

PRIMIX SOLUTIONS

Core Labs

Primix Virtual Library

CORE LABS

Primix Virtual Library

© Primix Solutions
One Arsenal Marketplace
Phone (617) 923-6639 • Fax (617) 923-5139

Table of Contents

An Improved Demonstration	2	Change WebLogic Configuration	9
Setting up the Application	4	Setup the Database	10
Setup the basic environment	4	Launch WebLogic	10
Change CLOUDSCAPE_INSTALL	5	A Tour of the Virtual Library	
Compile the VlibBeans module	5	Application	11
Compile the Vlib module	6		
Configure WebLogic environment	8		

An Improved Demonstration

The Primix Virtual Library is an improved Tapestry demonstration, demonstrating, in a realistic concepts, how Tapestry integrates into a J2EE environment.

The initial Tapestry demonstration, the so-called "blue border" demo, is really more of a test bench for Tapestry during its initial development. Since it didn't access a database, it was very much a toy application.

The Primix Virtual Library is a more full-fledged demonstration. It is based on an actual application developed internally at Primix; the idea being to have an application that allows everyone at the company to share their books. A database stores information about the users of the application and the books they own and are borrowing.

Users should be able to connect to the application and begin using it immediately to search for books. If they want to borrow books they must first log in (possibly, registering themselves on the fly).

The goal was to have a useful, streamlined system that could be used internally at Primix *and* to serve as a demonstration of the full power of Tapestry.

The application consists of two modules: VlibBeans which contains the entity and session beans for the application, and Vlib, which contains the Tapestry user interface for the application.

The Virtual Library uses the Cloudscape database to store its data. Cloudscape is an all-Java database. Free developer evaluation copies are available at <http://www.cloudscape.com>. The Virtual Library was developed using Cloudscape version 3.5.

At the time of this writing, the Virtual Library requires BEA WebLogic server as its Enterprise JavaBeans container. The Virtual Library was developed using WebLogic Server 5.1 Service Pack 4. The Virtual Library makes use of WebLogic's internal container managed persistence, so it is not immediately deployable on other application servers.

Note that WebLogic ships with an earlier version of Cloudscape; it is necessary to change the WebLogic configuration to make use of the more recent version.

An evaluation copy of BEA Weblogic Server is available at <http://www.bea.com/products/weblogic/server/index.html>.

The Tapestry portion of the Virtual Library can be adapted to any servlet container, though the source code currently available is also designed to work with WebLogic. It should be possible to deploy it on another server by creating a server-specific deployment descriptor.

Finally, all development has been under Windows NT and Windows 2000. We have not yet had a chance to setup under Linux or Solaris.

Setting up the Application

This discussion assumes that you have already installed WebLogic and the Java Build Environment¹ (JBE). In addition, you must have installed Cloudscape, and have set the `CLOUDSCAPE_INSTALL` environment variable.

Setup the basic environment

First, you must select a working directory. This document assumes the directory `D:\Work`.

Create a directory `D:\Work\lib`. Copy into this directory the following Jar files:

Name	Description
<code>gnu-regexp.jar</code>	GNU Regular Expression Library, available from the Giant Java Tree: http://www.gjt.org/servlets/JCVSlet/list/gjt/gnu/regexp
<code>j2ee.jar</code>	Java 2 Enterprise Edition libraries, available from Sun: http://java.sun.com/j2ee/download.html
<code>PrimixFoundation.jar</code>	Utility classes used by Tapestry and EJBs, available (as source) from the Tapestry project: http://sourceforge.net/projects/tapestry/
<code>Tapestry.jar</code>	Core Tapestry framework, available (as source) from the Tapestry project: http://sourceforge.net/projects/tapestry/
<code>xerces.jar</code>	Xerces XML parser, available from Apache.org: http://xml.apache.org/xerces-j/index.html

¹ The JBE is part of the Tapestry project. It is a set of GNU Makefiles used to compile, build and deploy Java projects.

In addition, extract the contents of the Example archive as `D:\Work\Examples`. There will be several sub-directories: Tutorial, SessionTracker, TapestryDemo, VlibBeans and Vlib. This document is concerned only with the final two (VlibBeans and Vlib).

To compile VlibBeans, the JBE needs to know where WebLogic was installed. This is done by setting the variable `WEBLOGIC_DIR`. The best way to do this is by editing the file `config/SiteConfig.mk` in your JBE directory.

```
config/SiteConfig.mk
TOOLS_DIR := C:/Cygwin/cygwin-b20/H-i586-cygwin32/bin

JDK_DIR := C:/jdk1.2.2

WEBLOGIC_DIR := C:/WebLogic
```

Change CLOUDSCAPE_INSTALL

When the Cloudscape package is installed, it sets an environment variable `CLOUDSCAPE_INSTALL` that points to the install directory.

