



IBM Software Group

IBM Rational Host Access Transformation Services (HATS)

March 2008

Rational software



Will Smythe
IBM Rational
smythew@us.ibm.com

© 2007 IBM Corporation

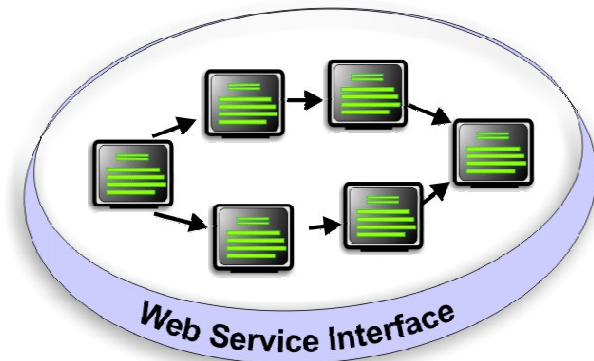
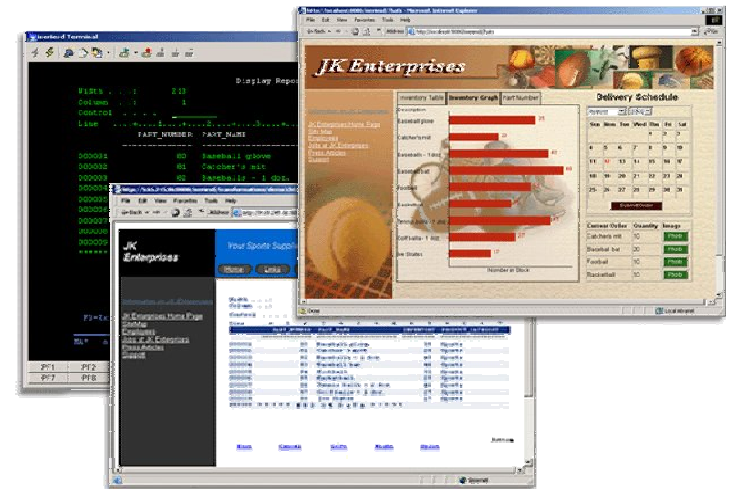
Agenda

- Overview of IBM Rational Host Access Transformation Services (HATS)
- Application modernization demonstration
- HATS 7.1 release highlights
- EGL and HATS



Host Access Transformation Services (HATS)

- What can you do with HATS?
 - ▶ Quickly and easily create Web, portal or rich client applications that provide an easy-to-use GUI for your green-screen applications
 - Low skills requirement
 - Highly customizable
 - Iterative development process
 - Transformation “on the fly”
 - ▶ Extend terminal application tasks as Web services
- Benefits
 - ▶ Extend host application to new users
 - ▶ Improve the navigation of your host application
 - ▶ Reuse your existing assets in a Service Oriented Architecture
 - ▶ Avoid rewriting applications (no impact to code)

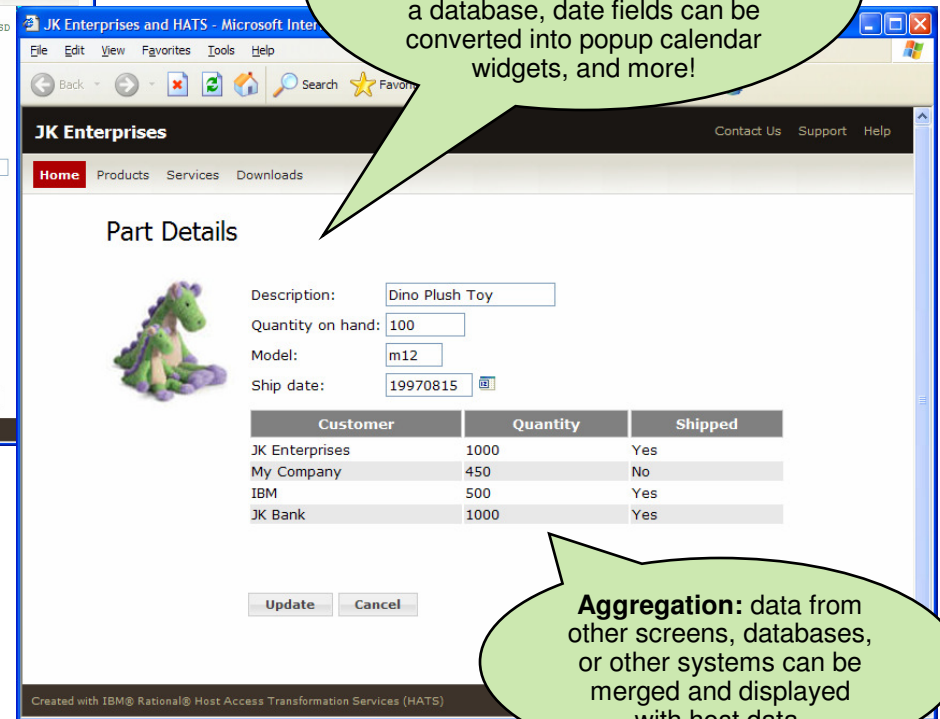
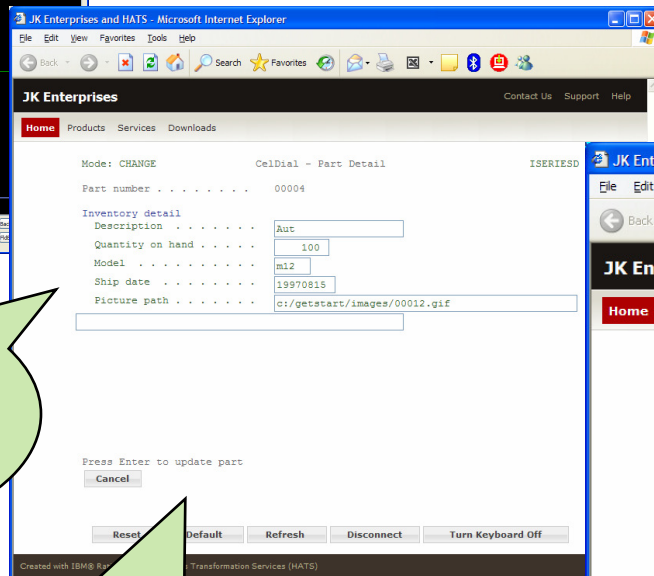
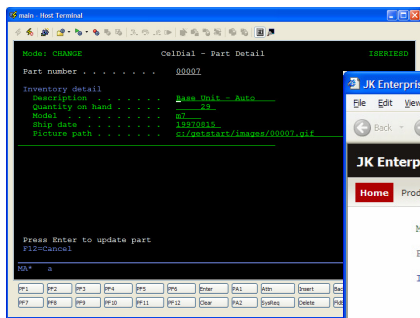


Modernization At Your Own Pace

Today

Day 1

Tomorrow



Modern UI: All pages share the same theme and style, which can be based on an existing Web site.

Instant Value: HATS default rendering automatically transforms actions into clickable links and buttons. No changes required to the host application.

Integration: Text can be replaced with images, input fields can be populated from data on other screens or from a database, date fields can be converted into popup calendar widgets, and more!

Aggregation: data from other screens, databases, or other systems can be merged and displayed with host data.

Host Access Transformation Services Values

Rich Client



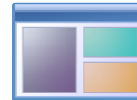
- Integration at the desktop with other Eclipse-based applications
- Client side processing
- Rich set of user interface widgets
- Supports Lotus Expeditor deployment

Browser



- Zero footprint
- Pure HTML
- Access through your favorite browser, including Internet Explorer and Firefox.

Portal



- Integration at the glass
- Cooperative portlet support
- JSR 168 compliant

Mobile



- Access host applications from mobile devices

Web Services



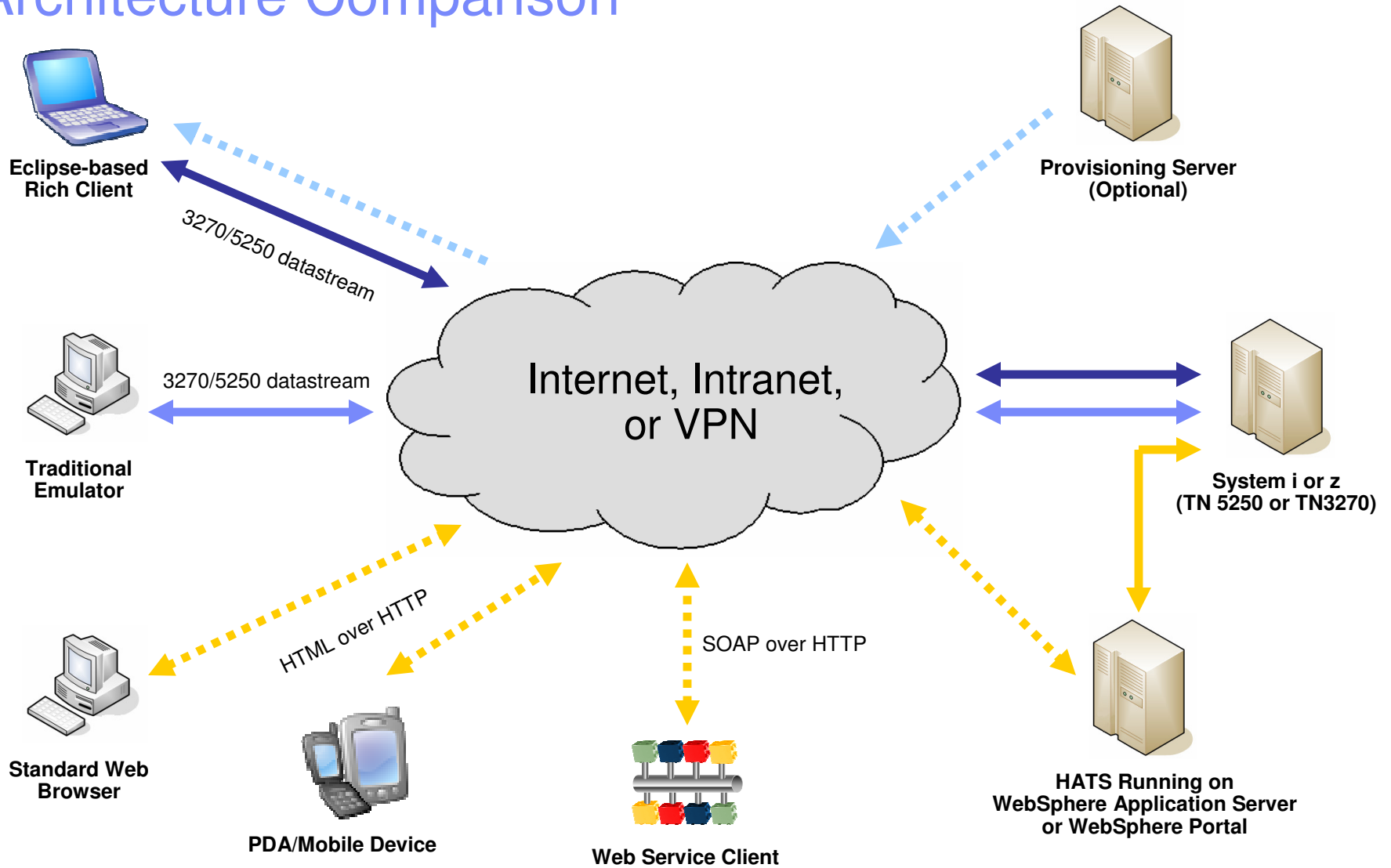
- Build self-service transactions
- Expose host business processes as Web Services
- Provide controlled access to vital host applications and host data.

- *Quick ROI : Host applications can be quickly deployed with a new user interface*
- *Low cost : No need to rewrite application*
- *Low risk : Leverage open, proven platforms and technologies*
- *Increase productivity and reduce training costs*
- *Improving work flow from multiple applications*
- *Provides integration of host business processes and data with other Web, portal, and rich client applications*



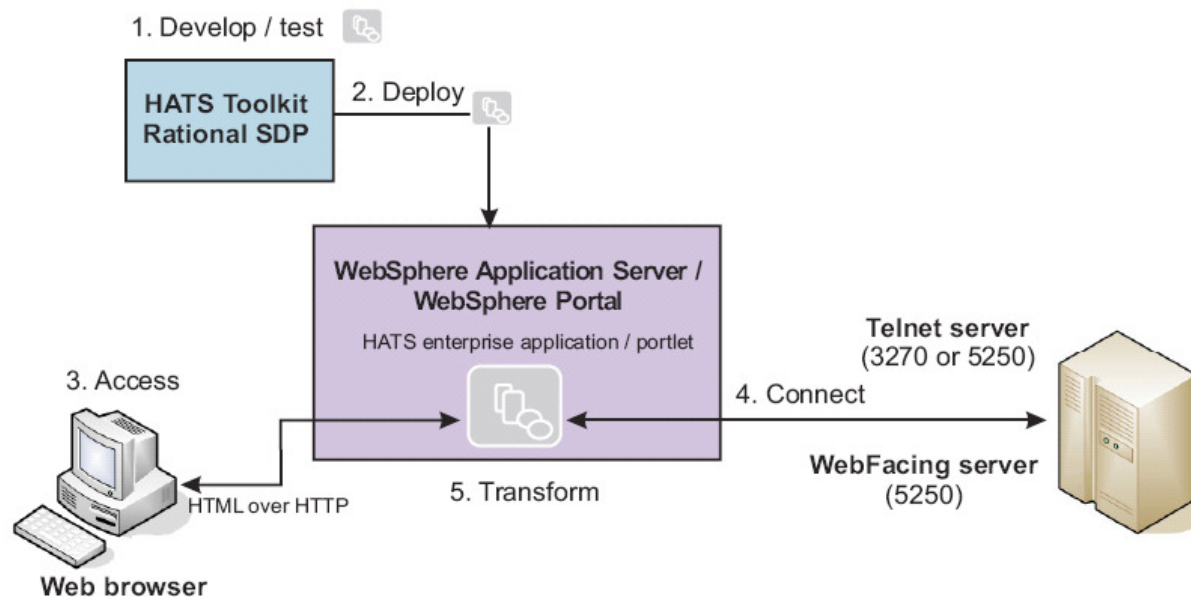
Persistent
Non-persistent

Architecture Comparison



Creating a HATS Web Application

1. The developer uses the HATS Toolkit to develop and test a HATS Web application.
2. When ready, the developer deploys the HATS application by exporting it as a J2EE application and installing and running it on a WebSphere Application Server system.
3. Using a Web browser, the user accesses the HATS application.
4. The HATS runtime connects to the target host system through either a Telnet or a WebFacing server.
5. As the user interacts with the host through the HATS application, the HATS runtime transforms host screens to a GUI.



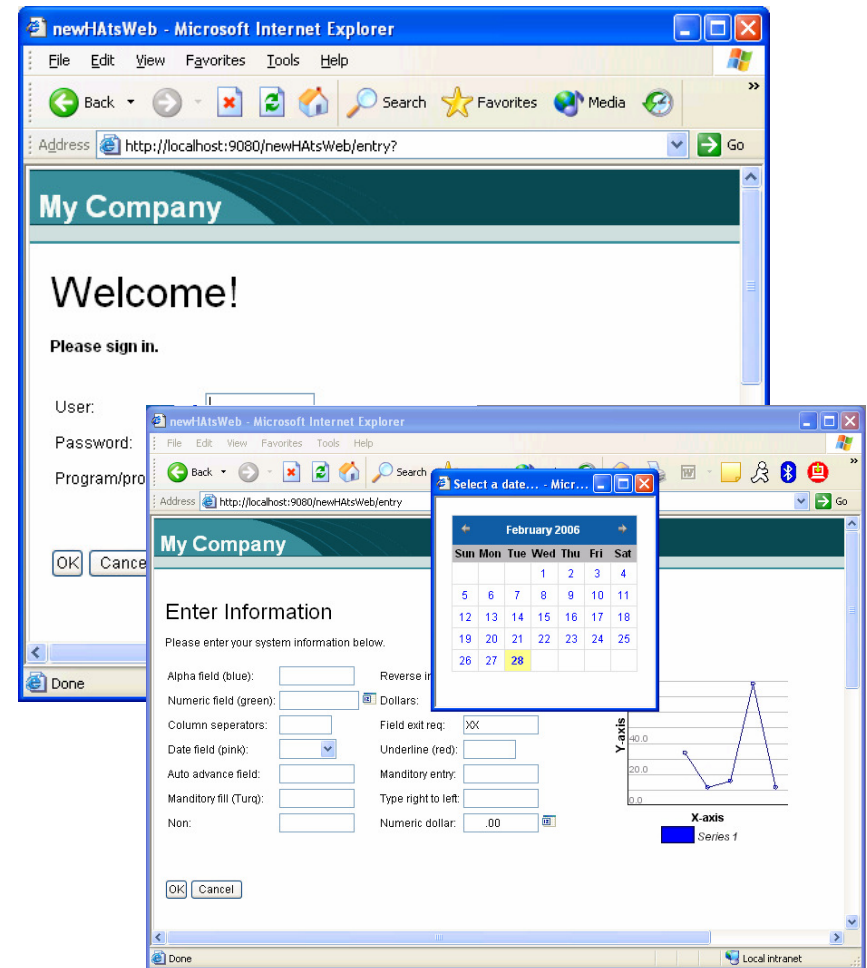
Achieving a Green Screen Look with HATS



- HATS supports host colors and extended field attributes, including:
 - ▶ Numeric only, alphabetic only, signed numeric, reverse image, column separator, underline, and blink fields
- Supports automatic field advance
- Host keypad and OIA can be displayed (and positioned anywhere)
- Native keyboard support (i.e. pressing F12 in the browser sends F12 to the host application).

Achieving a Modern Web Look with HATS

- Customize screens with intuitive web user controls, including:
 - ▶ Popup lists, drop-downs, checkboxes, lists, radio buttons, calendar popups, graphs, etc.
- Simplify screens and work flow by
 - ▶ Split one screen into multiple screens
 - ▶ Organize data in tabs
 - ▶ Remove functions / actions no longer needed by removing full screens or partial screens
- One customized Web page can be applied to more than 1 host screen!



HATS Customized Transformation

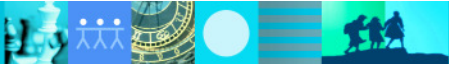
The screenshot shows a web browser window titled "Welcome to JK Enterprises - Microsoft Internet Explorer". The address bar shows "http://localhost:9080/oct21/entry". The main content area features a navigation menu on the left with links for "Home Page", "Map", "Employees", "Enterprises", "Articles", "Products", "Main Product", "Additional Products", "Downloads", and "Support". Below these are buttons for "Reset", "Default", "Refresh", "Disconnect", and "Turn Keyboard Off".

The central part of the page displays an "Inventory Graph" with a horizontal bar chart. The y-axis is labeled "Type" and lists various animals. The x-axis is labeled "Price" and ranges from 14.0 to 28.0. A legend indicates that the red bars represent "Sales Price".

Type	Sales Price
Bear, Koala	25.0
Dog, Dachshund	18.0
Cat, Manx	14.0
Cat, Persian	21.0
Cat, Siamese	22.0
Monkey	20.0
Wallaby	23.0
Manatee	24.0
Wombat	15.0

To the right of the graph is a "Submit Order" form with fields for "Product number:", "Quantity:", and "Date required:". Below the form is an "Order" button. A "Check Inventory" button is located below the form. A note states: "To see the inventory count for this product, click the **Check Inventory** button."

A "Select a date..." calendar window is open, showing the month of October 2003. The date 27 is selected.



Global Rules and Valid Value Lists

JK Enterprises

Search Request

To request a search, type the search data and press E

Corporate and Personal Directories

Name

Birthdate 01/22/1979

Node

User ID

Extension

Job Responsibilities

Location

*any field (Name Value)

*any field (Name Value)

Enter command: _____

United States
Japan
Germany
Australia
United Kingdom
Canada
Italy
France

January 1979

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

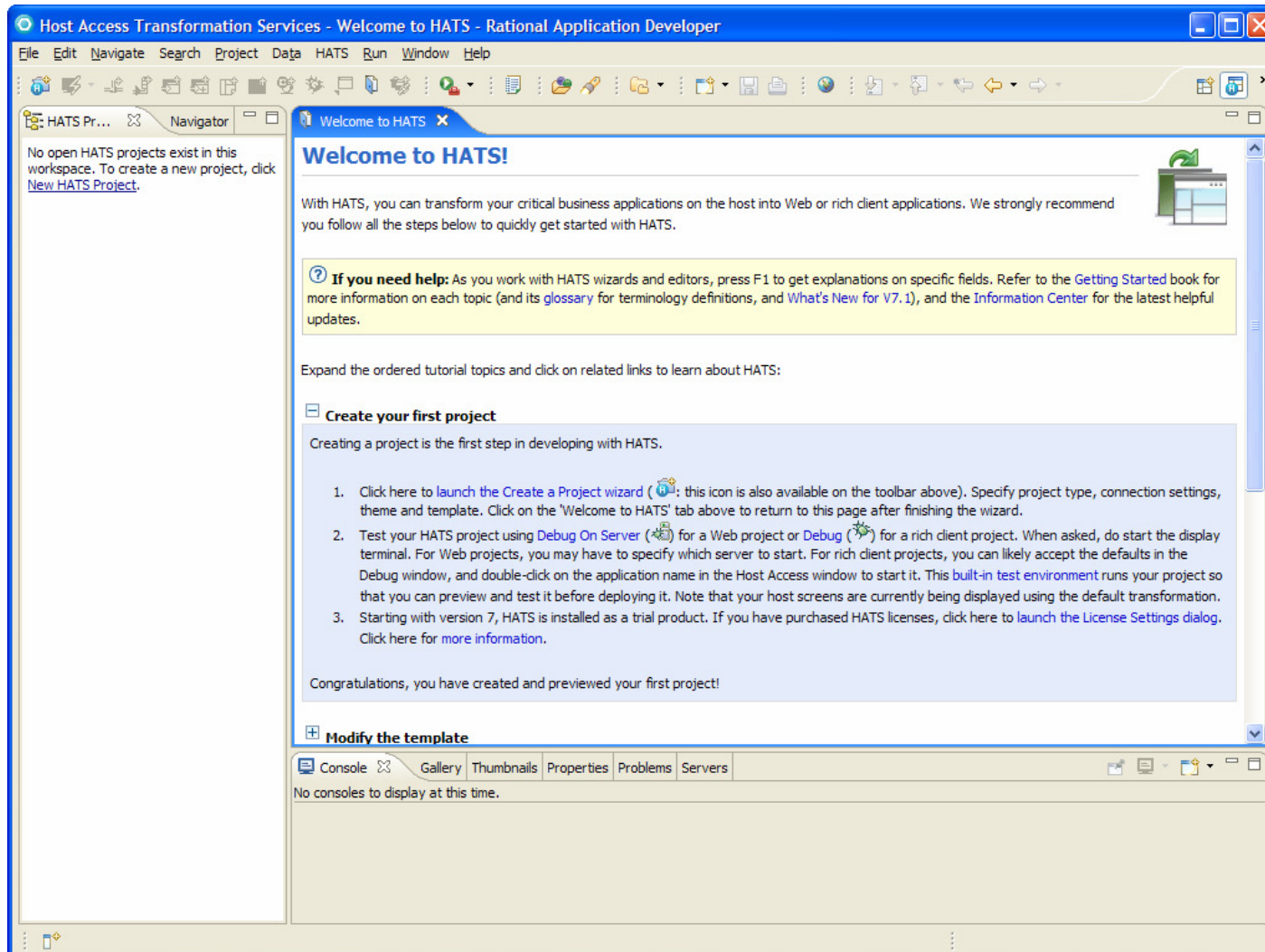
Reset
Default
Refresh
Disconnect
Turn Keyboard Off

Help | Set 2 | Exit | Profile | Refresh | Fuzzy | Forward | Command | Actions | Dist list | Cancel

Simple screen transformation demonstration



HATS Perspective



HATS New Project Wizard

Create a Project
✕

HATS Project

A HATS project is a set of resources used to transform host terminal applications.

