

# **Host Access Transformation Services (HATS) V5 Development Lab**

By Michael Sandberg

**IBM eServer Solutions Enablement**

June 2004

## Table of Contents

<b><i>What this exercise is about.....</i></b>	<b><i>1</i></b>
<b><i>Lab Requirements.....</i></b>	<b><i>1</i></b>
<b><i>What you should be able to do.....</i></b>	<b><i>1</i></b>
<b><i>Introduction.....</i></b>	<b><i>1</i></b>
Tips .....	2
<b><i>Exercise Instructions: .....</i></b>	<b><i>3</i></b>
Part 1: Create a Host Access Transformation Services Application .....	3
Success .....	8
Part 2: Preview your HATS project using Run On Server.....	9
Success .....	13
Part 3: Modify the Template .....	14
Success .....	19
Part 4: Working with the Default Rendering.....	20
Success .....	29
Part 6: Customizing a Screen .....	30
Success .....	36
Part 7: Working with Macros and Integration Objects .....	37
Success .....	44
<b><i>Conclusion .....</i></b>	<b><i>45</i></b>
<b><i>Trademarks and Disclaimer .....</i></b>	<b><i>46</i></b>

## **What this exercise is about**

The objective of this lab is to provide you with an understanding on the basics of creating and previewing a Host Access Transformation Services (HATS) application. This will be done through the use of the WebSphere® Development Studio Client for iSeries with the HATS plug-in and the integrated WebSphere Application Server Test environment.

## **Lab Requirements**

The list of systems and software required for the student to complete the lab are:

- iSeries™ servers and PC
- WebSphere Development Studio Client v5.1 on PC
- HATS Studio v5 installed into WebSphere Development Studio Client on PC

## **What you should be able to do**

At the end of this lab you should be able to:

- Create a Host Access Transformation Services Application
- Preview your HATS project in the WebSphere Application Server test environment
- Modify the default template
- Modify the default rendering
- Customizing a screen
- Create a macro

## **Introduction**

In this lab, you will create a HATS application in WebSphere Development Studio Client. After creating the HATS application, you will customize and enhance HATS while previewing it in the WebSphere Application Server test environment.

## ***Tips***

Here are a few things to keep in mind as you are completing this lab:

- WebSphere Development Studio Client is a very dynamic development environment capable of a large amount of user customization. Because of this, the screen captures in this lab may not exactly match what you see when you follow the steps of this lab.
- Host Access Transformation Services comes with a wealth of documentation covering everything from getting started to advanced programming in HATS. This documentation can be accessed in WebSphere Development Studio Client through the HELP interface, it can be found on your PC in IBM WebSphere HATS —> Library from the Start menu in the form of PDFs. This information is also available in the “Welcome to HATS” tab in WebSphere Development Studio Client in the “View the library of online information for HATS” section. In addition to the information that is shipped with HATS, there is additional information on the HATS v5 Information Center Web site at:  
<http://publib.boulder.ibm.com/infocenter/hatshelp/index.jsp>
- Throughout this lab, different actions will cause HATS Tip windows to pop up. These windows can provide useful information to the HATS novice. You can select to have a specific tip not show again or chose to not show any tips.
- In the lab instructions, substitute your team number for XX.

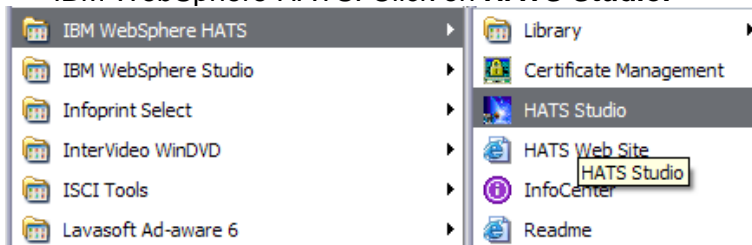
## Exercise Instructions:

### ***Part 1: Create a Host Access Transformation Services Application***

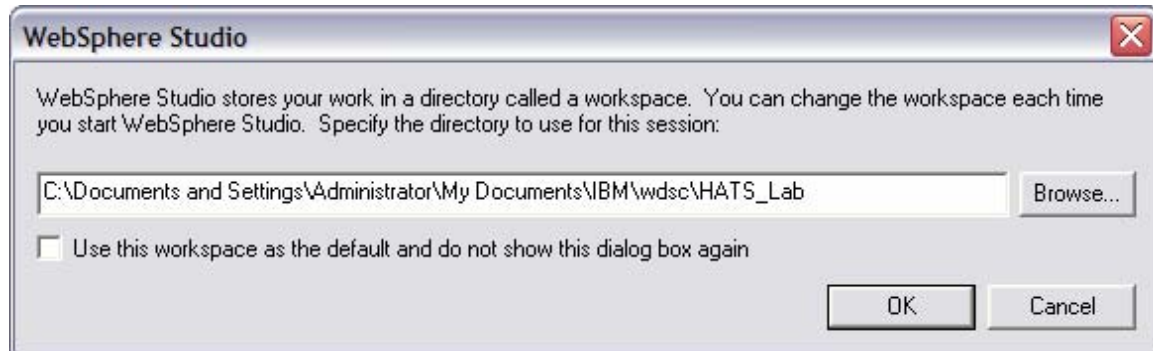
In Part 1, you will start HATS Studio (WebSphere Development Studio Client) and create a basic HATS project. After you have created your HATS project, it will be previewed in the next section.

- \_\_\_ 1. Start WebSphere Development Studio Client (HATS Studio).  
Using Microsoft Windows XP, HATS Studio can be found under Start —>All Programs —>IBM® WebSphere HATS. Click on **HATS Studio**.

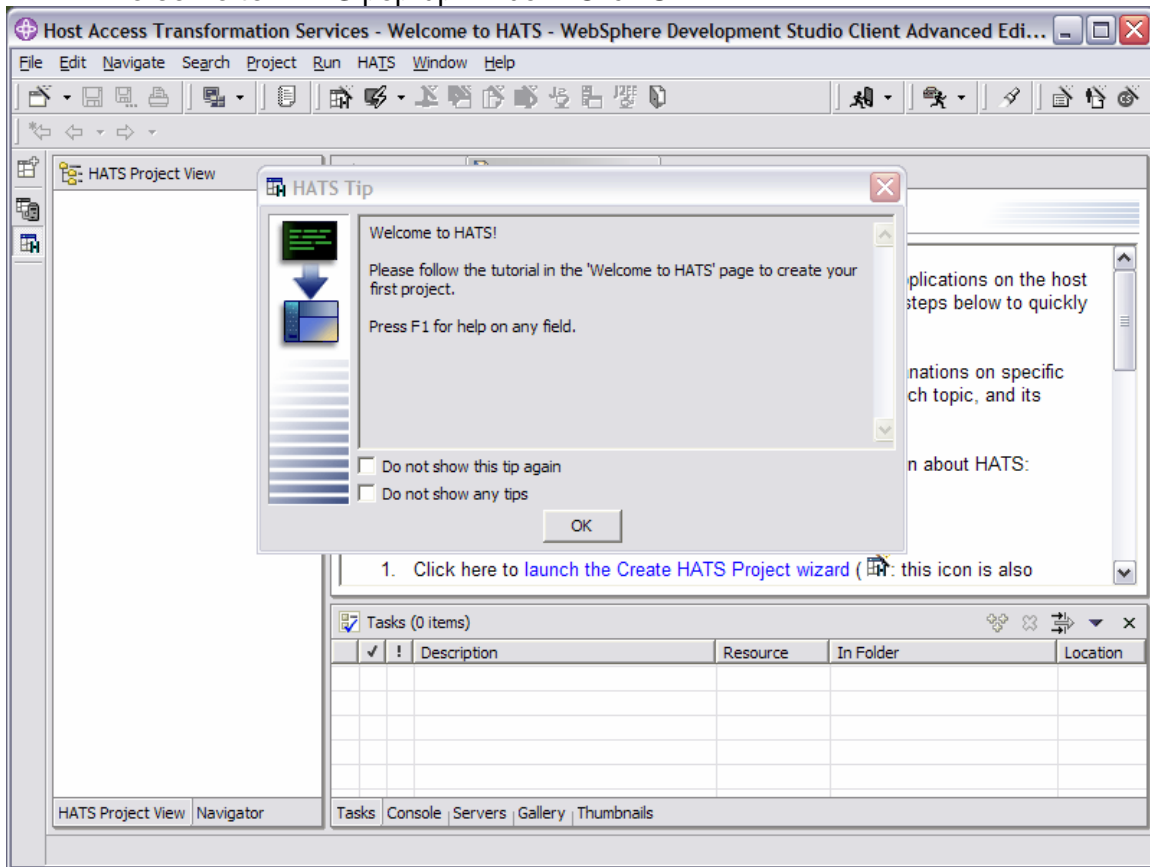
Using Windows 2000, HATS Studio can be found under Start —>Programs —> IBM WebSphere HATS. Click on **HATS Studio**.



- \_\_\_ 2. A WebSphere Studio window will pop up asking what workspace you would like to use. For this lab, we will create a new workspace called HATS\_Lab. Enter C:\Documents and Settings\Administrator\My Documents\IBM\wdsc\HATS\_Lab in the text box. Since the HATS\_Lab workspace does not exist, it will be created. This creation of the workspace makes WebSphere Development Studio Client slow to launch the initial time.



3. After WebSphere Development Studio Client starts, you should be greeted by the **Welcome to HATS** pop-up window. Click **OK**.



4. Click on **launch the Create HATS Project wizard**; this will open the **Create a Project** window. In the Name field, enter **TeamXXHATS** (where XX is your team number), the rest of the fields should remain at the default values. Click **Next>**.

**Create a Project**

Create a Project  
Press F1 for help on any field in the wizard.

Name: TeamXXHATS

Description:  
(optional)

☒ Use default location

Location: C:\Documents and Settings\Administrator\My Documents\IBM\wdsc\HATS\_Lab\TeamXXHATS Browse...

☒ Use default Enterprise Application project

Enterprise Application project name: HATS Lab

☐ Create as a Portlet

☒ Add administration support

< Back Next > Finish Cancel

5. In the Host name field, enter **iSeriesHostName** for your iSeries. In the Type field, select **5250** from the drop down box. Select **27 x 132** from the **Screen size:** drop down box. Click **Next>**.

**Create a Project**

**Connection Settings**  
Configure host connection settings for this new project.

You can specify advanced connection values later in the connection editor.

Host name:

Port:

Type:

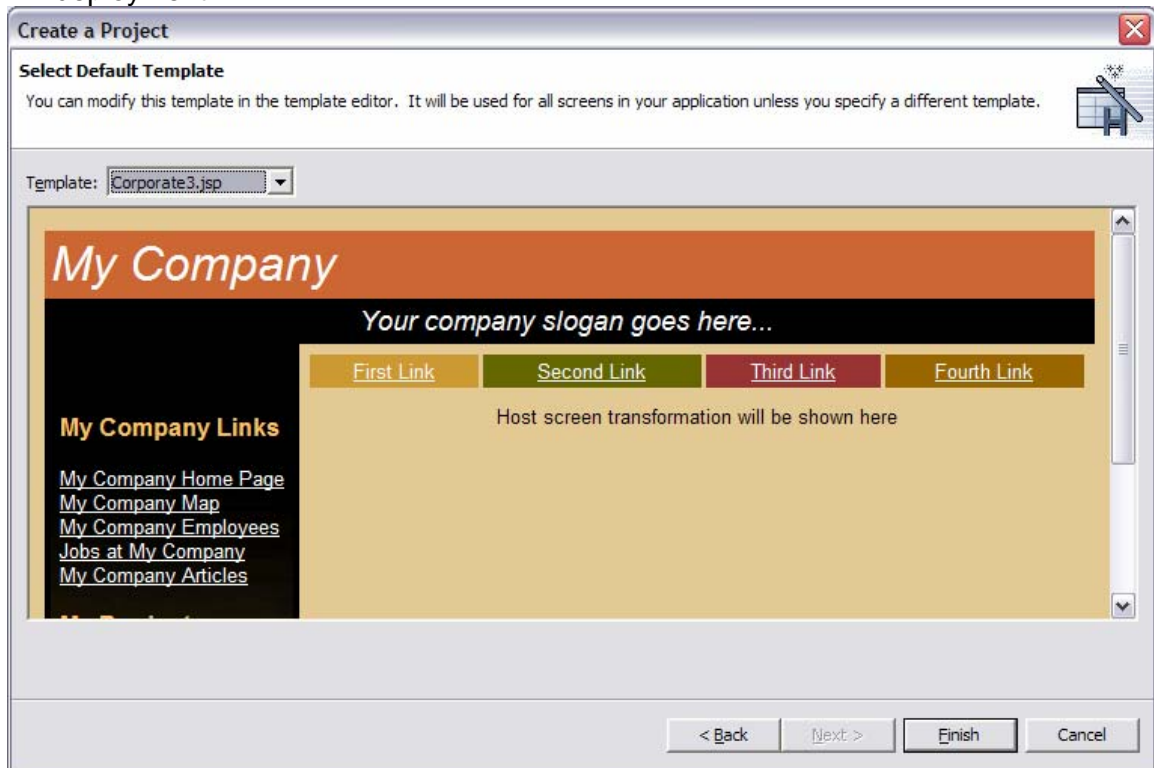
Code page:

Screen size:

< Back   Next >   Finish   Cancel

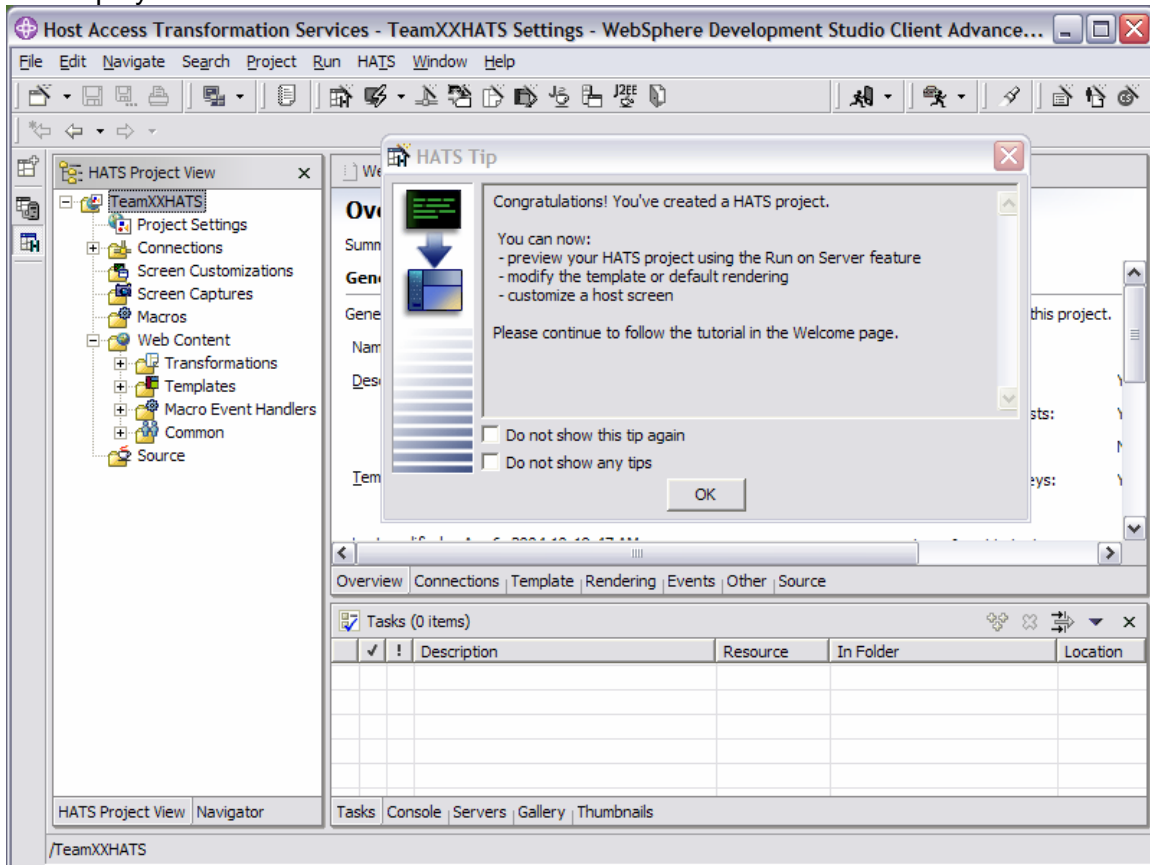


- \_\_\_6. Select the template that best fits your taste by selecting it from the **Template:** drop down box. Click **Finish**. You have just created a basic HATS project ready to run in the WebSphere Application Server test environment or export for deployment.



