

GETTING STARTED WITH BLACKBOARD BUILDING BLOCKS™ (PART 2)

Priyanka Gupta, Blackboard® Inc.



ABOUT THE PRESENTER

- Principal Software Engineer
- Joined Blackboard in 2004

AUDIENCE

- Intended for developers creating Blackboard Building Blocks™ for the *Blackboard Learn™*.
- A thorough understanding of Java®, Servlets, and Java Server Pages is assumed.
- Proficiency in Java server-side programming is also assumed.

INTRODUCTION

Presentation intended to show how to use the Blackboard Building Blocks APIs.

Will be going over examples from the “Blackboard Building Blocks Getting Started Guide” contained in the Blackboard Software Development Kit TM which can be downloaded from

<http://www.edugarage.com/display/BBDN/Documentation>

<http://www.blackboard.com/Communities/Partners/Building-Blocks.aspx>

Used EasyEclipse Server Java

Eclipse packaged with a bunch of useful plug-ins

Free from www.easyeclipse.org

TOPICS TO COVER

- Short introduction on Blackboard Building Blocks
- Overview of the APIs
- Examples

WHAT IS A BUILDING BLOCK?

A Building Block is a .zip or .war file that has a certain structure defined by Blackboard. This structure primarily consists of a file-directory hierarchy containing custom-code tied together by a Blackboard Learn manifest.

A Building Block is simply a set of files installed on the Blackboard application server in a way that is structured so the server has predefined entry points to call upon the functionality of the Building Block. The entry points are URL links that are tracked in the Blackboard Learn database and associated with key entities such as content handlers and navigation items.



HELLOWORLD

- Start with a course tool (control panel) Building Block
- Just says Helloworld
- Will go over the jsp

HELLOWORLD.JSP

```
<%@ page language="java" import="java.util.*"  
pageEncoding="UTF-8"%>
```

```
<html>
```

```
<body>
```

```
  This is my JSP page. <br>
```

```
  <% out.print("hello world"); %><br>
```

```
</body>
```

```
</html>
```


HELLOWORLD RESULT



TAG LIBRARIES

Use tag libraries to get the Blackboard software “look and feel”

bbUI.tld -Encapsulates Blackboard user interface elements to allow developers easy access to the Blackboard look and feel

bbData.tld (context tag) -Encapsulates interaction with the Blackboard data and logic layer to allow separation of the UI elements (bbUI) and other page logic. Specifically, bbData:context also allows you to determine the current user and course (assuming that you are in a course context)

bbNG.tld

TAG LIBRARIES (CONT'D)

Rewrite helloworld.jsp

```
<%@ page language="java" import="java.util.*"  
pageEncoding="UTF-8"%>
```

```
<%@ taglib uri="/bbUI" prefix="bbUI" %>
```

```
<%@ taglib uri="/bbData" prefix="bbData"%>
```

Tag Lib
Directives

```
<%
```

```
String iconUrl = "/images/ci/icons/bookopen_u.gif";
```

```
String page_title = "HelloWorld: helloworld.jsp";
```

```
%>
```

Variables
used later on

TAG LIBRARIES (CONT'D)

Gives you access to the context –
not actually needed in this example

Should put everything
inside here

```
<bbData:context id="ctx">
```

```
<bbUI:docTemplate title="<%= page_title %>">
```

```
<!-- Start Breadcrumb Navigation -->
```

```
<bbUI:breadcrumbBar environment="CTRL_PANEL"  
handle="control_panel">
```

```
<bbUI:breadcrumb>Hello World</bbUI:breadcrumb>
```

```
</bbUI:breadcrumbBar>
```

```
<!-- End Breadcrumb Navigation -->
```

```
<bbUI:titleBar iconUrl="<%=iconUrl%>">
```

```
<%= page_title %>
```

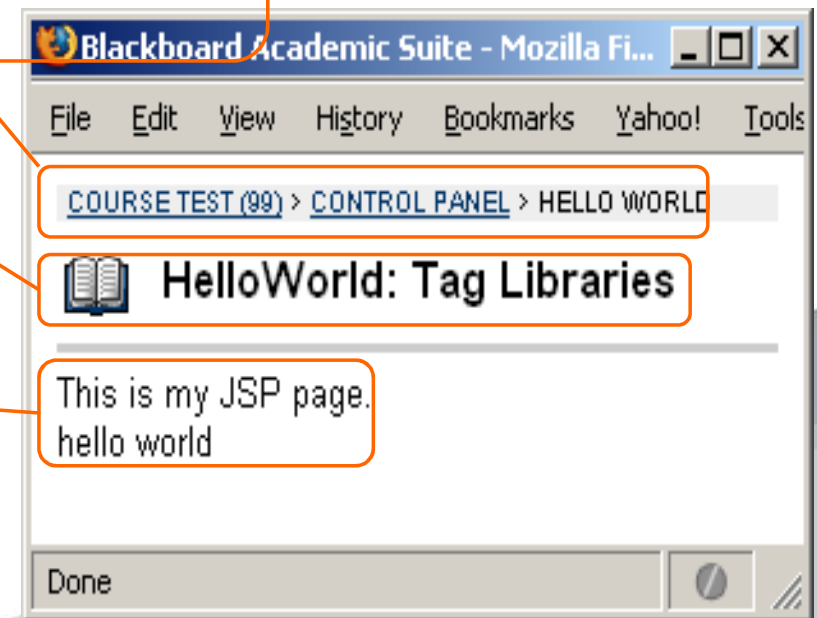
```
</bbUI:titleBar>
```

This is my JSP page.


```
<% out.print("hello world"); %><br>
```

```
</bbUI:docTemplate>
```

```
</bbData:context>
```