Name:

Description (optional):

Location: Use default location

Deployment

Select a deployment environment and configure options:

Web

Target server:

Enterprise application project:

Portlet API:

Optimize options for mobile devices

Add administrative console support

Rich client

Plug-in ID:

Plug-in version:

Target platform:

Press F1 for help on any field in this or any other HATS wizard.

?
< Back
Next >
Finish
Cancel



Connection Settings Page

Create a Project - sample3270

Connection Settings

Configure host connection settings for this new project. You can specify advanced connection values later in the connection editor.

Host name:

Type:

Port:

Code page:

Screen size:

Add light pen rendering support



Theme Selection Page

Create a Project - sample3270

Project Theme

A theme groups a set of common appearance and behavior characteristics for a project. These attributes can be individually modified later.

Appearance and behavior theme

Standard
 Modern Web application
 Classic terminal emulator
 Custom

Settings:

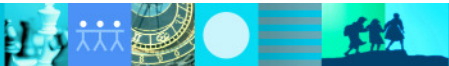
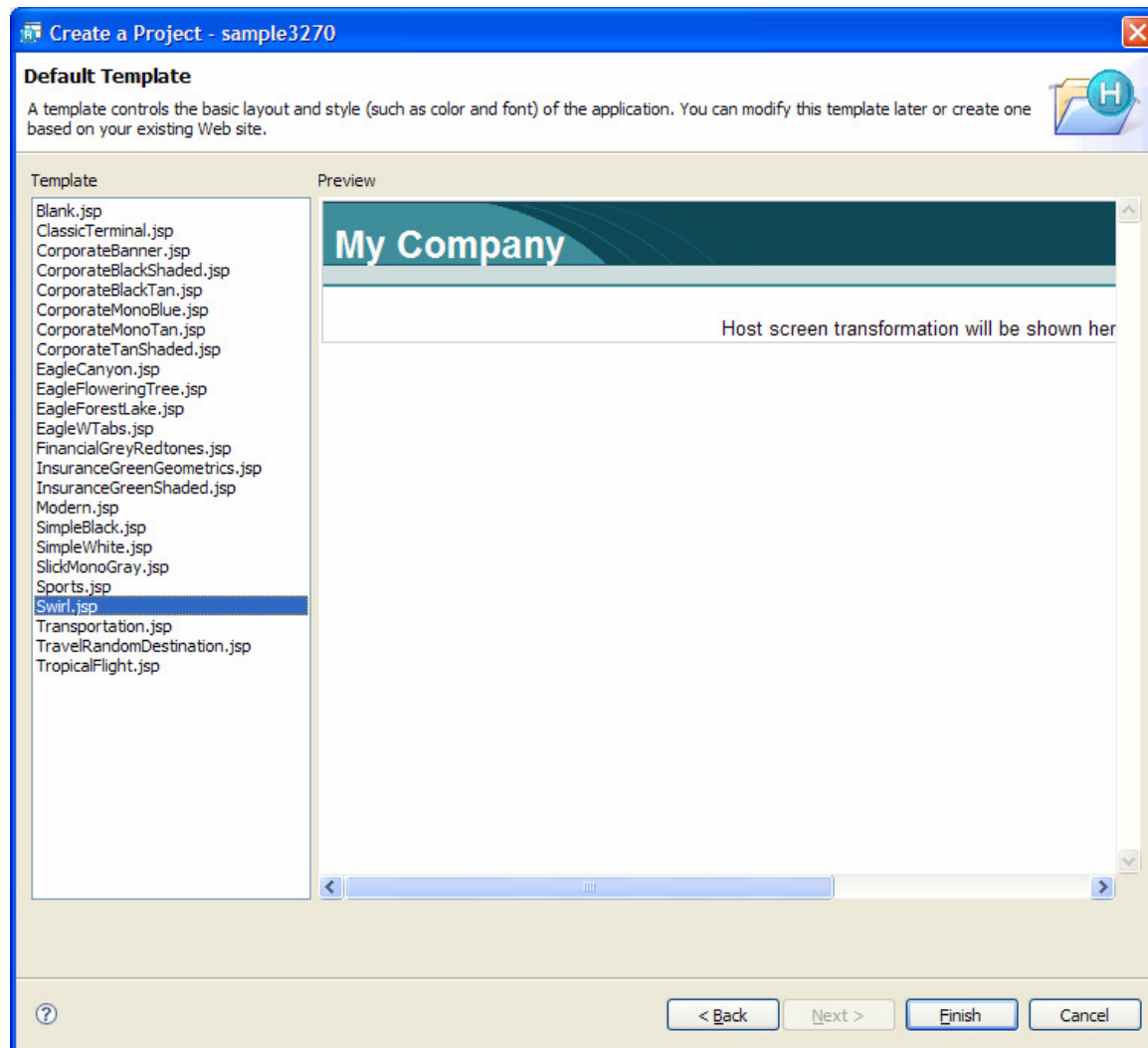
- Automatic field advance
- Cursor positioning on protected fields
- Field extended attributes
- Host keypad
- Keyboard support
- Operator information area
- Overwrite mode (initial)
- Rendering using monospaced font

Description:

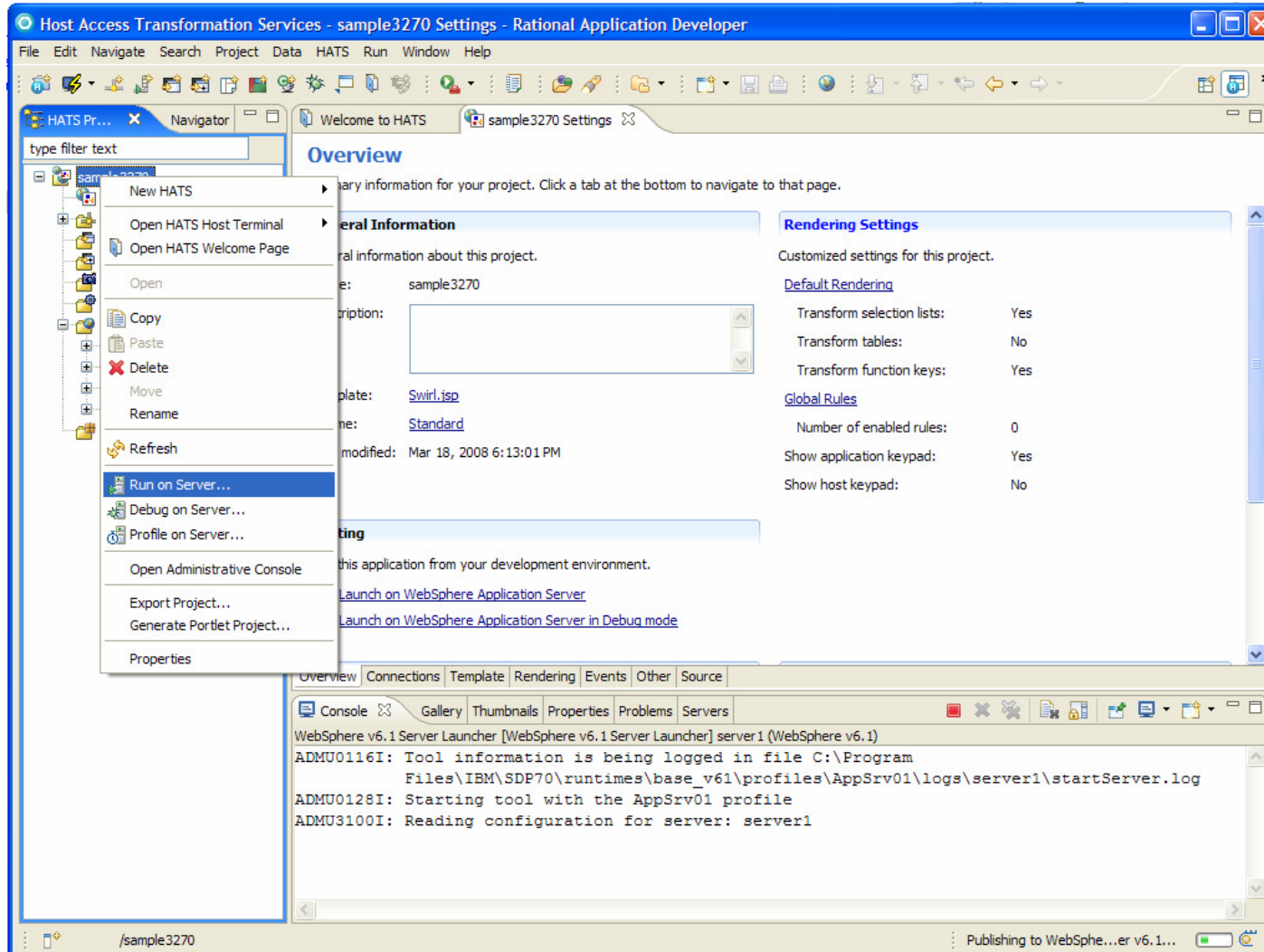
Move the cursor to the next input field if the cursor is located at the end of the current input field. If auto-advance is disabled, the cursor doesn't go to the next input field until the user explicitly moves it.