The default directory is `C:\Cloudscape_3.5`.

For compatibility with the Vlib makefiles, this must be edited to `C:/Cloudscape_3.5`, that is, the back slash must be converted to a forward slash.

Environment variables may be changed under Windows NT or Windows 2000 by right clicking the "My Computer" icon and selecting "Properties ...".

Compile the VlibBeans module

This module contains the EJBs used by the Vlib web application. It makes use of WebLogic's container managed persistence and includes a WebLogic specific deployment descriptor.

Change to the directory `D:\Work\Example\VlibBeans`

Execute the command `make install`:

```
make install

*** Cataloging package com.primix.vlib.ejb ... ***

*** Compiling ... ***

C:/jdk1.2.2/bin/javac.exe -d .build/classes -classpath
"D:/Work/Examples/VlibBeans;D:/Work/Examples/VlibBeans/.build/classes;C:/we
blogic/lib/weblogicaux.jar;C:/weblogic/classes;D:/Work/lib/j2ee.jar;D:/Work
/lib/PrimixFoundation.jar" com/primix/vlib/ejb/AbstractEntityBean.java
com/primix/vlib/ejb/Book.java com/primix/vlib/ejb/BookBean.java
```

```

com/primix/vlib/ejb/BookQueryBean.java com/primix/vlib/ejb/IBook.java
com/primix/vlib/ejb/IBookHome.java com/primix/vlib/ejb/IBookQuery.java
com/primix/vlib/ejb/IBookQueryHome.java
com/primix/vlib/ejb/IEntityBean.java com/primix/vlib/ejb/IKeyAllocator.java
com/primix/vlib/ejb/IKeyAllocatorHome.java
com/primix/vlib/ejb/IOperations.java
com/primix/vlib/ejb/IOperationsHome.java com/primix/vlib/ejb/IPerson.java
com/primix/vlib/ejb/IPersonHome.java com/primix/vlib/ejb/IPublisher.java
com/primix/vlib/ejb/IPublisherHome.java
com/primix/vlib/ejb/KeyAllocatorBean.java
com/primix/vlib/ejb/OperationsBean.java com/primix/vlib/ejb/Person.java
com/primix/vlib/ejb/PersonBean.java com/primix/vlib/ejb/Publisher.java
com/primix/vlib/ejb/PublisherBean.java
com/primix/vlib/ejb/RegistrationException.java

*** Copying META-INF resources ... ***

Copying: ejb-jar.xml weblogic-ejb-jar.xml Book-weblogic-cmp.xml Person-
weblogic-cmp.xml Publisher-weblogic-cmp.xml

*** Building VlibBeans.jar ... ***

C:/jdk1.2.2/bin/jar.exe cf VlibBeans.jar -C .build/classes .

*** Creating VlibBeans-deploy.jar ... ***

cd .build ; \
C:/jdk1.2.2/bin/java.exe -classpath
"C:/weblogic/lib/weblogicaux.jar;C:/weblogic/classes;D:/Work/lib/j2ee.jar;D
:/Work/lib/PrimixFoundation.jar" \
weblogic.ejbcb -keepgenerated ../VlibBeans.jar ../VlibBeans-deploy.jar
[EJB]: Creating output jar:../VlibBeans-deploy.jar

*** Installing VlibBeans.jar to ../../lib ***

*** Installing VlibBeans-deploy.jar to ../../lib ... ***

```

The last step, "Creating VlibBeans-deploy.jar" may take several minutes.

This creates two Jar files: `vlibBeans.jar` and `vlibBeans-deploy.jar`. The first is the client jar, used to compile the Vlib application². The second is the deployment jar, that contains the extra classes and resources needed to deploy the EJBs using WebLogic.

Compile the Vlib module

This module contains the Vlib web application as well as a WebLogic deployment descriptor.

² In some application servers, it is possible to create a true client Jar, that contains only the EJB interfaces (and data objects passed as paramters or return values). This client Jar also contains the bean implementation classes, which aren't used by the client.