## Success

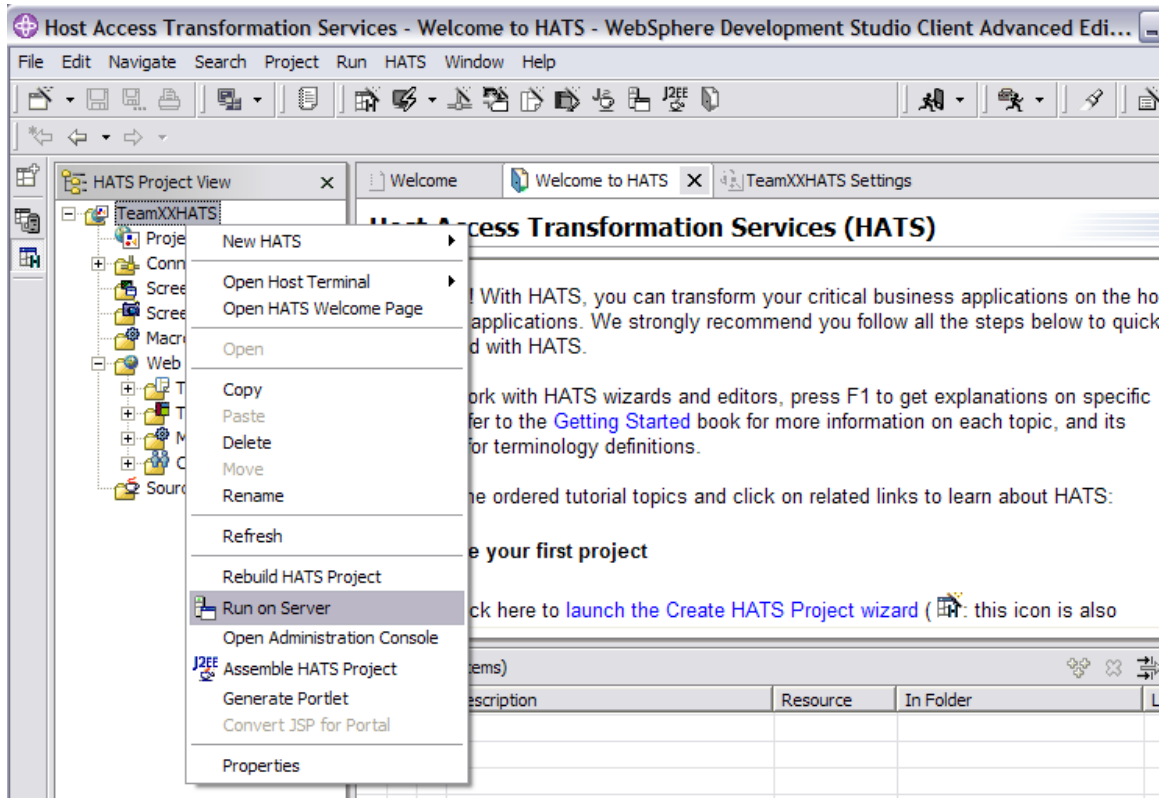
Congratulations! You have just created a HATS project that is ready to enhance, preview, and deploy.



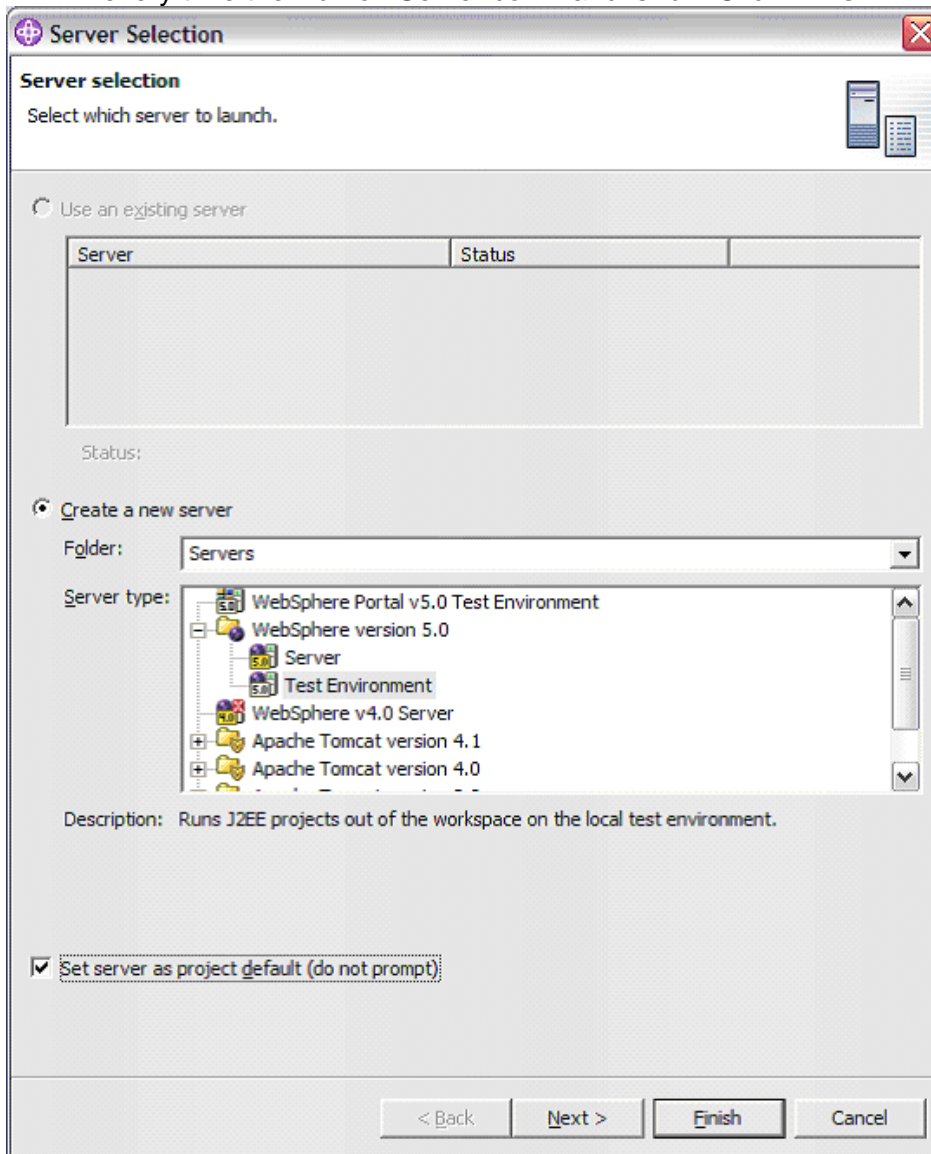
## Part 2: Preview your HATS project using Run On Server

In this section, you will use the Run On Server command to preview your HATS application in the integrated WebSphere Application Server test environment in WebSphere Development Studio Client. After verifying the project functions, you will enhance and customize it.

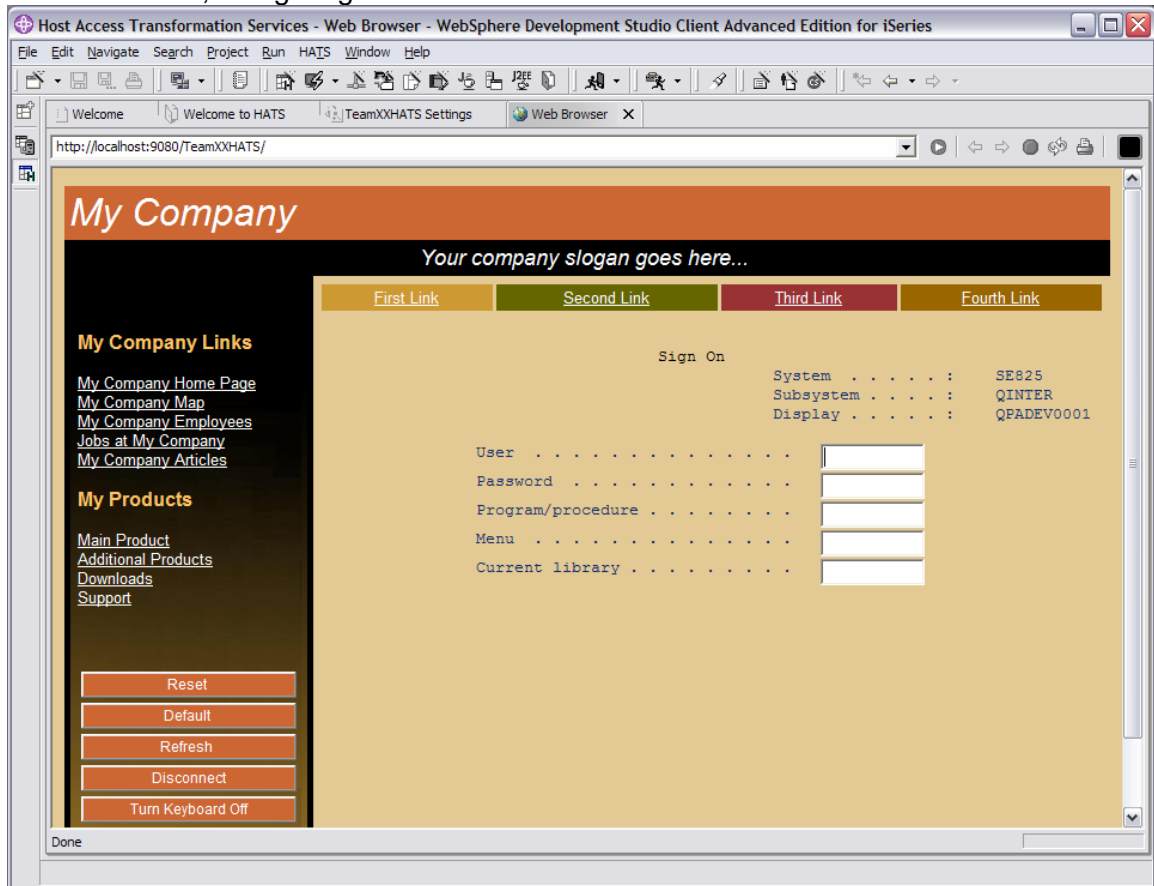
- \_\_\_ 1. Right click on your project in the **HATS Project View** window and select **Run on Server**.



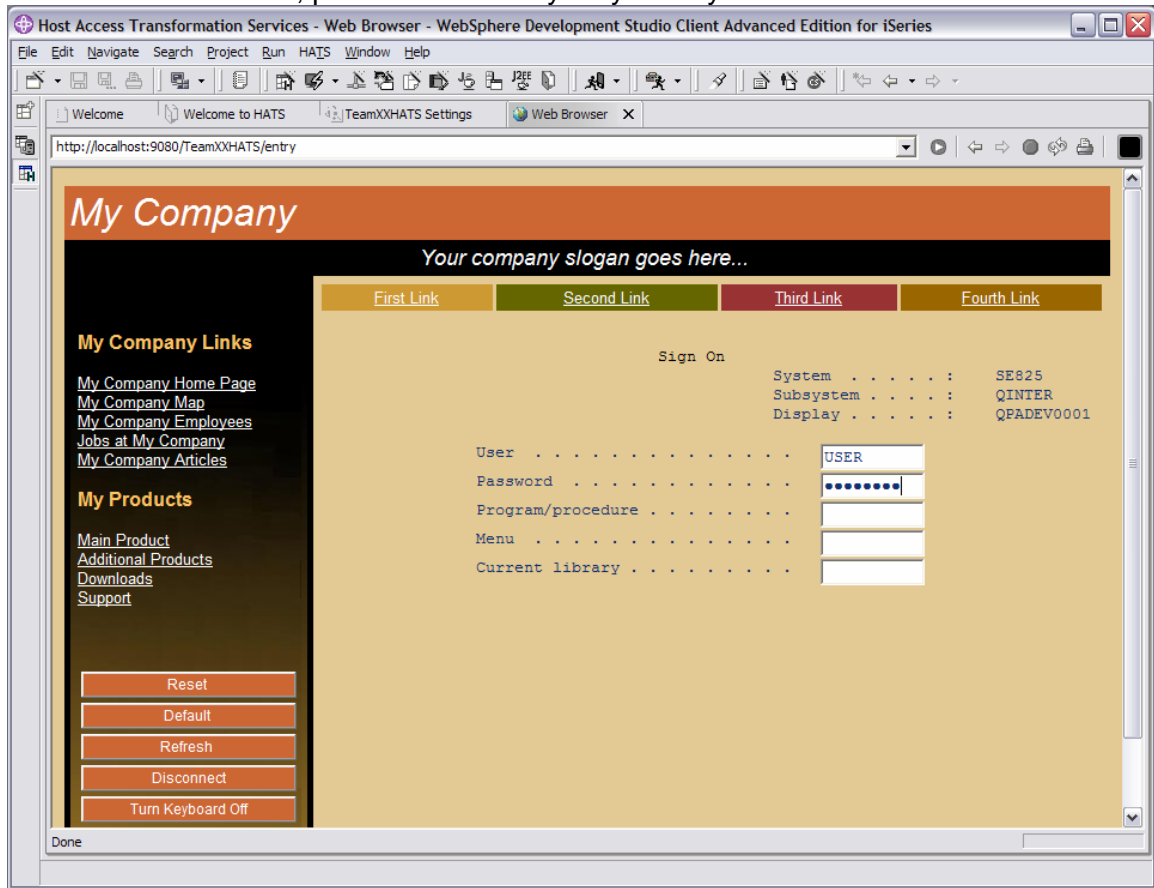
2. Having selected Run on Server, the **Server Selection** screen will open. Click the **Create a new Server** radio button, expand the **WebSphere version 5.0** folder and select **Test Environment**. Click the **Set server as project default (do not prompt)** check box. This will set this WebSphere Application Server to be used every time the Run on Server command is run. Click **Finish**.



- \_\_\_3. Your HATS application should be displayed in the Web Browser tab created by the Run on Server command. You can double click the Web Browser tab to maximize it, doing it again will reduce it to normal size.

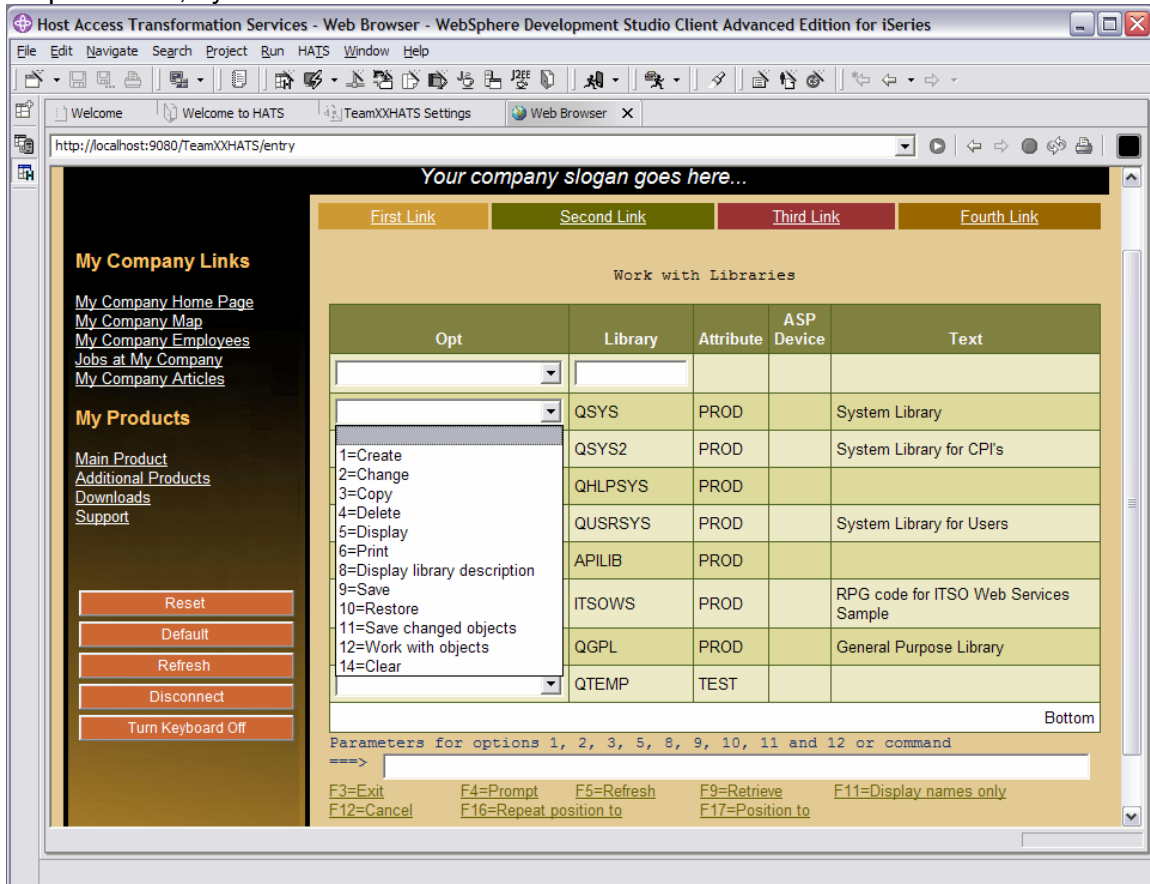


4. Log into your HATS application by filling in the iSeries user ID and password for your iSeries server in the **User** and **Password** text boxes. Once you have entered the values, push the **Enter** key on your keyboard.