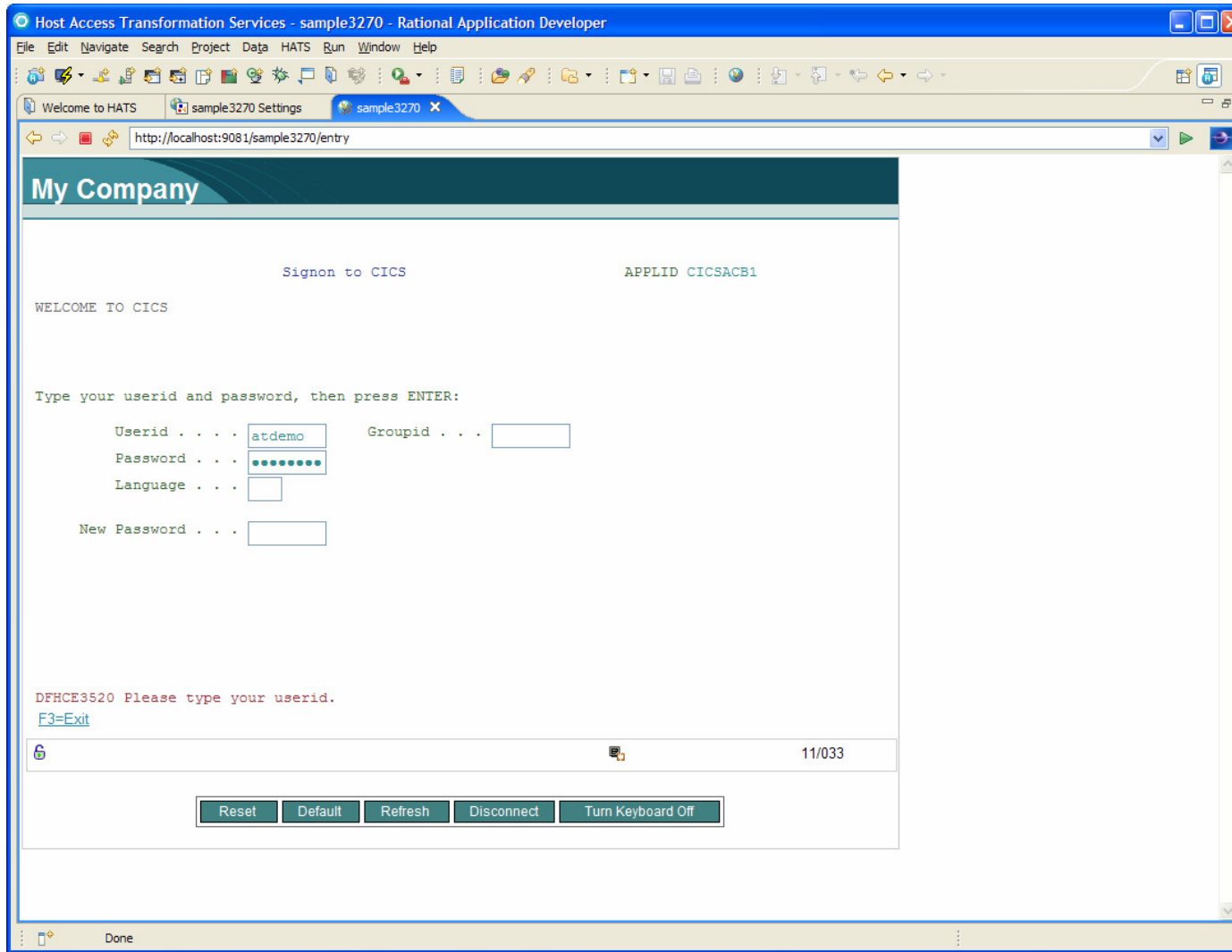
Template Selection Panel



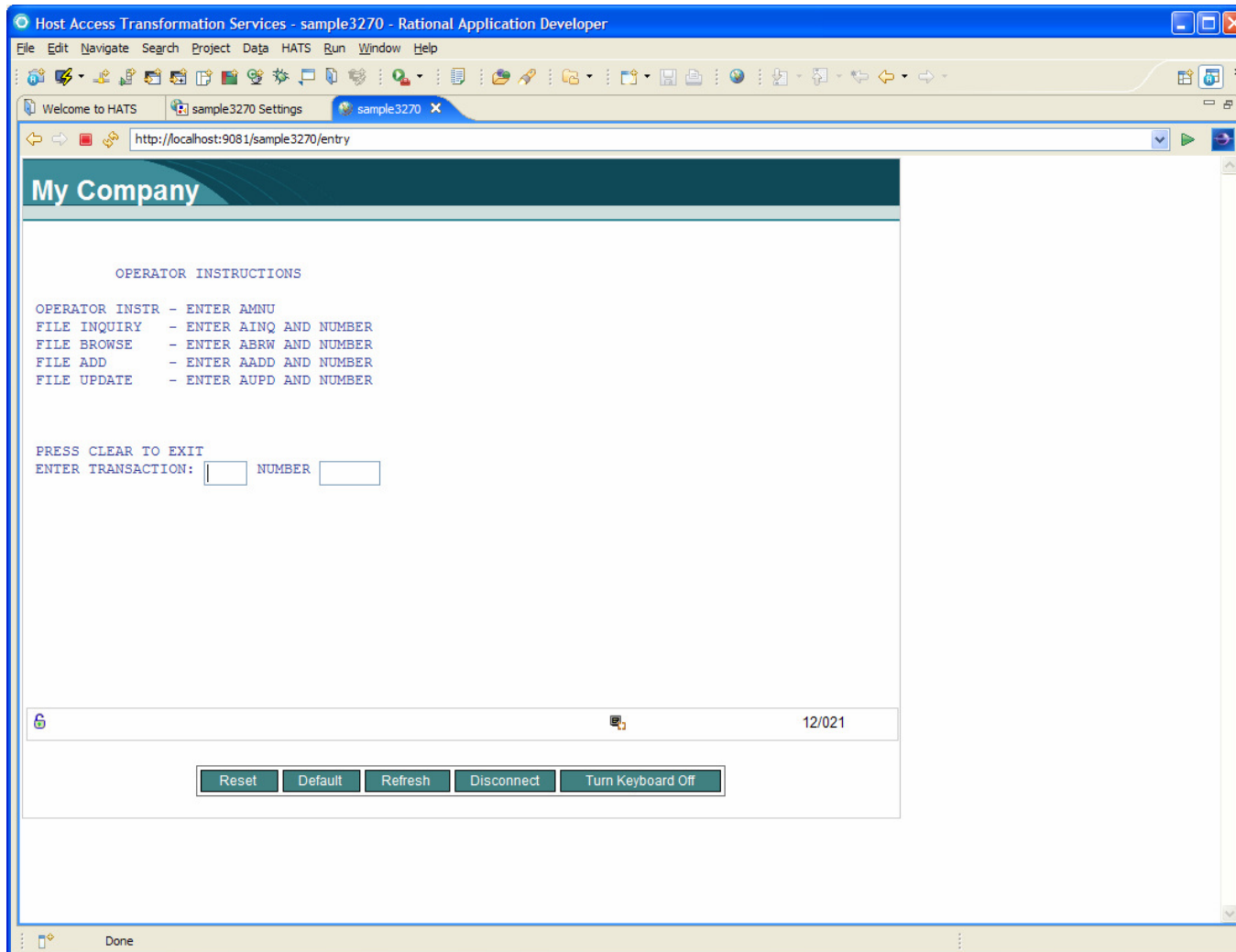
Project Creation / Run on Server



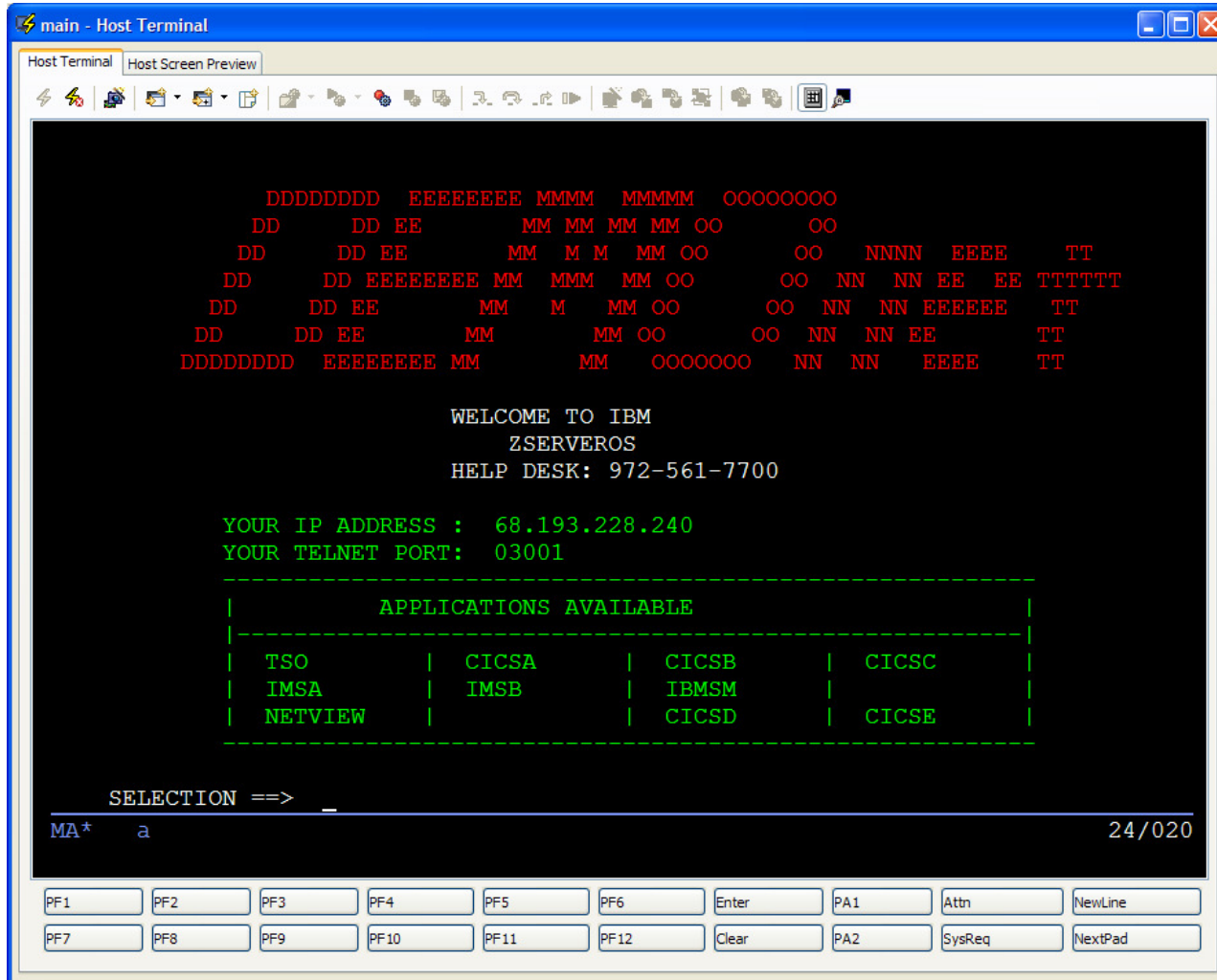
Default Rendering of Sign on Screen



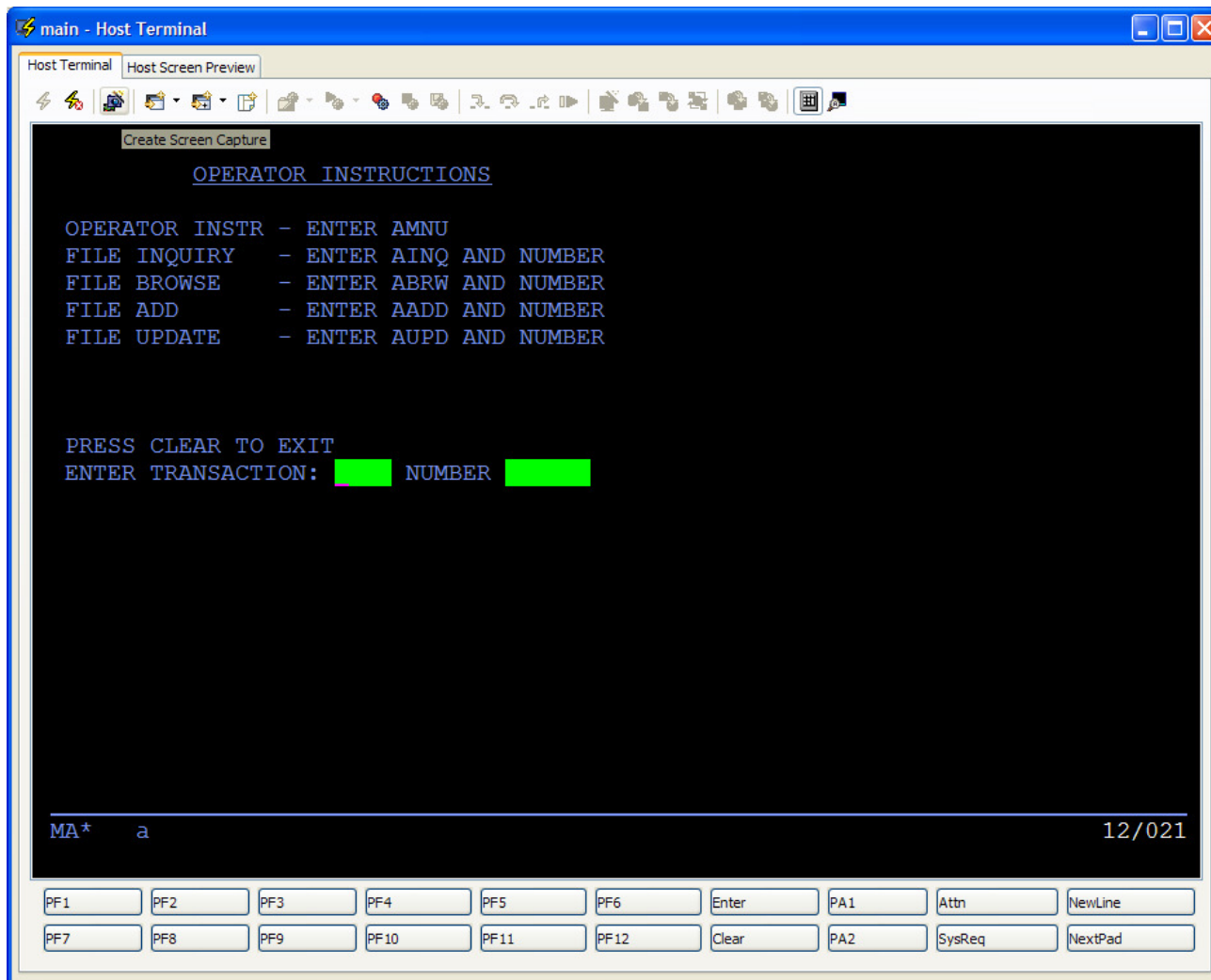
Default Rendering of Main Menu Screen



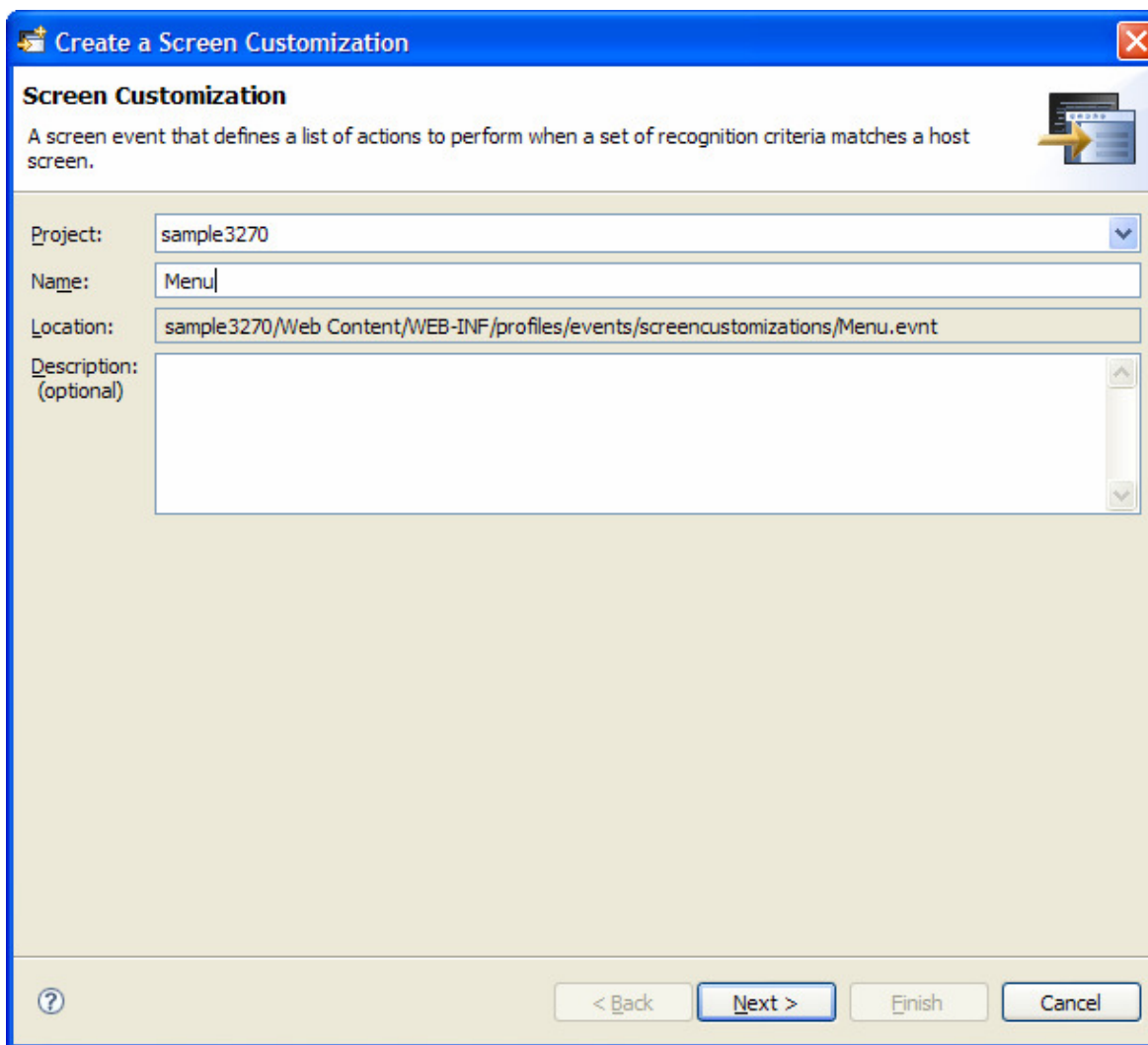
HATS Host Terminal – Sign On Screen



HATS Host Terminal – Menu Screen



New Screen Customization Wizard



The screenshot shows a dialog box titled "Create a Screen Customization" with a close button (X) in the top right corner. The dialog has a blue header bar. Below the header, the title "Screen Customization" is displayed in bold. A descriptive text reads: "A screen event that defines a list of actions to perform when a set of recognition criteria matches a host screen." To the right of this text is a small icon of a document with a yellow arrow pointing to it.

The main area of the dialog contains several input fields:

- Project:** A dropdown menu with "sample3270" selected.
- Name:** A text box containing "Menu".
- Location:** A text box containing "sample3270/Web Content/WEB-INF/profiles/events/screencustomizations/Menu.evnt".
- Description:** A large, empty text area with a vertical scrollbar on the right side. The label "Description: (optional)" is to the left of the text area.

At the bottom of the dialog, there is a row of buttons: a help button (question mark icon), a "< Back" button, a "Next >" button (which is highlighted with a blue border), an "Finish" button, and a "Cancel" button.



Screen Recognition Panel

Create a Screen Customization - Menu

Screen Recognition Criteria

Specify how to identify the screen. You can make the criteria more or less specific in order to match one or many screens. You can add additional criteria in the screen customization editor.

Select a screen source

Use the host terminal screen

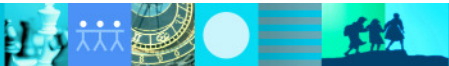
Use a previously captured screen

Screen: [Dropdown]

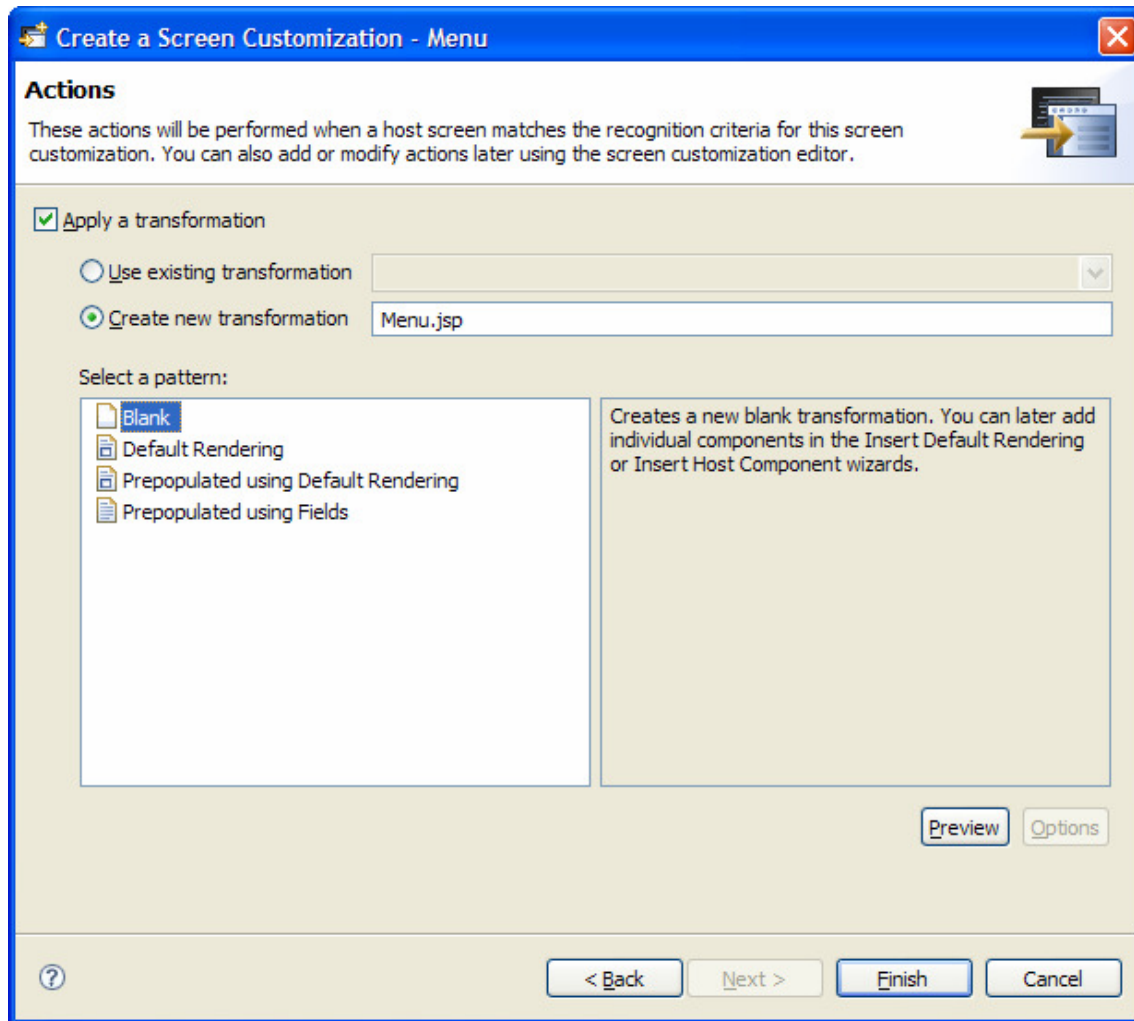
```
OPERATOR INSTRUCTIONS
OPERATOR INSTR - ENTER AMNU
FILE INQUIRY - ENTER AINQ AND NUMBER
FILE BROWSE - ENTER ABRW AND NUMBER
FILE ADD - ENTER AADD AND NUMBER
FILE UPDATE - ENTER AUPD AND NUMBER
PRESS CLEAR TO EXIT
ENTER TRANSACTION: [redacted] NUMBER [redacted]
```

Show Advanced >>

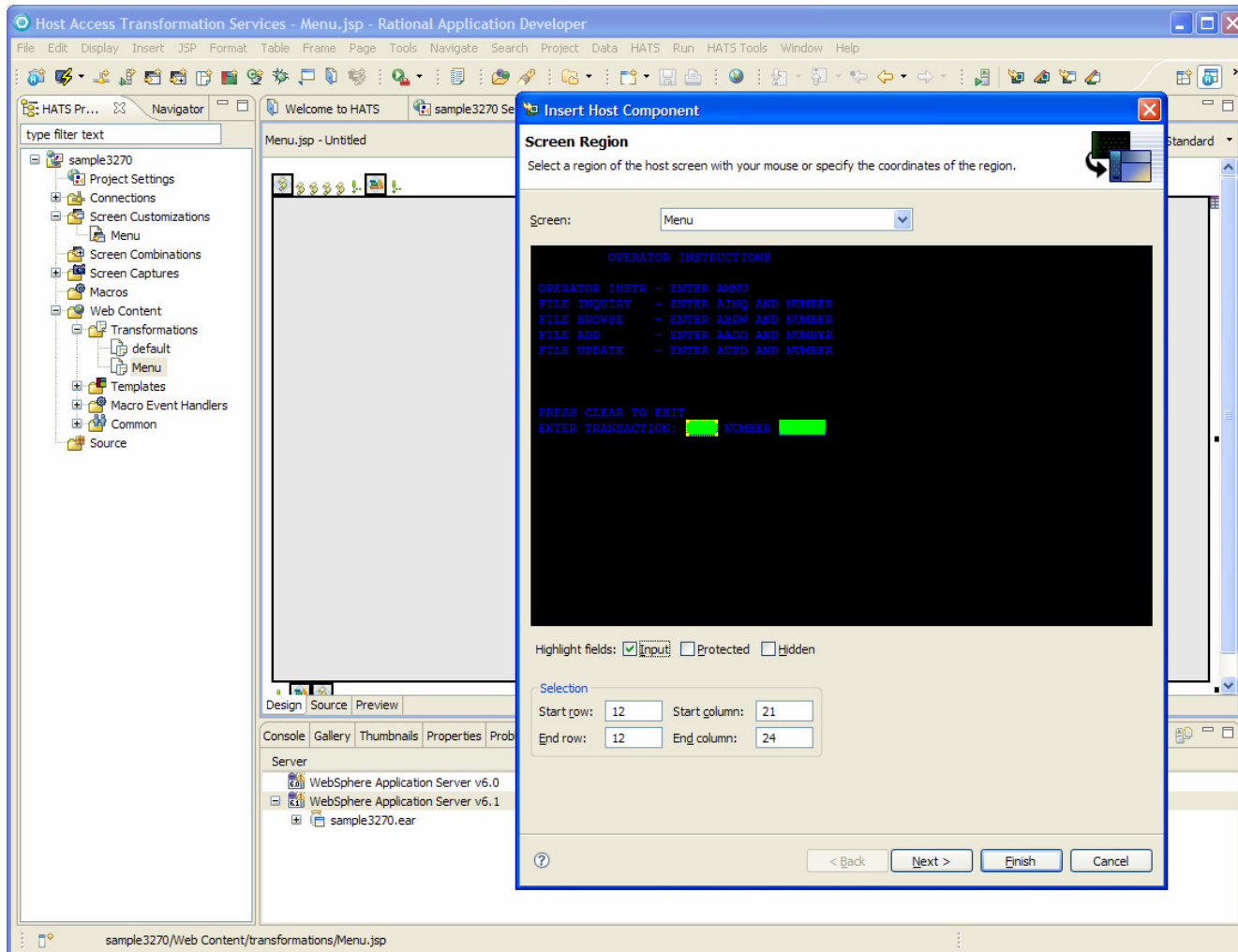
? < Back Next > Finish Cancel



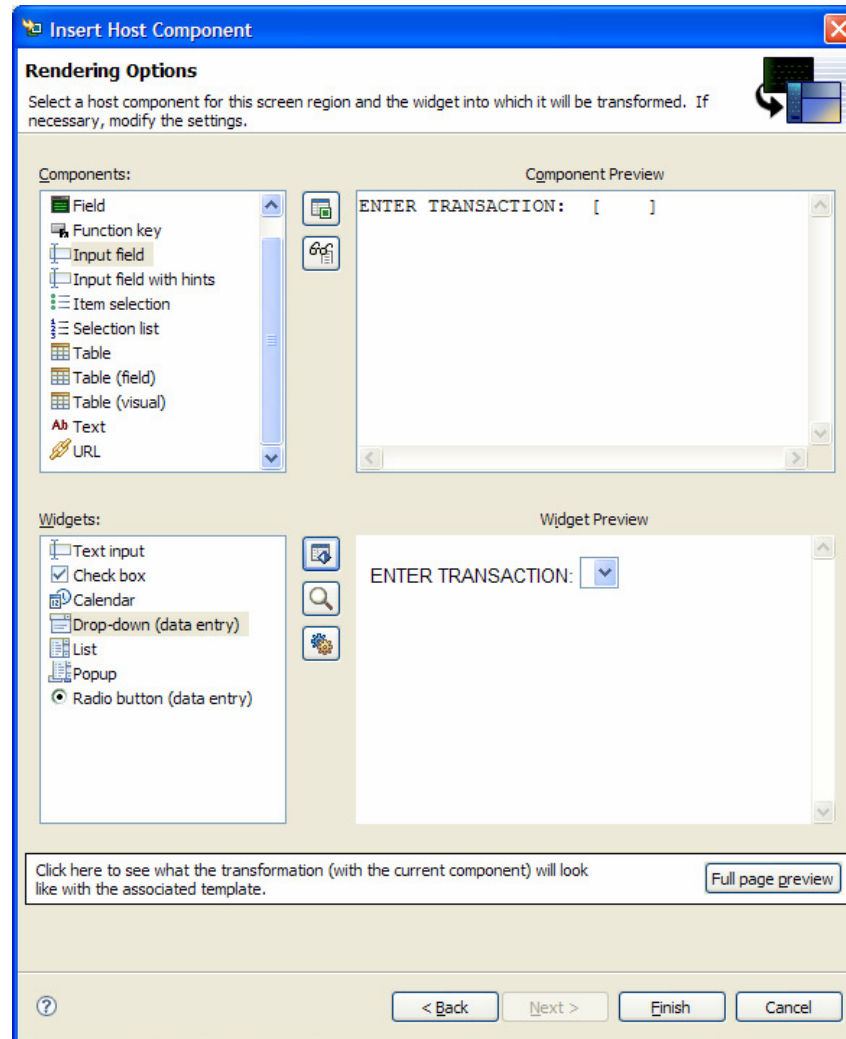
Screen Actions Panel



Transformation Editor



Component / Widget Selection Panel



Drop-down Widget Settings Panel

Settings - Drop-down (data entry)

Use project defaults

Fill from global variable

Global variable containing list values:

Shared

Global variable containing list captions (optional):

Fill from string

List items (example: Item 1=ITEM1KEY;Item 2):

Caption source:

Custom caption:

Number of columns per row:

Auto submit on select

Show submit button

Submit button caption:

Drop-down style class:

List option style class:

Caption style class:

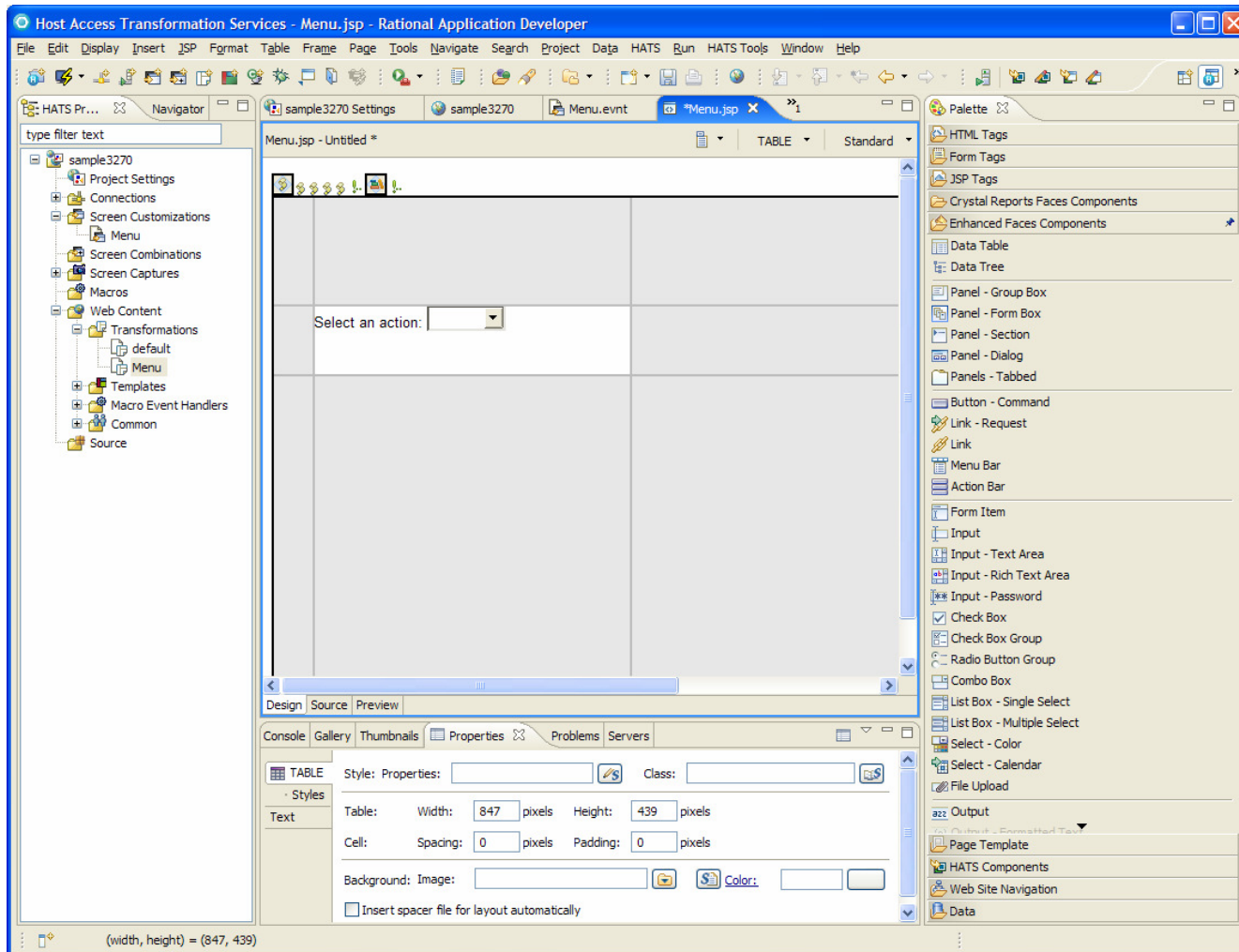
Table style class:

Style:

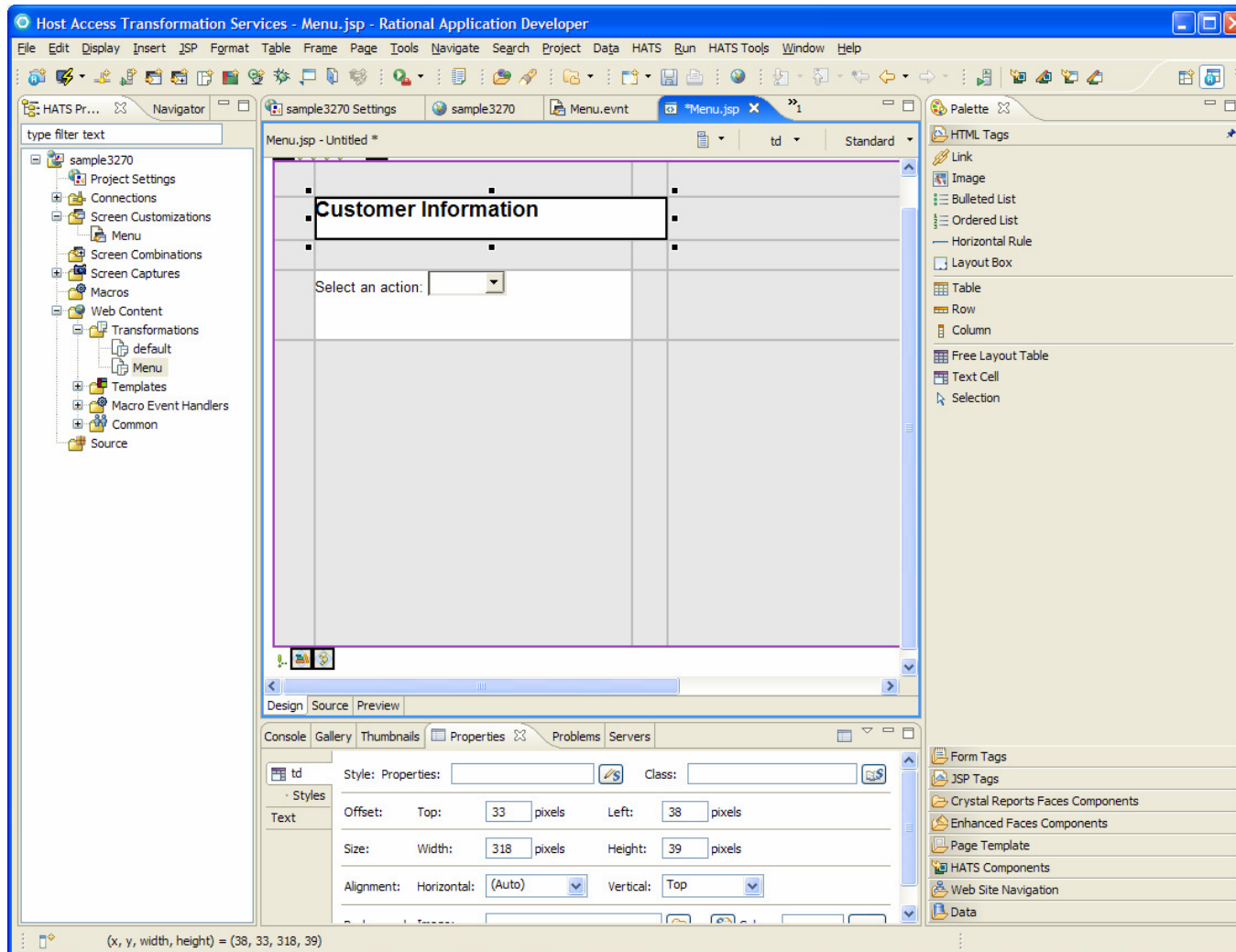
Select an action:



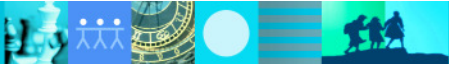
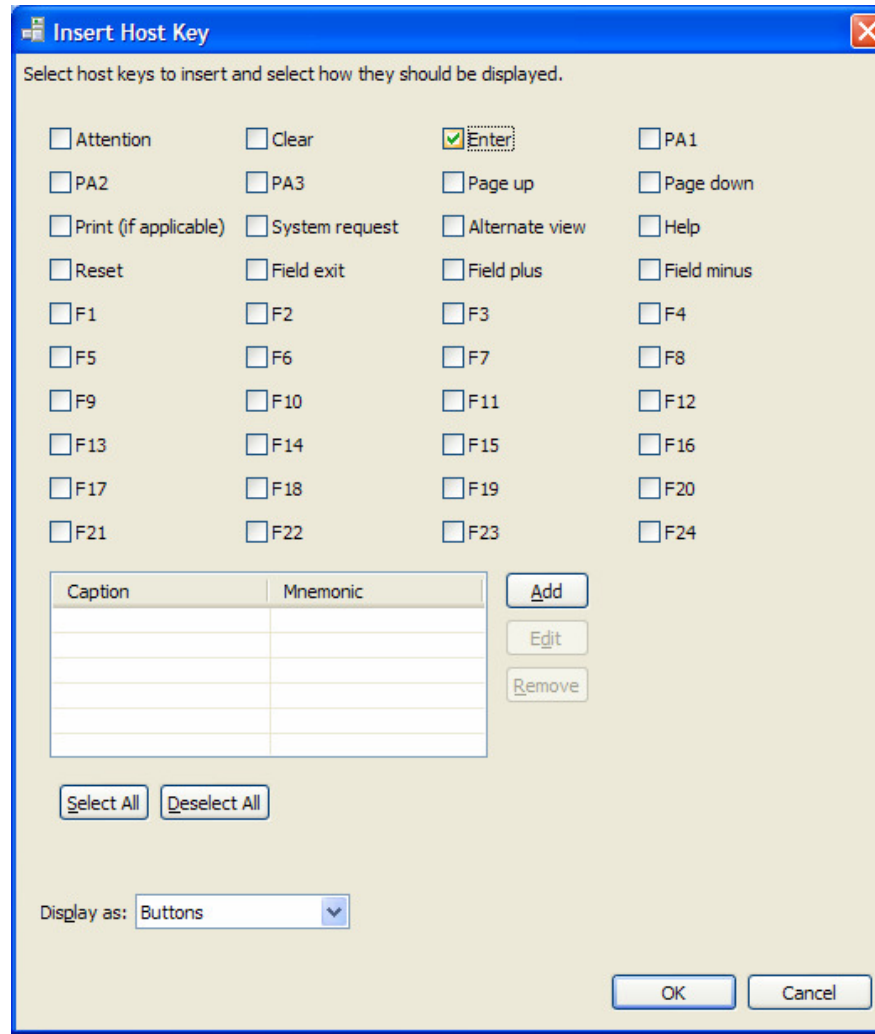
Transformation Editor



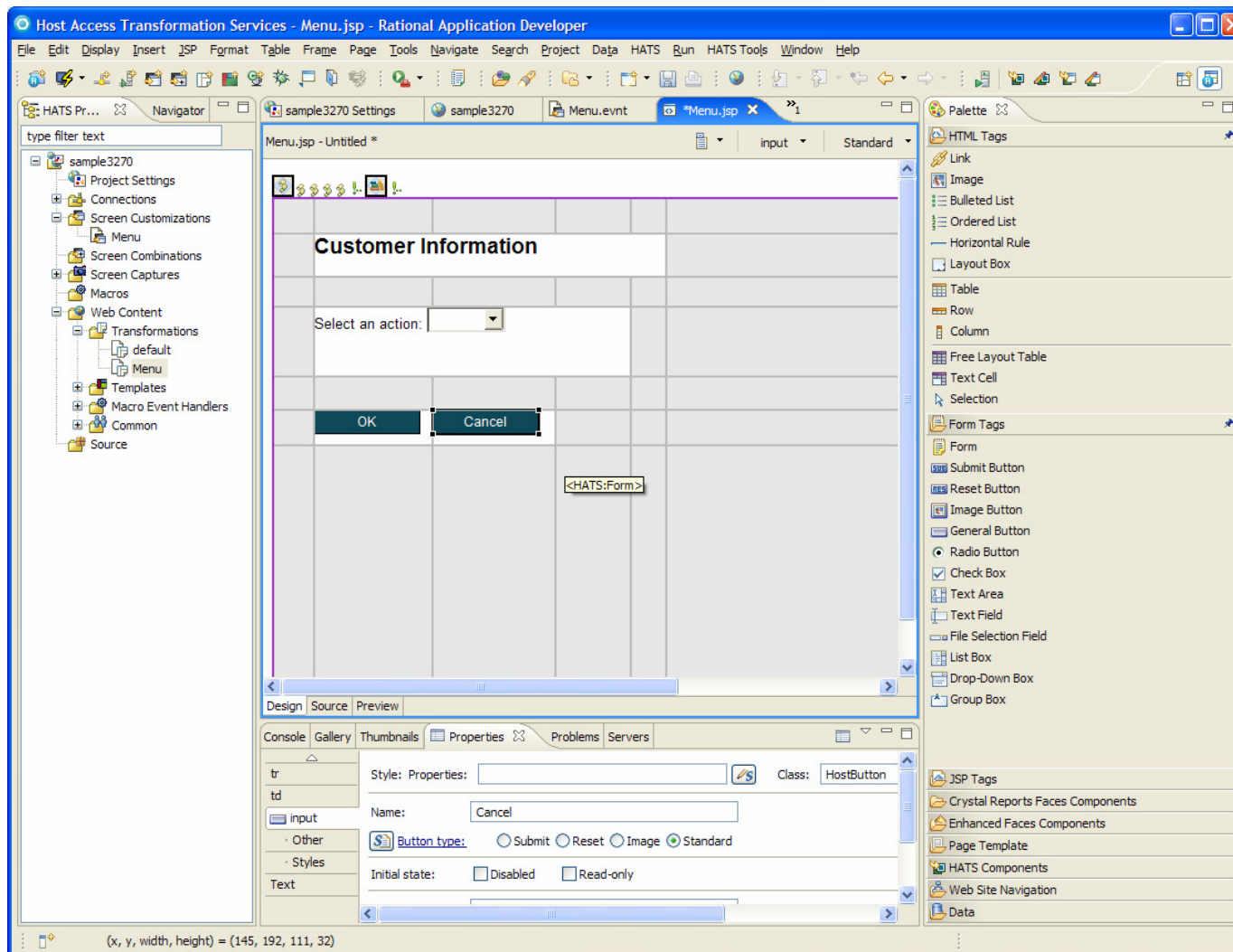
Transformation Editor – Adding a Label



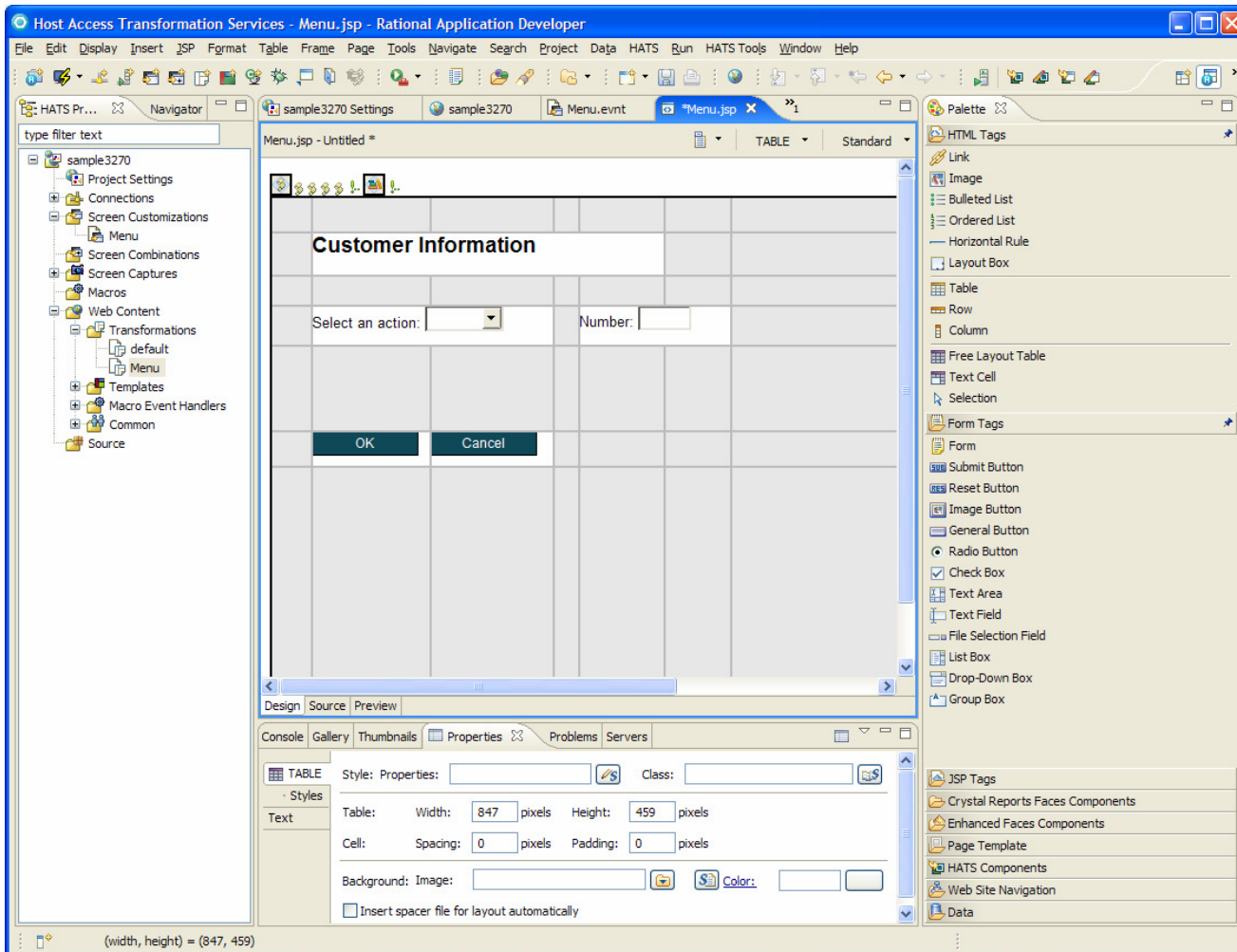
Transformation Editor – Adding an OK Button



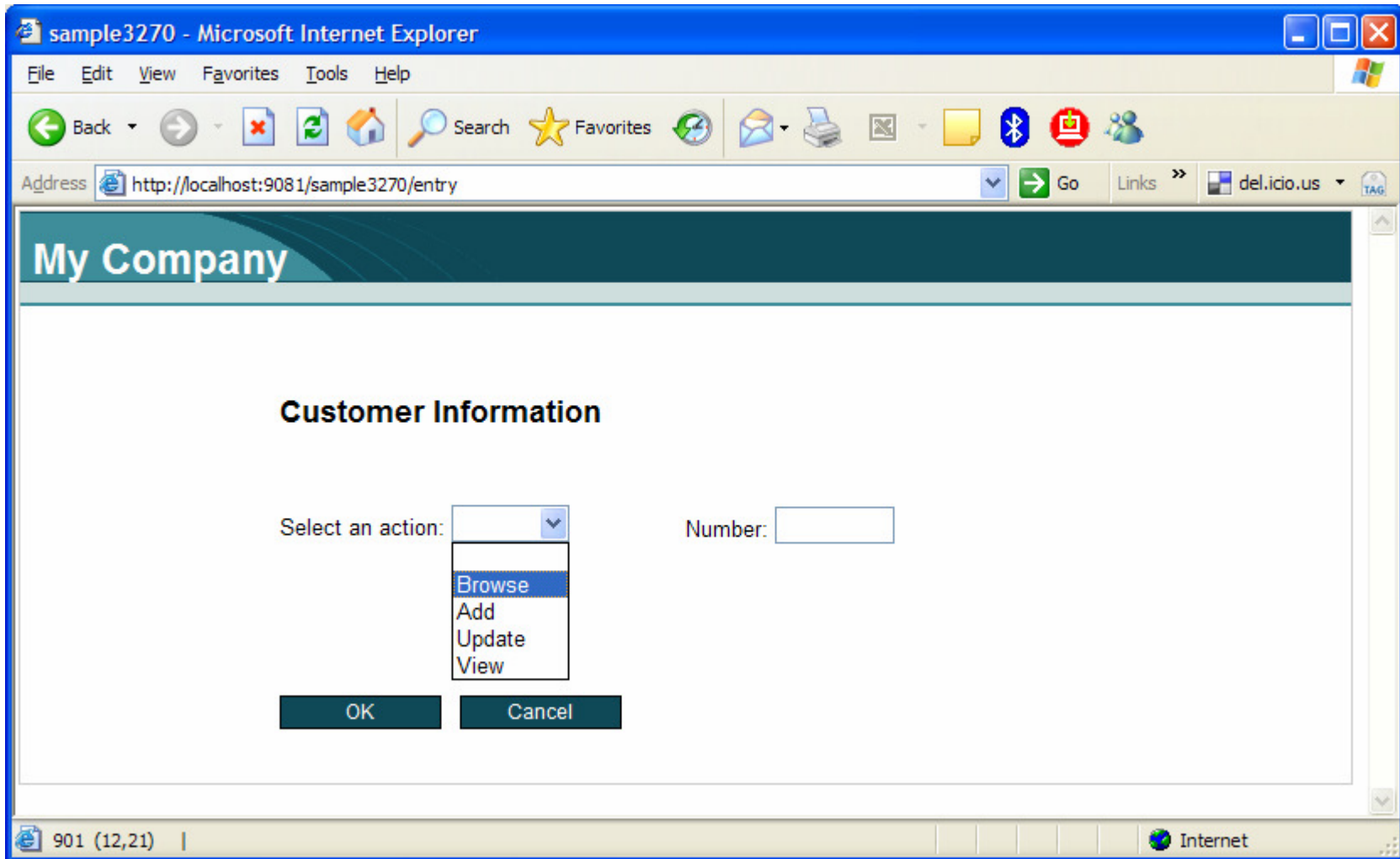
Transformation Editor – Adding a Cancel Button



Transformation Editor – Adding Number Field

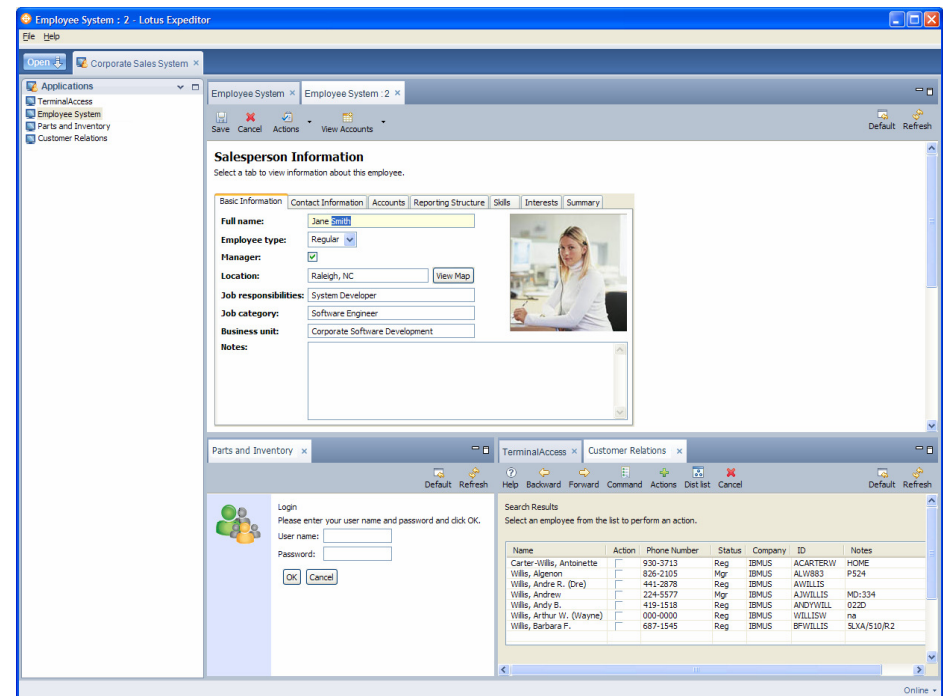


Results



HATS Rich Client Overview

- Extends host applications to the desktop. Applications run on:
 - ▶ Eclipse RCP 3.2 or 3.3
 - ▶ IBM Lotus Expeditor Client 6.1.x
- Provide a modern GUI interface to mission critical green screen applications.
- Processing occurs on the desktop (WebSphere Application Server is not required)
- Provides the capability to integrate host access applications with other applications on the desktop.
- Alternative to traditional terminal emulator products.
- Targeted at internal users in a controlled environment.



Comparing HATS Rich Client to Traditional Emulators

- What are the advantages of HATS RCP over a traditional “fat client” emulator?
 - ▶ Instant productivity improvements for end users (for example, automatic conversion of function keys into clickable toolbar buttons)
 - ▶ Information integration - multiple sources of information can be integrated and presented to the end user (increased productivity / less switching)
 - ▶ Platform support for automatic installation and updates
 - ▶ Platform independence (same HATS application can run on Windows and Linux)
 - ▶ Provides more control over what an end user can do
 - ▶ More attractive and usable to users not accustomed to a “green screen” or a terminal emulator
- Why is this important?
 - ▶ Allows you to continue using your proven, reliable terminal applications without costly rewrites, code changes, or major disruption.



Comparing HATS Rich Client to HATS Web

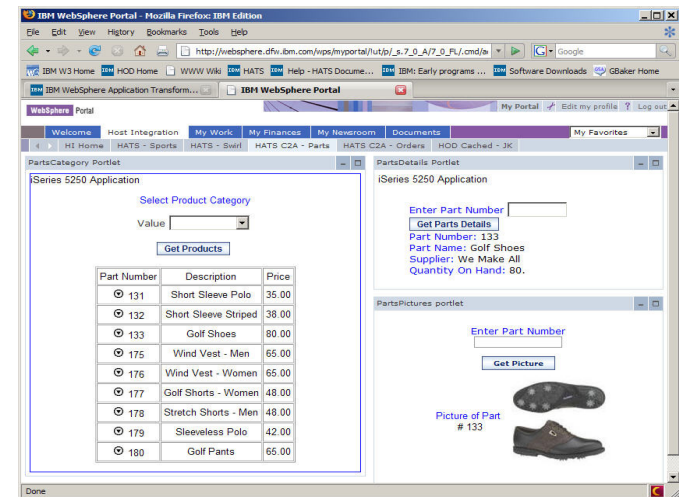
- What are the advantages of HATS RCP over HATS Web?
 - ▶ Integration at the desktop with other applications
 - ▶ Client side processing (no server bottlenecks) – no server required
 - ▶ Better response time (good for slow speed links)
 - ▶ Less network traffic
 - ▶ Rich set of user interface (UI) widgets
 - ▶ 3270e print directly to end user's printer
- HATS RCP is primarily targeted at internal users who need integration with other desktop applications or are looking for responses times more similar to that of a traditional emulator.
- End-user keyboard remapping support, and support for numeric keypad field exit.
- End users can easily work with multiple instances of the same application.
- Asynchronous updates supported (but not an applet)
- More flexibility in how a transformation is presented and validated.



