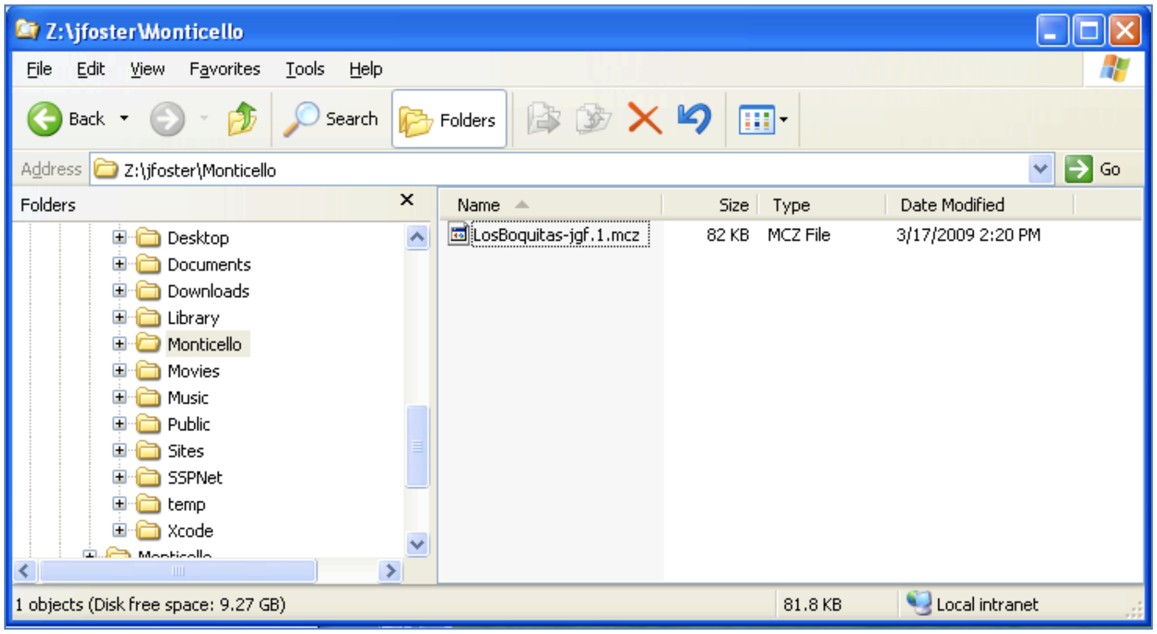
Save initial version of tutorial exercise.

4. Using your native OS tools, navigate to the directory being used as the Monticello repository. You should see something like the following (depending on your operating system).



Name	LosBoquitas-jgf.1.mcz
Kind	TextWrangler text file
Size	84 KB on disk
Created	Today at 2:20 PM
Modified	Today at 2:20 PM
Last opened	Today at 2:20 PM