Change to the directory D:\Work\Examples\Vlib **and execute** make install:

```
make install

*** Cataloging package com.primix.vlib ... ***

*** Cataloging package com.primix.vlib.pages ... ***

*** Cataloging package com.primix.vlib.components ... ***

*** Compiling ... ***

C:/jdk1.2.2/bin/javac.exe -d .build/wapp/WEB-INF/classes -classpath
"D:/Work/Examples/Vlib;D:/Work/Examples/Vlib/.build/wapp/WEB-
INF/classes;D:/Work/lib/j2ee.jar;D:/Work/lib/xerces.jar;D:/Work/lib/PrimixF
oundation.jar;D:/Work/lib/Tapestry.jar;D:/Work/lib/VlibBeans.jar"
com/primix/vlib/EntitySelectionModel.java
com/primix/vlib/IExternalPage.java com/primix/vlib/Protected.java
com/primix/vlib/VirtualLibraryApplication.java
com/primix/vlib/VirtualLibraryServlet.java
com/primix/vlib/pages/ConfirmBookDelete.java
com/primix/vlib/pages/EditBook.java com/primix/vlib/pages/EditProfile.java
com/primix/vlib/pages/Home.java com/primix/vlib/pages/Login.java
com/primix/vlib/pages/Logout.java com/primix/vlib/pages/Matches.java
com/primix/vlib/pages/MyBooks.java com/primix/vlib/pages/NewBook.java
com/primix/vlib/pages/PersonPage.java com/primix/vlib/pages/Register.java
com/primix/vlib/pages/ViewBook.java com/primix/vlib/components/Book.java
com/primix/vlib/components/Border.java
com/primix/vlib/components/ExternalLink.java
com/primix/vlib/components/Person.java

*** Copying WEB-INF resources ... ***

Copying: web.xml weblogic.xml

*** Copying package resources ...***

Copying: Vlib.application ConfirmBookDelete.html EditBook.html
EditProfile.html Home.html Login.html Logout.html Matches.html MyBooks.html
NewBook.html PersonPage.html Register.html ViewBook.html
ConfirmBookDelete.jwc EditBook.jwc EditProfile.jwc Home.jwc Login.jwc
Logout.jwc Matches.jwc MyBooks.jwc NewBook.jwc PersonPage.jwc Register.jwc
ViewBook.jwc Border.html Book.jwc Border.jwc Person.jwc

*** Building Vlib.war ... ***

C:/jdk1.2.2/bin/jar.exe cf Vlib.war -C .build/wapp .

*** Installing Vlib.war to ../../lib ***
```

This creates the file vlib.war **in** D:\Work\lib.

Configure WebLogic environment

The Vlib web application requires several frameworks to be in the Weblogic class path at execution time (Tapestry.jar, PrimixFoundation.jar, xerces.jar, gnu-regexp.jar). There's also some additional configuration necessary to use Cloudscape 3.0.

This involves executing the wlconfig command to set the necessary parameters. Normally, this is a time consuming and error prone process, but using the JBE we can perform the necessary operations automatically.

In the D:\Work\Example\Vlib directory, execute the command `make configure-weblogic`:

```
make configure-weblogic

*** Configuring WebLogic Server options ***

C:/WebLogic/bin/wlconfig.exe -mx128M -mx128M -
Dweblogic.class.path="C:/weblogic/
lib/weblogic510sp4.jar;C:/weblogic/license;C:/weblogic/classes;C:/weblogic/
lib/weblogicaux.jar;D:/Work/lib/gnu-
egexp.jar;D:/Work/lib/xerces.jar;D:/Work/lib/Tapestry.jar;D:/Work/lib/Primi
xFoundation.jar" -Dweblogic.system.home=C:/WebLogic -
Dcloudscape.system.home="C:\Cloudscape_3.0_Eval/demo/databases" -classpath
"C:/w
eblogic/lib/weblogic510sp4boot.jar;C:/Cloudscape_3.0_Eval/lib/cloudscape.jar;C:/
Cloudscape_3.0_Eval/lib/license.jar"
Updating CLASSPATH from:

"C:/weblogic/lib/weblogic510sp4boot.jar;C:/Kawa4.1/cmd/Cloudscape_3.0_Ev
al/lib/cloudscape.jar;C:/Kawa4.1/cmd/Cloudscape_3.0_Eval/lib/license.jar"
to

"C:/weblogic/lib/weblogic510sp4boot.jar;C:/Cloudscape_3.0_Eval/lib/cloud
scape.jar;C:/Cloudscape_3.0_Eval/lib/license.jar"
WebLogic new settings are:

CLASSPATH Prefix
C:/weblogic/lib/weblogic510sp4boot.jar;C:/Cloudscape_3.0_Eva
l/lib/cloudscape.jar;C:/Cloudscape_3.0_Eval/lib/license.jar
CLASSPATH
C:/weblogic/lib/weblogic510sp4boot.jar;C:/Cloudscape_3.0_Eva
l/lib/cloudscape.jar;C:/Cloudscape_3.0_Eval/lib/license.jar;C:\weblogic\jre1_2\l
ib\tools.jar;C:\weblogic\jre1_2\jre\lib\rt.jar;C:\weblogic\jre1_2\jre\lib\i
18n.j
ar;C:\weblogic\license;C:\weblogic\classes\boot;C:\weblogic\classes;C:\webl
ogic\
lib\weblogicaux.jar;C:\weblogic\eval\cloudscape\lib\cloudscape.jar
JAVA_HOME C:\weblogic\jre1_2
WEBLOGIC_LICENSEDIR C:\weblogic\license
WEBLOGIC_HOME C:\weblogic
system properties:
    java.security.manager
    java.security.policy==C:\weblogic\weblogic.policy
```

```

weblogic.system.home=C:/WebLogic
java.compiler=symcjit

weblogic.class.path=C:/weblogic/lib/weblogic510sp4.jar;C:/weblogic/license;C:/weblogic/classes;C:/weblogic/lib/weblogicaux.jar;D:/Work/lib/gnu-regexp.jar;D:/Work/lib/xerces.jar;D:/Work/lib/Tapestry.jar;D:/Work/lib/PrimitiveFoundation.jar

cloudscape.system.home=C:\Cloudscape_3.0_Eval/demo/databases

INITIAL_HEAP          128 MB
MAX_HEAP              128 MB
SERVERCLASSPATH
C:/weblogic/lib/weblogic510sp4boot.jar;C:/Cloudscape_3.0_Eval/lib/cloudscape.jar;C:/Cloudscape_3.0_Eval/lib/license.jar;C:\weblogic\jre1_2\jre\lib\rt.jar;C:\weblogic\jre1_2\jre\lib\i18n.jar;C:\weblogic\classes\boot;C:\weblogic\eval\cloudscape\lib\cloudscape.jar

Type "wlconfig -help" for program usage.