## Success

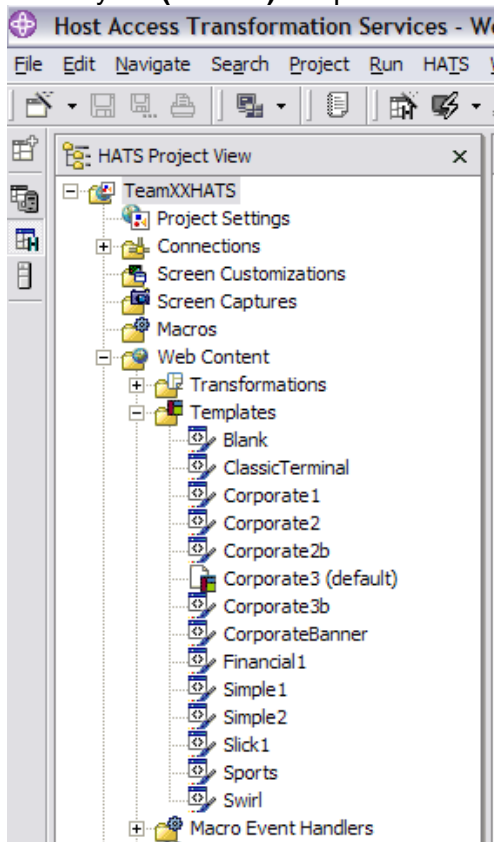
You are now previewing your HATS application running in the test environment. Remember that this is just a default transformation of the 5250 green screen. Take a look around and see what the default transformation gives you. In the screen shot below, the Work with Libraries menu is shown where the options have been transformed into a dropdown list, by default.



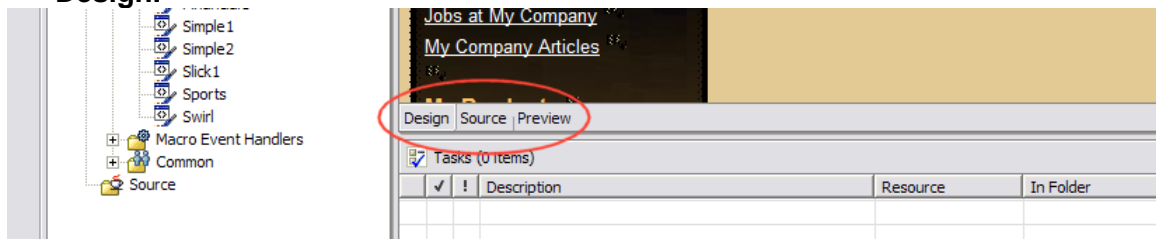
### Part 3: Modify the Template

The template you selected in the Create HATS Project wizard is the default used for each screen. In this section, we will locate the templates and then modify the default template.

1. Expand your HATS project, **TeamXXHATS**, expand the **Web Content** folder, and then expand the **Templates** folder. The template you selected during the creation of your HATS project should have **(default)** to the right of it. In the screenshot below, the default template was **Corporate3(default)**. Double-click on your **(default)** template. This will open the template in the editor.

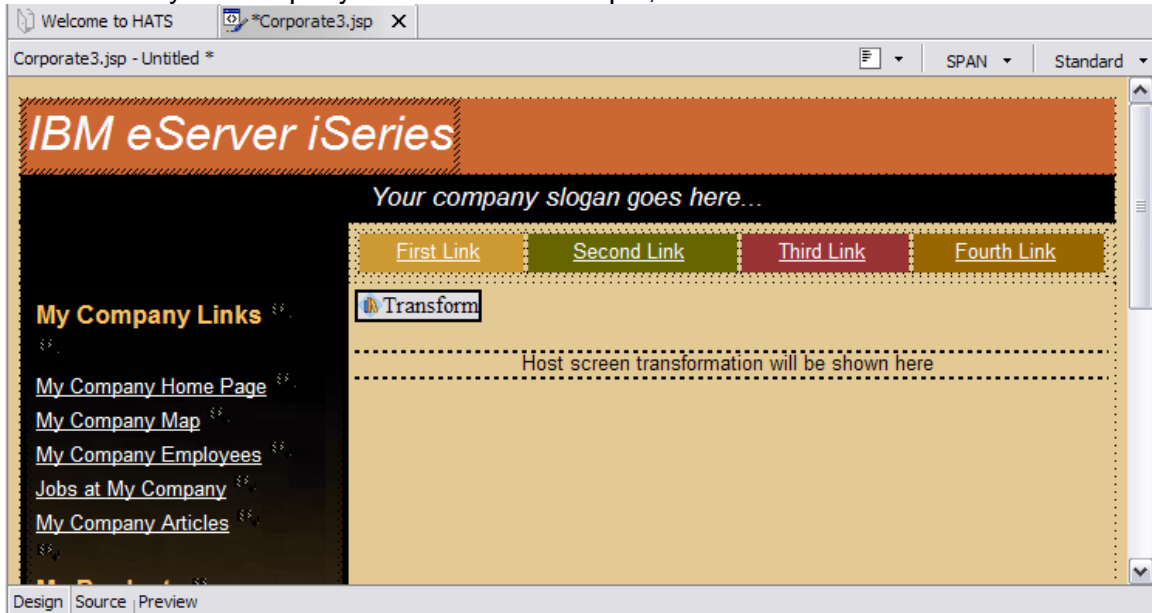


2. You should now see your default template in the editor. Notice that the editor has a **Design**, **Source**, and **Preview** view (see red circle below). Take a little time to familiarize yourself with the different views. Return to the Design view by clicking **Design**.

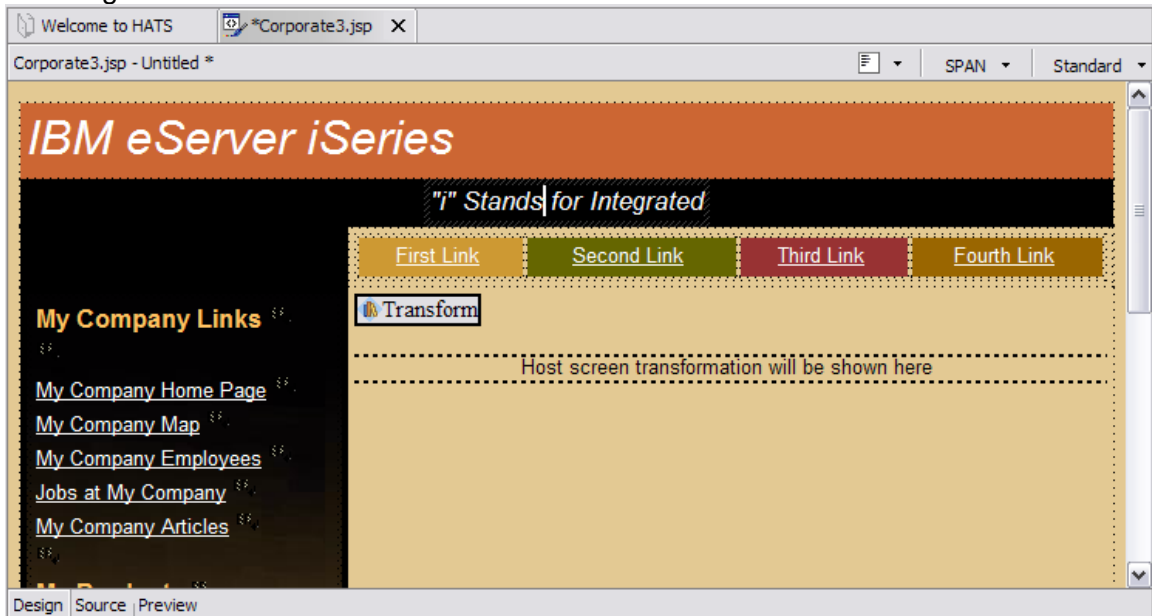




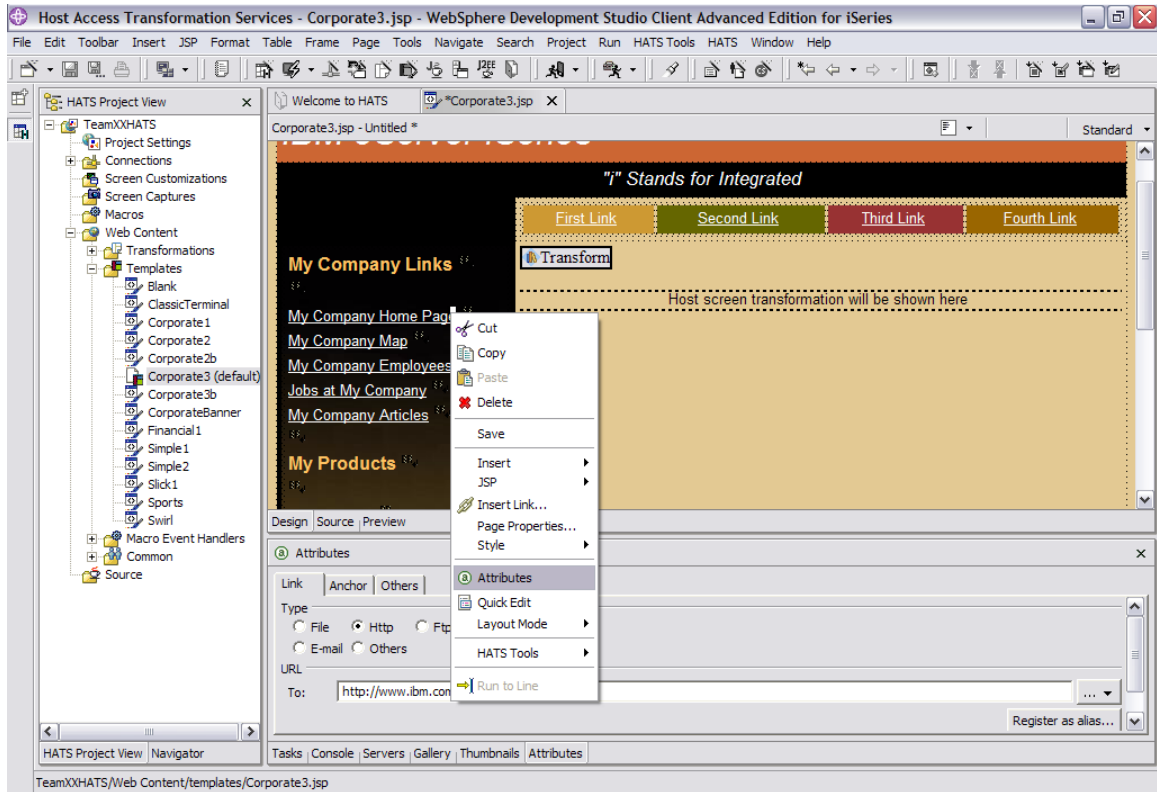
- \_\_\_3. Click anywhere on **My Company** in the editor. Using the keyboard, replace the text with your company name. In this example, we used IBM eServer™ iSeries.



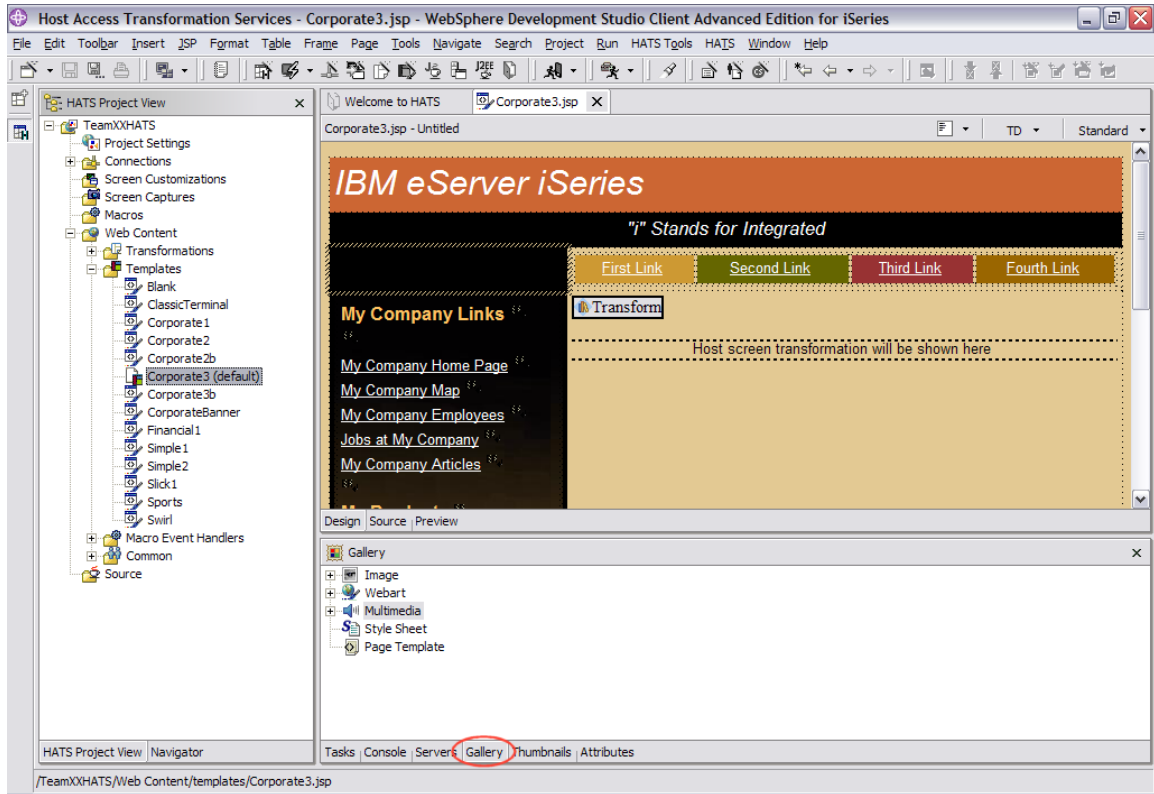
- \_\_\_4. Click anywhere on **Your company slogan goes here...** in the editor. Using the keyboard, replace the text with a slogan of your choice. We used, "I" Stands for Integrated.



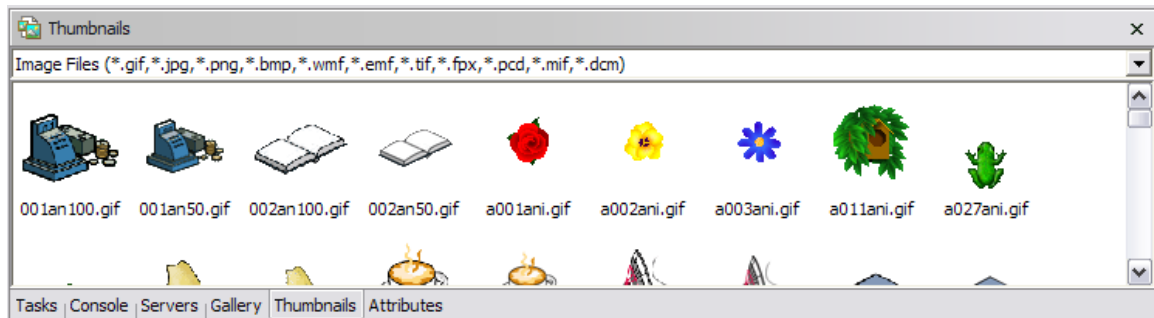
5. So far, we have done a simple text change. On the default template, there are many other attributes for the fields. Right click on **My Company Home Page** and select **Attributes** from the list. The bottom section will then show the Attributes for a HTML Link field. Notice that there is a **Link**, **Anchor**, and **Others** tab. By selecting different fields, you will see the attributes change, depending upon the field selected. Try clicking on different parts of the template and study the attributes.