Built on Rational, WebSphere, and Lotus Technologies

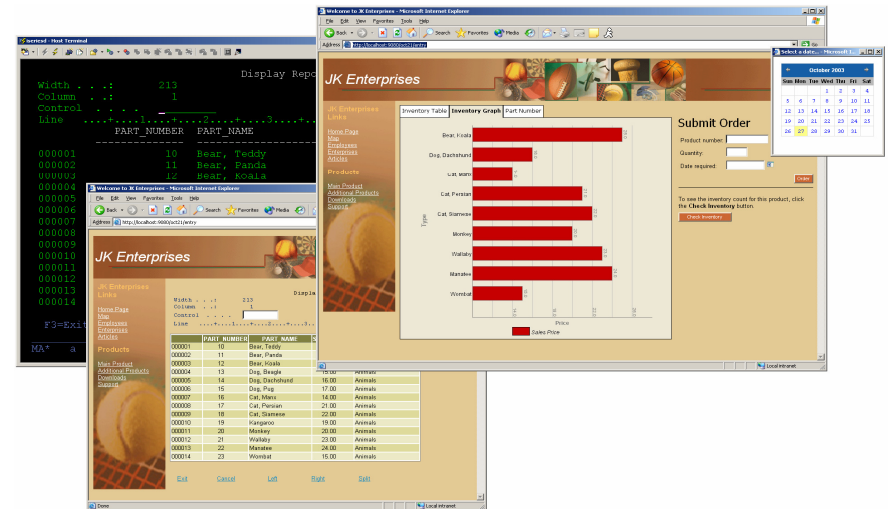
- IBM Rational Software Development Platform Tool
 - ▶ HATS Toolkit is an Eclipse plug-in to the IBM Rational Software Delivery Platform Tool
 - ▶ Wizard-based
 - ▶ Creates HATS applications
- WebSphere Application Server, WebSphere Portal Server
 - ▶ HATS leverages WebSphere Application Server (WAS) or WebSphere Portal Server at runtime
 - ▶ HATS Toolkit creates J2EE .war and .ear files for deployment
 - ▶ HATS runtime is contained in the HATS application
- Lotus Expeditor, Eclipse Rich Client
 - ▶ Open, standards-based platform
 - ▶ HATS runs on the client machine, not on a server

Building the technical embedded software that controls devices



Summary of HATS Features and Value Propositions

- Host screens are converted to GUIs on the fly, in real time
- Improved navigation of host application
- Zero footprint on the desktop with Web deployment
- Support deployment to Rich Client
- Access host assets from mobile devices
- Combine screens and data from multiple applications
- Eliminates need to customize every screen
- Generate Web services from host transactions
- Exploits security & scalability of WebSphere Application Server and WebSphere Portal
- Extensible through IBM Rational Software Delivery Platform
- Supports iterative development



HATS 7.1 Release Highlights

- Rebranding
- Mobile device support
- JSR 168 portlet support
- Improved SOA tooling
- Other new features
 - ▶ Arrow key navigation support (rich client only)
 - ▶ Automatic setting of WAR class loader policy
 - ▶ New Remove Global Variable action
- Dates:
 - ▶ 3/11/2008 – Announcement
 - ▶ 3/28/2008 – eGA
 - ▶ 4/10/2008 – GA



HATS 7.1 - Rename/Rebrand/Reprice/Repackage

- **Rename**

- ▶ From WDHT to Host Access Transformation Services for 5250 Applications

- **Rebrand**

- ▶ From WebSphere to Rational

- **Reprice**

- ▶ From Multiple Tier prices to Standard Value Unit measures

- **Repackage**

- ▶ WDSC and WDSC AE will not advance – no future release to include WebFacing Tool
- ▶ WebFacing Tool will be an installable option in HATS Toolkit



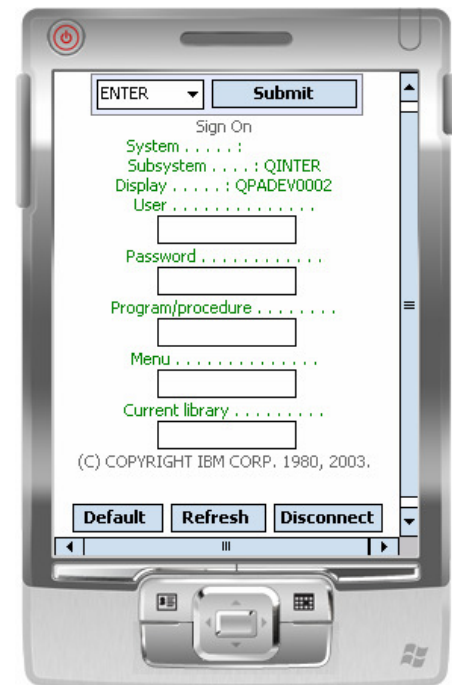
Packaging

- **HATS for Multiplatforms and HATS for Linux on zSeries**
 - ▶ Available for use with 3270 and 5250 applications
 - ▶ HATS applications can be deployed for Web, portal, rich client and mobile access.
 - ▶ HATS applications can be deployed on any supported platform (refer to server runtime requirements)
 - ▶ Ships with a restricted WebSphere Application Server (WAS) Network Deployment license
- **HATS for 5250 applications on Multiplatforms**
 - ▶ Available for use with 5250 applications only
 - ▶ HATS applications can be deployed for Web, portal, rich client, and mobile access
 - ▶ HATS applications can be deployed on any supported platform with the exception of i5/OS (refer to server runtime requirements)
 - ▶ Ships with a restricted WebSphere Application Server (WAS) - Express license
- **HATS for 5250 applications on i5/OS**
 - ▶ Available for use with 5250 applications only
 - ▶ HATS applications can be deployed for Web, portal, rich client, and mobile access
 - ▶ HATS applications can be deployed on i5/OS (refer to server runtime requirements)
- **HATS is also included as a component of Host Integration Solution (HIS)**



What is HATS mobile device support for v7.1?

- HATS transformation applications can now be accessed by pervasive devices such as cellular phones, data collection terminals, and Personal Digital Assistants
- This allows for the HATS developer to:
 - ▶ Customize and transform individual screens
 - ▶ Add business logic to screen interactions
 - ▶ Enhance the presentation, layout, and flow of the original host systems
- Supported browser: Internet Explorer Mobile 5.0 or later
- Reminder: HATS Integration Object or Web service data can also be used in conjunction with mobile devices



Compact Default Rendering Example

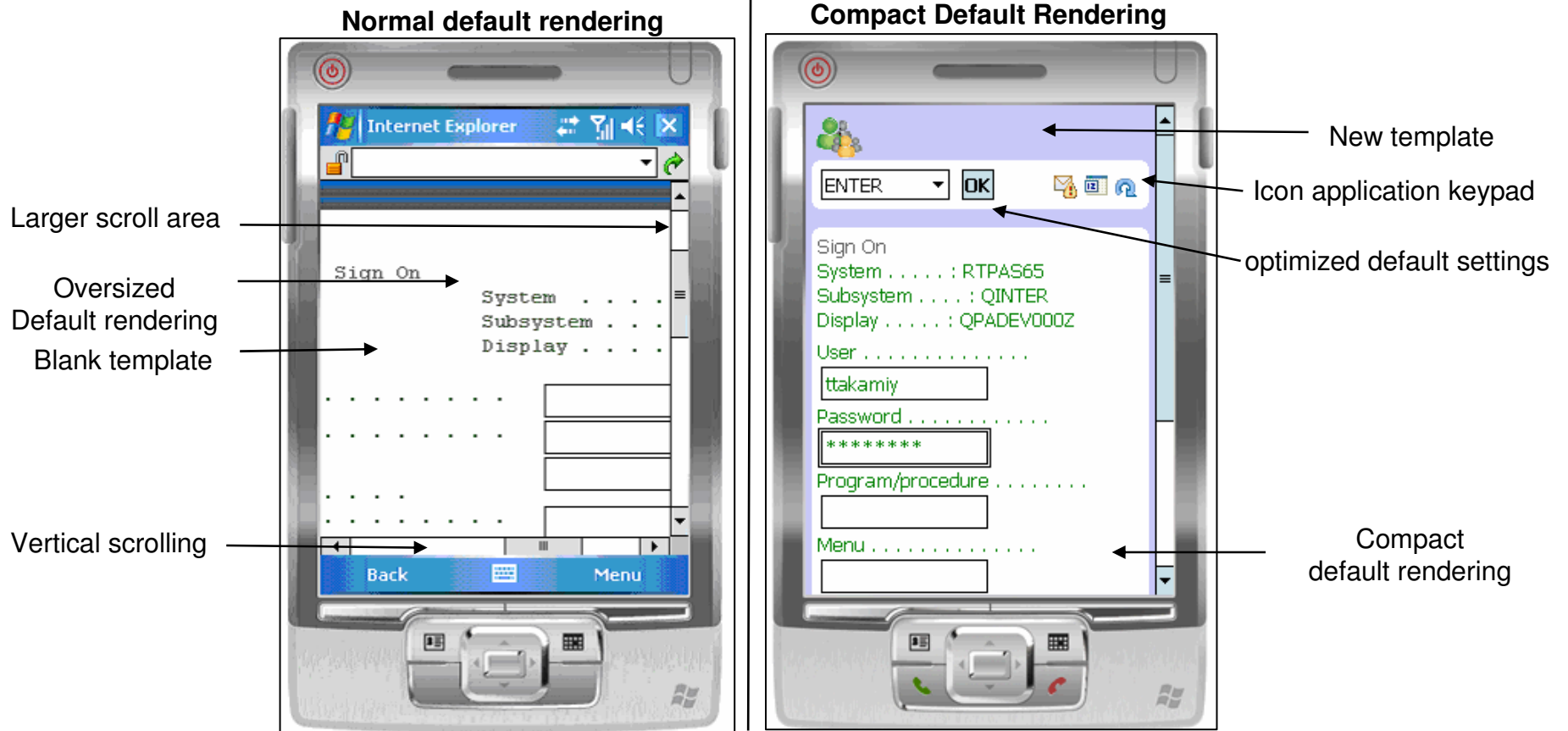


Table and Subfile column arrangement enhancement

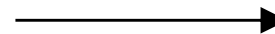
- This enhancement enables tables to be collapsed and expanded so that more data can fit in the table or subfile representation without resulting in horizontal scrolling.
- As an additional option, the table and subfile widgets will create the expandable areas for retrieval by the browser only when requested
 - ▶ Uses AJAX technology
 - ▶ Reduces the memory and size footprint on the client device
- The feature can also be used in standard HATS Web applications.

2=Change 3=Hold 4=End 5=Work with 6=Release 7=Display message
8=Work with spooled files 13=Disconnect ...

Opt	Subsystem/Job	User	Type	CPU %	Function	Status
▼	QBATCH	QSYS	SBS	.0		DEQW
▼	QCMN	QSYS	SBS	.0		DEQW
▼	QCTL	QSYS	SBS	.0		DEQW
▼	QSYSSCD	QPGMR	BCH	.0	PGM-QEZSCNEP	EVTW
▼	QHITPSVR	QSYS	SBS	.0		DEQW
▼	ADMIN	QTMHHTTP	BCH	.0	PGM-QZHBMAIN	SIGW
▼	ADMIN	QTMHHTTP	BCI	.0	PGM-QZSRLOG	SIGW
▼	ADMIN	QTMHHTTP	BCI	.0	PGM-QZSRHTTP	SIGW
▼	QINTER	QSYS	SBS	.0		DEQW

More...

PF1 Submit



2=Change 3=Hold 4=End 5=Work with 6=Release 7=Display message 8=Work with spooled files 13=Disconnect ...

Opt	Subsystem/Job	
▼	QBATCH	details
▼	QCMN	hide
	User	QSYS
	Type	SBS
	CPU %	.0
	Function	
	Status	DEQW
▼	QCTL	details
▼	QSYSSCD	details
▼	QHITPSVR	details
▼	ADMIN	details
▼	ADMIN	details
▼	ADMIN	details
▼	QINTER	details

More...

PF1 Submit

JSR 168 Portlet Support

- IBM Portlet API deprecated in WebSphere Portal 6.0
 - ▶ No new functionality will be added
- JSR 168 Portlet Support allows:
 - ▶ Creation of JSR 168 portlet that can run in the following environments:
 - WebSphere Portal Server v5.1.x
 - WebSphere Portal Server v6.0.x
 - WebSphere Application Server (WAS) v6.1
 - WAS v6.1 is shipped with a JSR 168 portlet container that you can use to deploy and test your JSR 168 portlets. This portlet container has limited functionality compared to WebSphere Portal servers



The Value of HATS SOA

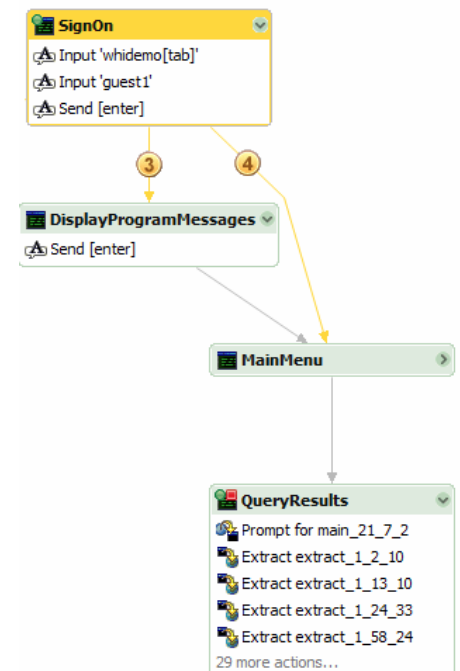


- **HATS is an important part of the IBM SOA strategy!**
 - ▶ IBM SOA Foundation helps to extend the value of the applications and business processes that currently run your business, not replace them.
- **HATS does not require application reengineering or rewriting**
 - ▶ Rewriting can consume significant amounts of time and resources.
 - ▶ Rewriting requires access to and understanding of application source code.
 - ▶ Rewriting inherently introduces new bugs that must be found and fixed.
 - ▶ Simply put - rewriting is not an option for some customers!
- **HATS is unique**
 - ▶ HATS is the only IBM product that Web Service-enables 3270, including CICS and IMS, 5250, and VT terminal applications – great for customers with a mix of applications (one tool can be used to enable all of them).
- **HATS generates standard Web Service interfaces**
 - ▶ HATS services can be integrated into an existing SOA.

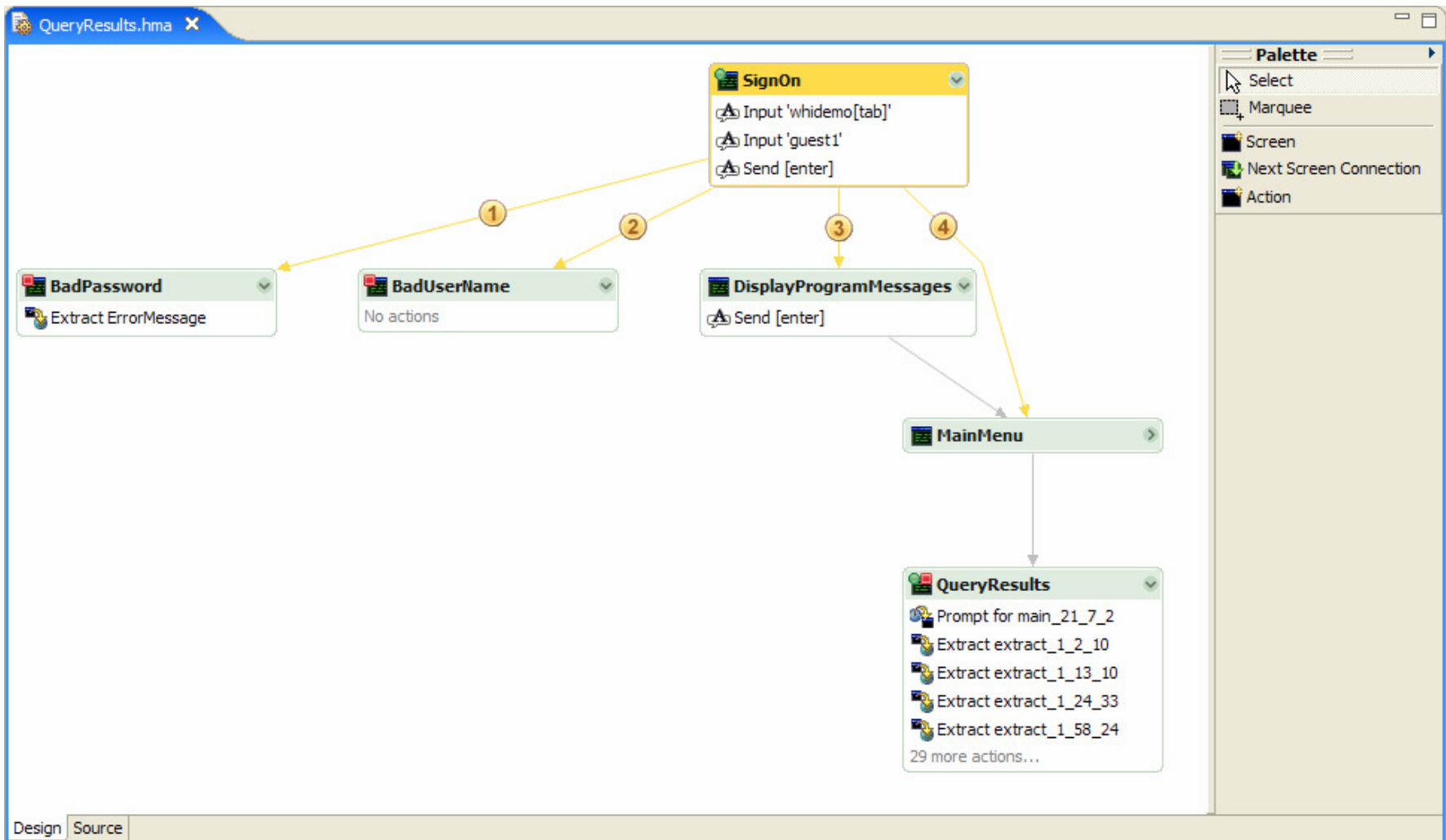


Visual Macro Editor (VME) Technology Preview

- The VME is tool for visually developing HATS macros
 - ▶ Macros are the scripts that navigate and interact with 3270, 5250, VT host applications.
 - ▶ Macros are the foundation for the Web Service support provided by HATS.
 - ▶ Inputs and outputs in a HATS macro are eventually manifested as inputs and outputs in a Web Service (screen interactions are hidden behind the Web Service interface).
- The VME provides value by:
 - ▶ Significantly increasing the productivity of HATS macro developers.
 - ▶ Allowing for easier handling of alternate / error flows.
 - ▶ Helping decrease the number of logic / flow problems (since the flow can be clearly analyzed by both developers, expert users, and business analysts).

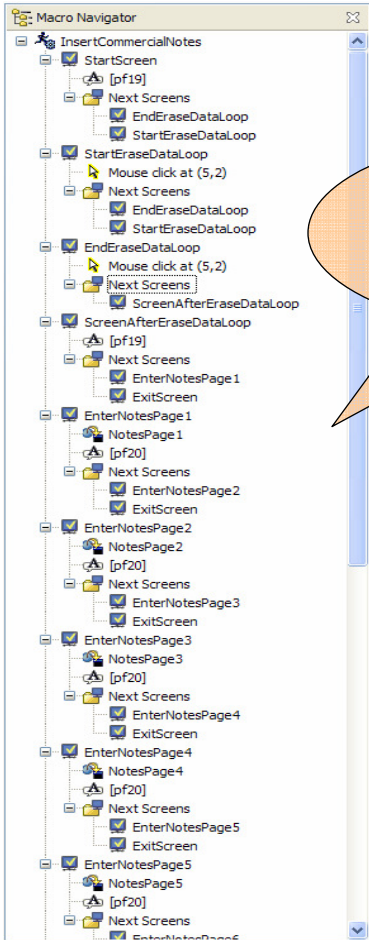


Sample (from HATS VME 7.1 M2)

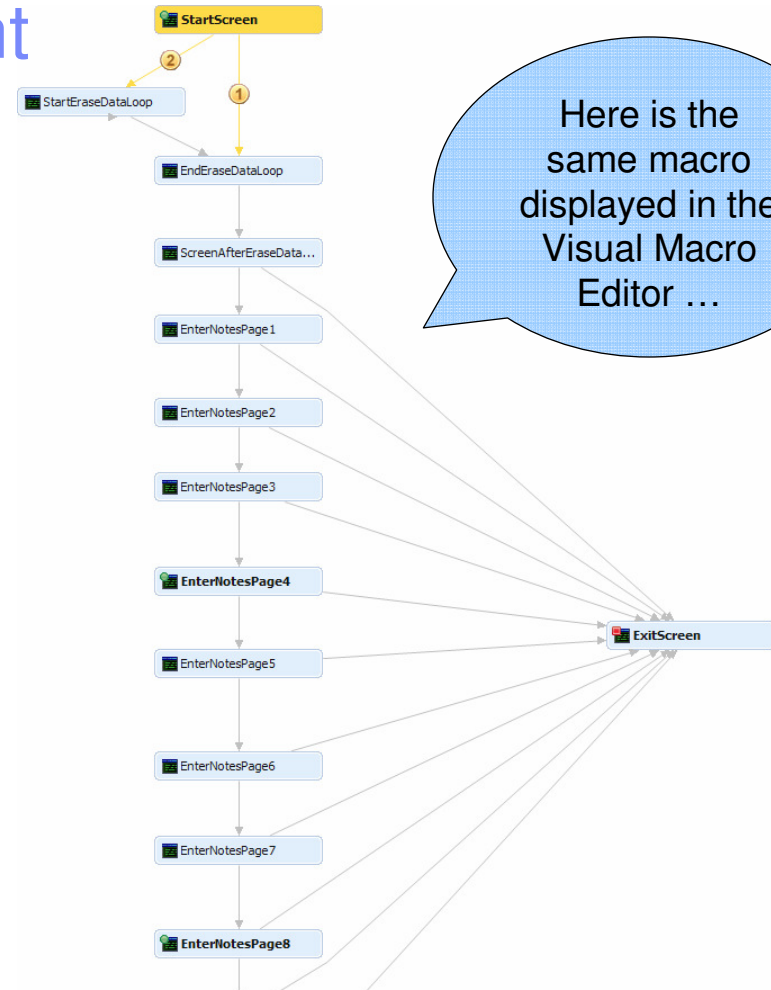


Why Visualization is Important

Current HATS macro tooling:



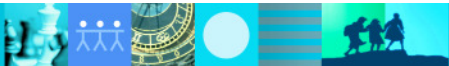
Can you *easily* determine the flow of this macro?



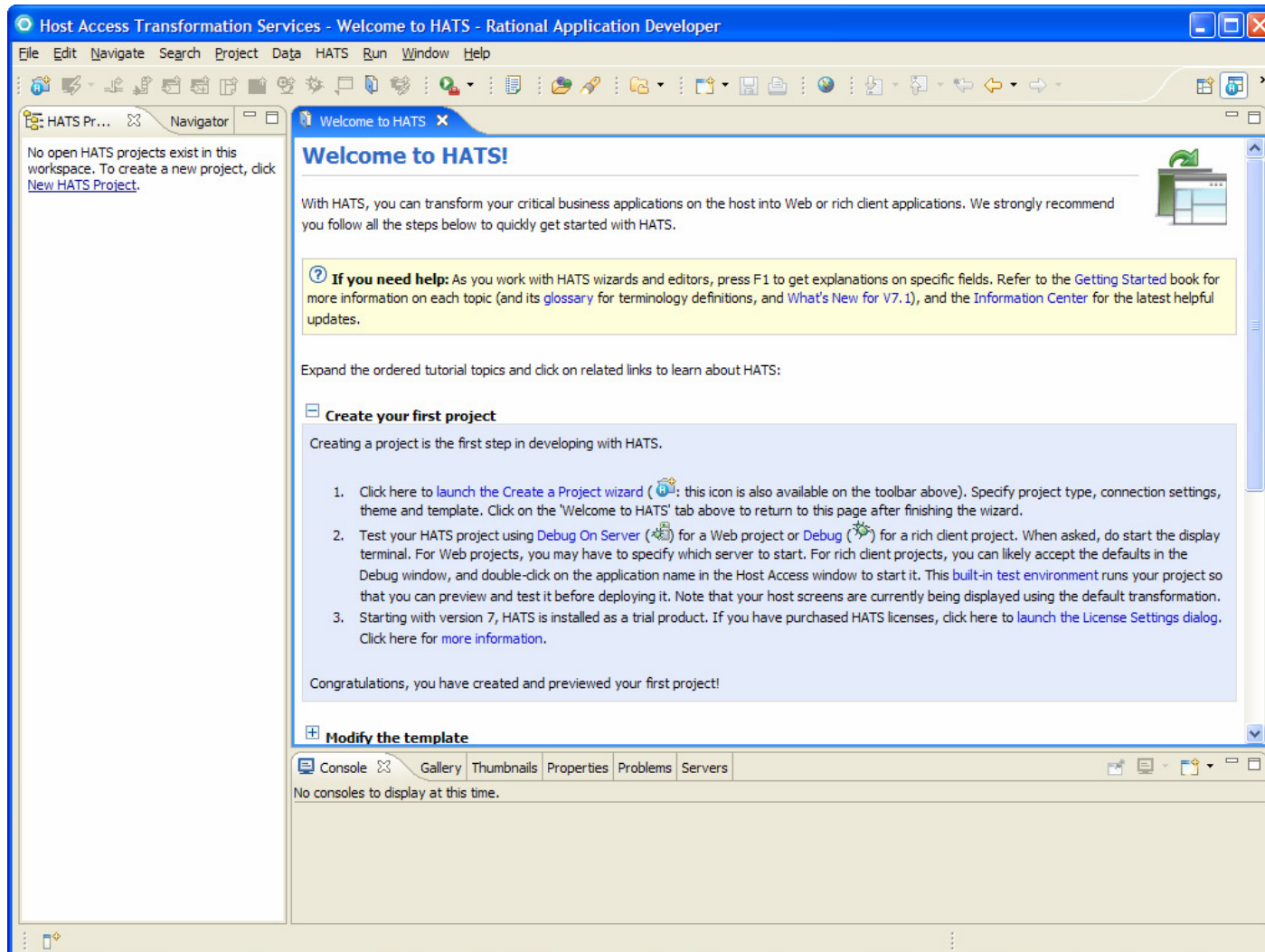
Here is the same macro displayed in the Visual Macro Editor ...