```

This assumes that you have installed patch SP4 for WebLogic, and that the two path Jars (weblogic510sp4.jar and weblogic510sp4boot.jar) are in C:\WebLogic\lib.

Having done all this, it will be possible launch WebLogic as an NT service or using its Start Menu icon.

Change WebLogic Configuration

Before WebLogic can be run, we must add deployment information for the EJBs and web application, and set up pools and data sources.

Edit the weblogic.properties file and add the following lines:

```

weblogic.httpd.webApp.vlib=D:/Work/Examples/Vlib/.build/wapp

weblogic.ejb.deploy=\
    D:/Work/lib/VlibBeans-deploy.jar

weblogic.jdbc.connectionPool.com.primix.vlib.Pool=\
    url=jdbc:cloudscape:vlib,\
    driver=COM.cloudscape.core.JDBCDriver,\
    initialCapacity=1,\
    maxCapacity=2,\
    capacityIncrement=1,\
    props=user:none;password:none;server:none

weblogic.allow.reserve.weblogic.jdbc.connectionPool.com.primix.vlib.Pool=everyone

```

```
weblogic.jdbc.TXDataSource.com.primix.vlib.DataSource=com.primix.vlib.Pool
```

In addition, you should edit the configuration to remove the reference to the Cloudscape install directory (that was set in the previous step), and comment out all the demos that are included with WebLogic.

Due to a WebLogic bug³, we can't deploy the `vlib.war` file itself, instead we deploy its build image (generated by the JBE), from which the `vlib.war` file is generated.

Setup the Database

The database is shipped as a Zip archive.

Open the file `D:\Work\Examples\VlibBeans\Vlib.zip`.

Extract its contents to `C:\Cloudscape_3.0_Eval\demo\databases` (this may vary, depending on where you installed Cloudscape).

This will create the directory `C:\Cloudscape_3.0_Eval\demo\databases\vlib`, which is the Vlib database. It comes with three users and a dozen or so books:

Name	E-Mail	Password
Alternate Test User	alt@primix.com	secret
Howard Ship	hship@primix.com	secret
Test User	test@primix.com	secret

Launch WebLogic

Finally, launch WebLogic from the Start menu. If all the steps have been followed, it will launch without error.