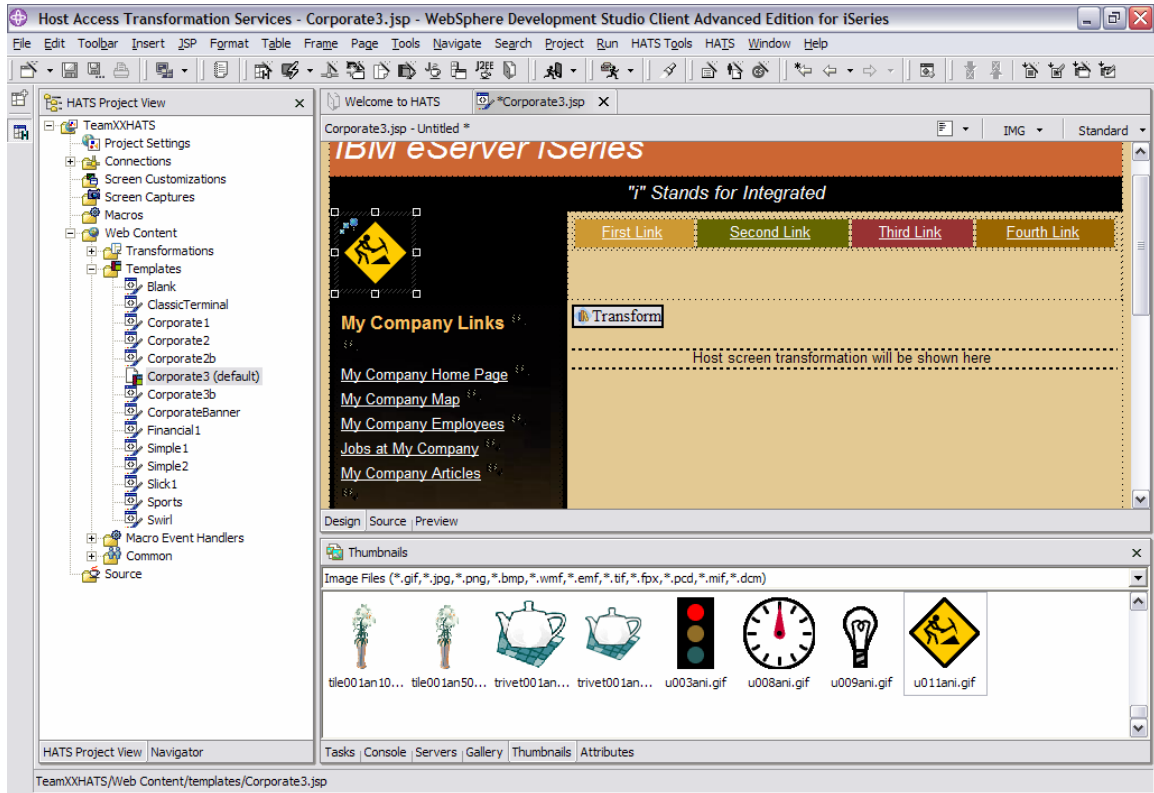
6. Now insert an image into our template. Click on the **Gallery** tab at the bottom of the Web browser.




7. In the **Gallery** window, expand **Multimedia** and click on **Animation**. This will automatically change you to the **Thumbnails** view where you can see the contents of the **Multimedia** folder.




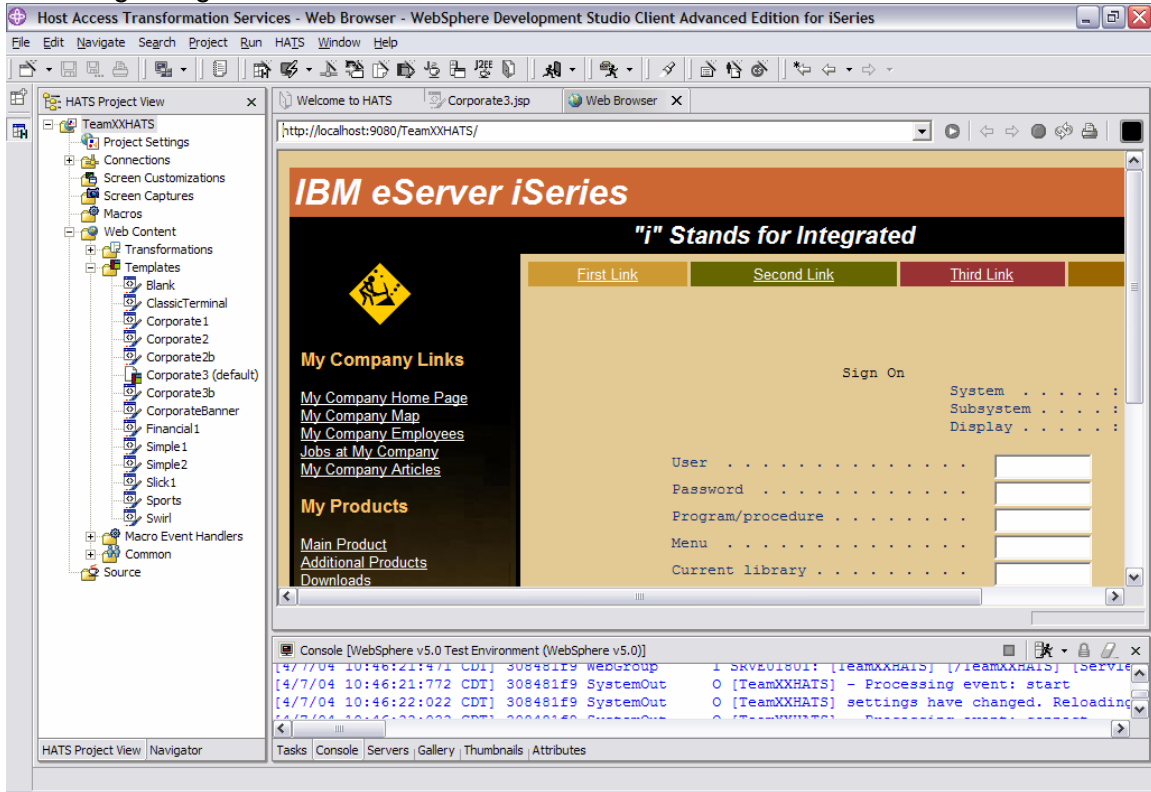
8. Scroll to the bottom of the Thumbnails window and locate u011ani.gif. Click on u011ani.gif and drag it onto your template and drop it above **My Company Links**. You can view how the template will be displayed by clicking on the **Preview** tab at the bottom of the editor window.



9. Click the  button at the top of Web browser, select File —> Save, or type Ctrl+S. This will save the work you have completed so far. Once you have done saved your work, preview your HATS Application. **Right click** on your **TeamXXHATS** project and select **Run on Server** just like in Part 2. Since you created the configuration in Part two and told it not to prompt, your application will run in the Web browser in WebSphere Development Studio Client with just the one click.

## Success

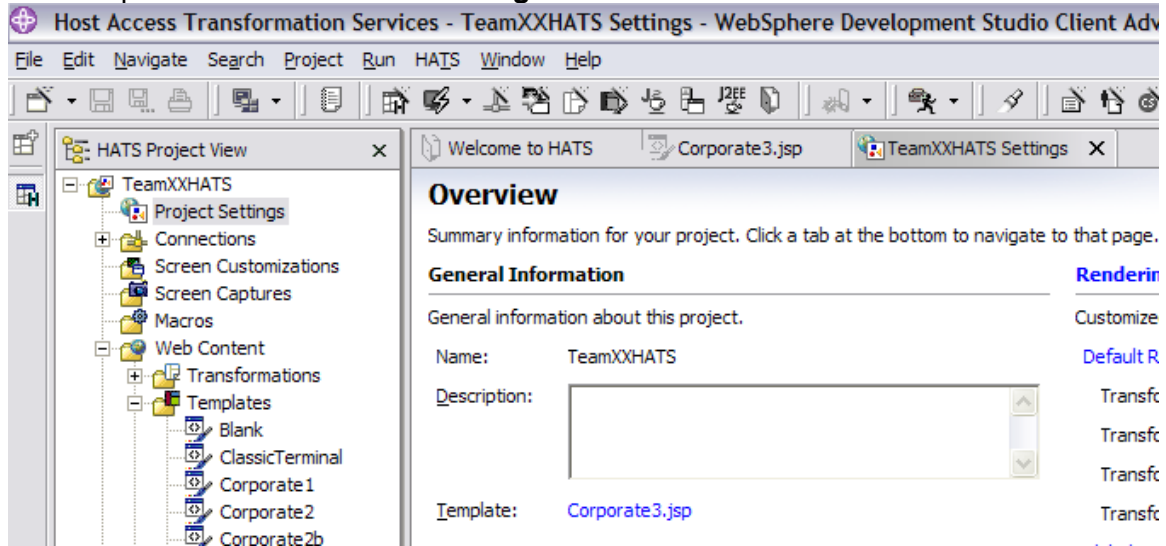
You have now seen how the template can be easily modified and enhanced. New templates can even be created based on an existing Web site using the Create Template wizard (  ), allowing easy use of company style and content. Working with the template is the simplest method of changing the look of your HATS application. Much more powerful and dramatic enhancements are available working with the default rendering and global rules.



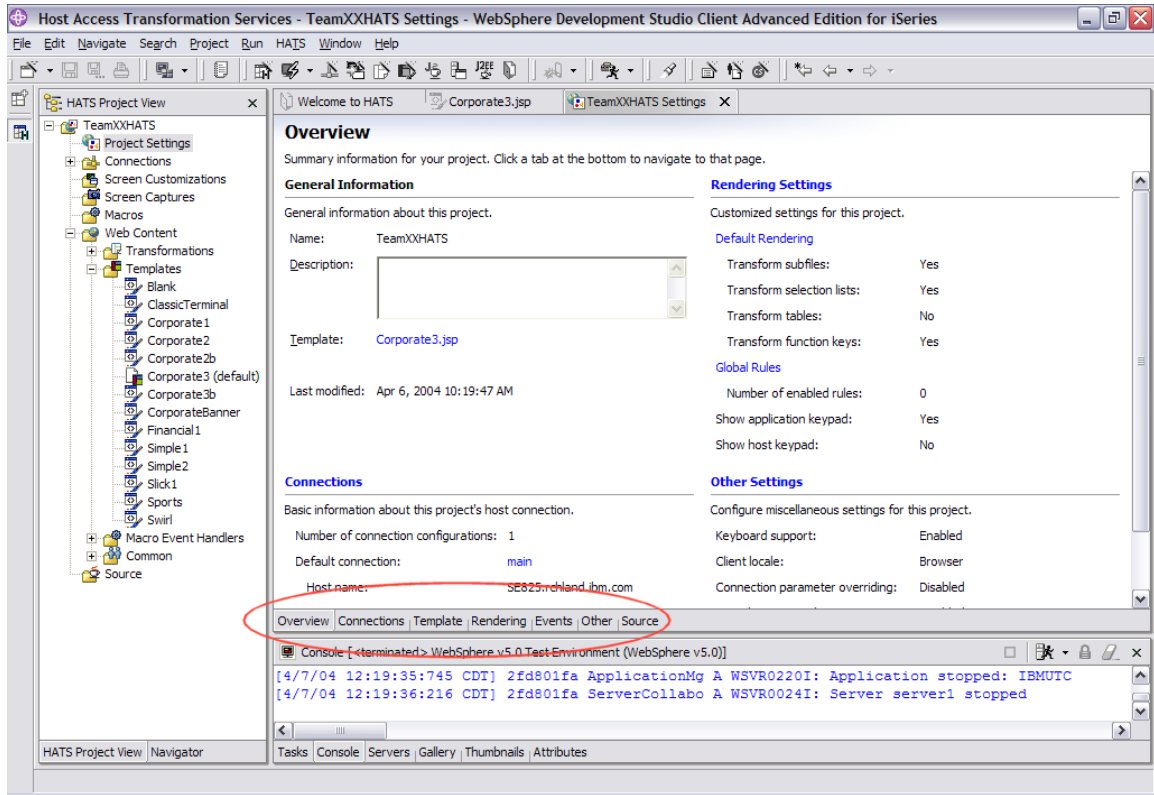
## Part 4: Working with the Default Rendering

The default rendering is an ordered set of conversion pairings, and is used by the default transformation to change host components into HTML widgets. You saw the results in Part 2 when you ran your HATS application in the WebSphere Application Server test environment. In this section, we will view and modify the default renderings.

- \_\_\_1. Expand your **TeamXXHATS** project and double click on **Project Settings**. This will open the **TeamXXHATS Settings** to be viewed and modified.



2. Notice that like other windows in WebSphere Development Studio Client, the **TeamXXHATS Settings** window has a set of tabs in the lower left corner of the window (see the red circle). In this example, the tabs are **Overview**, **Connections**, **Template**, **Rendering**, **Events**, **Other**, and **Source**. Click through the tabs and take some time to review the information available in each tab.



3. To work with the default rendering, click on the **Rendering** tab. The default rendering works with the default transformation to change host components on the screen to widgets on the Web page. Certain host components are mapped to specific widgets for conversion in the default transformation. A default list of mapping items is defined for projects. You can modify the list of items to suit your needs. These items are processed in the order in which they are listed. After a rendering item has been applied to a region of the host screen, that region has been "consumed" and will not be processed again.

Welcome to HATS Corporate3.jsp TeamXXHATS Settings X

### Rendering

Configure default rendering, global rules, text replacement, and default component and widget settings.

- Default Rendering
- Global Rules
- Text Replacement
- Components
- Widgets
- Application Keypad
- Host Keypad

Description	Component	Widget
<input type="checkbox"/> Transform dialogs	Dialog	Dialog
<input checked="" type="checkbox"/> Transform subfiles	Subfile	Subfile
<input checked="" type="checkbox"/> Transform selection lists	Selection list	Link
<input checked="" type="checkbox"/> Transform function keys	Function key	Link
<input type="checkbox"/> Transform field tables	Field table	Table
<input type="checkbox"/> Transform visual tables	Visual table	Table
<input checked="" type="checkbox"/> Transform remaining text and input fields	Field	Field

Add Edit Remove Up Down

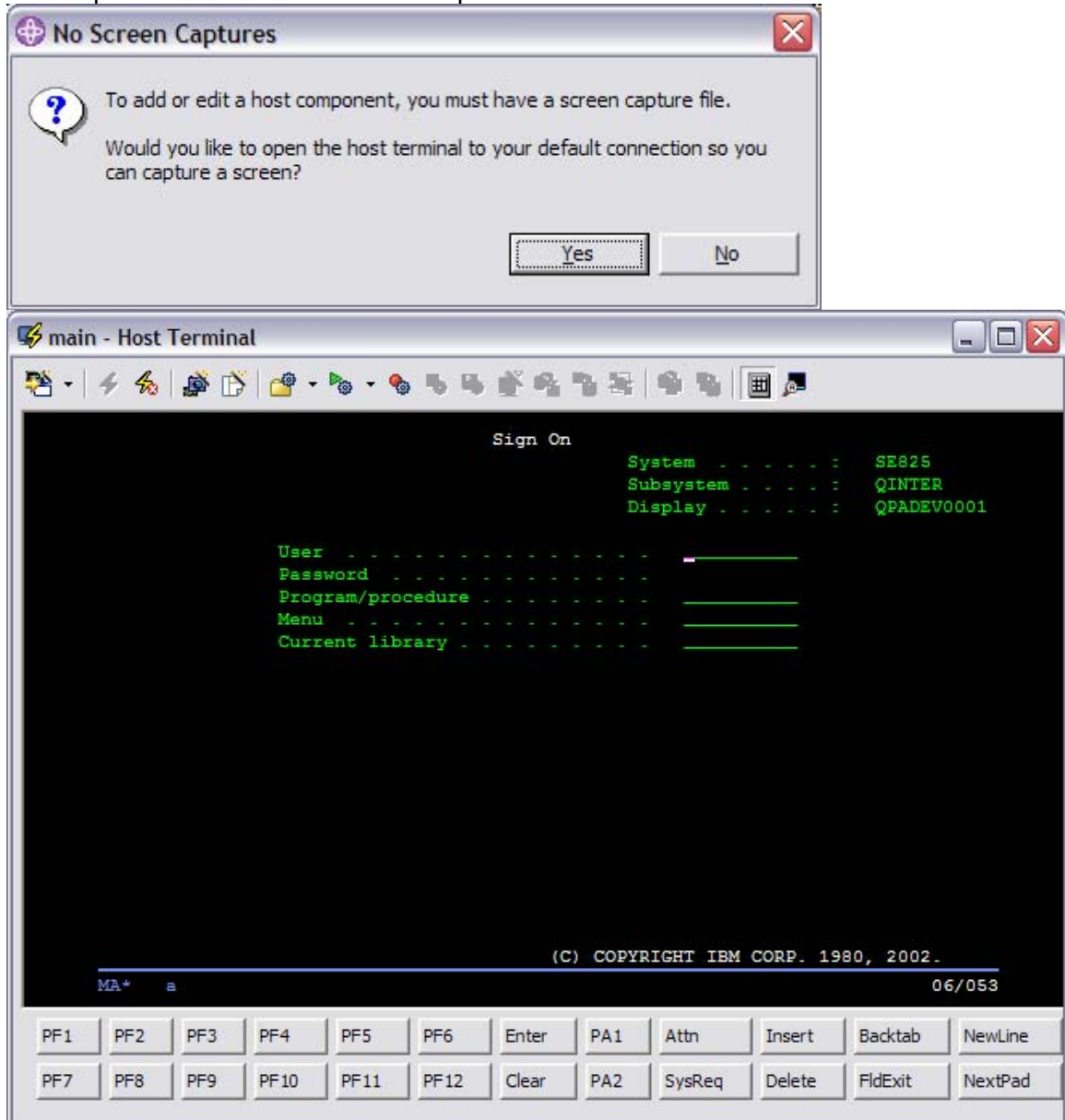
Overview | Connections | Template | **Rendering** | Events | Other | Source



4. We will now change the default rendering for function keys. Select **Transform function keys** by clicking on the box in the Description column.

