

IBM HATS PORTLET MESSAGING

IBM WebSphere **Host Access Transformation Services** is an easy-to-implement Web-to-host solution that delivers HTML to users' Web browsers, extending legacy applications to end users on the Web.

HATS applications can run directly within WebSphere Portal as portlets. By integrating HATS with WebSphere Portal server, data can flow freely between HATS, WebSphere Portal, and other portal applications. IBM WebSphere® Host Access Transformation Services (HATS) portlet messaging is an integration of HATS, WebSphere Portal, and the IBM Portal Toolkit.

This article provides an overview of the implementation of HATS portlet messaging.

We shall discuss two applications:

MessageSender Portlet

MessageReceiver Portlet

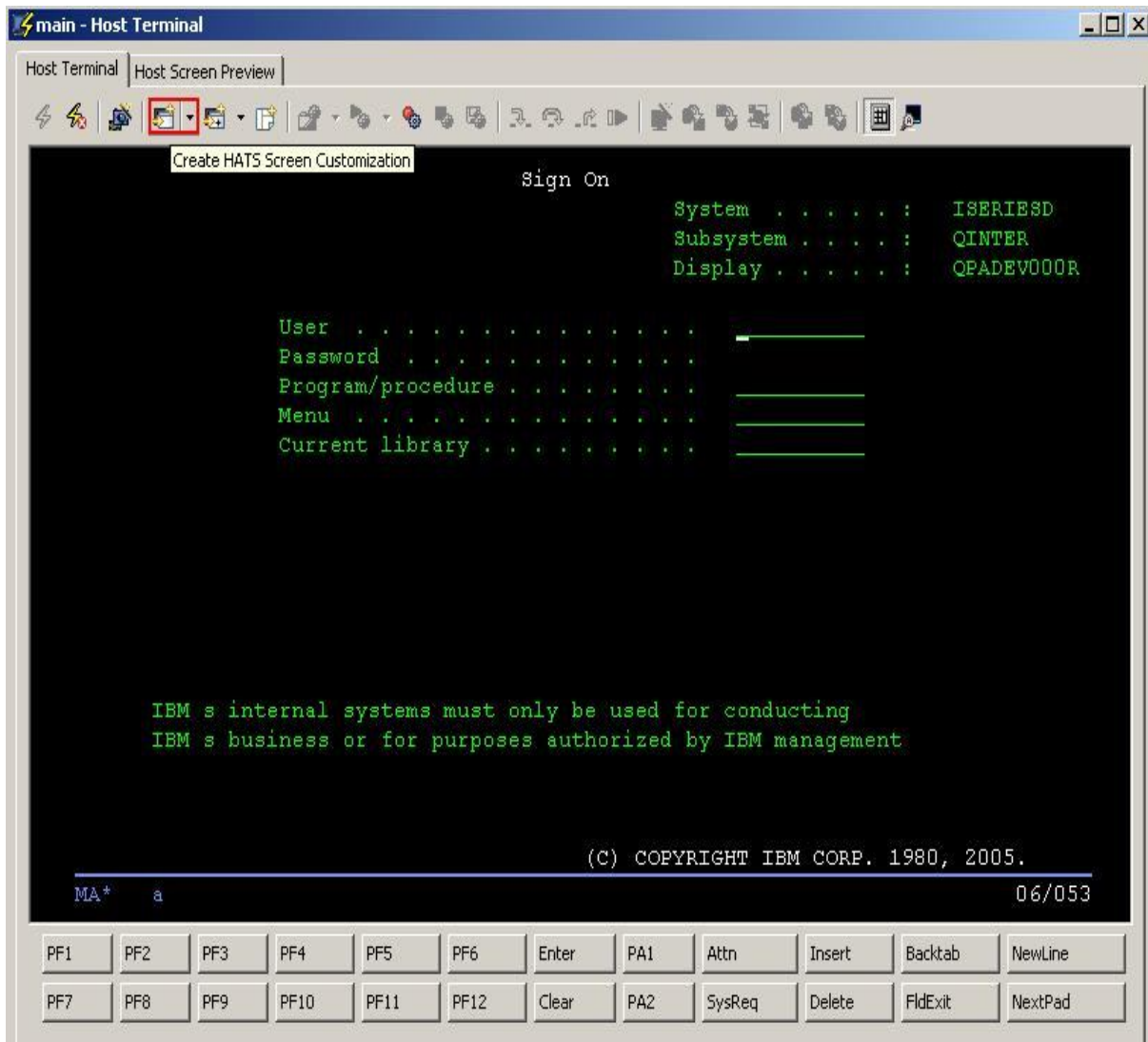
Let us consider the **MessageSender** Portlet first:

CREATING THE *MessageSender* PORTLET:

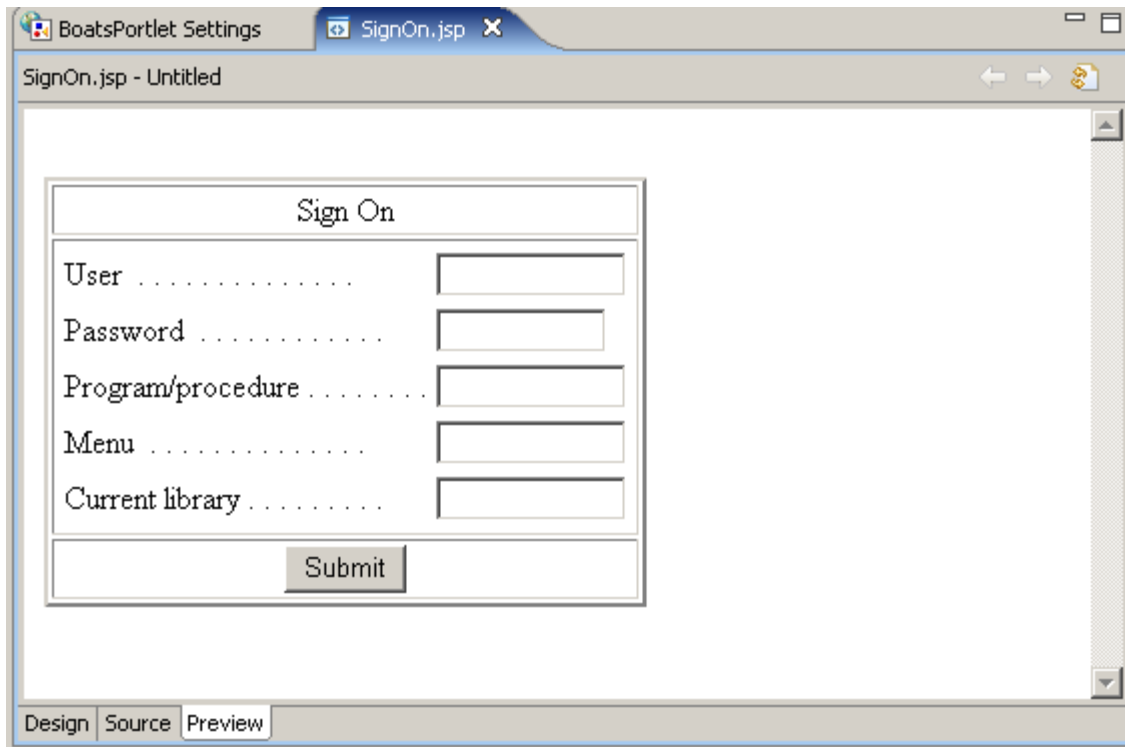
We create a new HATS project in which we specify the target server to be the **WebSphere Portal Server**.

In this project, we create simple customizations of IBM **iseries** server's **Sign On** screen and the **boats** application.