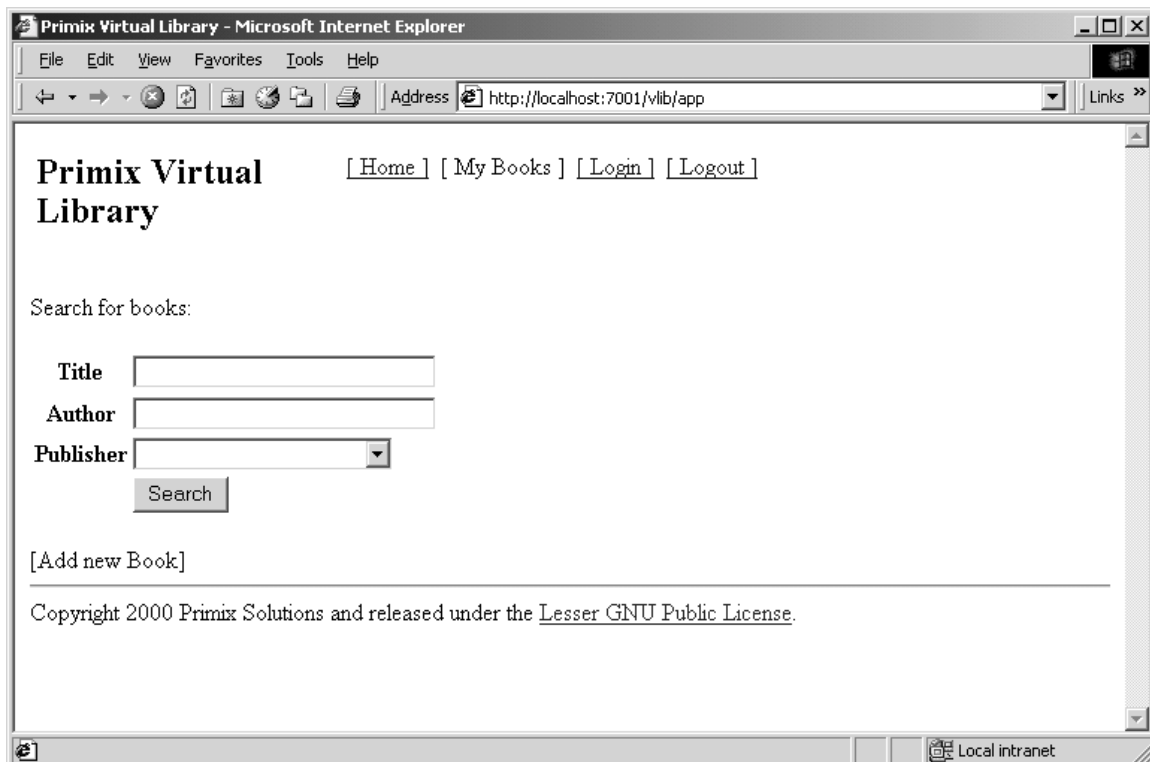
³ When WebLogic deploys a War file, it partially unpacks it, extracting all the `.class` files. Unfortunately, it doesn't unpack all the class resource files Tapestry needs, such as HTML templates and specification files.

A Tour of the Virtual Library Application

You can start the Vlib application with the URL:

<http://localhost:7001/vlib/app>

You'll then see the home page for the application, from which you may search for books.

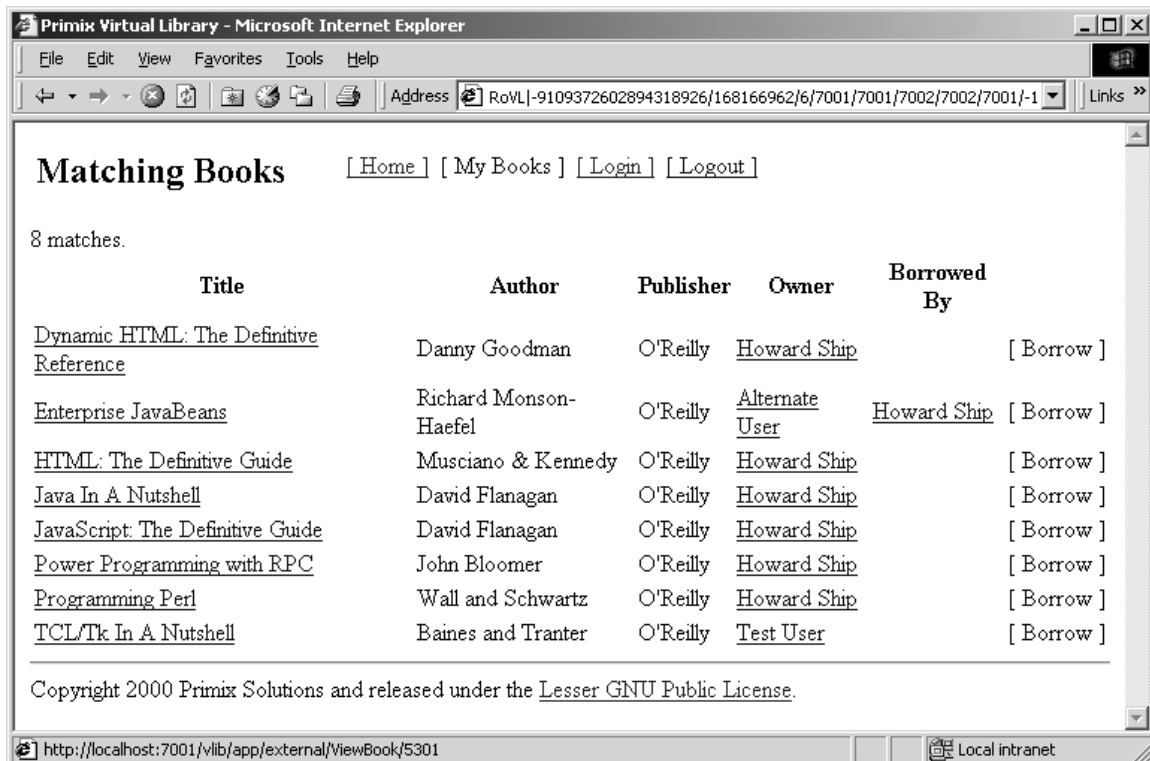


The current version of the Primix Virtual Library is simple, presentation wise. We will shortly be sprucing it up with graphics and colors.