Description	Component	Widget	
<input type="checkbox"/> Transform dialogs	Dialog	Dialog	Add
<input checked="" type="checkbox"/> Transform subfiles	Subfile	Subfile	Edit
<input checked="" type="checkbox"/> Transform selection lists	Selection list	Link	Remove
<input checked="" type="checkbox"/> Transform function keys	Function key	Link	
<input type="checkbox"/> Transform field tables	Field table	Table	Up
<input type="checkbox"/> Transform visual tables	Visual table	Table	Down
<input checked="" type="checkbox"/> Transform remaining text and input fields	Field	Field	

5. After selecting Transform function keys, click the **Edit** button. A window will pop up stating that you need to have a screen capture to add or edit a host component. Click **Yes** — this will open the **main – Host Terminal** window.



- \_\_\_6. Since we want to work with function key transformation, we need to capture a screen with function keys. Sign on to the iSeries server using your **User Name** and **Password** for the iSeries server.

The screenshot shows a Host Terminal window titled "main - Host Terminal". The terminal displays a "Sign On" screen with the following information:


```
Sign On
System . . . . . : SE825
Subsystem . . . . : QINTER
Display . . . . . : QPADEV0001

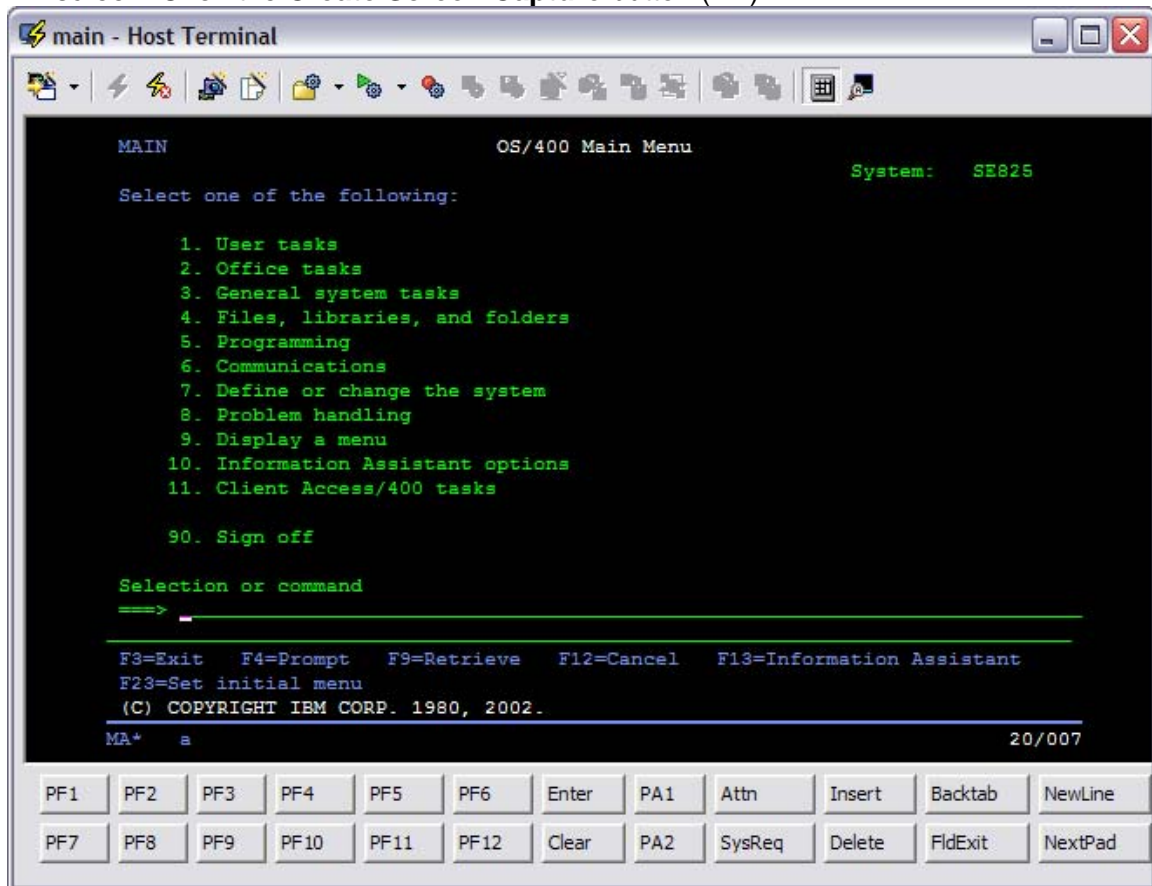
User . . . . . : USER
Password . . . . : 
Program/procedure . . . . : 
Menu . . . . . : 
Current library . . . . : 
```

At the bottom of the terminal, there is a copyright notice: "(C) COPYRIGHT IBM CORP. 1980, 2002." and a date "07/061".

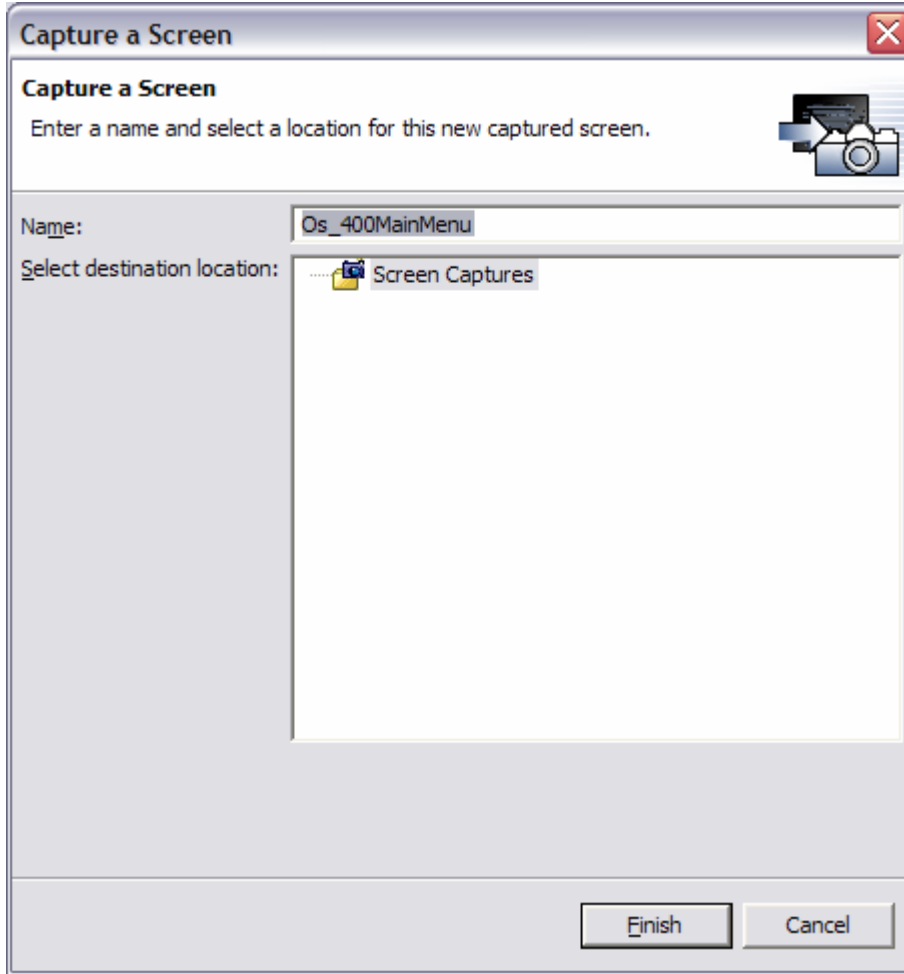
Below the terminal window is a row of function keys:

PF1	PF2	PF3	PF4	PF5	PF6	Enter	PA1	Attn	Insert	Backtab	NewLine
PF7	PF8	PF9	PF10	PF11	PF12	Clear	PA2	SysReq	Delete	FldExit	NextPad

- \_\_\_7. After logging on to the iSeries server, you should be on the OS/400 Main Menu screen. Click the **Create Screen Capture** button ().



- \_\_\_8. Accept the defaults on the **Capture a Screen** window and click finish. Sign off of the iSeries **Host Terminal** session and close the **main – Host Terminal** window.



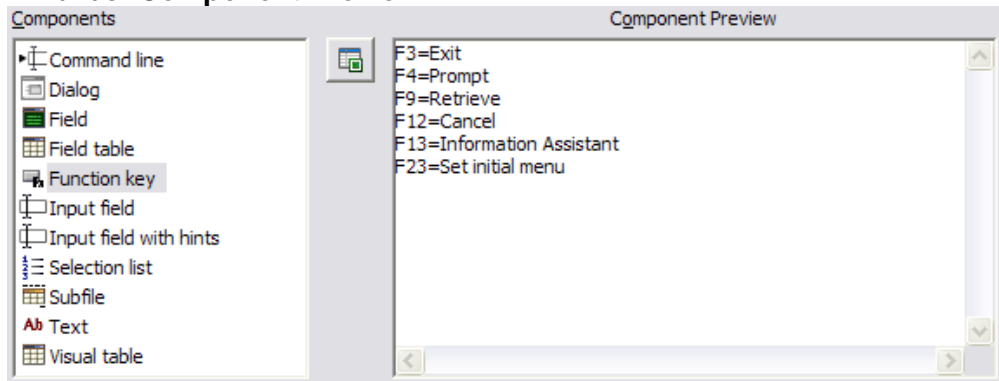
- \_\_\_9. We now can edit the **Transform function keys Default Rendering**. Once again, select the **Transform function keys** and click the **Edit** button.

Description	Component	Widget	
<input type="checkbox"/> Transform dialogs	Dialog	Dialog	<input type="button" value="Add"/>
<input checked="" type="checkbox"/> Transform subfiles	Subfile	Subfile	<input checked="" type="button" value="Edit"/>
<input checked="" type="checkbox"/> Transform selection lists	Selection list	Link	<input type="button" value="Remove"/>
<input checked="" type="checkbox"/> Transform function keys	Function key	Link	
<input type="checkbox"/> Transform field tables	Field table	Table	<input type="button" value="Up"/>
<input type="checkbox"/> Transform visual tables	Visual table	Table	<input type="button" value="Down"/>
<input checked="" type="checkbox"/> Transform remaining text and input fields	Field	Field	

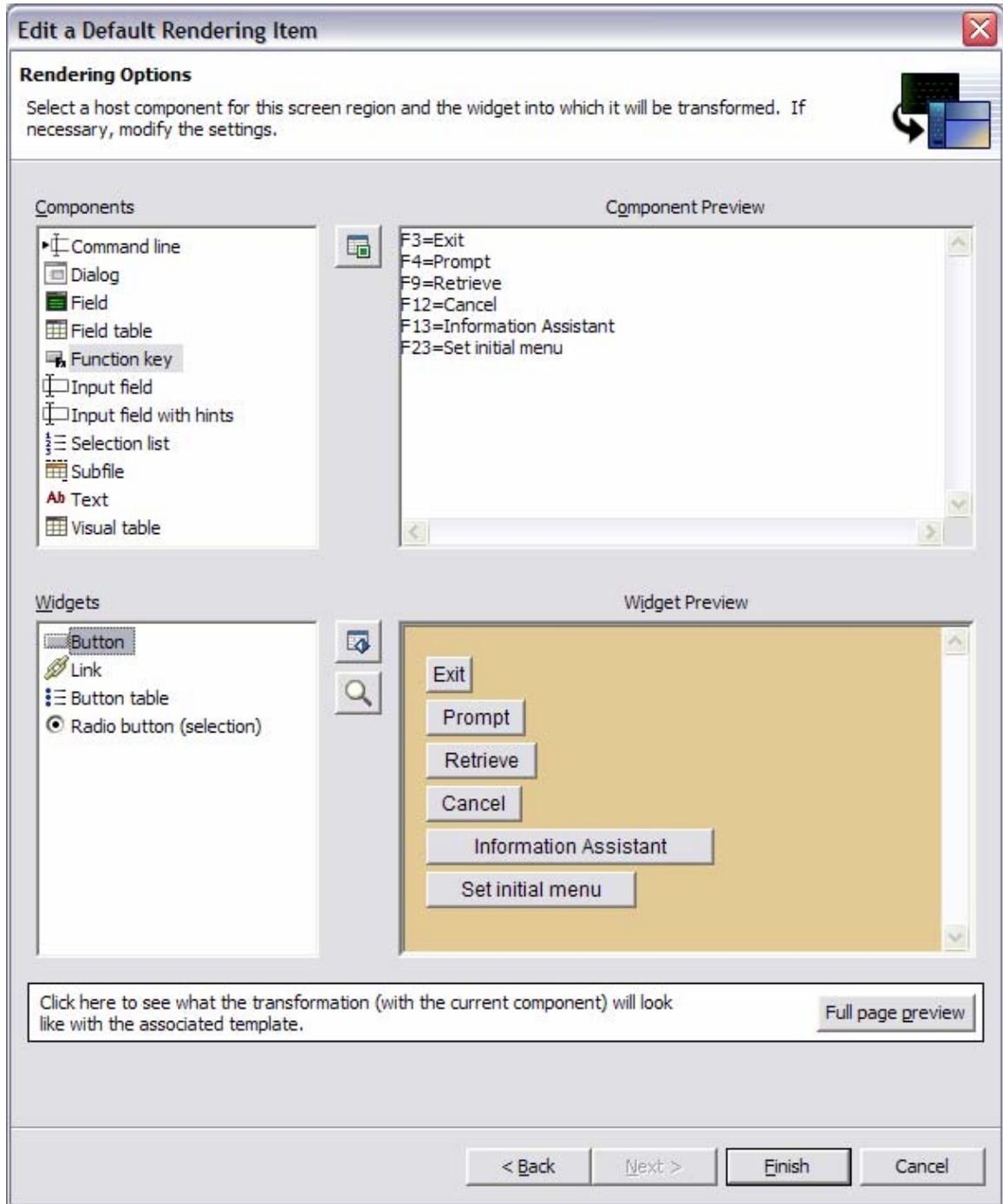
- \_\_\_10. In the **Edit a Default Rendering Item** window, accept the **Default Description** and click **Next**.

- \_\_\_11. In **Select Screen Region**, accept the defaults and click **Next**.

- \_\_\_12. The Rendering Options is where we are able to make the connection between components on the iSeries host screen and widgets that get displayed in the browser. Click on **Function key** under Components. It will then be previewed under **Component Preview**.



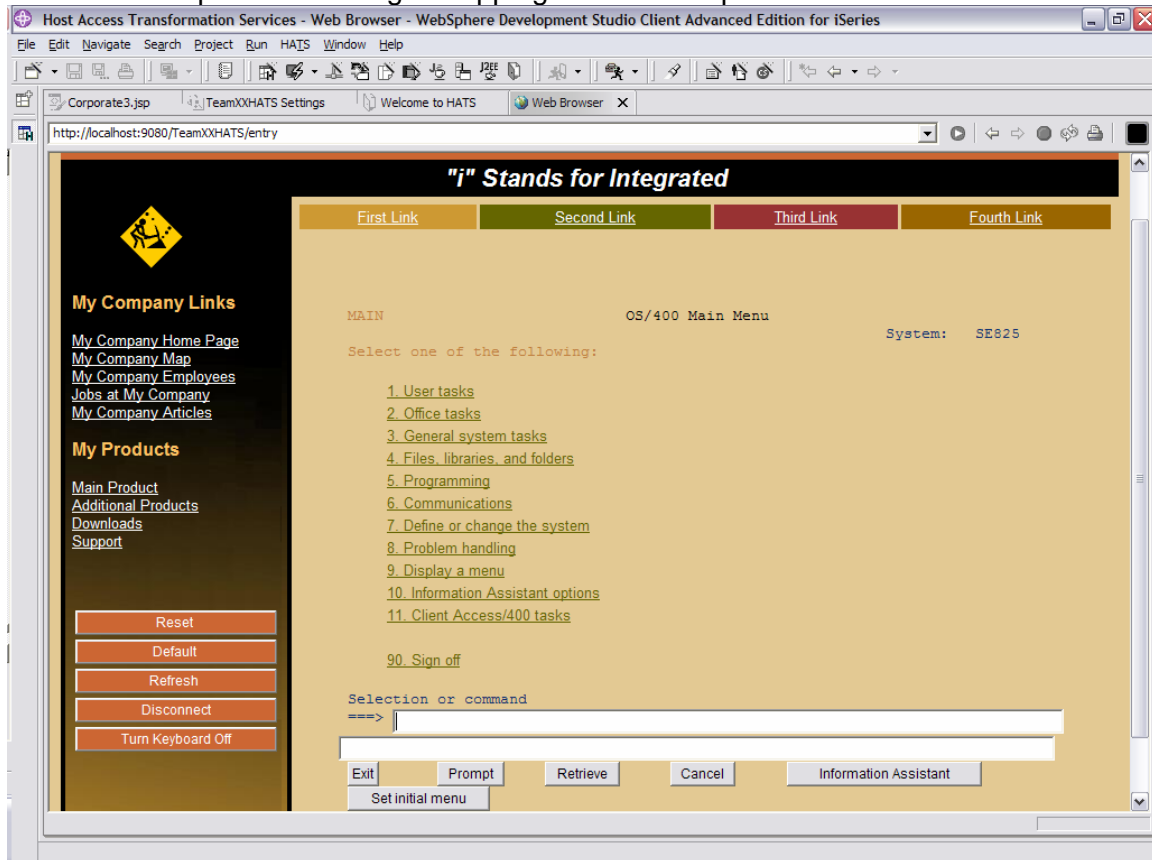
13. Under Widgets, click on **Button**, **Link**, **Button table**, and **Radio button** (selection) one after another. A preview of each will be shown under the **Widget Preview**. Select your preferred method for displaying Function Keys, then click **Finish**.