It is much easier to understand the logic of a macro and find logic errors and missing flows when you can see the entire macro!

Simple HATS Web Service Creation / Visual Macro Editor Demonstration



HATS Perspective



HATS New Project Wizard

Create a Project

HATS Project
 A HATS project is a set of resources used to transform host terminal applications.

Name:

Description: (optional)

Location: Use default location

Deployment
 Select a deployment environment and configure options:

Web

Target server:

Enterprise application project:

Portlet API:

Optimize options for mobile devices
 Add administrative console support

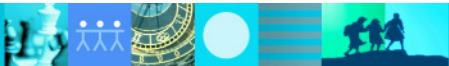
Rich client

Plug-in ID:

Plug-in version:

Target platform:

Press F1 for help on any field in this or any other HATS wizard.



Connection Settings Page

Create a Project - sample3270

Connection Settings

Configure host connection settings for this new project. You can specify advanced connection values later in the connection editor.

Host name:

Type:

Port:

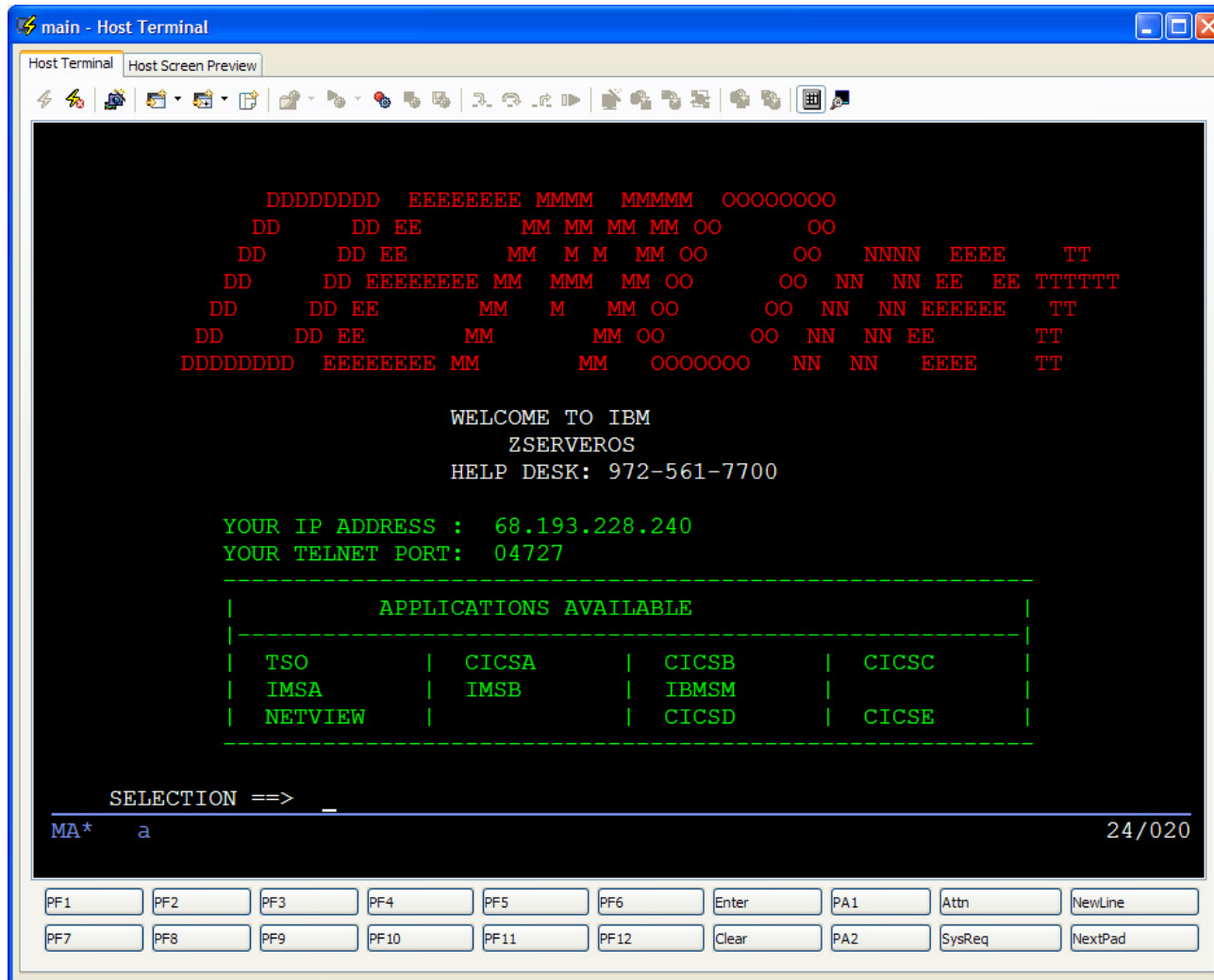
Code page:

Screen size:

Add light pen rendering support



Host Terminal



New Macro Wizard

Record Macro
✕

Record Macro

The macro will start recording after you click finish.

Project:

Name:

Location:

Description: (optional)

Create screen captures automatically while recording (VME technology preview)

?
Finish
Cancel



Macro Recording to Main Menu Screen

The screenshot shows a 'main - Host Terminal [Recording...]' window. On the left is a 'Macro Navigator' tree with the following structure:

- GetCustomerInformation
 - Screen1
 - cicsa[enter]
 - Next Screens
 - Screen2
 - Screen2
 - atdemo[tab][tab]
 - *****
 - [enter]
 - Next Screens
 - Screen3
 - Screen3
 - amnu[enter]
 - Next Screens
 - Screen4
 - Screen4

The main terminal area displays the following text:

```

OPERATOR INSTRUCTIONS

OPERATOR INSTR - ENTER AMNU
FILE INQUIRY   - ENTER AINQ AND NUMBER
FILE BROWSE    - ENTER ABRW AND NUMBER
FILE ADD       - ENTER AADD AND NUMBER
FILE UPDATE    - ENTER AUPD AND NUMBER

PRESS CLEAR TO EXIT
ENTER TRANSACTION: [redacted] NUMBER [redacted]
    
```

At the bottom of the terminal, the prompt 'MA * a' is visible on the left and '12/021' on the right. Below the terminal is a keyboard control panel with buttons for PF1 through PF12, Enter, Clear, PA1, PA2, Attn, SysReq, NewLine, and NextPad.



Screen Name / Recognition Dialog

Define Screen Recognition Criteria

Screen Recognition Criteria

Define specific criteria so that only this specific screen will be matched.
You can add additional criteria in the screen customization editor.

Screen Name:

This is a transient screen

```
OPERATOR INSTRUCTIONS
OPERATOR INSTR - ENTER AMNU
FILE INQUIRY - ENTER AINQ AND NUMBER
FILE BROWSE - ENTER ABRW AND NUMBER
FILE ADD - ENTER AADD AND NUMBER
FILE UPDATE - ENTER AUPD AND NUMBER

PRESS CLEAR TO EXIT
ENTER TRANSACTION: [redacted] NUMBER [redacted]
```

Highlight fields: Input Protected Hidden

Total number of fields:

Number of input fields:

Cursor position: Row Column

Strings:

OPERATOR INSTRUCTIONS

String position

Anywhere on the screen

At a specified position:

Row Column

Within a rectangular region:

Start row Start column

End row End column

< Back Next > Finish Cancel

Add Macro Prompt Action Dialog

The screenshot shows the 'main - Host Terminal [Recording...]' window. On the left is the 'Macro Navigator' tree with a selected path: GetCustomerInformation > Screen1 > dicsa[enter] > Next Screens > Screen2 > atdemo[tab][tab] > ***** > [enter] > Next Screens > Screen3 > amnu[enter] > Next Screens > MainMenu > [tab] > Next Screens. The main terminal area displays the following text:

```

OPERATOR_INSTRUCTIONS

OPERATOR INSTR - ENTER AMNU
FILE INQUIRY - ENTER AINQ AND NUMBER
FILE BROWSE - ENTER ABRW AND NUMBER
FILE ADD - ENTER AADD AND NUMBER
FILE UPDATE - ENTER AUPD AND NUMBER

PRESS CLEAR TO EXIT
ENTER TRANSACTION:            NUMBER           
    
```

The 'Add Prompt Action' dialog box is open, showing the following configuration:

- Name: customerNumber
- Default value: (empty)
- Password protect input
- Position: Row: 12, Column: 33
- Handle Macro Prompt:
 - Show handler (Handler: default.jsp)
 - Set prompt to string (String: (empty))
 - Set prompt to global variable (Global variable: (empty), Advanced... button)
 - Set prompt to property from User List (User Profile: (empty), User List property: (empty))

At the bottom of the terminal window is a keyboard layout with buttons for PF1-PF12, Enter, PA1-PA2, Attn, SysReq, NewLine, and NextPad. The terminal status bar shows '12/033'.

Supply Prompt Value

The screenshot shows a 'Host Terminal' window with a 'Macro Navigator' on the left. The terminal displays the following text:

```

OPERATOR INSTRUCTIONS

OPERATOR INSTR - ENTER AMNU
FILE INQUIRY   - ENTER AINQ AND NUMBER
FILE BROWSE    - ENTER ABRW AND NUMBER
FILE ADD       - ENTER AADD AND NUMBER
FILE UPDATE    - ENTER AUPD AND NUMBER

PRESS CLEAR TO EXIT
ENTER TRANSACTION:  NUMBER
    
```

A 'Prompt' dialog box is open, showing 'Enter a value:' with the input '000001'. Below the terminal, a row of function keys is visible: PF1, PF2, PF3, PF4, PF5, PF6, Enter, PA1, Attn, NewLine, PF7, PF8, PF9, PF10, PF11, PF12, Clear, PA2, SysReq, NextPad.

Define Name / Recognition Dialog

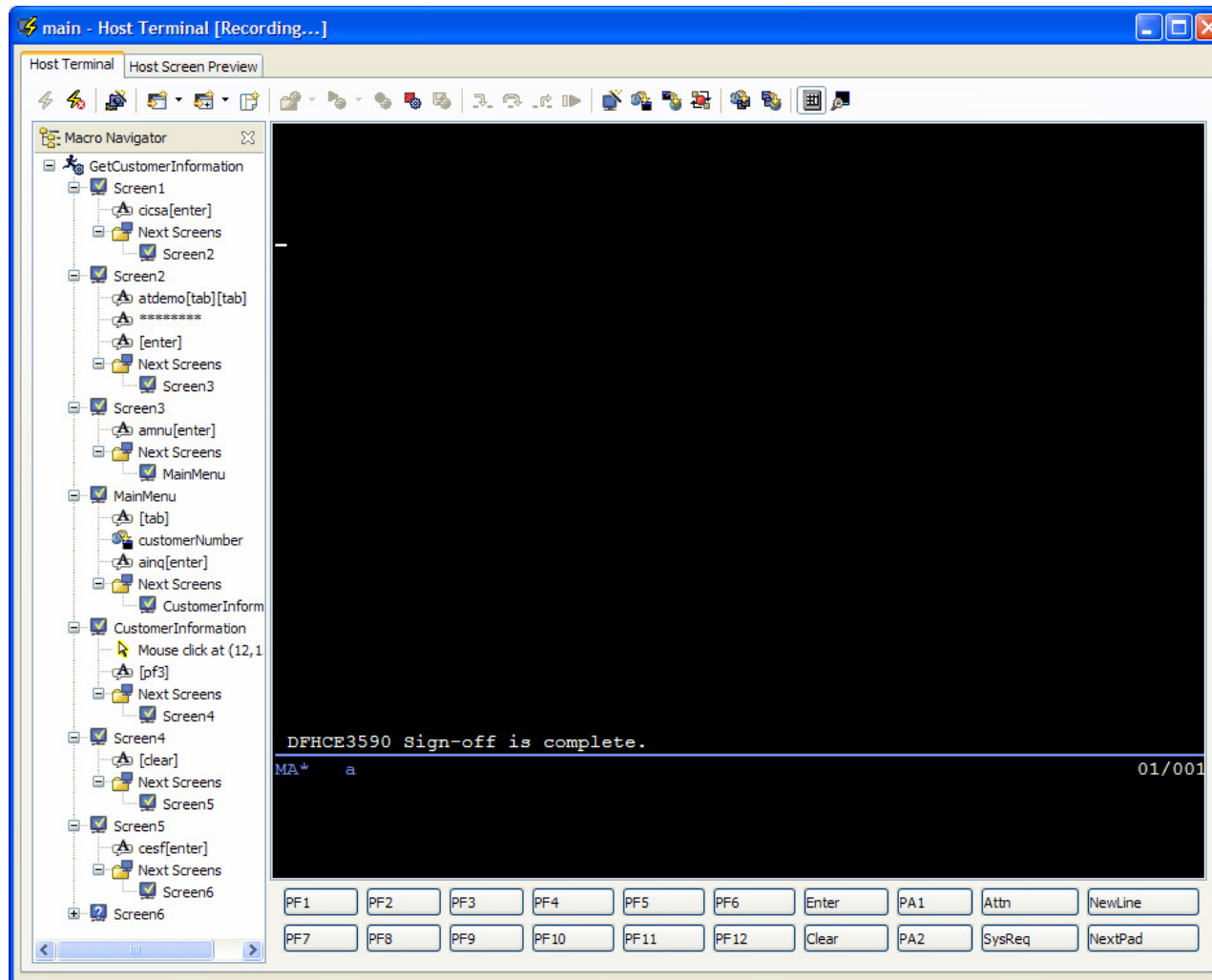
The screenshot displays the IBM Rational software interface. On the left, a 'Macro Navigator' tree shows a sequence of screens: GetCustomerInformation, Screen1, Screen2, Screen3, MainMenu, and Screen4. The main 'Host Terminal' window shows a 'FILE INQUIRY' screen with the following data:

```

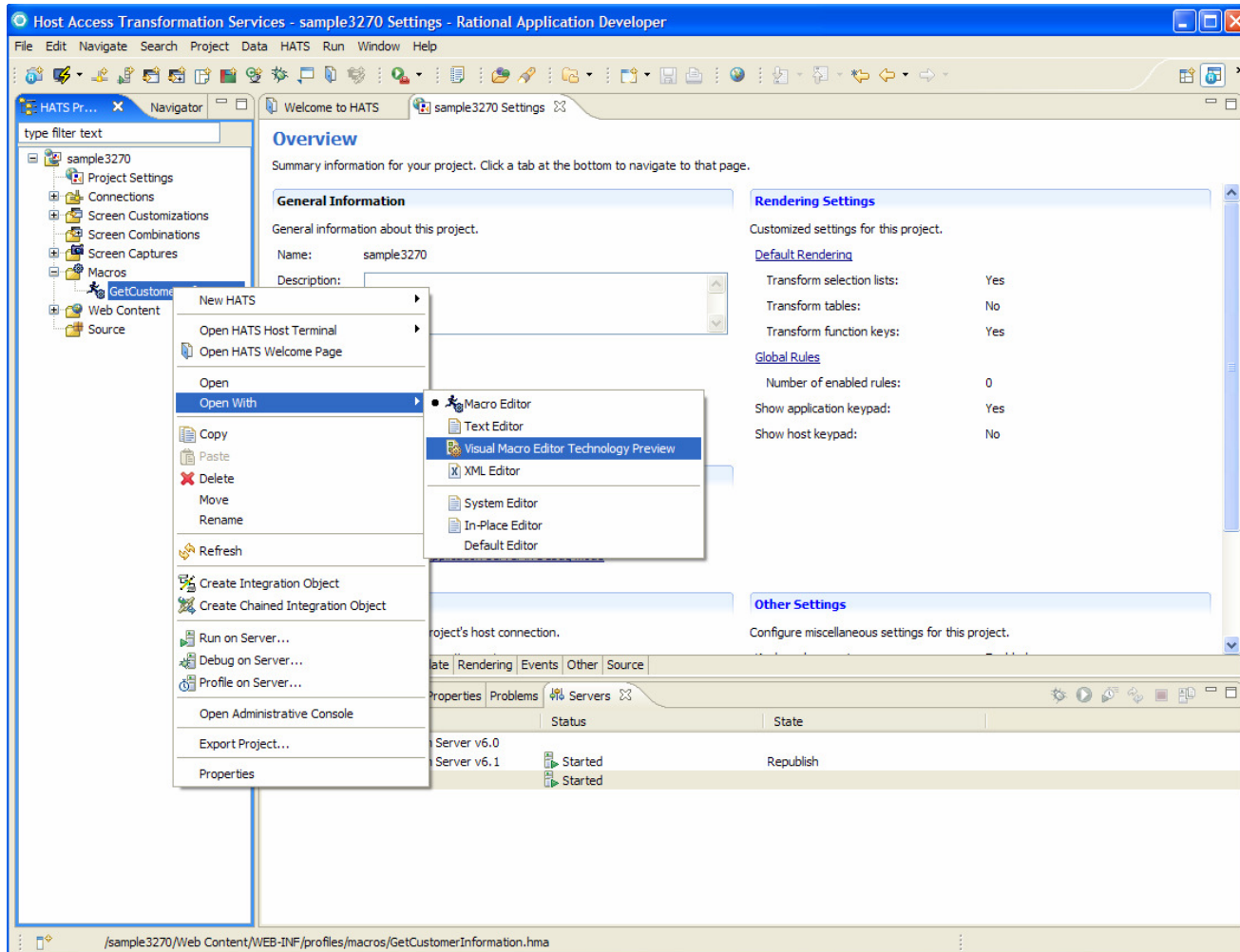
FILE INQUIRY
NUMBER: 000001
NAME: BMO - AMP
ADDRESS: BMO
PHONE: 555-5555
DATE: 01082008
AMOUNT: 99999.99
COMMENT:
    
```

Below the terminal, there are function key buttons labeled PF1 through PF10. A 'Define Screen Recognition Criteria' dialog box is open in the foreground, showing the 'Screen Recognition Criteria' for the screen named 'CustomerInformation'. The dialog includes a checkbox for 'This is a transient screen' and a preview of the screen content with the 'FILE INQUIRY' title highlighted.