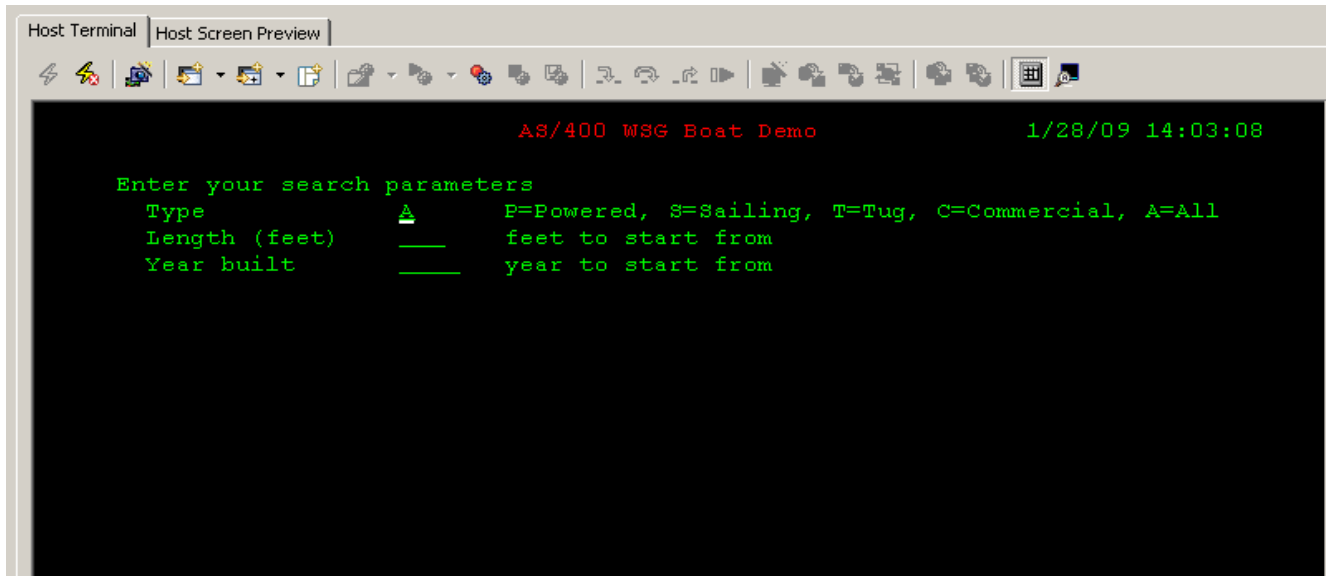
Firstly we create a simple customization for the **Sign On** screen that runs a transformation as an action:



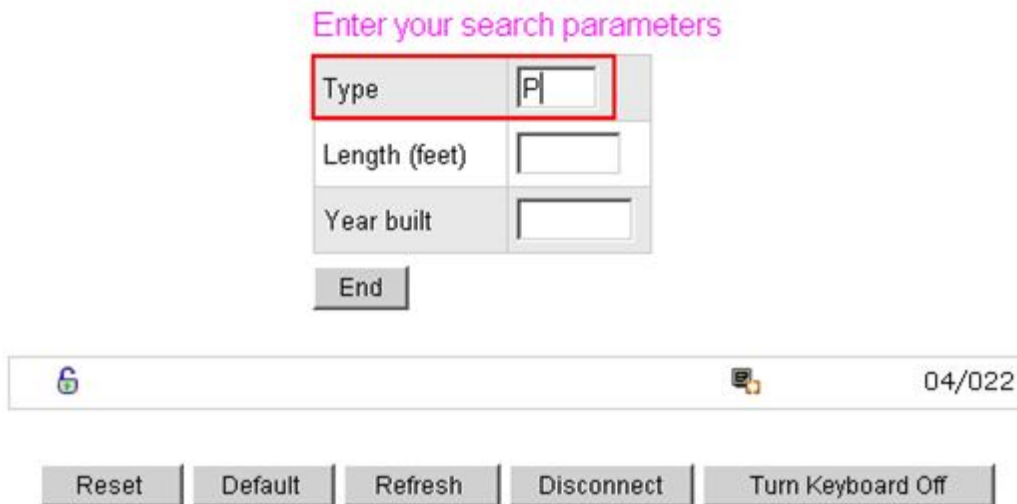
The transformation of the above screen appears as below:



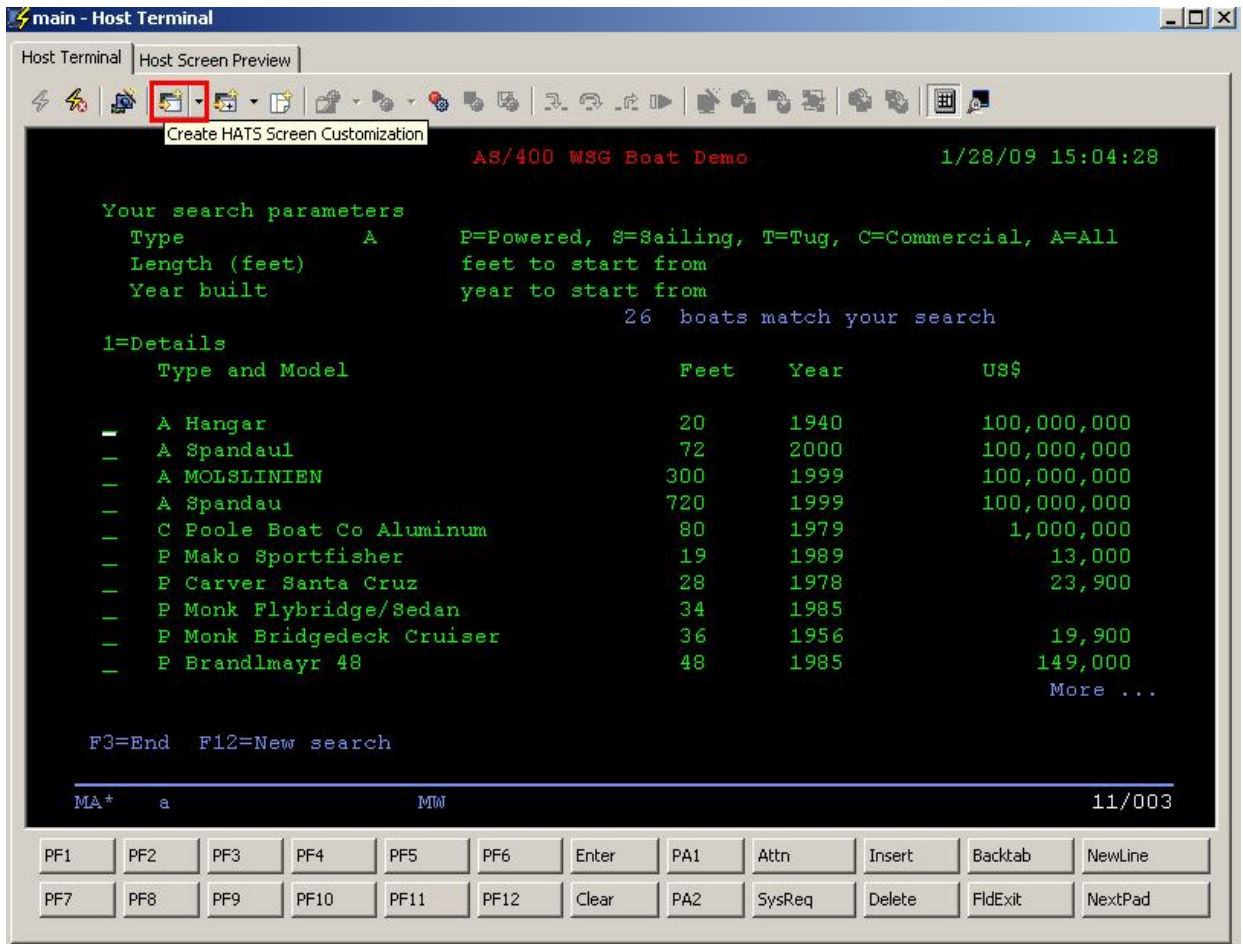
Now we create a customization for the **boats** search screen:



The above screen is transformed as shown below:



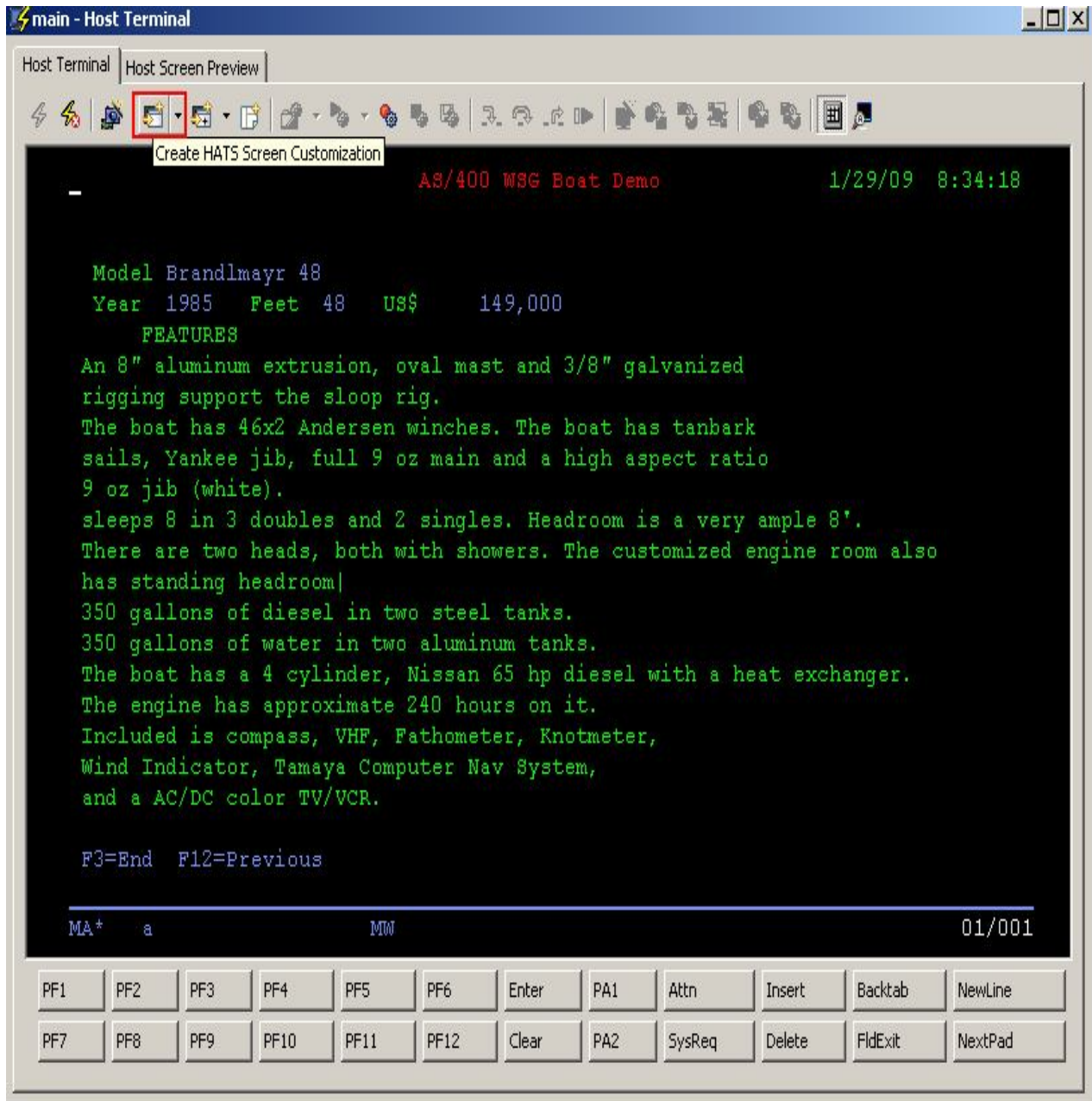
Similarly, we create a customization for the **boats** details screen:



And the transformation appears as below:

Boat Types
Hangar
Spandau1
MOLSLINIEN
Spandau
Poole Boat Co Aluminum
Mako Sportfisher
Carver Santa Cruz
Monk Flybridge/Sedan
Monk Bridgedeck Cruiser
Brandlmayr 48
<input type="button" value="Search"/>

Similarly, a customization for the details screen of any particular boat is also created:



The following information is extracted from the above screen:

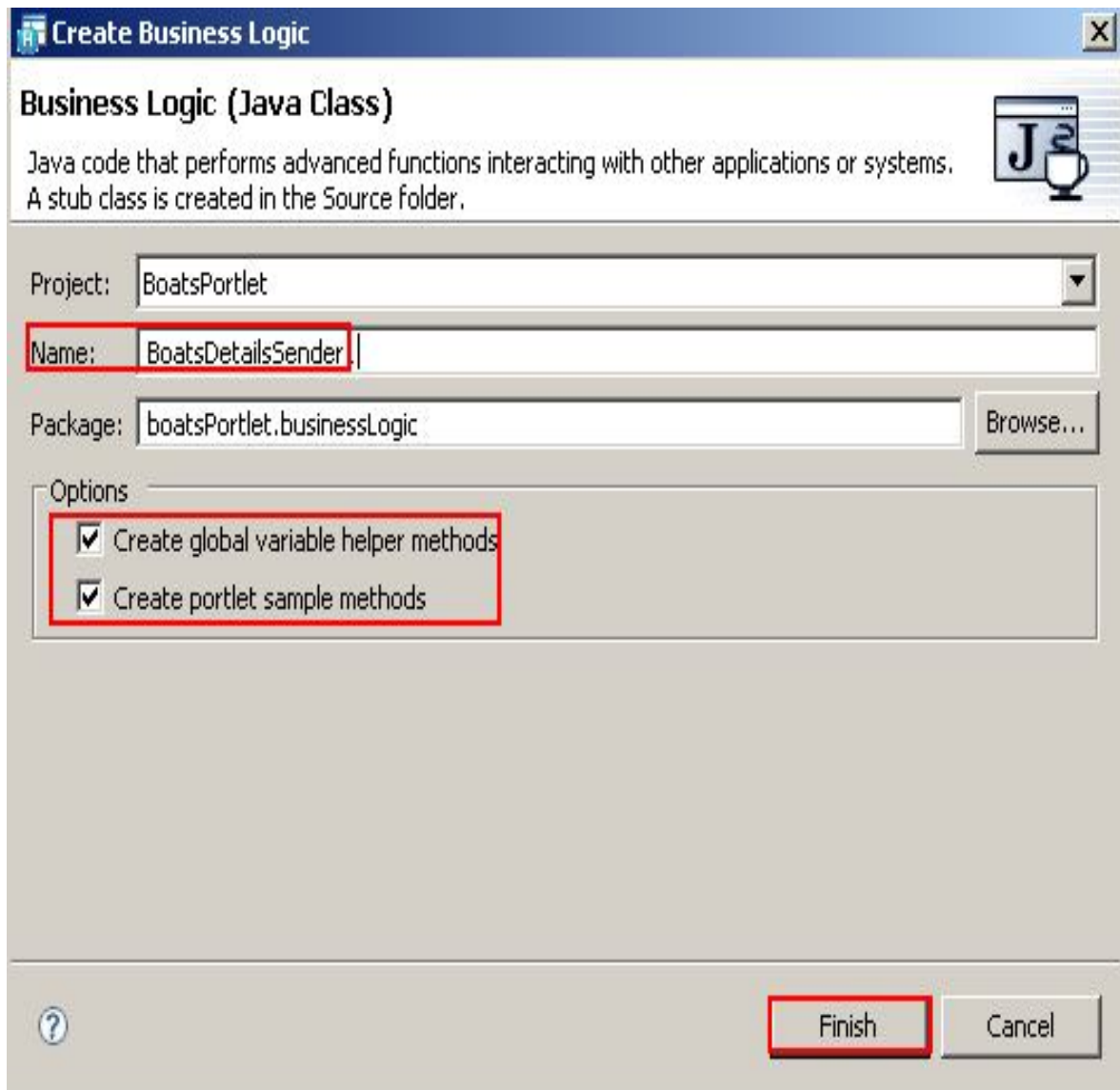
The screenshot shows a web browser window titled "BoatsPortlet Settings" with a sub-tab for "*Boats_Details.evnt". The main heading is "Actions". Below the heading is a paragraph: "Use the Up and Down buttons to modify the order in which your actions are applied (top to bottom) when this event occurs. Note that a macro must always be run last." Below this is a table with two columns: "Type" and "Description". The table contains five rows of actions, each with a checked checkbox in the "Type" column. The fifth row is highlighted. To the right of the table are five buttons: "Add", "Edit", "Remove", "Add Block Delimiter", "Up", and "Down". Below the table is a checkbox labeled "Send a host key when action list completes to cause screen to change:" followed by a dropdown menu showing "ENTER". At the bottom of the window is a navigation bar with tabs: "Overview", "Screen Recognition Criteria", "Actions", "Global Rules", "Text Replacement", "Next Screen", and "Source".

Type	Description
<input checked="" type="checkbox"/> Extract a global variable	Get Features (shared) from region (7, 2) to (21, 73)
<input checked="" type="checkbox"/> Extract a global variable	Get Year (shared) from region (5, 9) to (5, 12)
<input checked="" type="checkbox"/> Extract a global variable	Get Cost (shared) from region (5, 31) to (5, 41)
<input checked="" type="checkbox"/> Extract a global variable	Get Feet (shared) from region (5, 21) to (5, 23)
<input checked="" type="checkbox"/> Extract a global variable	Get Model (shared) from region (4, 9) to (4, 38)
<input checked="" type="checkbox"/> Apply a transformation	Apply Boats_Details.jsp using the default template