1 of 1 selected, 9.27 GB available



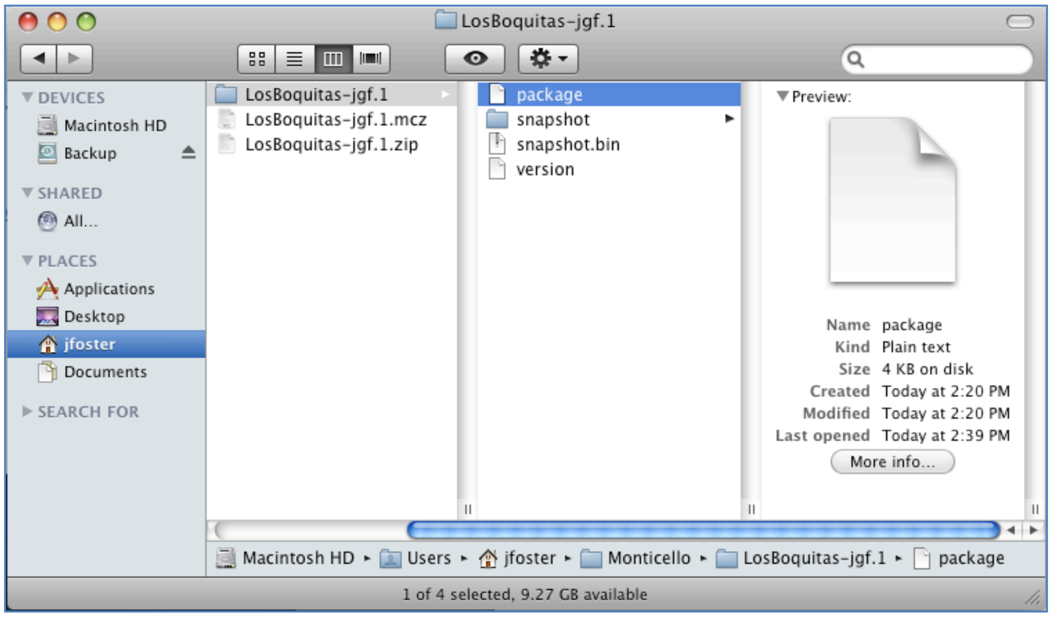
Address: Z:\jfoster\Monticello

Name	Size	Type	Date Modified
LosBoquitas-jgf.1.mcz	82 KB	MCZ File	3/17/2009 2:20 PM

1 objects (Disk free space: 9.27 GB) 81.8 KB Local intranet

a. Duplicate the MCZ file and rename it with a ZIP extension (e.g., ‘LosBoquitas.zip’).

b. Unzip the new file and view the contents. You should see three files and a directory in the new directory.



LosBoquitas-jgf.1
LosBoquitas-jgf.1.mcz
LosBoquitas-jgf.1.zip

package
snapshot
snapshot.bin
version

Name	package
Kind	Plain text
Size	4 KB on disk
Created	Today at 2:20 PM
Modified	Today at 2:20 PM
Last opened	Today at 2:39 PM

1 of 4 selected, 9.27 GB available

c. The ‘package’ file is a text file that contains the following:

```
(name 'LosBoquitas')
```

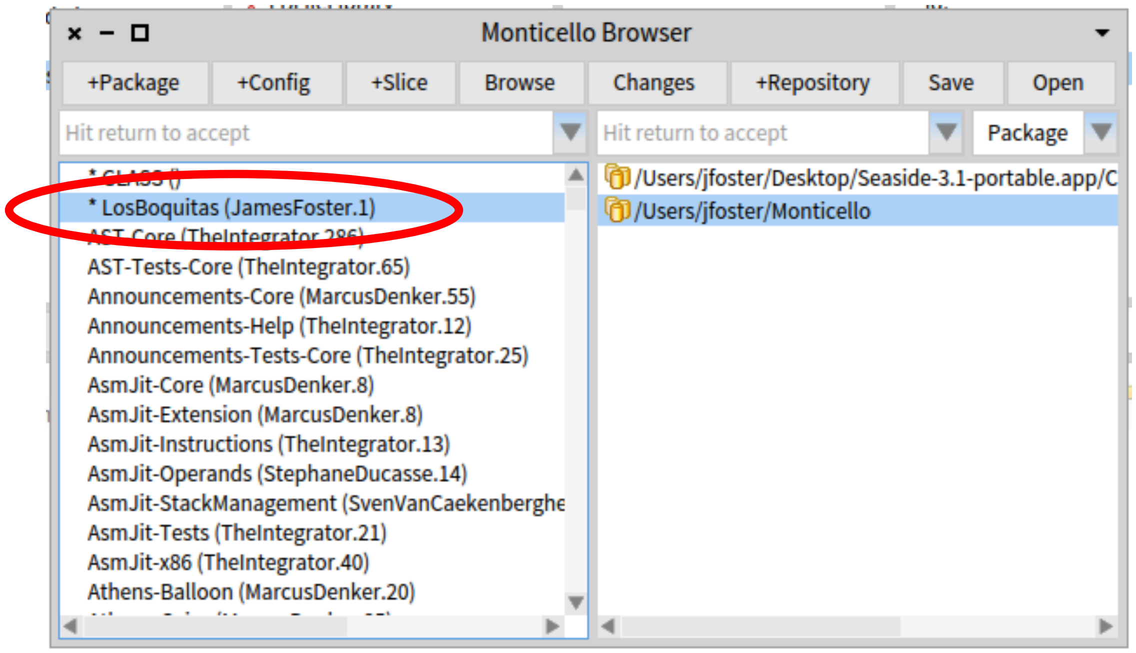
d. The ‘snapshot.bin’ file is a binary file used by the Monticello tools to load and compare packages.