Let's start by finding all the O'Reilly books in the database.

Select "O'Reilly" in the Publisher pulldown, then click the Search button.

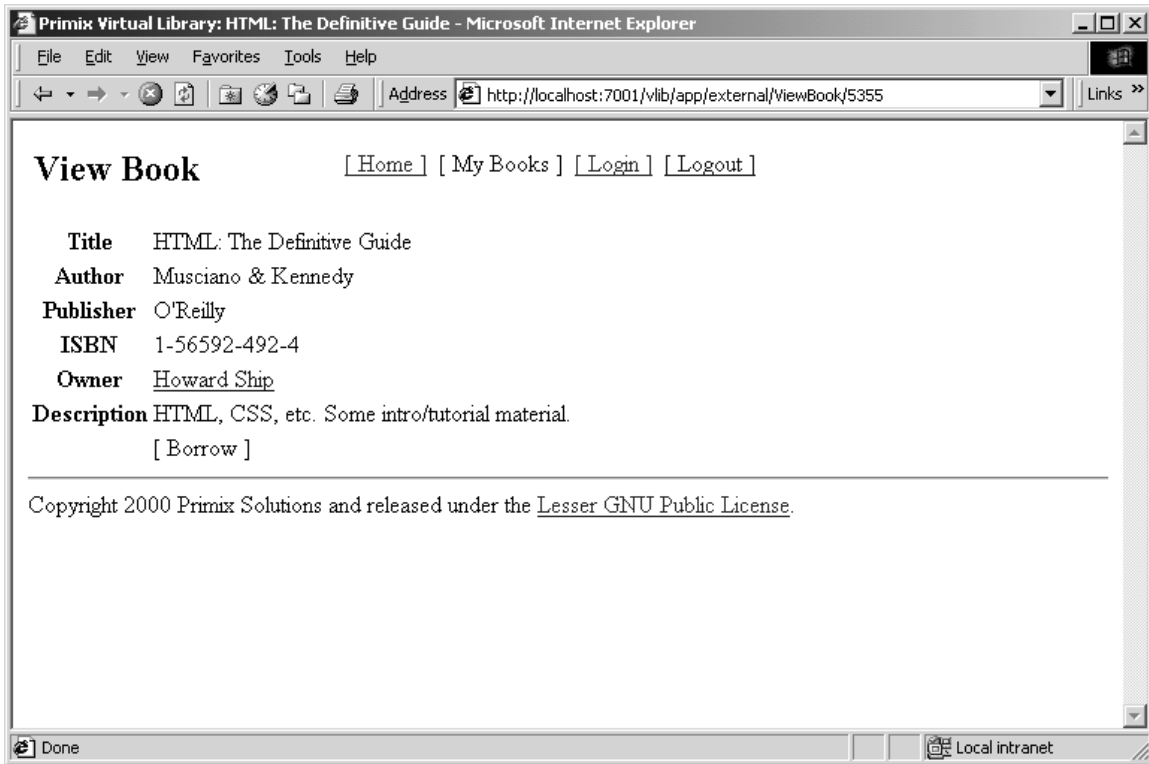
You'll get a list of all the books, sorted by title.