IDENTIFYING THE USER

Get a user object from the bbData:context

Need Blackboard software APIs to manipulate the user object

Add the Blackboard software JARs to your project. Copy following files to your eclipse workspace from /blackboard/systemlibs

bb-cms-admin.jar

bb-platform.jar

bb-taglibs.jar

Add the Blackboard software JARs to your project's Java Build Path.

Javadoc for all APIs is in the bbas_r7_1_building_blocks_api_spec directory of the Blackboard Software Development Kit.

IDENTIFYING THE USER (CONT'D)

Modify helloworld.jsp as follows:

```
<%@ page language="java" import="java.util.*,
```

```
    blackboard.data.user.*,
```

```
    blackboard.persist.*"
pageEncoding="UTF-8"%>
```

```
<%@ taglib uri="/bbUI" prefix="bbUI" %>
```

```
<%@ taglib uri="/bbData" prefix="bbData"%>
```

Imports needed
java classes

IDENTIFYING THE USER

Modify helloworld.jsp (cont'd)

```
<bbData:context id="ctx">
```

```
<%
```

```
//Get a User instance via the page context
```

```
User sessionUser = ctx.getUser();
```

```
//Get the User's Name and Id
```

```
String sessionUserName = sessionUser.getUserName();
```

```
Id sessionUserId = sessionUser.getId();
```

```
String sessionUserGivenName = sessionUser.getGivenName();
```

```
String sessionUserFamilyName = sessionUser.getFamilyName();
```

```
String sessionUserEmailAddress = sessionUser.getEmailAddress();
```

```
String sessionUserBatchID = sessionUser.getBatchUid();
```

```
%>
```

```
<bbUI:docTemplate title="<%= page_title %>">
```

A
Blackboard
API

Use the context we didn't
need in first example

Identifying the User(cont'd)

– Modify helloworld.jsp (cont'd)

```
</bbUI:titleBar>
```

```
This is my JSP page. <br>
```

```
Hi <%= sessionUserGivenName %> <%= sessionUserFamilyName %>, <br>
```

```
Your User Name is: <%= sessionUserName %>, <br>
```

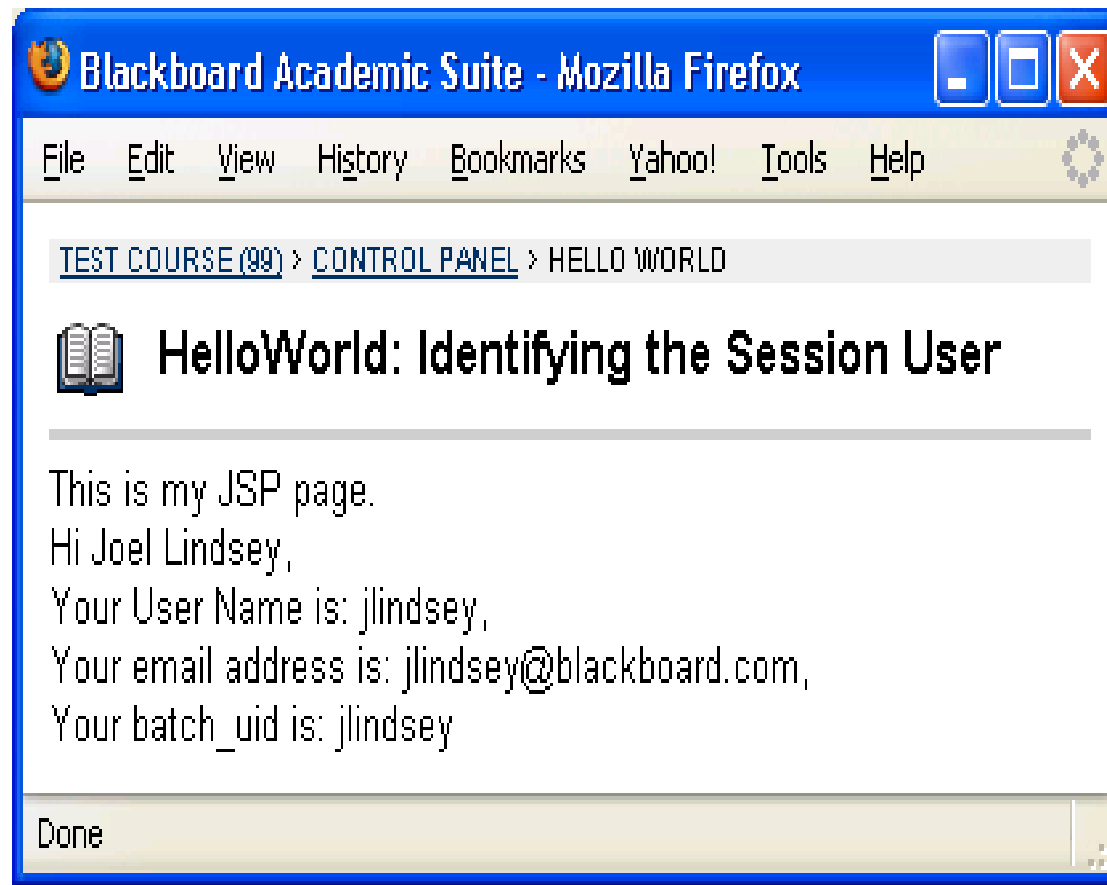
```
Your email address is: <%= sessionUserEmailAddress %>, <br>
```

```
Your batch_uid is: <%= sessionUserBatchID %> <br>
```

```
</bbUI:docTemplate>
```