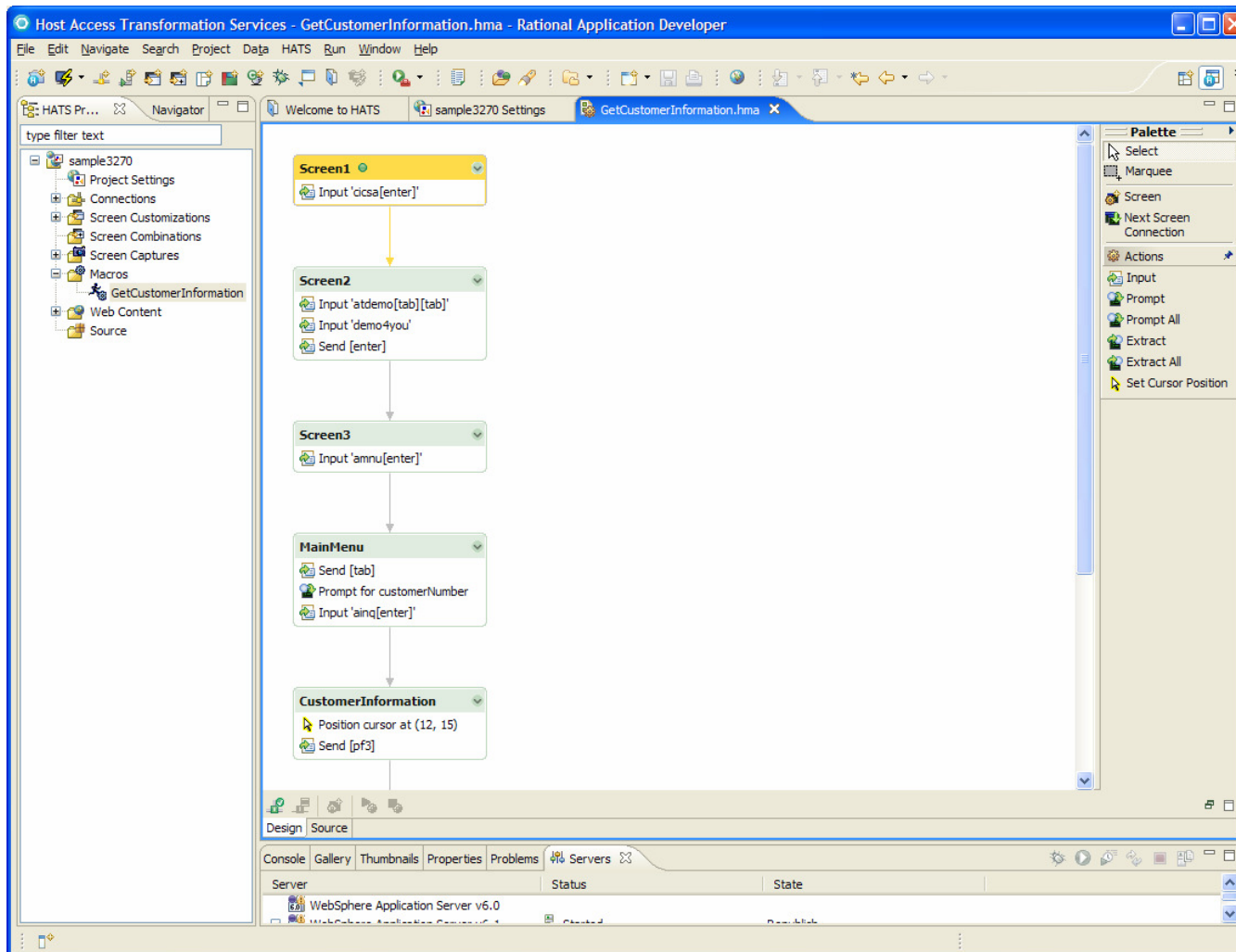
Sign Off System



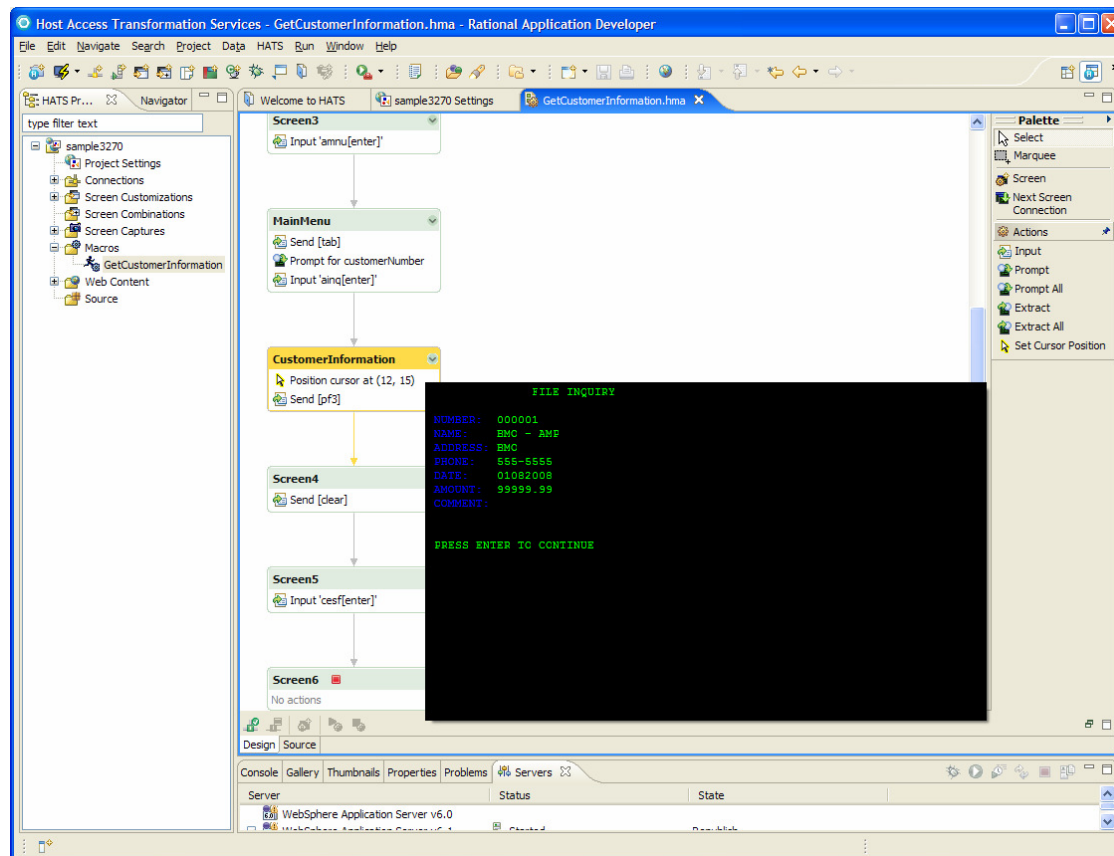
Open Macro in Visual Macro Editor



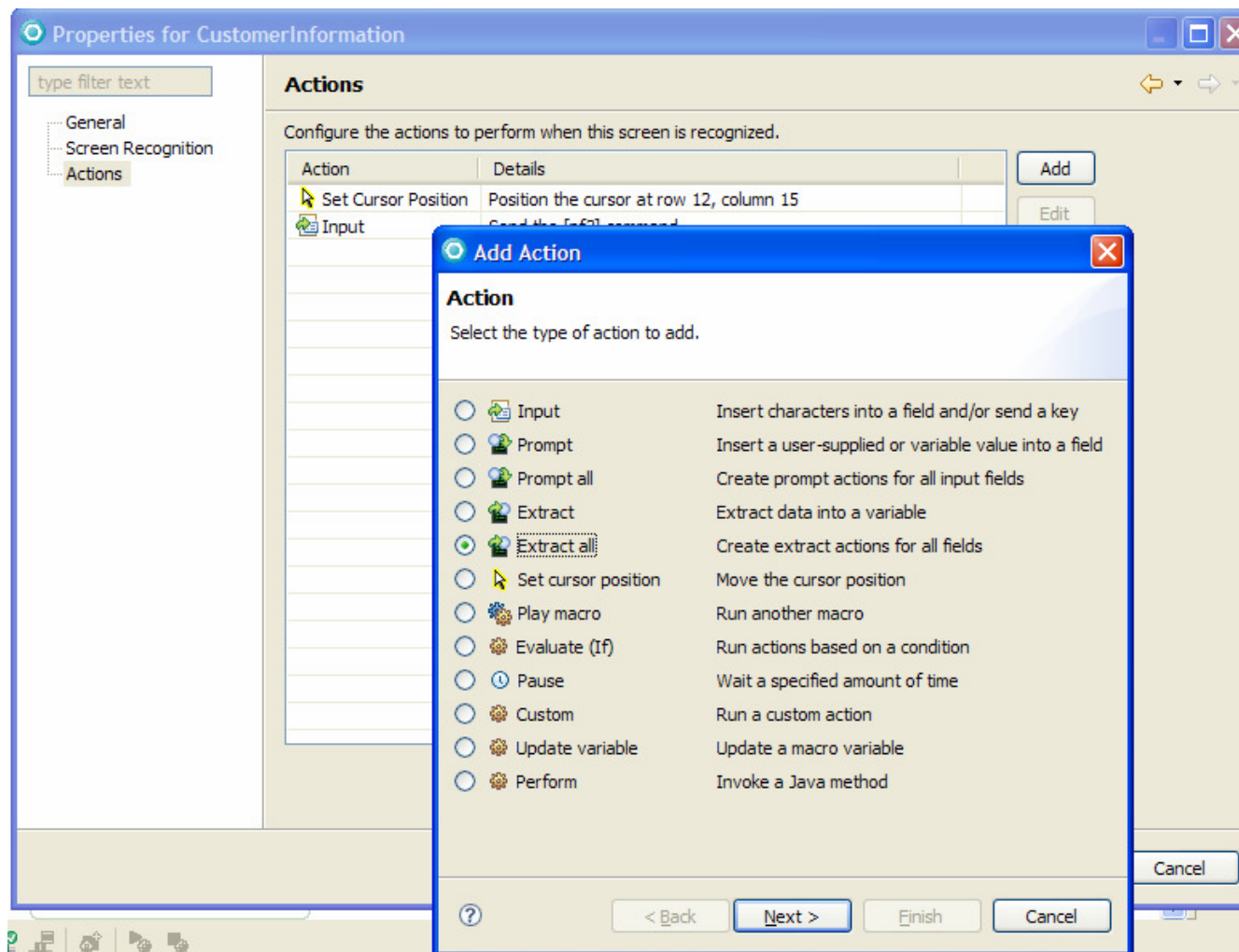
Visual Macro Editor



Visual Macro Editor / Customer Info Screen



Add Action Panel



Extract All Action Panel

⊕ Add Action
✕

Extract All
Configure the common settings for the new extract actions.

```

FILE INQUIRY
NUMBER: 000001
NAME: BMO - AMP
ADDRESS: BMO
PHONE: 555-5555
DATE: 01082008
AMOUNT: 99999.99
COMMENT:

PRESS ENTER TO CONTINUE
                
```

Handler

Show handler

Handler:

Save as global variable

Check the boxes next to the fields you want to create actions for.

Name	Row	Column	Length	Current Value
<input type="checkbox"/> in_16_145	1	16	145	FILE INQUIRY
<input type="checkbox"/> in_162_8	3	2	8	NUMBER:
<input checked="" type="checkbox"/> number	3	11	6	000001
<input type="checkbox"/> in_242_8	4	2	8	NAME:
<input checked="" type="checkbox"/> name	4	11	20	BMO - AMP
<input type="checkbox"/> in_322_8	5	2	8	ADDRESS:
<input checked="" type="checkbox"/> address	5	11	20	BMO
<input type="checkbox"/> in_402_8	6	2	8	PHONE:
<input checked="" type="checkbox"/> phone	6	11	8	555-5555
<input type="checkbox"/> in_482_8	7	2	8	DATE:
<input checked="" type="checkbox"/> date	7	11	8	01082008
<input type="checkbox"/> in_562_8	8	2	8	AMOUNT:
<input checked="" type="checkbox"/> amount	8	11	8	99999.99

Filters

Exclude empty protected fields Exclude input fields

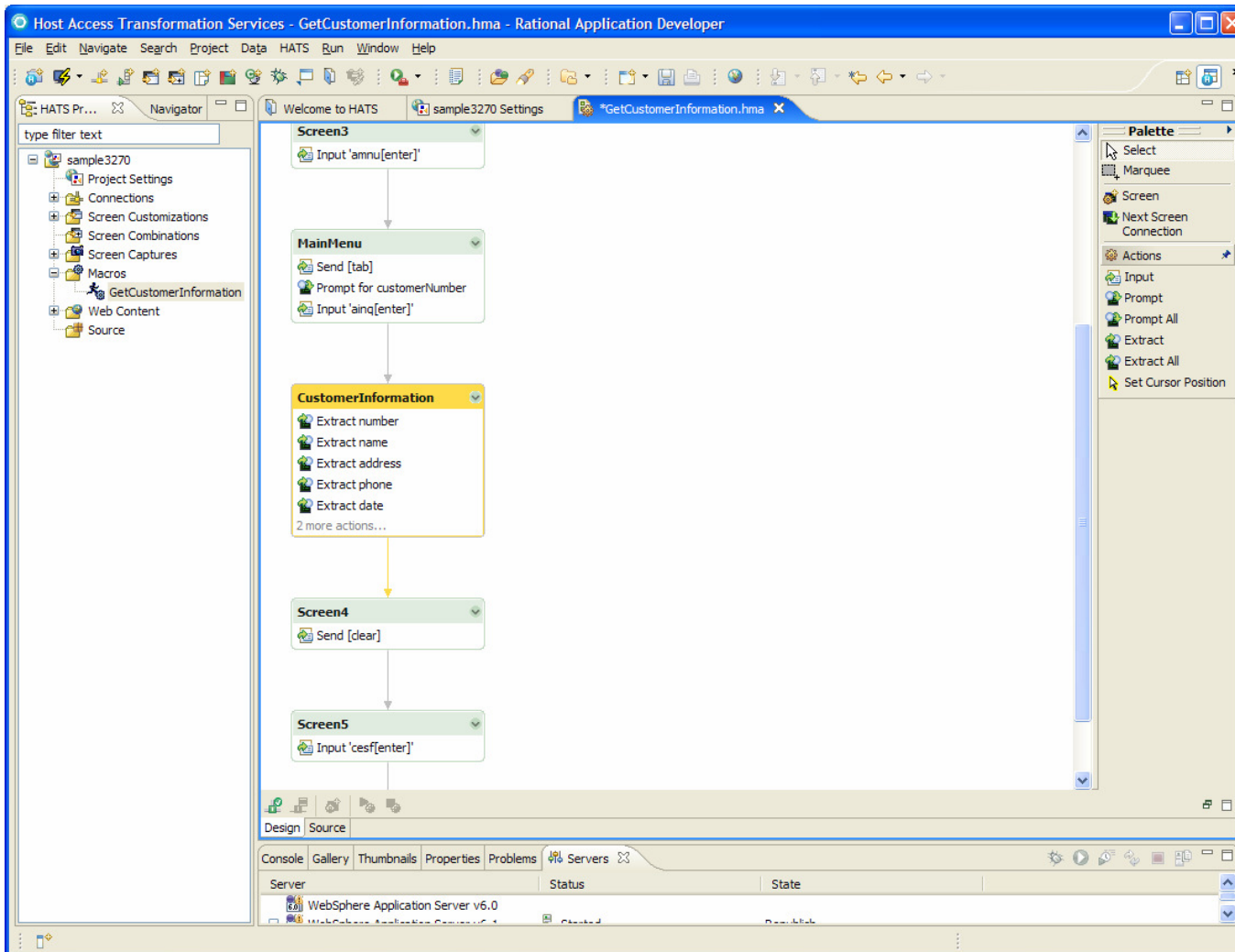
15 shown, 6 selected

Synchronize prompt names and global variable names

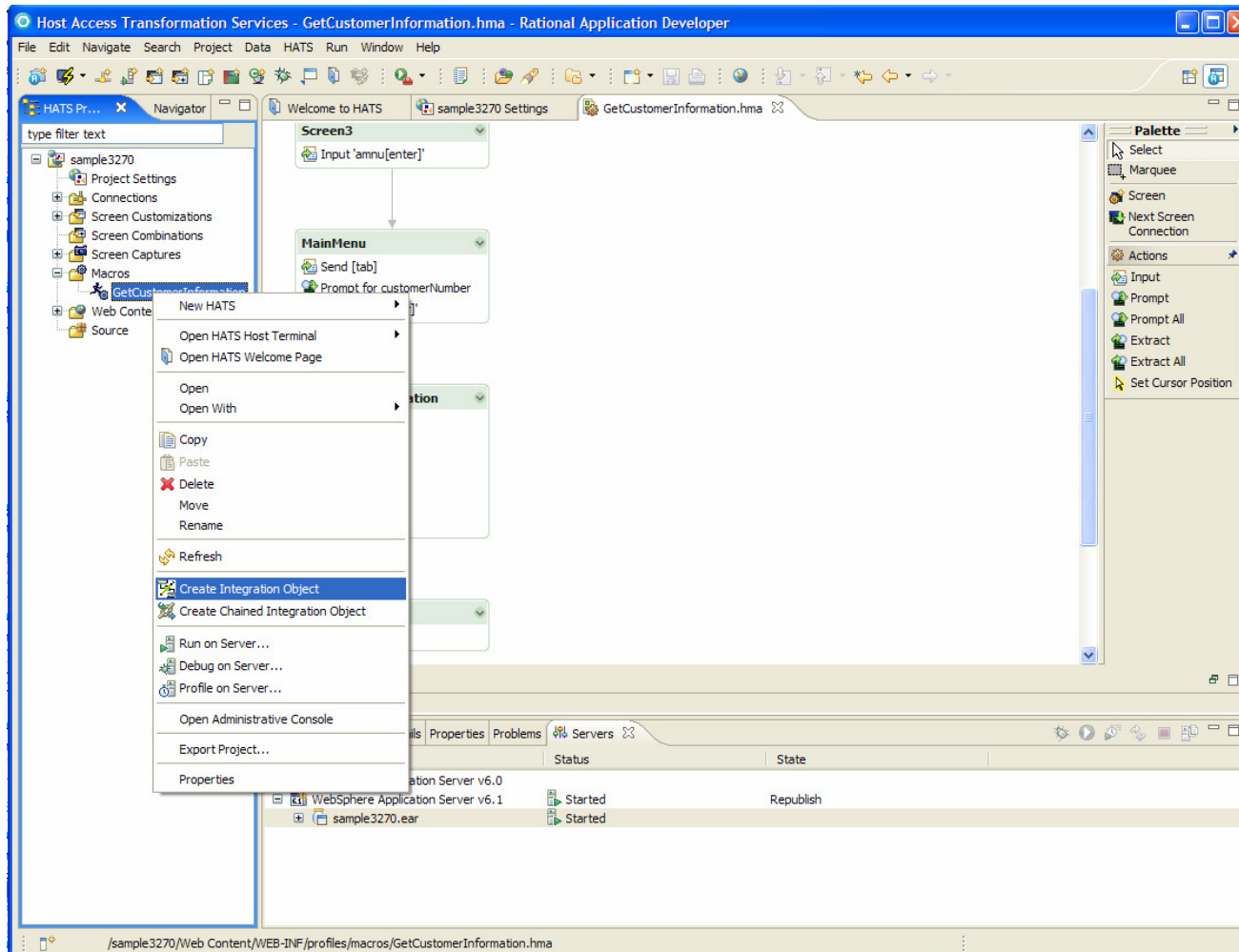
?