e. The ‘version’ file is a text file containing a full version history of the package. In our case the history is quite simple. In the following, the formatting has been changed to make the text more readable.

```
(
   name 'LosBoquitas-JamesFoster.1'
   message 'Save initial version of tutorial exercise.'
   id 'acc13957-a73b-4112-a1d5-9152062a7e08'
   date '12 February 2011'
   time '8:05:10 pm'
   author 'JamesFoster'
   ancestors ()
   stepChildren ()
)
```

f. The ‘snapshot’ directory contains a text file, ‘sources.st’ that is suitable for filing in to a Pharo image to generate the classes and methods in the package.

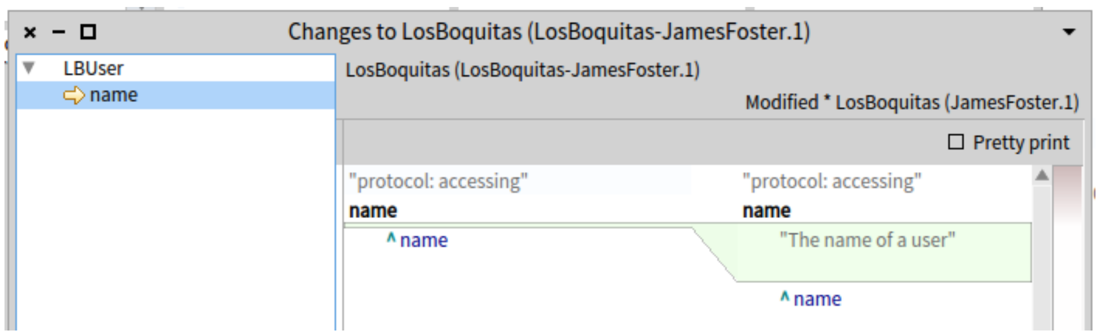
5. View changes using Monticello.
    a. Select a method in the LosBoquitas package and modify it. For this example, I've modified LBUser>>#'name' to add a comment. Note that the Monticello Browser now shows an asterisk next to the package name.



* CLASS ()
* LosBoquitas (JamesFoster.1)
AST-Core (TheIntegrator.286)
AST-Tests-Core (TheIntegrator.65)
Announcements-Core (MarcusDenker.55)
Announcements-Help (TheIntegrator.12)
Announcements-Tests-Core (TheIntegrator.25)
AsmJit-Core (MarcusDenker.8)
AsmJit-Extension (MarcusDenker.8)
AsmJit-Instructions (TheIntegrator.13)
AsmJit-Operands (StephaneDucasse.14)
AsmJit-StackManagement (SvenVanCaekenberghe
AsmJit-Tests (TheIntegrator.21)
AsmJit-x86 (TheIntegrator.40)
Athens-Balloon (MarcusDenker.20)

/Users/jfoster/Desktop/Seaside-3.1-portable.app/C
/Users/jfoster/Monticello

    b. With a package and repository selected, click on the 'Changes' button to open a Patch Browser. The saved package contents are shown in red type and the new package contents are shown in green type.



LosBoquitas (LosBoquitas-JamesFoster.1)
Modified * LosBoquitas (JamesFoster.1)

```
"protocol: accessing"              "protocol: accessing"
name                               name
    ^ name                             "The name of a user"

                                       ^ name
```

In subsequent chapters we will look at how Monticello can be used to transfer code from Pharo to GemStone.