IDENTIFYING THE USER(CONT'D)

Result



IDENTIFYING CURRENT COURSE

- Identify the course whose control panel the building block is invoked from
- Get the course object from the context

IDENTIFYING CURRENT COURSE (CONT'D)

Modify the jsp file again

```
<%@ page language="java"
    import="java.util.*,
        blackboard.data.user.*,
        blackboard.persist.*,
        blackboard.data.course.*,
        blackboard.persist.course.*"
    pageEncoding="UTF-8"%>
<%@ taglib uri="/bbUI" prefix="bbUI" %>
<%@ taglib uri="/bbData" prefix="bbData"%>
```

IDENTIFYING CURRENT COURSE (CONT'D)

Modify the jsp file again(cont'd)

```
String sessionUserBatchID = sessionUser.getBatchUid();
```

```
//Get the Course instance from the page context
```

```
Course sessionCourse = ctx.getCourse();
```

```
String sessionCourseId = sessionCourse.getCourseId();
```

```
String sessionCourseBatchUID = sessionCourse.getBatchUid();
```

```
String sessionCourseCourseTitle = sessionCourse.getTitle();
```

```
String sessionCourseDescription = sessionCourse.getDescription();
```

```
long sessionCourseUploadLimit = sessionCourse.getUploadLimit();
```

```
boolean sessionCourseIsAvailable = sessionCourse.getIsAvailable();
```

```
boolean sessionCourseAllowGuests = sessionCourse.getAllowGuests();
```

```
%>
```

```
<bbUI:docTemplate title="<%= page_title %>">
```

IDENTIFYING CURRENT COURSE (CONT'D)

Modify the jsp file again(cont'd)

Your batch_uid is: <%= sessionUserBatchID %>

<p>Information about this course.

<%= sessionCourseCourseTitle %> [BatchUID=<%= sessionCourseBatchUID %> : CourseId=<%= sessionCourseID %>]

 <%= sessionCourseDescription%>

Is this course available? <%= sessionCoursesAvailable?"Yes":"No" %>

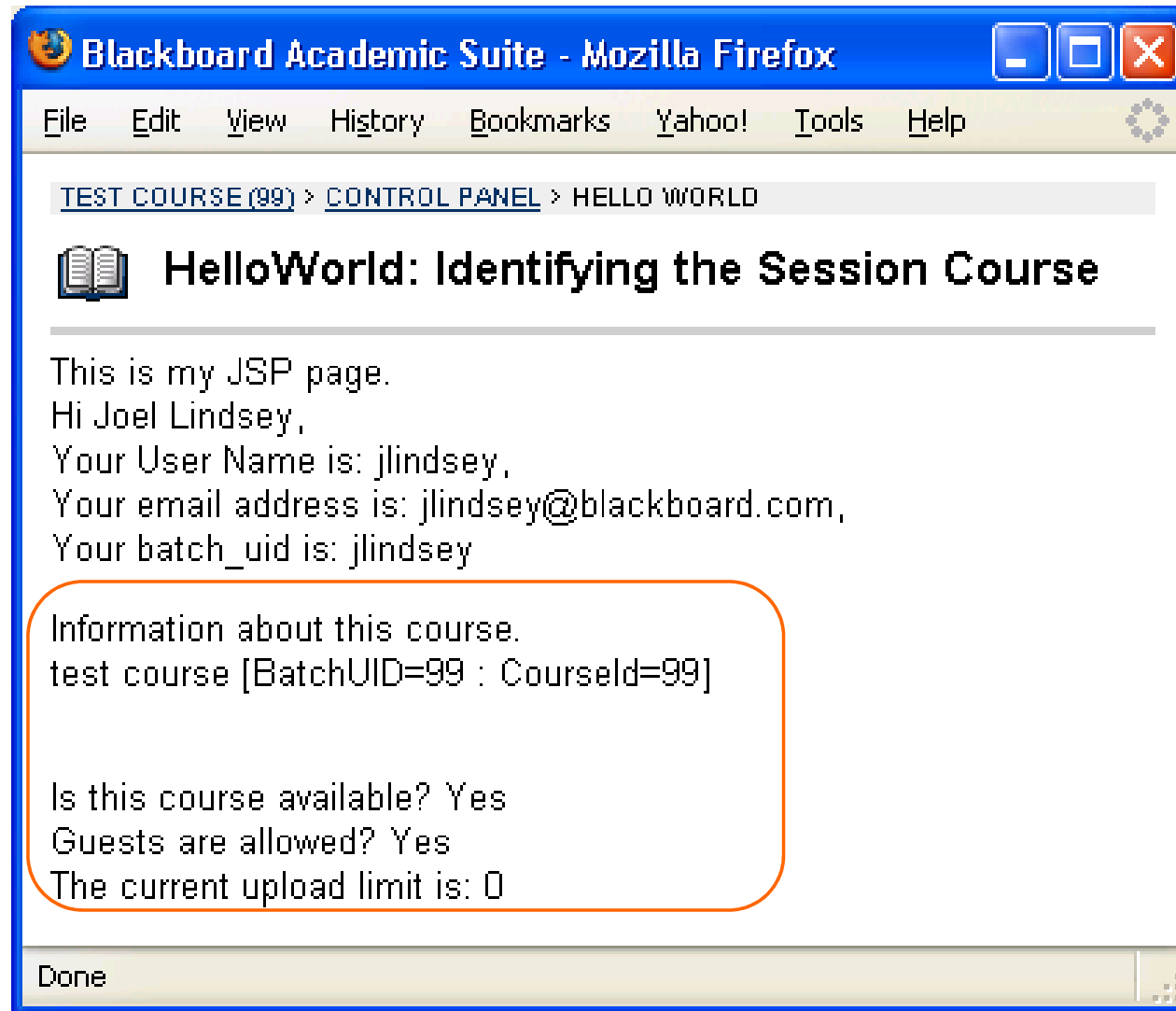
Guests are allowed? <%= sessionCourseAllowGuests?"Yes":"No" %>