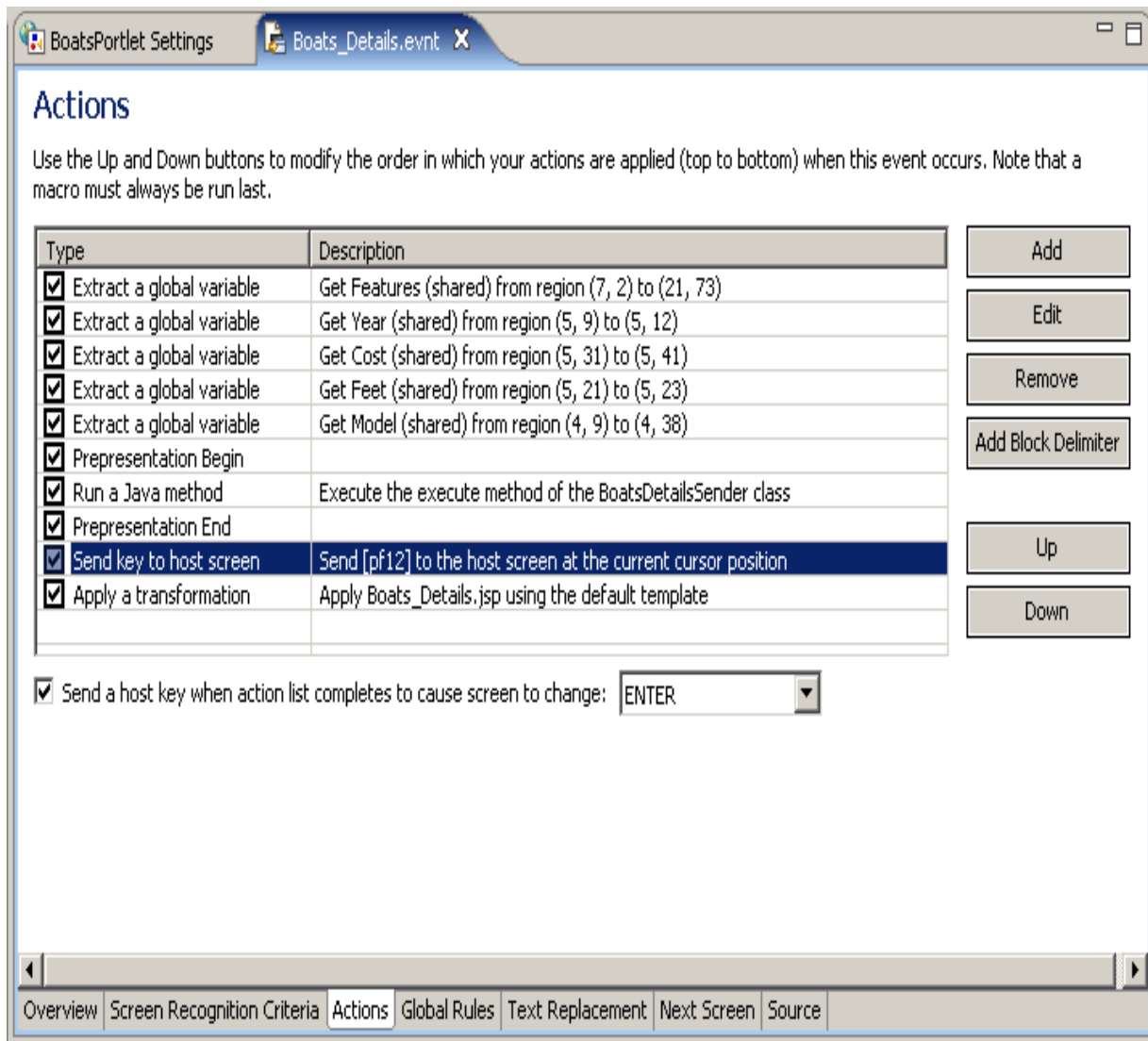
Send a host key when action list completes to cause screen to change:

Navigation tabs: Overview | Screen Recognition Criteria | **Actions** | Global Rules | Text Replacement | Next Screen | Source

Now we create **BoatsDetailsSender (MessageSender)** business logic as follows:



Now we insert our business logic and other necessary actions into the last created customization as shown below:



Next, we modify our business logic Java file to broadcast the Portlet message to the receiver Portlet application. This completes the development of **MessageSender** application and the WAR can now be generated which will be deployed later.