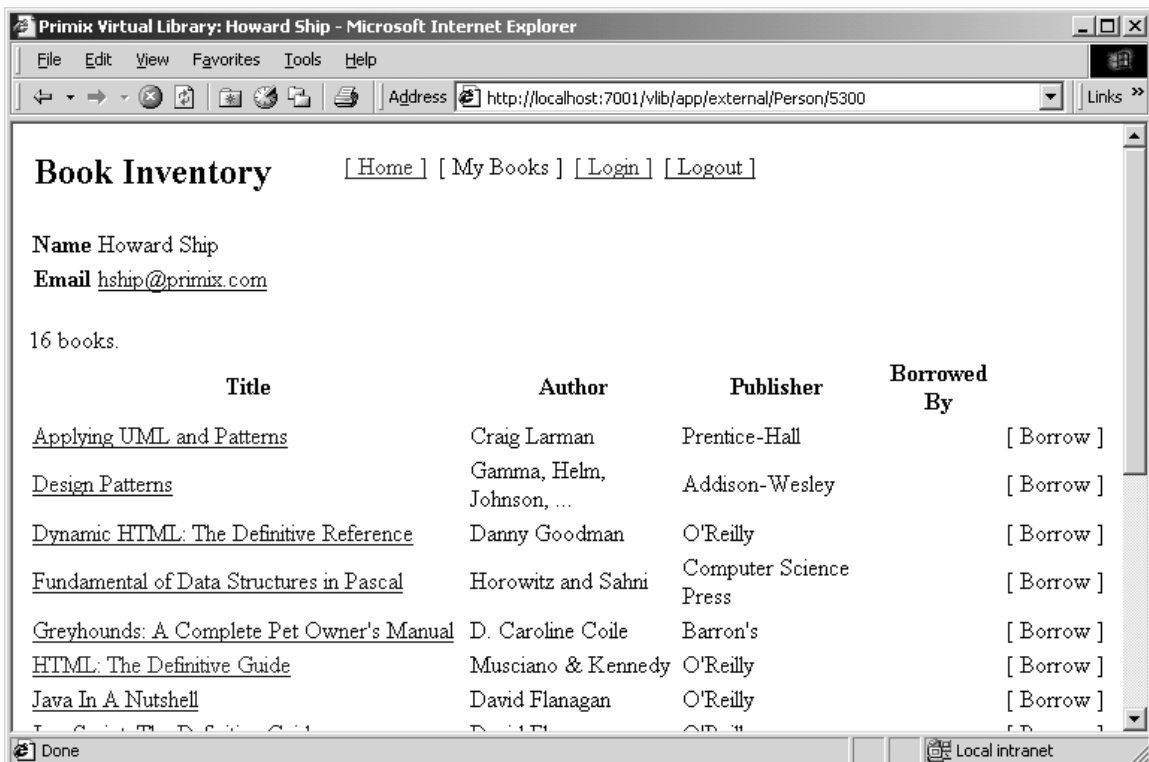
The screenshot shows a Microsoft Internet Explorer window titled "Primix Virtual Library - Microsoft Internet Explorer". The address bar contains a long URL: "RoVL|-9109372602894318926/168166962/6/7001/7001/7002/7002/7001/-1". The page content includes navigation links: "[Home] [My Books] [Login] [Logout]". Below this, it says "8 matches." and displays a table of search results. The table has five columns: Title, Author, Publisher, Owner, and Borrowed By. The results list eight books, all published by O'Reilly, with various owners and a "[Borrow]" link for each. At the bottom of the page, there is a copyright notice: "Copyright 2000 Primix Solutions and released under the Lesser GNU Public License." The browser's status bar shows the URL "http://localhost:7001/vlib/app/external/ViewBook/5301" and "Local intranet".

Title	Author	Publisher	Owner	Borrowed By
Dynamic HTML: The Definitive Reference	Danny Goodman	O'Reilly	Howard Ship	[Borrow]
Enterprise JavaBeans	Richard Monson-Haefel	O'Reilly	Alternate User	Howard Ship [Borrow]
HTML: The Definitive Guide	Musciano & Kennedy	O'Reilly	Howard Ship	[Borrow]
Java In A Nutshell	David Flanagan	O'Reilly	Howard Ship	[Borrow]
JavaScript: The Definitive Guide	David Flanagan	O'Reilly	Howard Ship	[Borrow]
Power Programming with RPC	John Bloomer	O'Reilly	Howard Ship	[Borrow]
Programming Perl	Wall and Schwartz	O'Reilly	Howard Ship	[Borrow]
TCL/Tk In A Nutshell	Baines and Tranter	O'Reilly	Test User	[Borrow]

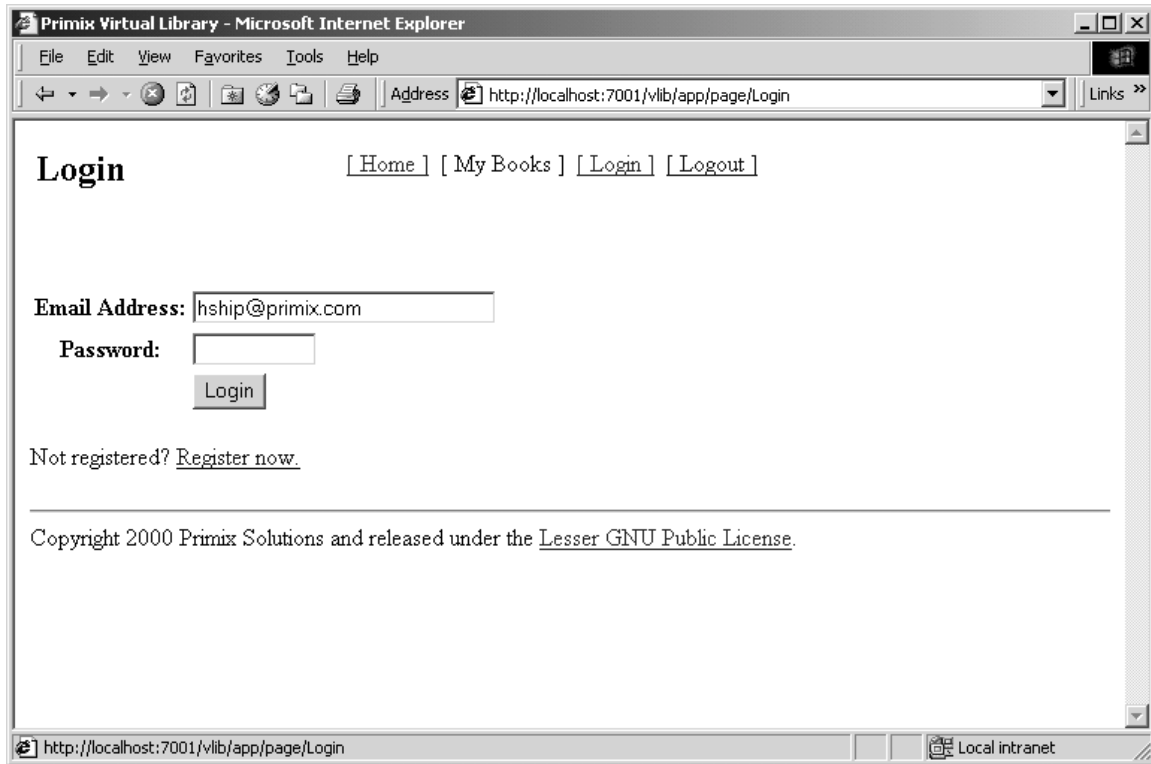
You can click on the title of a book to get more information about that book:



Or, you can click on the name of a person to see the books they own:

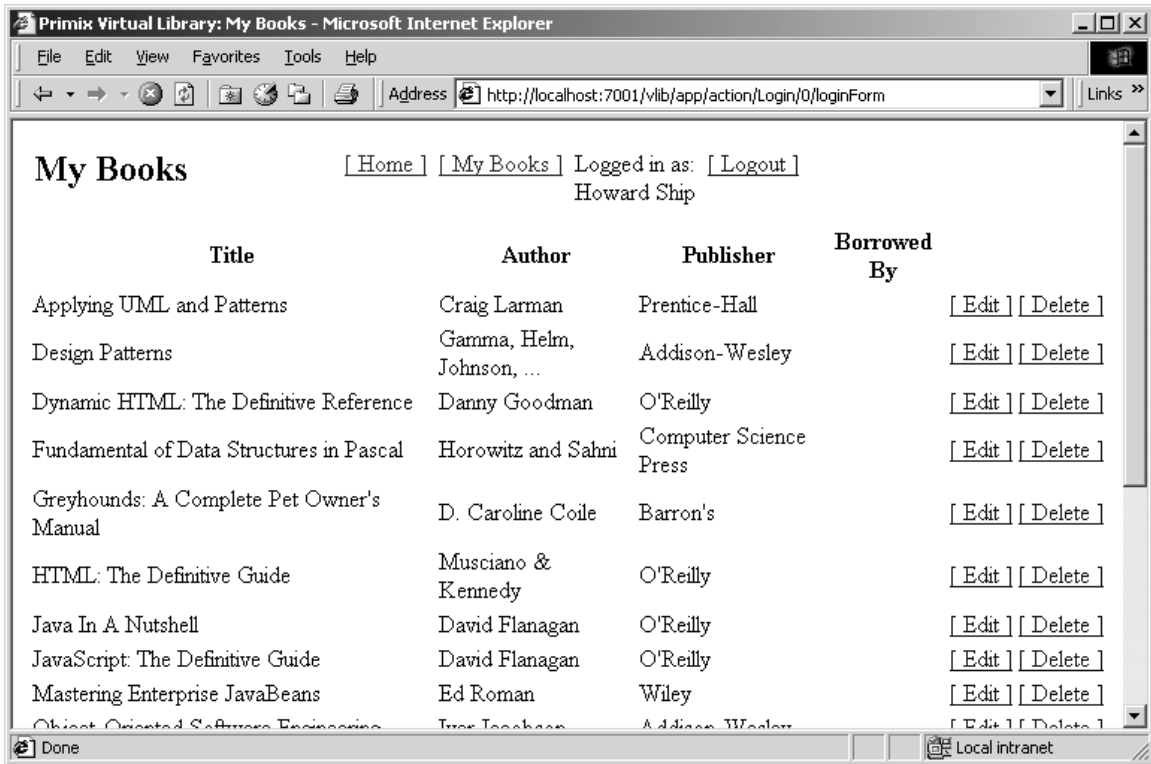


You might notice the "[Borrow]" that appears on several pages ... this is the borrow button, and is only enabled once you login. Use the "[Login]" button to log into the application.



The application uses a cookie to remember your email address (which is used as a unique identity when logging in). Enter the email address "hship@primix.com" and the password "secret".

This brings you to the "My Books" page, where you can manage the books you own:



The remainder of the application, adding and editing books, is basically self-explanatory.