The current upload limit is: <%= sessionCourseUploadLimit %> <p>

</bbUI:docTemplate>

IDENTIFYING COURSES(CONT'D)

Result



USING PERSISTENCE MANAGER

- Use persistence APIs to access course data.
 - Instead of bbData:context
- Just changes how we get the data. Display is unchanged.
- Can use persistence APIs to access user data also.
 - Can't determine current user without context though.

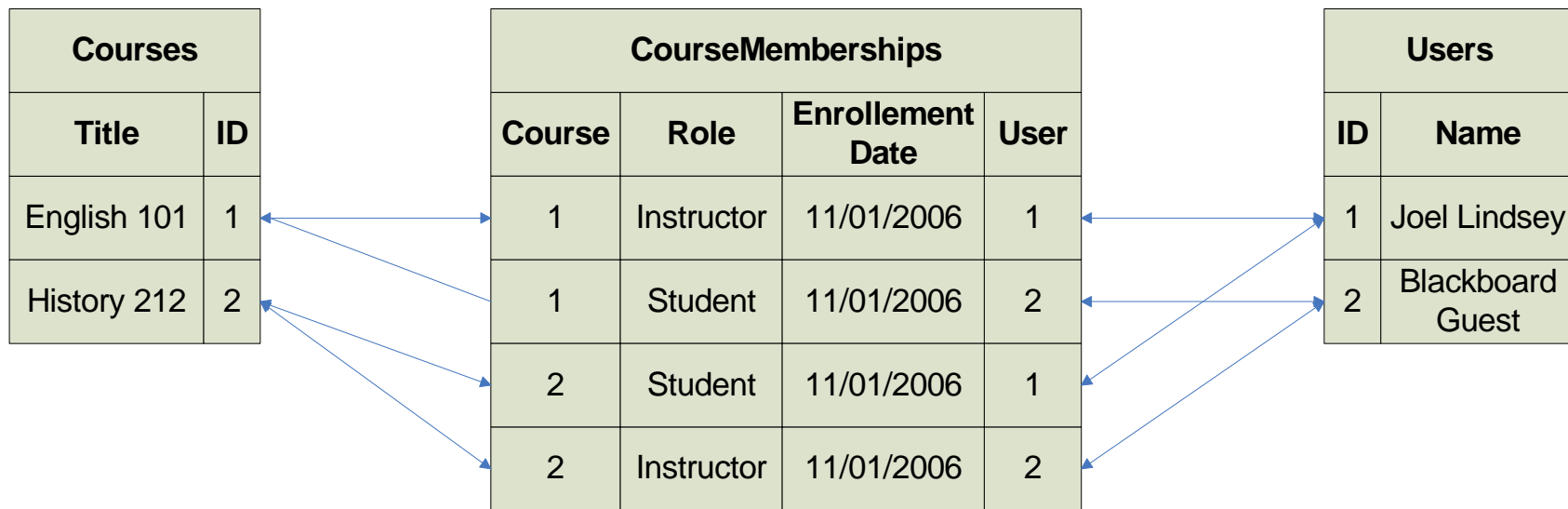
USING PERSISTENCE MANAGER

Modify the jsp file again

```
<%@ page language="java"
import="java.util.*,
    blackboard.data.user.*,
    blackboard.persist.*,
    blackboard.data.course.*,
    blackboard.persist.course.*,
    blackboard.platform.*"
pageEncoding="UTF-8"%>
<%@ taglib uri="/bbUI" prefix="bbUI" %>
<%@ taglib uri="/bbData" prefix="bbData"%>
```