\_\_\_14. **Save** your project. Then preview it using the **Run on Server** command.

## Success

You have now changed the default rendering for function keys. As you saw as you work through the steps in this section, there are far more options than we used. In addition to default rendering, there are global rules. As default rendering works with the default transformation, global rules enable you to define component to widget mappings for all your transformations. Although the mappings are similar to default rendering, global rules apply only to input field components, and there are no default mapping items defined. When you define global rules, they take precedence over any default rendering items and components to widget mappings defined in specific transformations.



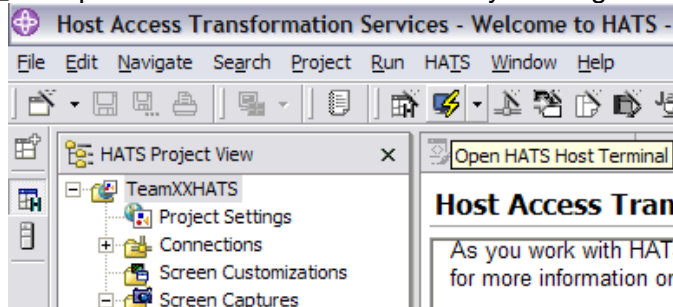
## Part 6: Customizing a Screen

A screen customization has two parts: a set of criteria for matching host screens and a list of actions to be taken when a host screen matches these criteria. The screen customization is at the heart of the HATS application's rules-based processing of host screens. Each screen customization includes a set of criteria that determine which host screens should be applied. When a host screen matches these criteria, the HATS application executes the actions that are defined in the screen customization. These actions can include:

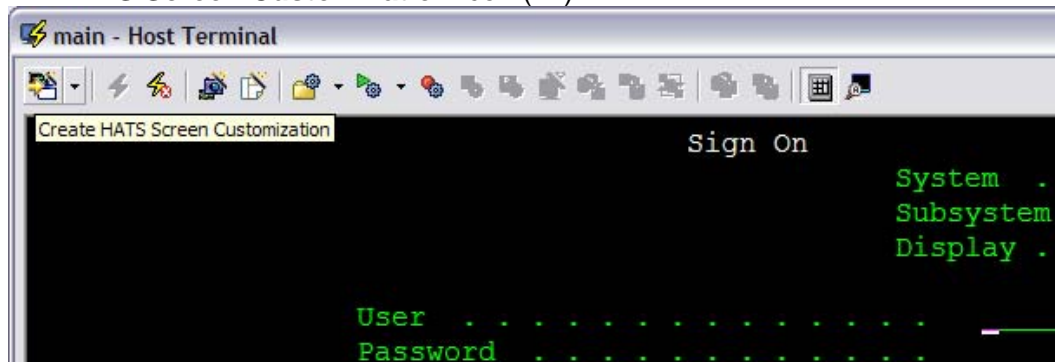
- Applying a transformation to the host screen
- Extracting, setting, or inserting data or the value of a global variable
- Running a macro or macro transaction
- Calling Java methods to execute business logic
- Displaying a specified Web page (URL)
- Sending a key to the host application
- Forwarding control to a URL

In this section of this lab, we are going to customize the Sign On screen. We will do this by creating a set of criteria to recognize the screen and then develop a list of actions to be taken when the Sign On screen is recognized.

1. Open the HATS **host Terminal** by clicking on the icon in the toolbar (⚡)



2. The **main – Host Terminal** window will open to the **Sign On** screen of the iSeries server. (This is the screen we want to customize.) Click on the **Create HATS Screen Customization** icon (🔧).





3. Accept the defaults on **New Screen Customization** and click **Next**.

The screenshot shows a Windows-style dialog box titled "Create a Screen Customization". Inside, there's a section titled "New Screen Customization" with a subtitle "Press F1 for help on any field in the wizard." and a small icon of a screen with a pencil. Below this, there are three main input areas: 1. "Name:" with a text box containing "SignOn". 2. "Description: (optional)" with a large empty text area. 3. "Enter or select the destination:" with a text box showing the path "/TeamXXHATS/Web Content/WEB-INF/profiles/events/screencustomizations" and a file explorer view below it. The file explorer shows a folder named "TeamXXHATS" which contains a sub-folder named "Screen Customizations". At the bottom right, there are four buttons: "< Back", "Next >" (which is highlighted with a black border), "Finish", and "Cancel".

Create a Screen Customization

**New Screen Customization**  
Press F1 for help on any field in the wizard.

Name: SignOn

Description: (optional)

Enter or select the destination: /TeamXXHATS/Web Content/WEB-INF/profiles/events/screencustomizations

TeamXXHATS  
Screen Customizations

< Back Next > Finish Cancel

4. The next step in the wizard is **Select Screen Recognition Criteria**. We will identify our **Sign On** screen by the string value **Sign On**. Under **Within a rectangular region**, enter the row and column values. Enter these values in their corresponding fields:  
**Starting row 1 Starting column 36**  
**Ending row 1 Ending column 42.**  
 Click **Finish**.

**Create a Screen Customization**


**Select Screen Recognition Criteria**

You can make the criteria more or less specific in order to match one or many screens.  
 You can refine the criteria (more strings, AND/OR/NOT logic) later in the screen customization editor.

Select a screen:

☒ Use the host terminal screen  
☐ Use a previously captured screen

Select a screen:



☐ Total number of fields:

☐ Number of input fields:

☐ Cursor position: Row  Column

☒ Strings:

String position

☐ Anywhere on the screen

☐ At a specified position:

Row  Column

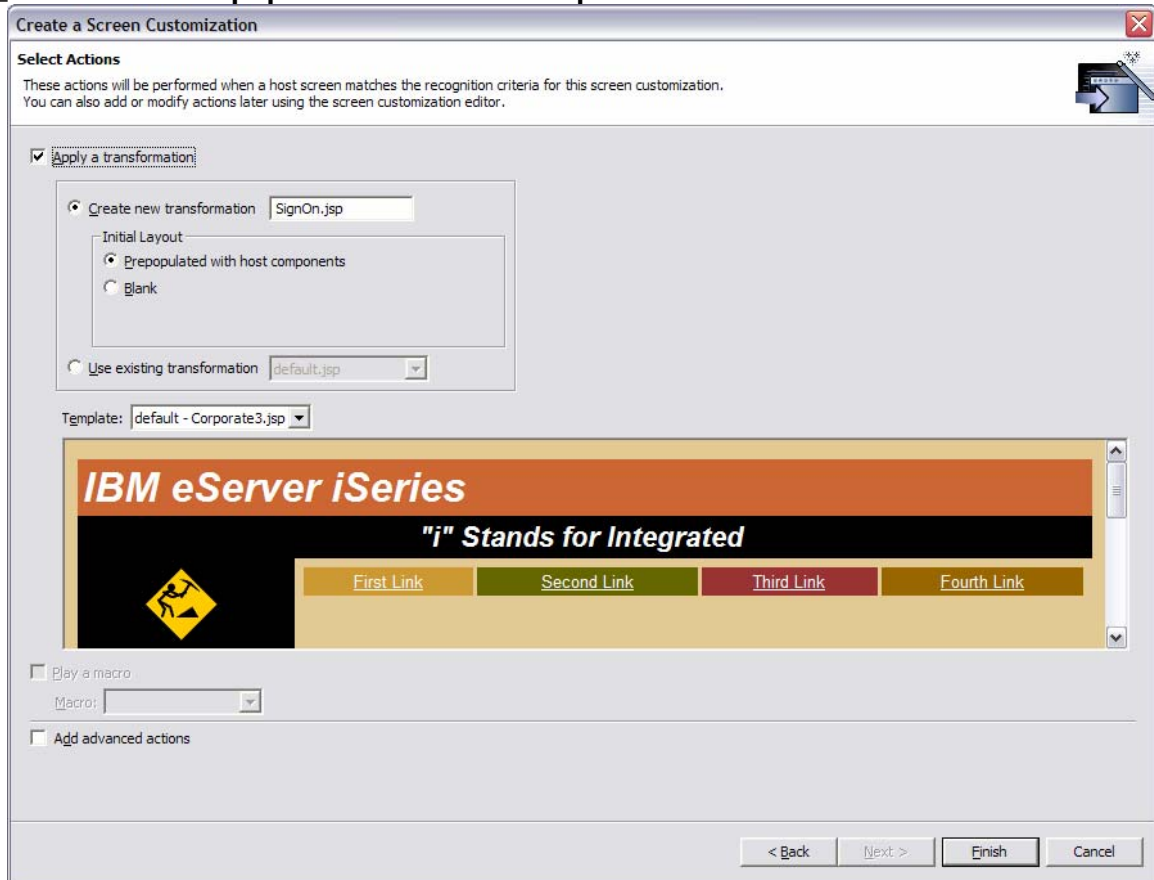
☒ Within a rectangular region:

Starting row  Starting column

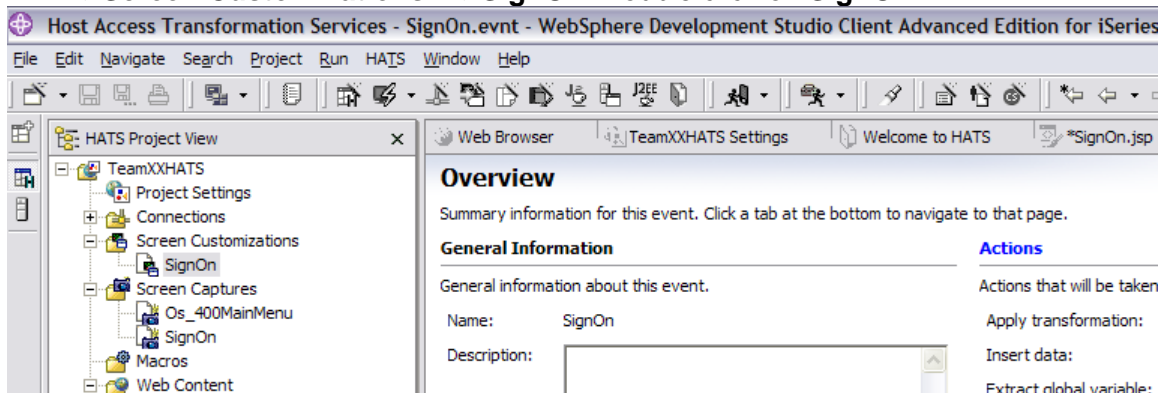
Ending row  Ending column

< Back   Next >   Finish   Cancel

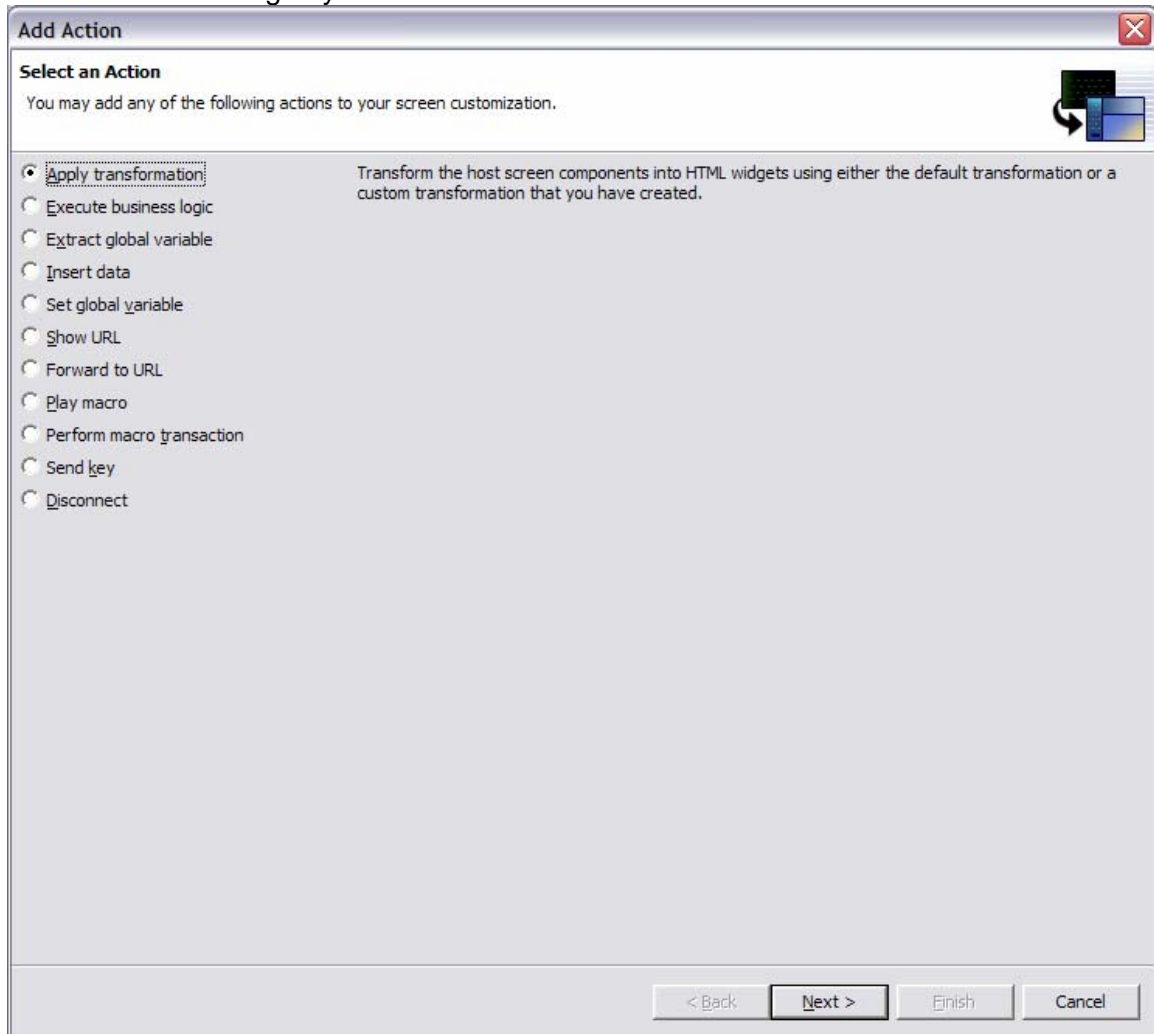
5. Click the **Prepopulated with host components** radio button. Click **Finish**.



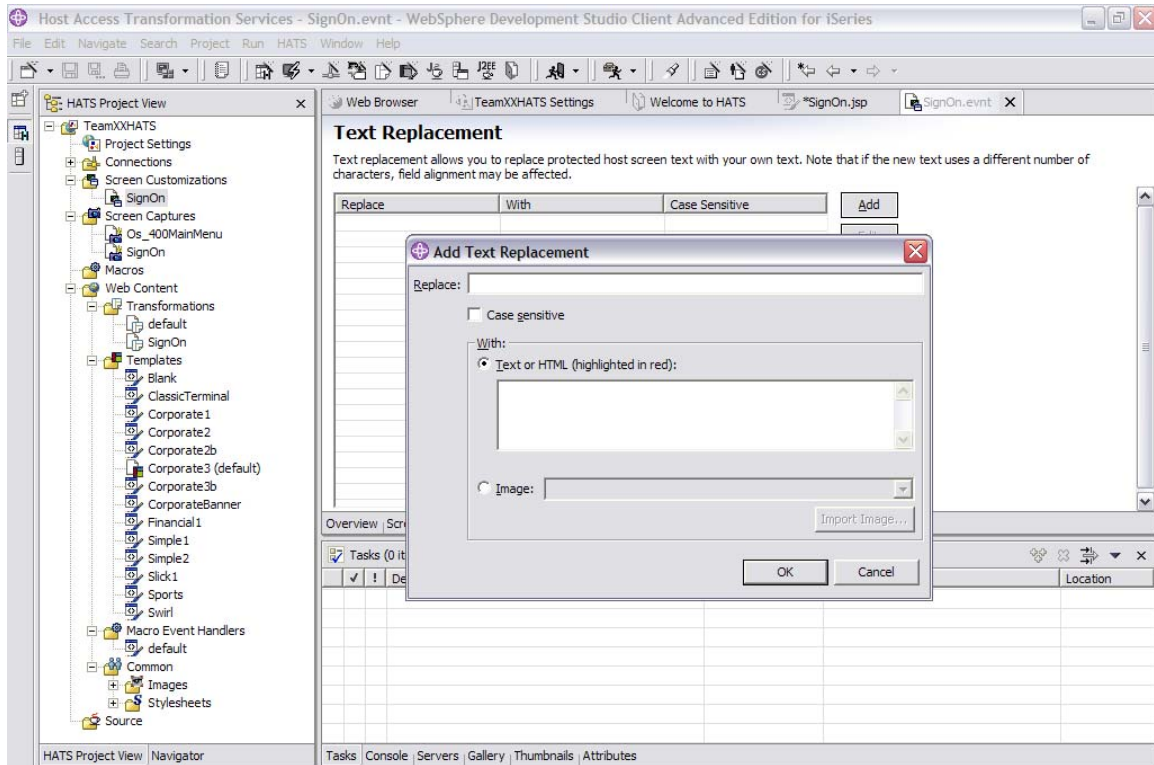
6. In the **HATS Project View**, expand your project and navigate to **TeamXXHATS** —>**Screen Customizations** —>**SignIn**. Double click on **SignIn**.