We now move on to create the **MessageReceiver** portlet.

CREATING THE *MessageReceiver* PORTLET:

We create a new portlet project:

New Portlet Project

Portlet Project
Specify a name and location for the new portlet project.

Project name: BoatsDetailsReceiver

Project contents:
 Use default
Directory: C:\Workspaces\HATS\Iserver\Mudasir\BoatsDetailsReceiver

Target Runtime
WebSphere Portal v6.0

EAR Membership
 Add project to an EAR
EAR Project Name: BoatsDetailsReceiverEAR

Portlet API: IBM Portlet
IBM portlet API supported by WebSphere Portal 4.0 and later.

Create a portlet
Portlet name: BoatsDetailsReceiver
Portlet type: Basic Portlet
Create a portlet that extends the PortletAdapter class defined in the IBM Portlet API. You will be asked to selected several options for generating sample code.

Show advanced settings

< Back Next > Finish Cancel

We now modify the **BoatsDetailsReceiverPortlet.java** class to include the message receiving functionality.

Next, we modify the portlet JSP to receive the sent message from the **MessageSender** portlet and to display the details.

Now we can generate the WAR file of our **MessageReceiver** portlet.

DEPLOYMENT AND TESTING ON THE WEBSHERE PORTAL SERVER.

The WAR files of our **MessageSender** and **MessageReceiver** portlets can now be deployed on the WebSphere Portal Server. The screen below shows the login screen on the **MessageSender** portlet running on the Portal Server and its corresponding **MessageReceiver** screen. Since there is no data on the **Sign On** screen that can be displayed on the **MessageReceiver** portlet, it displays the JSP without any details as shown below

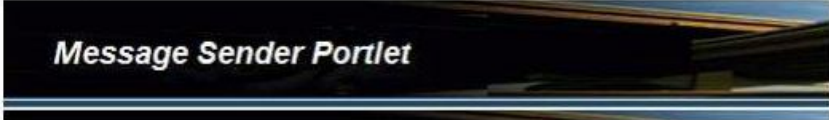
The image shows two side-by-side portlets from a WebSphere Portal. The left portlet, titled "BoatsPortlet", displays the "Message Sender Portlet" with a "Sign On" form. The form includes fields for "User" (containing "ATDEMO"), "Password" (masked with dots), "Program/procedure", "Menu", and "Current library", along with a "Submit" button. The right portlet, titled "BoatsDetailsReceiver", displays the "Message Receiver Portlet" with a "Boats Details" section. This section contains a table with columns for "Model", "Year", "Feet", and "Cost", and a "Features" section below it. The "Boats Details" section is currently empty.

Boats Details	
Model	
Year	
Feet	
Cost	
Features	

:

After signing on and navigating to the boat search results and clicking on one of the results in the **MessageSender** (left) portlet, the details of our selected item are displayed in the **MessageReceiver** (right) portlet JSP as shown below:

BoatsPortlet



Boat Types

[Mako Sportfisher](#)

[Carver Santa Cruz](#)

[Monk Flybridge/Sedan](#)

[Monk Bridgedeck Cruiser](#)

[Brandlmayr 48](#)

[Merlin's Magic](#)

[Monterey Marine Custom](#)

[Katemeran](#)

[william's boat](#)

[dragon's boats](#)

BoatsDetailsReceiver

Message Receiver Portlet

Model	Mako Sportfisher
Year	1989
Feet	19
Cost	13,000
Features	<p>Located in Anacortes, WA. Great fishing boat Center console with wind shield, Depth sounder, VHF, CB. Fiberglass hull. Dock bumpers, flares, full Canvas cover. 130 hp V-4 Yamaha gas outboard oil injected with 8 hp Honda kicker. Includes galvanized trailer.</p>

11/003



© Copyright IBM Corporation 2010
IBM Global Services
Route 100
Somers, NY 10589
U.S.A.
Produced in the United States of America
08-10
All Rights Reserved

IBM, the IBM logo, ibm.com, Lotus®, Rational®, Tivoli®, DB2® and WebSphere® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml Other company, product and service names may be trademarks or service marks of others. The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software. This document illustrates how one organization uses IBM products. Many factors have contributed to the results and benefits described; IBM does not guarantee comparable results elsewhere.