ACCESSING MEMBERSHIPS

Add course membership information to the Building Block



Use CourseMembershipDbLoader's
loadByCourseAndUserId() method

ACCESSING MEMBERSHIPS

Modify the jsp

```
boolean sessionCourseAllowGuests = sessionCourse.getAllowGuests();
```

```
// get the membership data to determine the User's Role
CourseMembership crsMembership = null;
CourseMembershipDbLoader crsMembershipLoader = null;
String errMsg = null;
crsMembershipLoader =
    (CourseMembershipDbLoader)
bbPm.getLoader(CourseMembershipDbLoader.TYPE);
try {
    crsMembership = crsMembershipLoader.loadByCourseAndUserId(courseId,
sessionUserId);
} catch (KeyNotFoundException e) {
    // There is no membership record.
    errMsg = "There is no membership record. Better check this out:" + e;
} catch (PersistenceException pe) {
    // There is no membership record.
    errMsg = "An error occurred while loading the User. Better check this out:" + pe;
}
```

Same bbPm we
used for the
course loader

ACCESSING MEMBERSHIPS (CONT'D)

- Modify the jsp (cont'd)

```
} catch (PersistenceException pe) {  
    // There is no membership record.  
    errMsg = "An error occurred while loading the User. Better check  
this out:" + pe;  
}  
CourseMembership.Role crsMembershipRole =  
crsMembership.getRole();  
String crsMembershipRoleStr = crsMembershipRole.toString();  
java.util.Calendar crsMembershipEnrollmentDate =  
crsMembership.getEnrollmentDate();  
boolean crsMembershipsAvailable =  
crsMembership.getIsAvailable();  
%>  
<bbUI:docTemplate title="<%= page_title %>">
```

ACCESSING MEMBERSHIPS (CONT'D)

- Modify the jsp (cont'd)

```
The current upload limit is: <%= sessionCourseUploadLimit %> <p>  
<%  
    // Get date information  
    int enrYear = crsMembershipEnrollmentDate.get(Calendar.YEAR);  
    int enrMonth = crsMembershipEnrollmentDate.get(Calendar.MONTH);  
    int enrDay = crsMembershipEnrollmentDate.get(Calendar.DATE);  
    String EnrDateStr = "" + enrMonth + "/" + enrDay + "/" + enrYear;  
%>  
<p>Information about this membership. <br>  
The availability of this membership is:  
<%= crsMembershipsIsAvailable?"Available":"Unavailable"%> <br>  
Date membership created: <%=EnrDateStr%> <br>  
The User's Role is: <%=crsMembershipRoleStr%> </p>  
</bbUI:docTemplate>
```



BUILDING BLOCK MANAGEMENT CONFIGURATION & REMOVAL

Configuration:

Not required

Configuration data may be stored in the file system under the directory returned by `PlugUtil.getConfigDirectory()`

Removal:

Not required.

Can perform necessary cleanup prior to removal

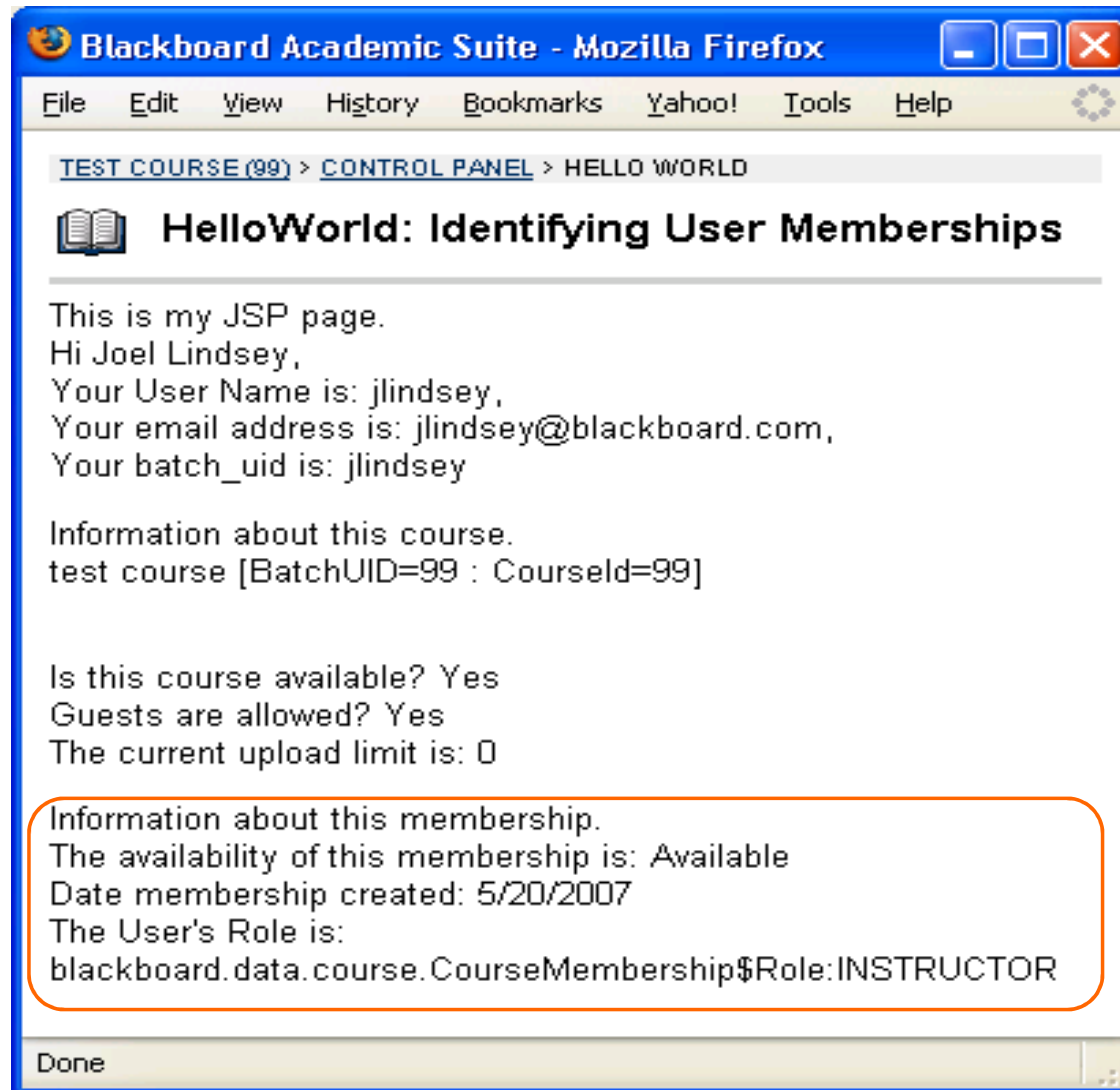
Blackboard Learn handles removal of all associated content or file system components