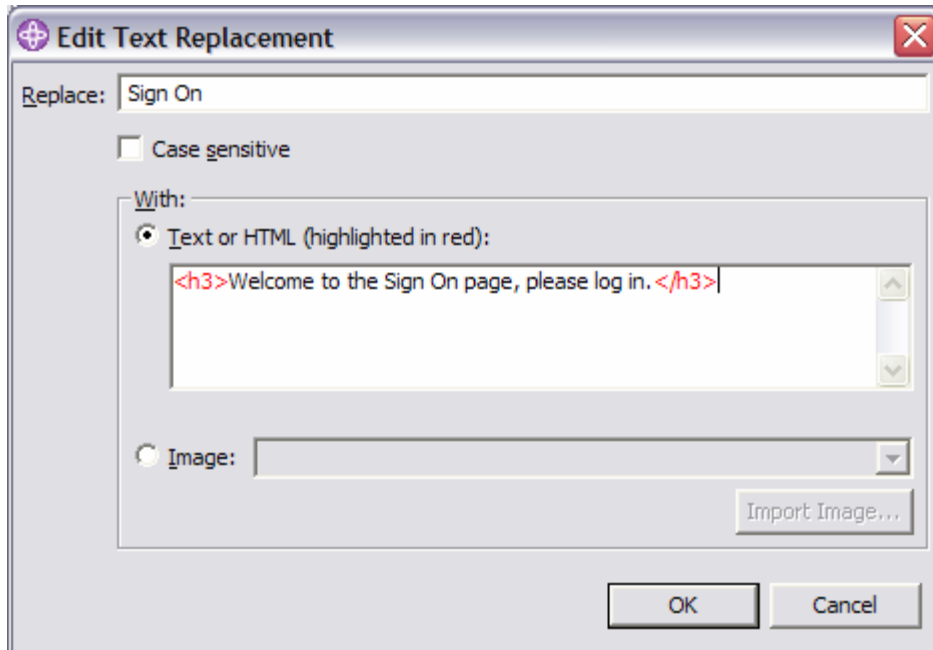
7. Click on the **Actions** tab in the **SignOn.evnt** window, and then click the **Add** button. This is a list of actions that can be added to the screen customization. We will not be adding any action to ours. Click **Cancel**.



8. Click the **Text Replacement** tab in the **SignOn.evnt** window. Click the **Add** button.



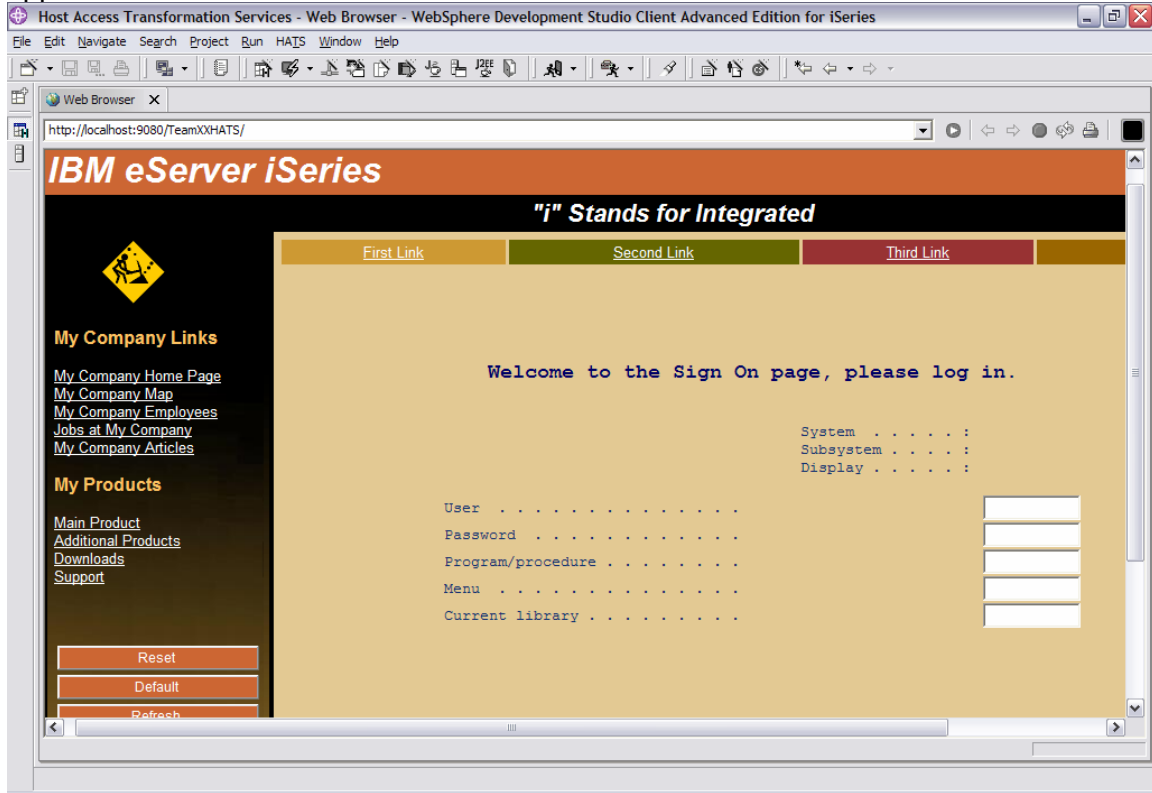
9. In the **Replace:** text field, enter **Sign On**. In the text field under **Text or HTML (highlighted in red):**, enter **<h3>Welcome to the Sign On page, please log in.</h3>**. Click **OK**.



\_\_\_ 10. **Save** your project and preview it using **Run On Server**.

## Success

Congratulations you have successfully completed a Screen Customization. With Screen Customization, you can affect single screen or sets of screens depending on the Screen Recognition Criteria. This is a very powerful technique that when combined with Transformation and other functions in HATS allow almost limitless customization of your application.



## Part 7: Working with Macros and Integration Objects

Macros can be used for many useful purposes, such as:

- Skipping host screens
- Automating common user paths (including repetitive looping)
- Combining data from many host screens into one JSP
- Prompting for and extracting data

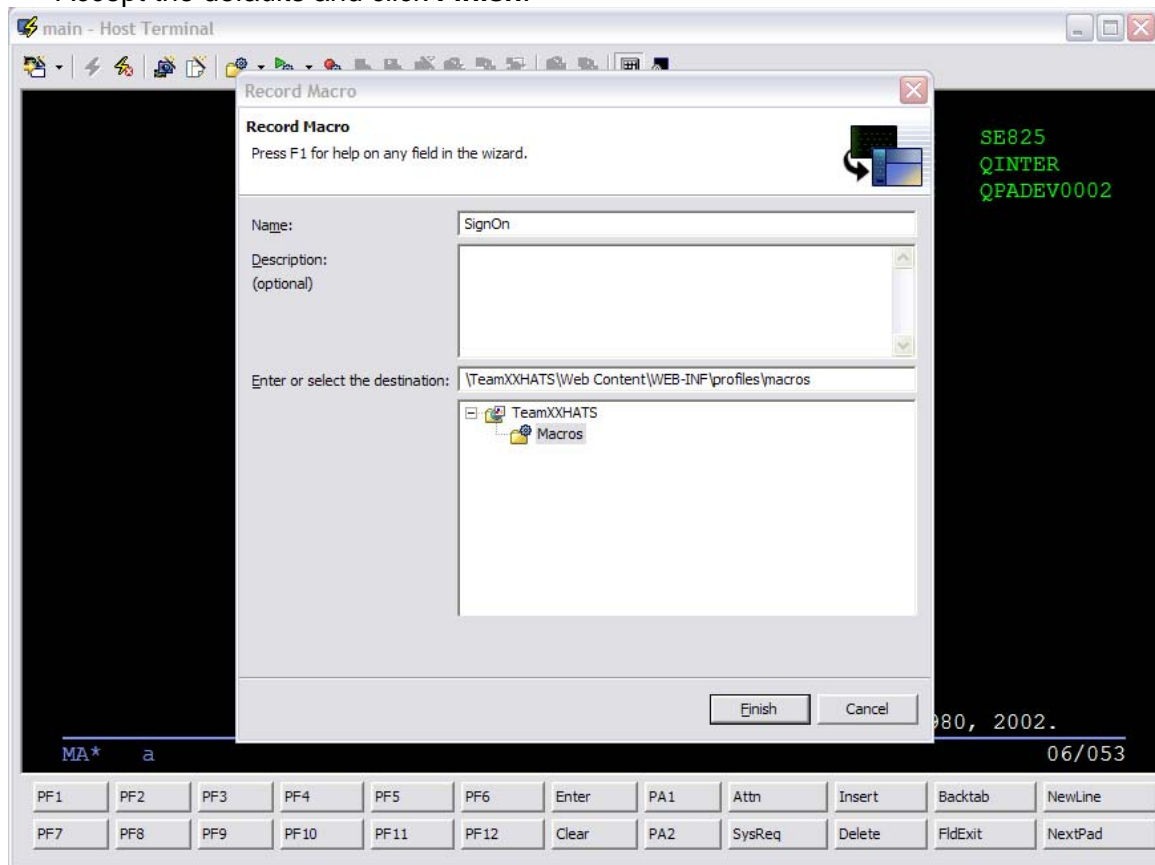
Integration Objects are macros in a Java wrapper. You can create one by right-clicking over a macro in the navigation pane and selecting from the context menu.

Macros and Integration Objects can be created on "secondary connections" rather than the HATS default (or primary) connection, which is used for all transformations.

Secondary connections allow back-end data operations to be specified as actions in screen customizations or other events.

Let's create a simple macro logon macro.

1. Open main – Host Terminal by clicking the **Open HATS Host Terminal** icon in the toolbar (⚡).
2. Click the **Record Macro** icon (⚙️); the **Record Macro** window will pop up. Accept the defaults and click **Finish**.



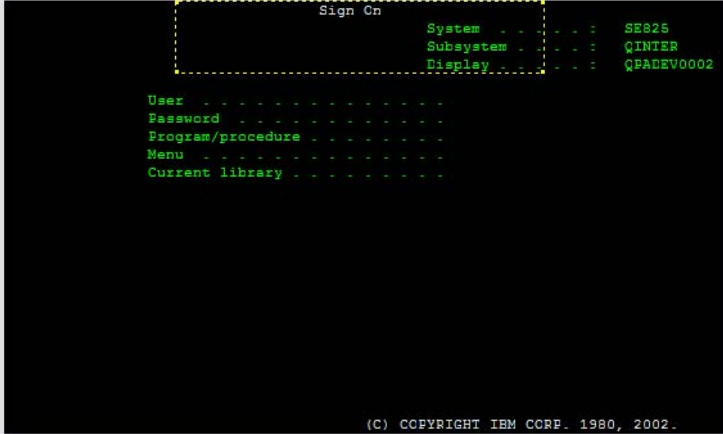
3. On the **Define Screen Recognition Criteria**, define the starting screen of the macro window and accept the defaults. Click **Finish**.

**Define Screen Recognition Criteria**

**Define the starting screen of the macro**  
Select the criteria which will be used to identify this screen when the macro is played. Select Next to add additional criteria.

Screen Name:

☐ This is a transient screen.



☐ Total number of fields:

☐ Number of input fields:

☐ Cursor position: Row  Column

☒ Strings:

Sign On  
System ...

String position:

☐ Anywhere on the screen

☒ At a specified position:

Row  Column

☐ Within a rectangular region:

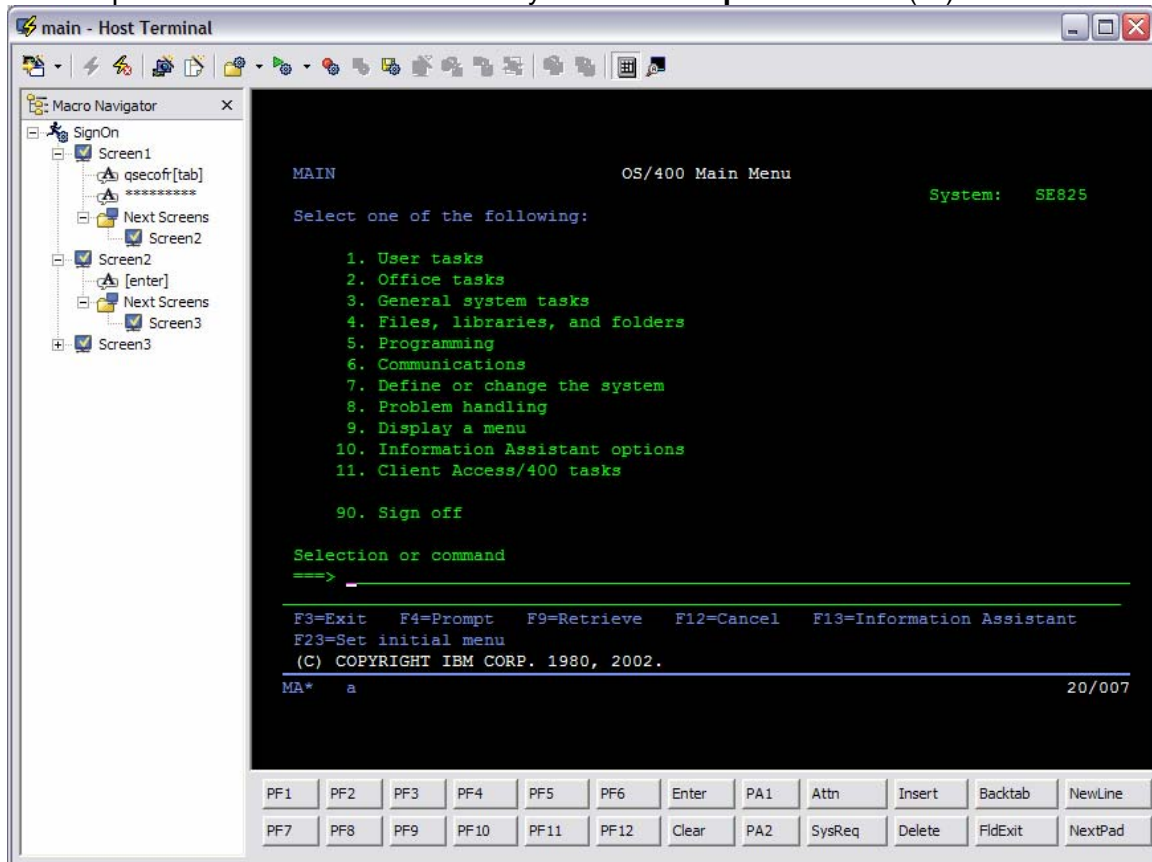
Starting row  Starting column

Ending row  Ending column

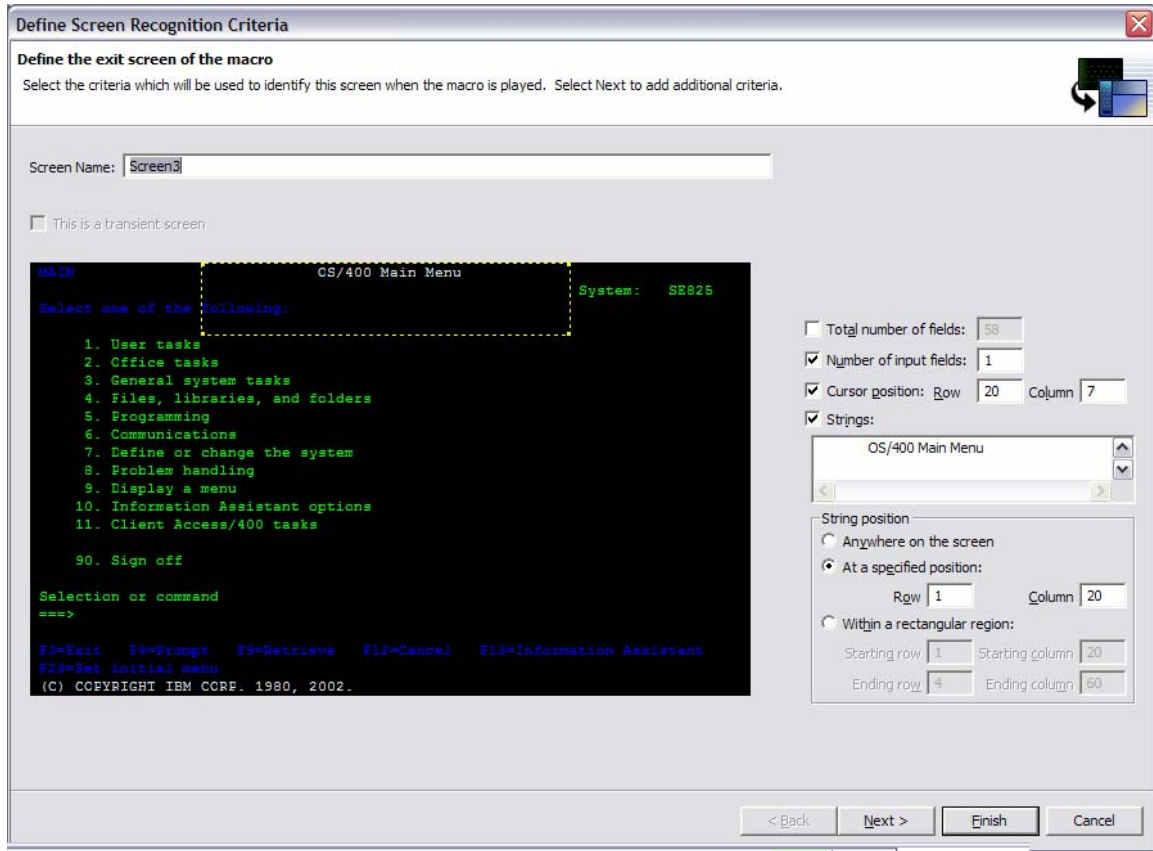
< Back   Next >   Finish   Cancel





4. HATS is now recording what you do in the terminal session. Enter your user ID and password and click the **enter** key. Click the **stop macro** icon (■).