Invoked via `RequestDispatcher.include()`

Must NOT display any user interface or interact with users

ACCESSING MEMBERSHIPS (CONT'D)

Result



DISPLAYING TABLES

- Modify the Building Block to show all memberships (not just the current user's)
- Use `crsMembershipLoader`'s `loadByCourseId()` method
 - Returns all memberships for the input course ID.
- Display membership list in a table
- Format the role name to fit in the table.

DISPLAYING TABLES (CONT'D)

Add a function to format role name

<%!

```
static public String getRoleString(Object role) {  
    // return a User Friendly String for the type Role passed in  
    String uRole = "";  
    // get role based on coursemembershipRole  
    (CourseMembership.Role)  
    if( (CourseMembership.Role)role ==  
        CourseMembership.Role.COURSE_BUILDER ) {  
        uRole="Course Builder";  
    } else if ( (CourseMembership.Role)role ==  
        CourseMembership.Role.DEFAULT) {  
        uRole="Student(Default)";  
    } else if ( (CourseMembership.Role)role ==  
        CourseMembership.Role.GRADER) {  
        uRole="Grader";  
    } ...  
}
```

Will be
returned
at end

DISPLAYING TABLES (CONT'D)

Use the new function

The User's Role is: `<%=crsMembershipRoleStr%>` `
`

`<!-- return User Friendly Role -->`

Your UI friendly Course Membership Role is: "`<%= getRoleString(crsMembershipRole) %>`"

Load the memberships into a list and sort it.

```
<%  
BbList allMembershipsList = crsMembershipLoader.loadByCourseId( courseId, null,  
true );  
Collections.sort(allMembershipsList, cmSortByFamilyName);  
%>
```

Heavyweight –
causes it to load
user objects also

Comparator

DISPLAYING TABLES (CONT'D)

Declare cmSortByFamilyName

```
Comparator cmSortByFamilyName = new Comparator() {  
    public int compare(Object o1, Object o2) {  
        CourseMembership cm1 = (CourseMembership) o1;  
        CourseMembership cm2 = (CourseMembership) o2;  
        String s1 = (String)((User)cm1.getUser()).getFamilyName();  
        String s2 = (String)((User)cm2.getUser()).getFamilyName();  
        int compare = s1.toLowerCase().compareTo(s2.toLowerCase());  
        if( compare == 0 ) {  
            s1 = (String)((User)cm1.getUser()).getGivenName();  
            s2 = (String)((User)cm2.getUser()).getGivenName();  
            compare = s1.toLowerCase().compareTo(s2.toLowerCase());  
        }  
        return compare;  
    }  
};
```

DISPLAYING TABLES (CONT'D)

Display the list

```
<bbUI:list
```

```
collection="<%= allMembershipsList %>"
```

```
objectId="cm"
```

```
className="CourseMembership">
```

```
<bbUI:listElement
```

```
label="Name"
```

```
name="Name"
```

```
comparator="<%= cmSortByFamilyName %>">
```

```
<%= cm.getUser().getGivenName() %> <%=  
cm.getUser().getFamilyName()  
%>
```

```
</bbUI:listElement>
```

One for each
column

Name of
current
row

Actual
column
value

DISPLAYING TABLES (CONT'D)

Add remaining columns

```
<bbUI:listElement
  label="Username"
  name="Username"
  comparator="<%= cmSortByUsername %>"
  <%= cm.getUser().getUserName() %>
</bbUI:listElement>
<bbUI:listElement
  label="Email"
  name="Email"
  comparator="<%= cmSortByEmailAddress %>"
  <%= cm.getUser().getEmailAddress() %>
</bbUI:listElement>
```

Another comparator – in case user want to sort by this column

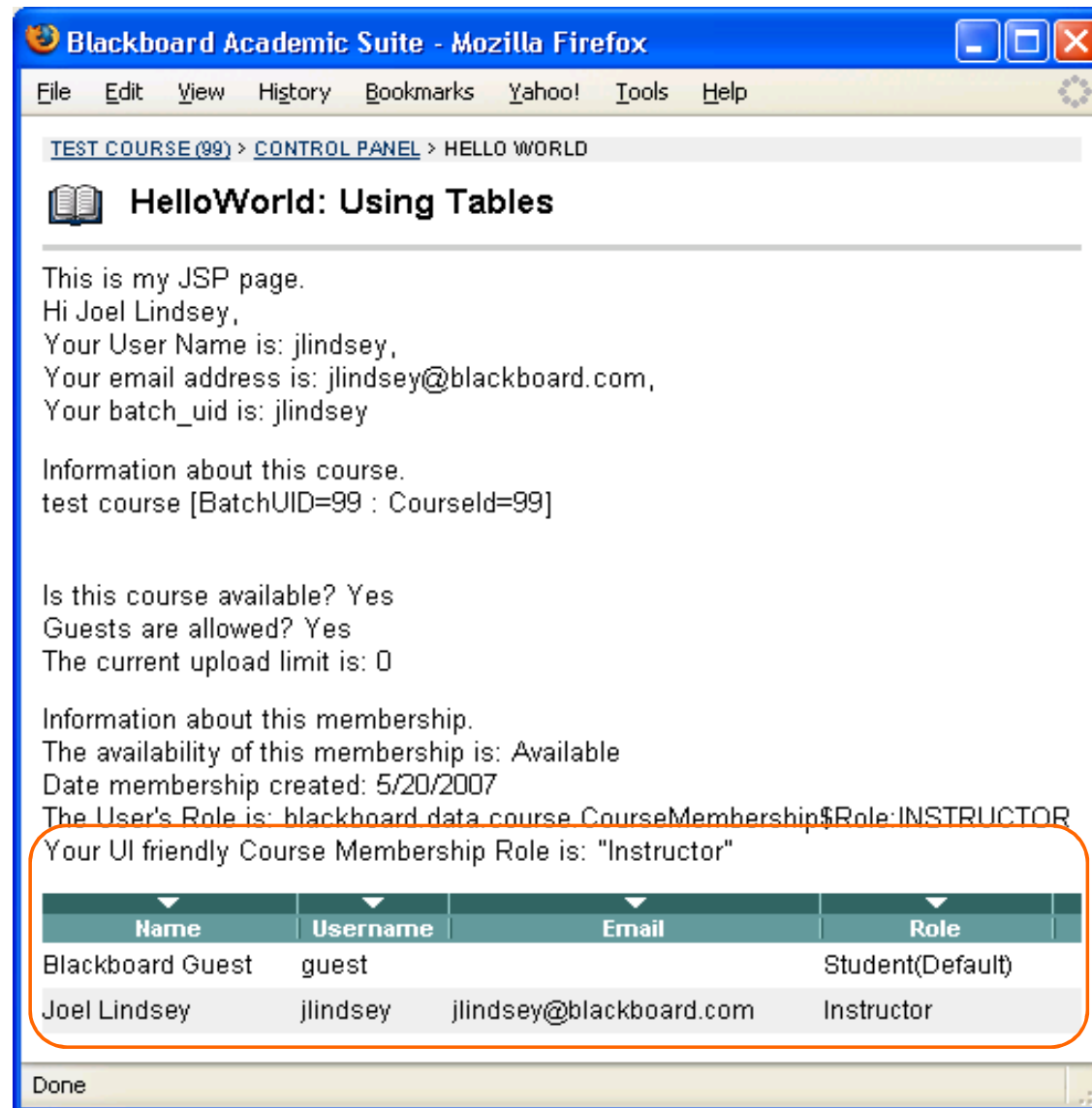
DISPLAYING TABLES(CONT'D)

Add remaining columns (cont'd)

```
<bbUI:listElement
  label="Role"
  name="Role"
  comparator="<%= cmSortByRole %>">
  <%= getRoleString( cm.getRole() ) %>
</bbUI:listElement>
</bbUI:list>
</p>
</bbUI:docTemplate>
```

DISPLAYING TABLES(CONT'D)

Results



Blackboard Academic Suite - Mozilla Firefox

File Edit View History Bookmarks Yahoo! Tools Help

TEST COURSE(99) > CONTROL PANEL > HELLO WORLD

HelloWorld: Using Tables

This is my JSP page.
Hi Joel Lindsey,
Your User Name is: jlindsey,
Your email address is: jlindsey@blackboard.com,
Your batch_uid is: jlindsey