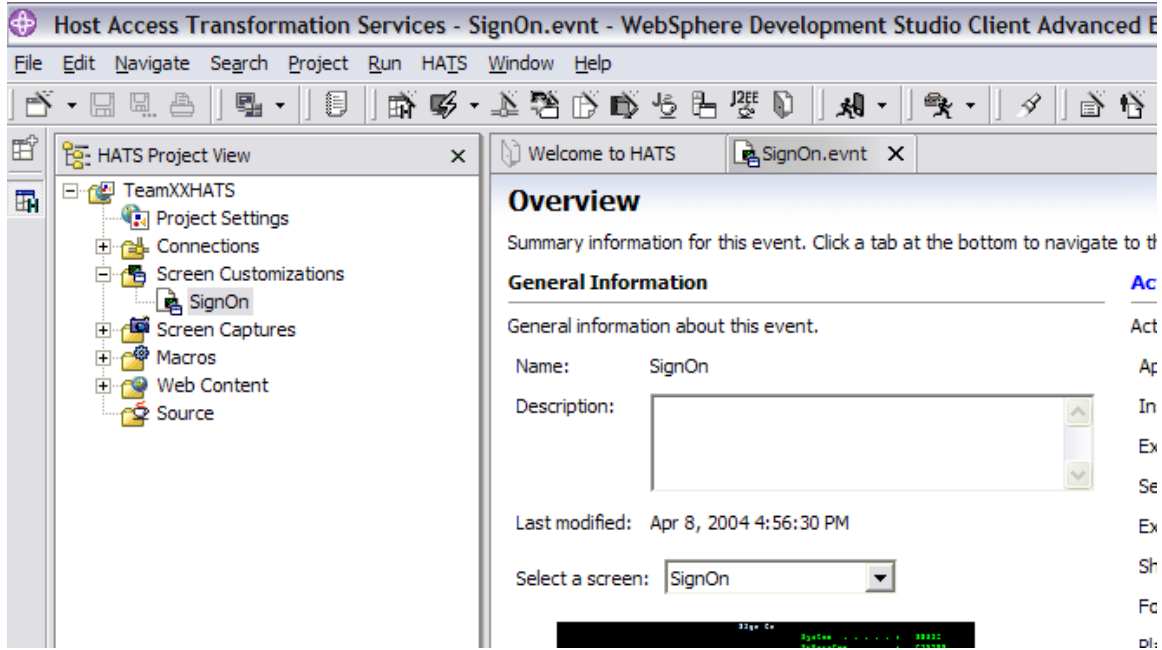


5. After you click the **stop macro** icon () , the **Define Screen Recognition Criteria / Define the exit screen of the macro** window pops up. Accept the defaults and click **Finish**.

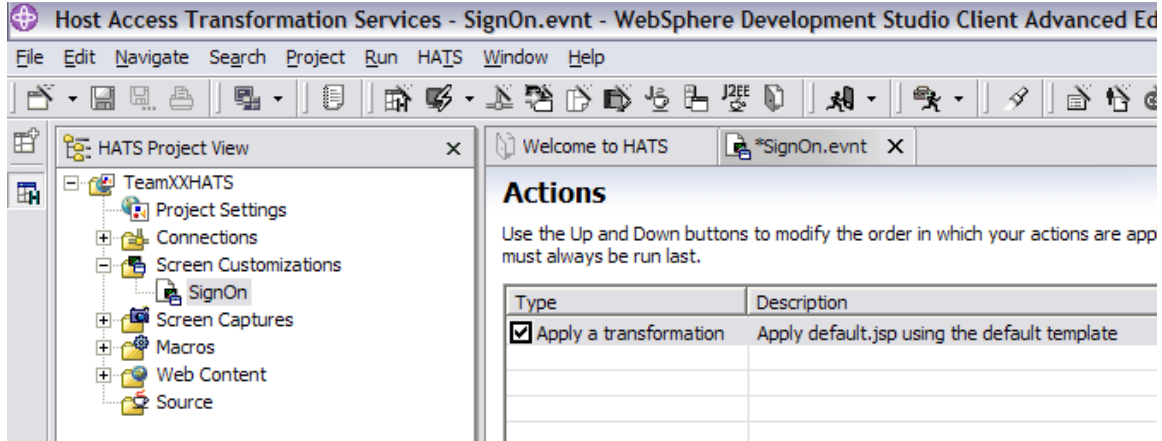


6. Click the **save macro** icon () .
7. You can test your macro by signing off on the 5250 session and then clicking the **play macro** icon () . It should log you on and leave you at the OS/400 Main Menu. Make sure you have saved your macro and then close the **main – Host Terminal** window.

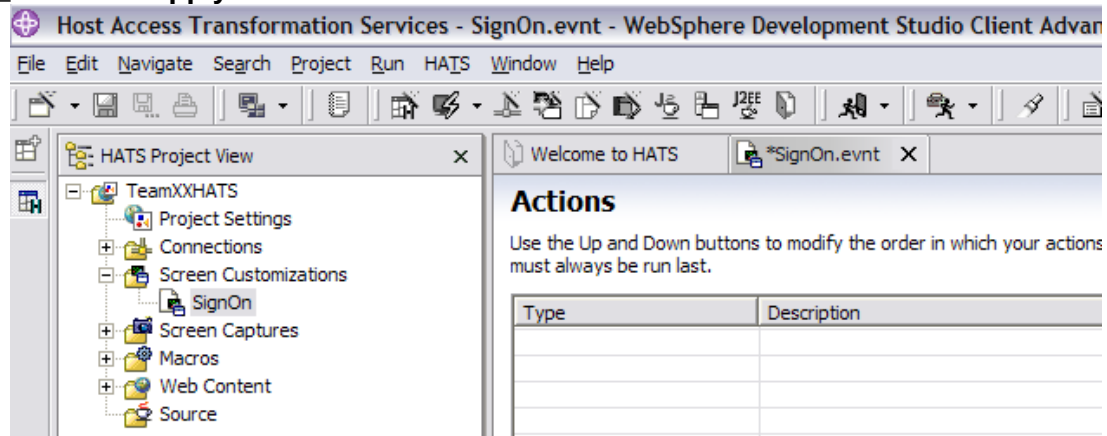
8. You will now add your macro to your **SignOn** Screen Customization. Navigate to **SignOn** under **TeamXXHATS** → **Screen Customizations**, double click **Sign On**.



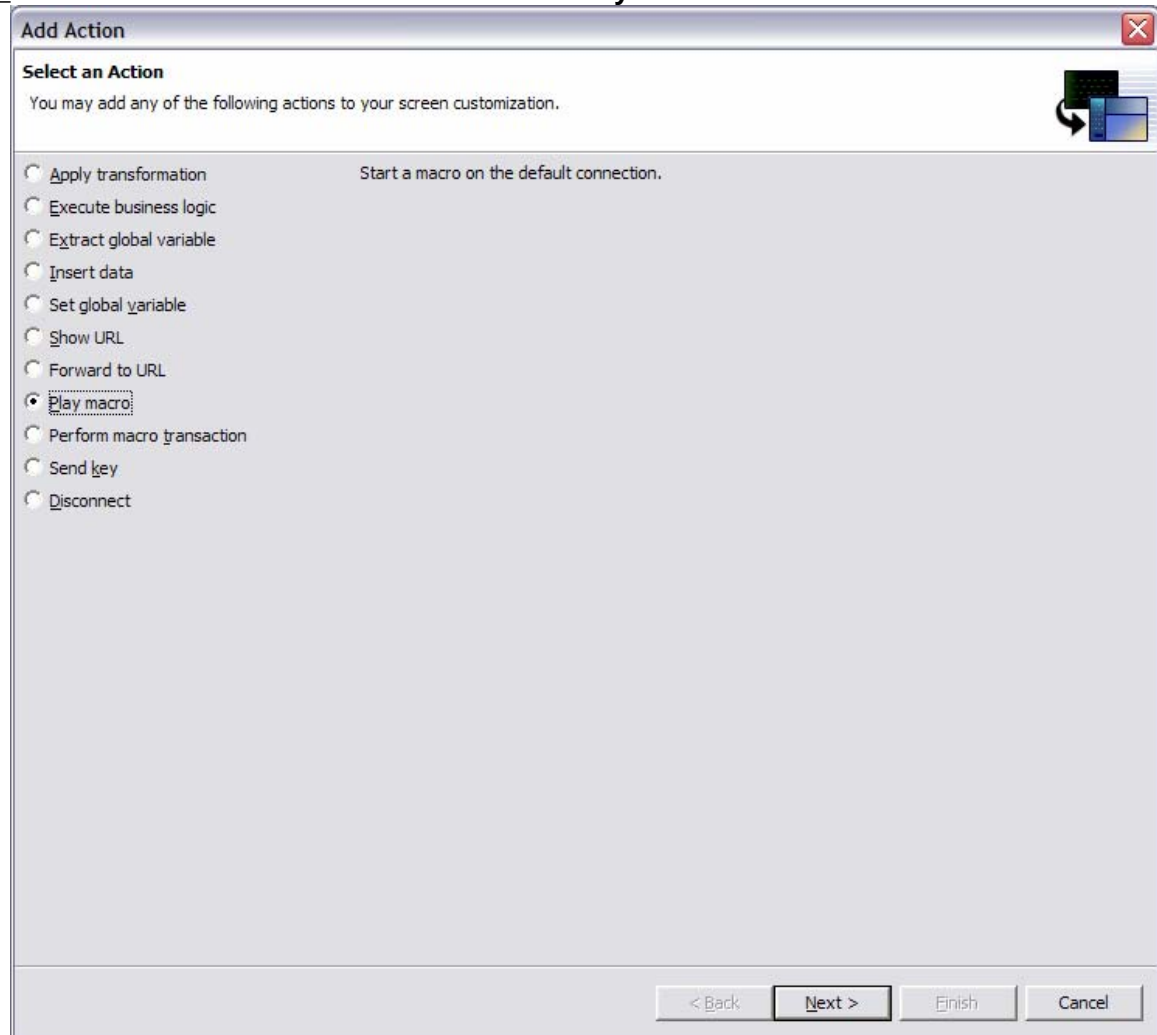
9. Click the **Actions** tab.



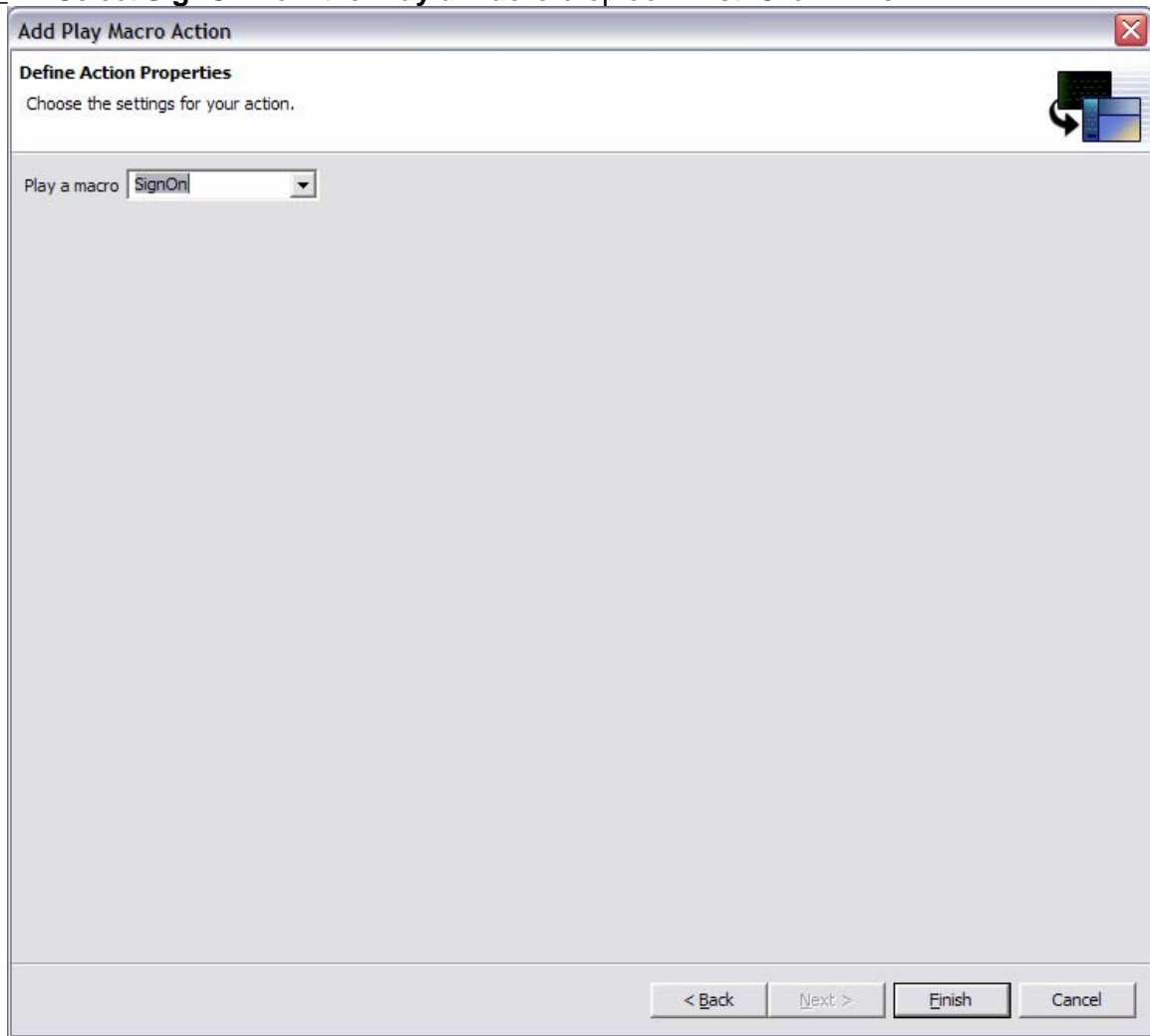
10. Select **Apply a transformation** and click the **Remove** button.



11. Click the **Add** button and then click the **Play macro** radio button. Click **Next**.



12. Select **SignOn** from the **Play a macro** drop down list. Click **Finish**.



13. **Save** your project and preview it using **Run on Server**. Your application should run using the macro logging you. You should see is the OS/400 Main Menu.

## Success

You have just used a macro to change the flow of a 5250 application. It took six steps to create a macro and only another six to integrate that macro into your screen customization. Not only are macros easy to use in HATS but the power of macros allow you to change the flow of your application.



## Conclusion

IBM WebSphere Host Access Transformation Services (HATS) gives you all the tools you need to quickly and easily extend your legacy applications to business partners, customers, and employees. HATS makes your 3270 and 5250 applications available as HTML through the most popular Web browsers, while converting your host screens to a Web look and feel. This lab only touched on a small amount of the function available in HATS. With HATS it is easy to improve the workflow and navigation of your host applications without any access or modification to source code. The HATS rules-based transformation engine makes it possible to extend your host applications to the Web within hours of installing the software. HATS is a zero-footprint, zero-download Web-to-host solution. The only software needed on the client is a Web browser.

## Trademarks and Disclaimer

© IBM Corporation 1994-2004. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both: IBM, eServer, iSeries, WebSphere

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.