Information about this course.
test course [BatchUID=99 : CourseId=99]

Is this course available? Yes
Guests are allowed? Yes
The current upload limit is: 0

Information about this membership.
The availability of this membership is: Available
Date membership created: 5/20/2007
The User's Role is: blackboard data course CourseMembership\$Role: INSTRUCTOR
Your UI friendly Course Membership Role is: "Instructor"

Name	Username	Email	Role
Blackboard Guest	guest		Student(Default)
Joel Lindsey	jlindsey	jlindsey@blackboard.com	Instructor

Done

DEBUGGING

Create a batch file as follows (assumes windows):

```
net stop Bb-Tomcat
```

```
set JPDA_TRANSPORT=dt_socket
```

```
set JPDA_ADDRESS=2222
```

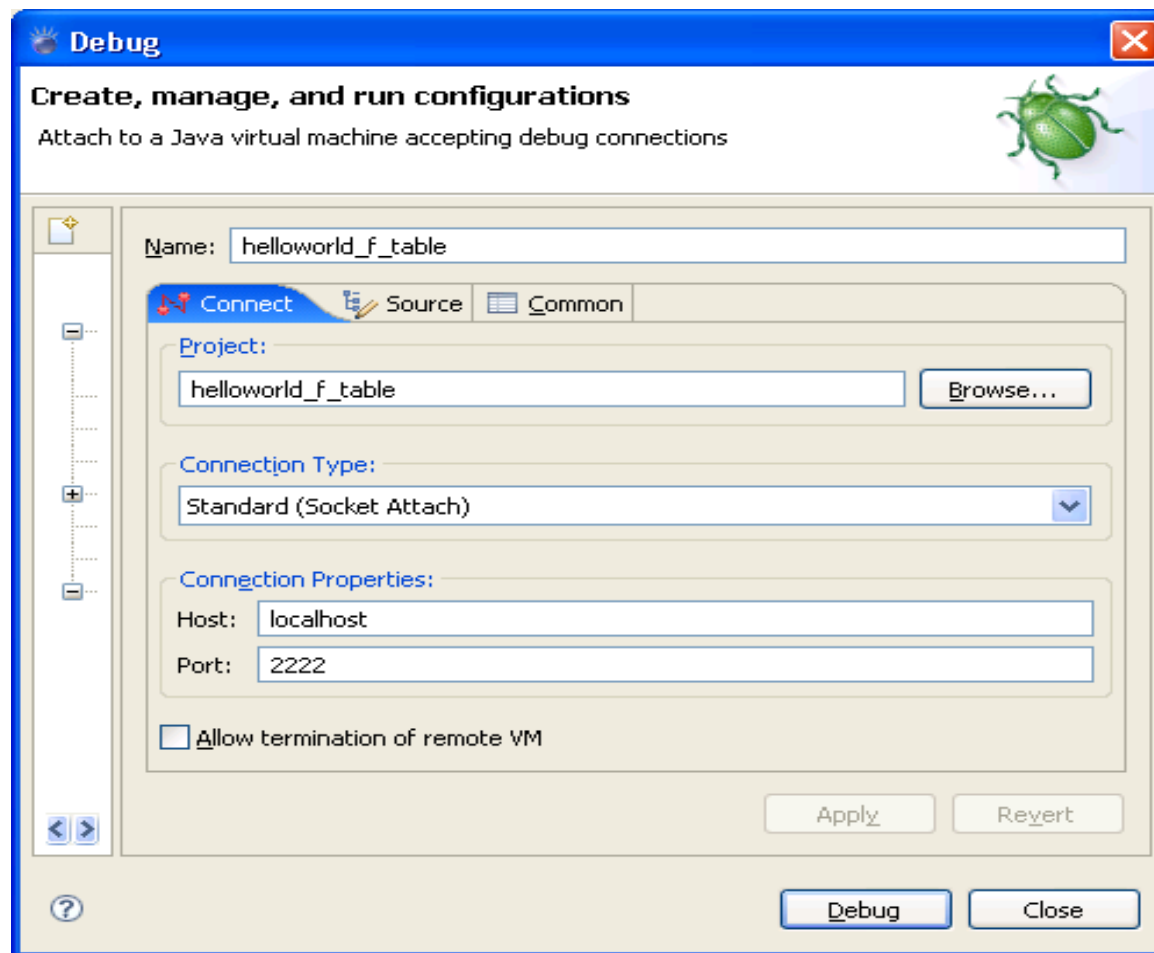
```
c:\blackboard\apps\tomcat\bin\blackboard-tomcat
```

```
jpda start
```

And execute it

DEBUGGING (CONT'D)

In eclipse set up a debug configuration:



RESOURCES

Blackboard Building Blocks Catalog

[**http://www.blackboard.com/extend/b2/**](http://www.blackboard.com/extend/b2/)

Documentation (pdf guides) and APIs (html)

[**http://behind.blackboard.com**](http://behind.blackboard.com) (Reference Center)

Software Development Kit (SDK)

[**http://www.blackboard.com/extend/dev/SDK.htm**](http://www.blackboard.com/extend/dev/SDK.htm)