< Back
Next >
Finish
Cancel

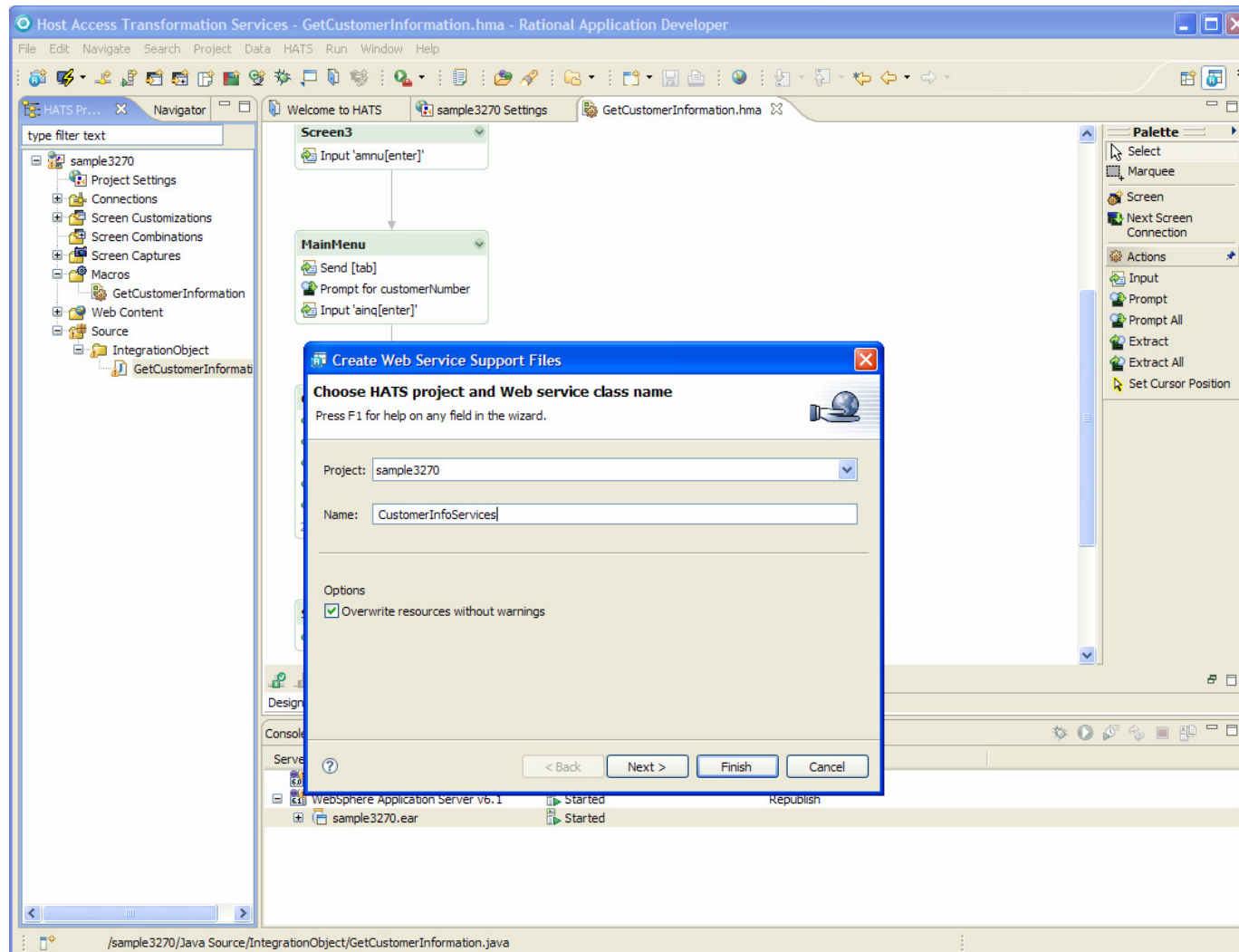
Screen Visualization



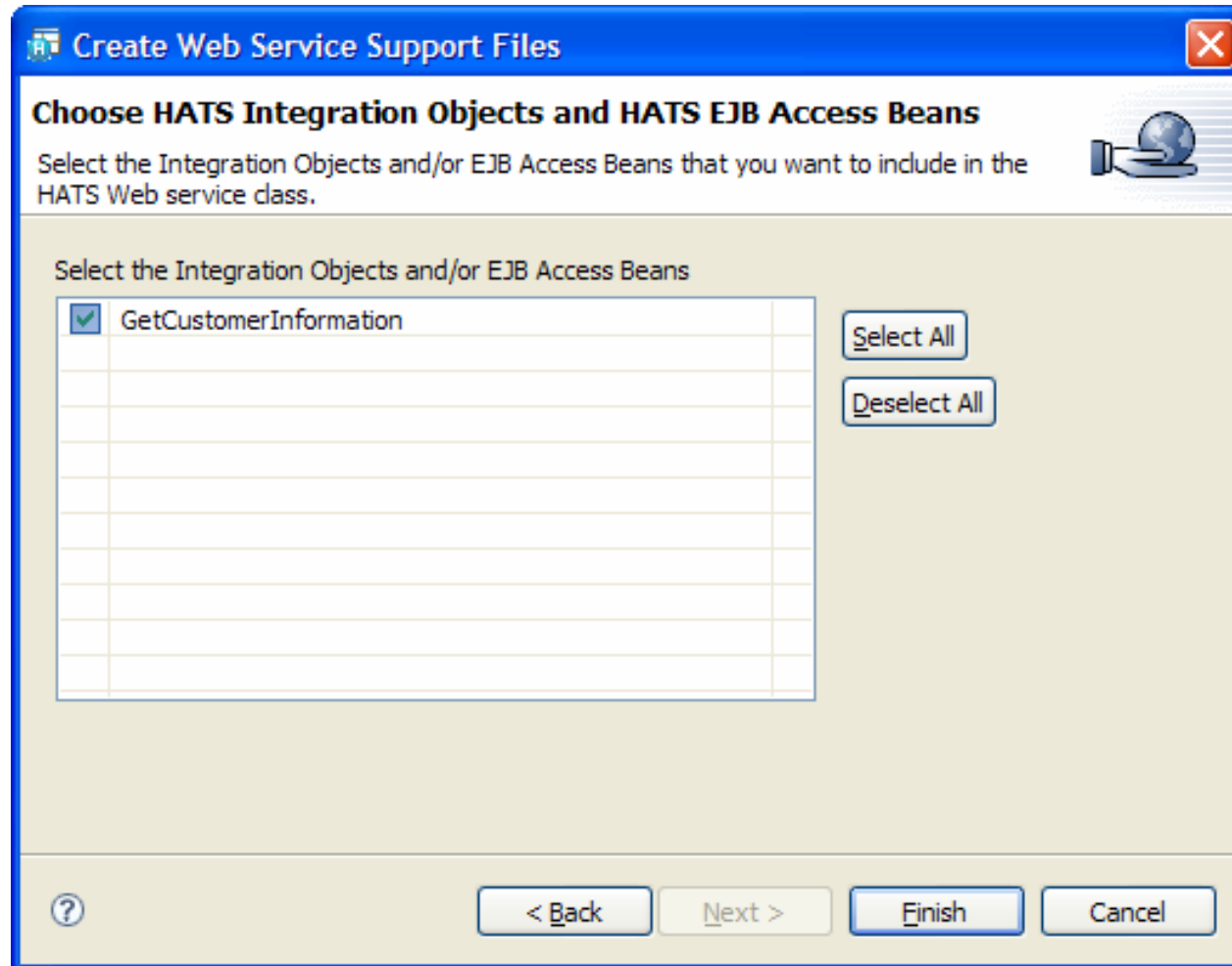
Create Integration Object



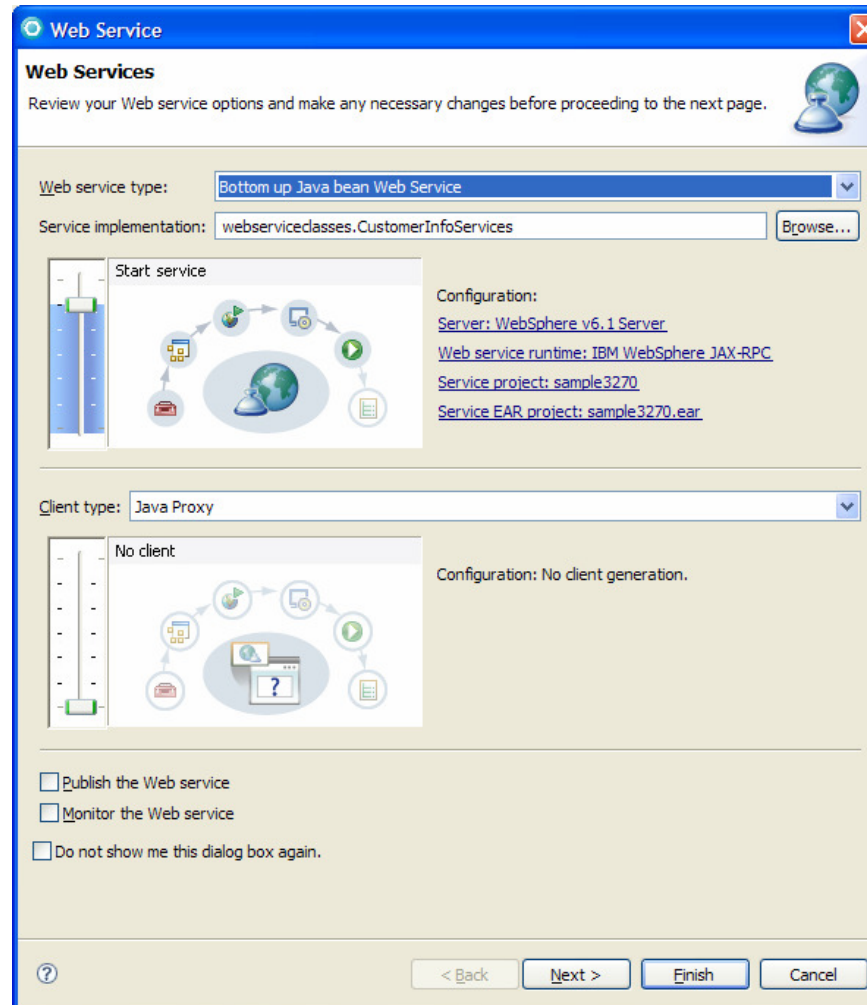
Create Web Service Support Files



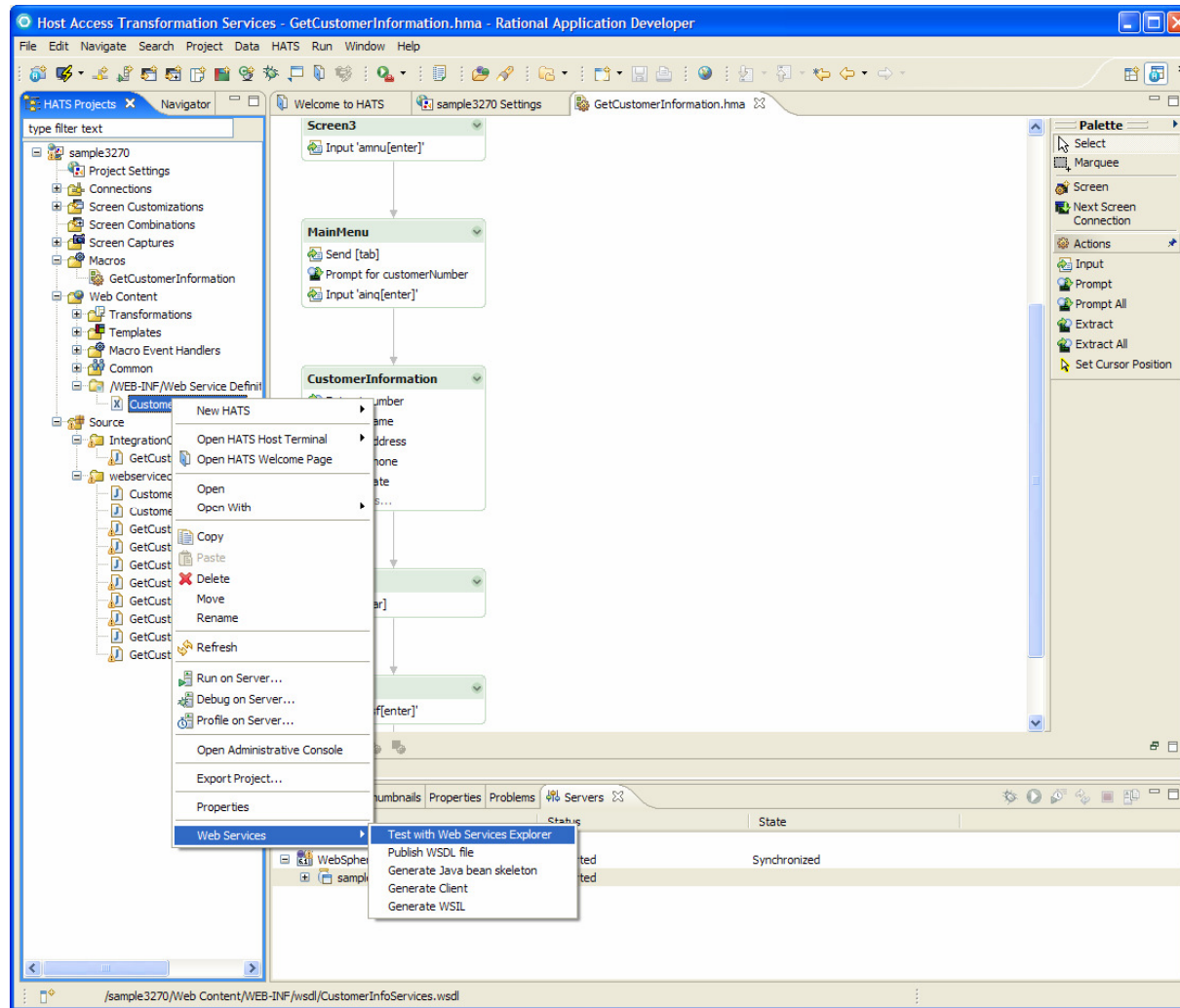
Web Service Support Files Wizard



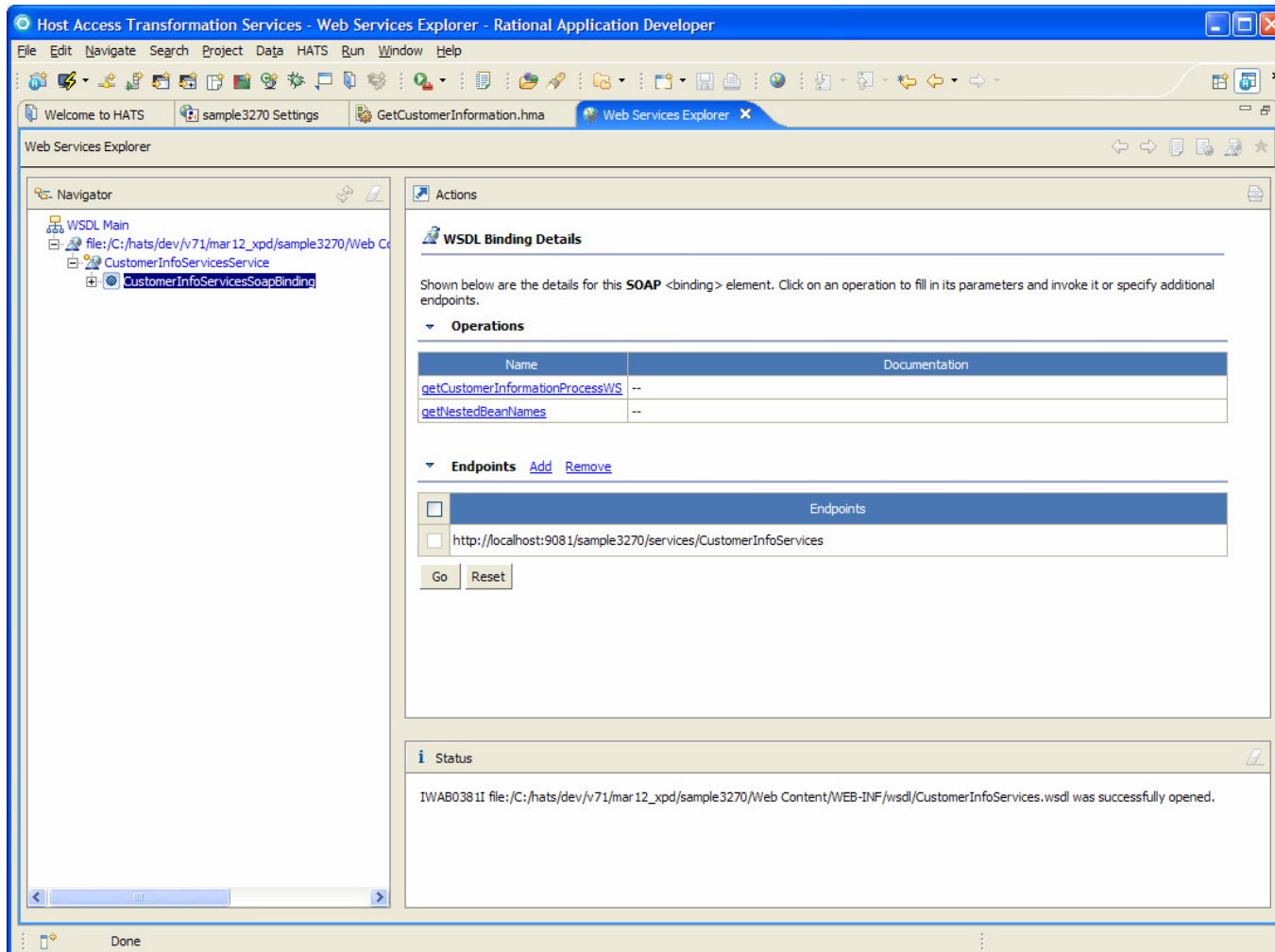
New Web Service Wizard



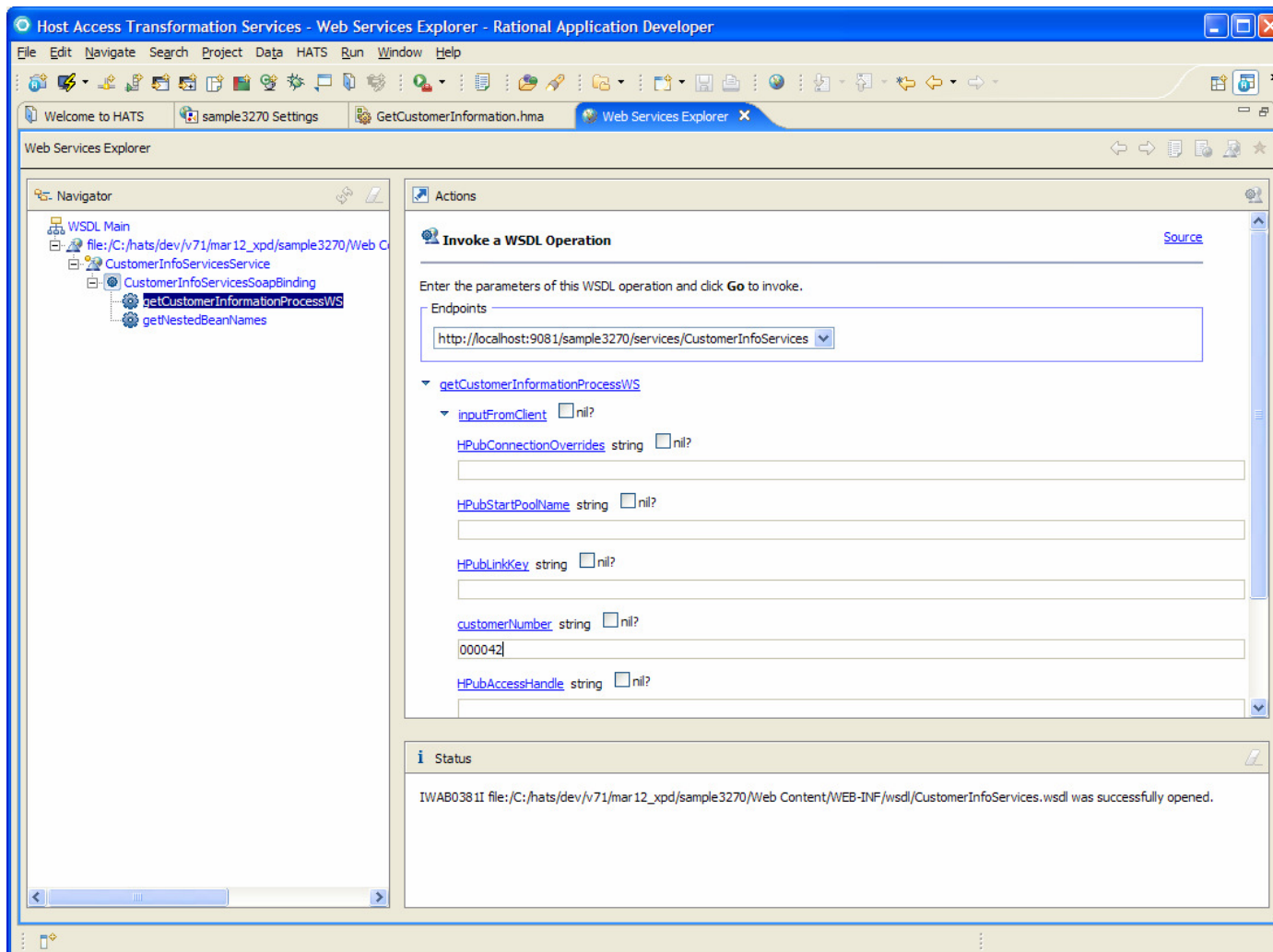
Launching the Web Services Explorer



Web Services Explorer



Test the Web Service



Web Service Invocation Results

The screenshot displays the Rational Application Developer interface for 'Host Access Transformation Services - Web Services Explorer'. The 'Web Services Explorer' pane shows a tree view with the following structure:

- WSDL Main
 - file:/C:/hats/dev/v71/mar12_xpd/sample3270/Web C
 - CustomerInfoServicesService
 - CustomerInfoServicesSoapBinding
 - getCustomerInformationProcessWS** (selected)
 - getNestedBeanNames

The 'Actions' pane shows the 'Invoke a WSDL Operation' dialog. The 'Endpoints' field is set to `http://localhost:9081/sample3270/services/CustomerInfoServices`. The selected operation is `getCustomerInformationProcessWS`. The 'inputFromClient' checkbox is unchecked, and the 'HPubConnectionOverrides' field is set to `string nil?`.

The 'Status' pane displays the following invocation results:

- HPubErrorMessage (string): null
- address (string): ROYAL CYBER
- HPubBeanName (string): GetCustomerInformation
- HPubScreenState (string): Screen6
- phone (string): 523687
- date (string): 250707
- HPubEndChainName (string): null
- HPubBeanType (string): HOD
- HPubErrorOccurred (int): 0
- amount (string): \$7478.73**
- HPubStartPoolName (string): sample3270/main
- HPubOutputParmSuffix (string): _



Other HATS SOA Capabilities

- Connection pooling
 - ▶ Automatically connects a specified number of connections at application startup. Connections can be reused (this decreases connect/disconnect activity on the system).
- Connect / disconnect macros
 - ▶ Automatically signs on sessions at startup and signs them out when no longer needed.
- User lists
 - ▶ Signs on sessions using a set of predefined credentials.
 - ▶ Starting in HATS 7.1, user list information can be encrypted.
- Stateful or stateless interaction (even for Web services)
 - ▶ Enables clients to have stateful interactions (i.e. the host session stays active between interactions) with host applications using Web Services.

* These capabilities are not new to HATS 7.1.



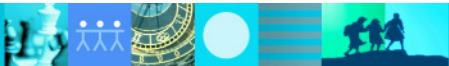
Connection Parameter Overrides For Integration Objects and Web Services

- Starting in HATS 7.0.0.2, customers can override individual connection parameters (for example: host IP address or port) when making a call to a HATS Web Service or Integration Object.
- Provides customers and ISVs with more flexibility since Web Services can be easily directed to interact with a different host system.
- Customers can also define multiple connections within the HATS application (and configure pooling, user lists, etc. for each connection) and then specify which connection pool to use when running the Integration Object, Web Service, or HATS transformation application.
- **Note:** Connection override for HATS Web transformation applications has been available since HATS V5.



Arrow Key Navigation Support (Rich Client Only)

- Provides end users the ability to navigate between protected and non-protected fields using their keyboard arrow keys.
- This behavior is comparable to how users can move around a green screen in a traditional terminal emulator using their arrow keys today.
- This function provides power users with similar keyboard navigation behavior (which helps power users more easily adapt to a HATS rendered application).
- Enabled by selecting the Classic terminal emulator theme or by enabling via **Project Settings > Other > Client Settings**.
- Function is only available in rich client applications (not Web applications).



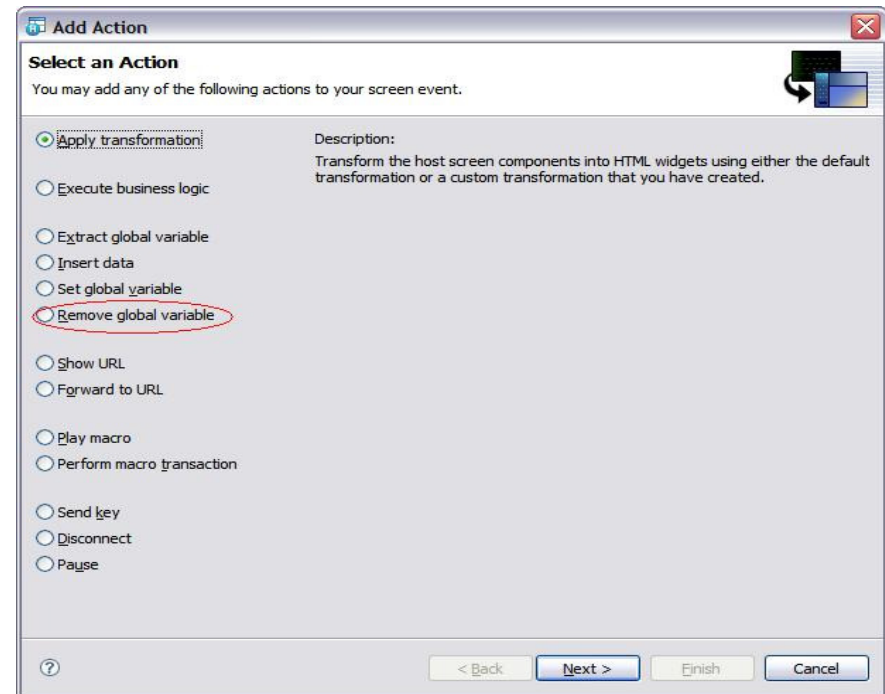
Automatically Set WAR Class Loader Policy

- In general, the WAR class loader policy for a HATS Web application must be set to “Application” for applications using:
 - ▶ Business logic
 - ▶ Custom Components
 - ▶ Custom Widgets
- Starting in HATS 7.1, this is automatically done for the developer when using the HATS Toolkit wizards to create these HATS artifacts.
- This feature helps avoid the situation where a HATS application fails to run because the WebSphere administrator forgot (or did not know) to set the class loader policy after deployment.
- Because this policy is set within the application, it does not need to be set every time the application is installed on the application server.
- **Note:** You must continue to manually set the WAR class loader policy when running a HATS application targeted for WebSphere V5.1.



Remove Global Variable Action

- A new action, **Remove Global Variable**, has been added to HATS.
- This new action makes it easier for the developer to completely clear or remove a global variable.
- Previously the HATS developer had to write custom Java business logic code (sample code was documented in the HATS Web Programmer's Guide).



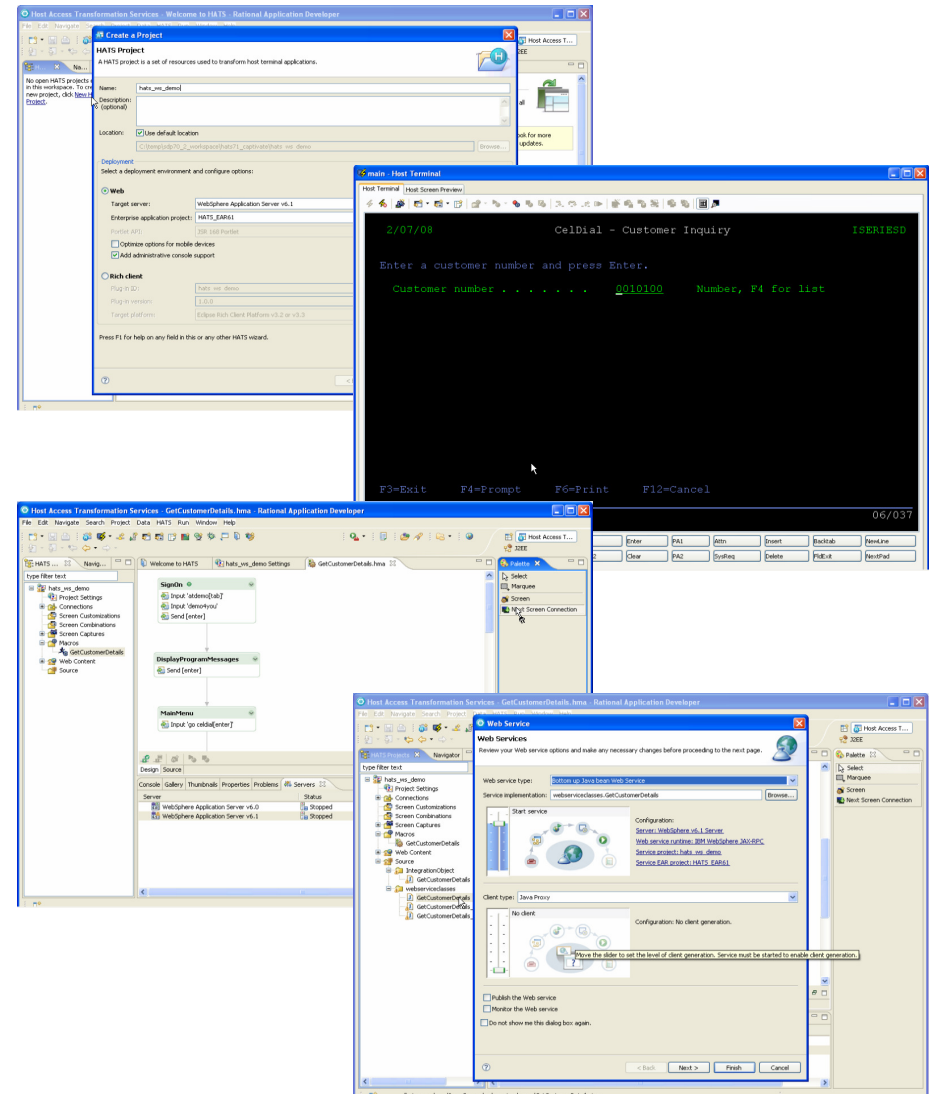
HATS and EGL

- HATS and EGL can be used together as part an overall modernization effort.
- Use HATS to service-enable programs you do not want to touch or impact
 - ▶ Reuse proven logic and abstract the details behind a Web service
 - ▶ Consume HATS Web services in an EGL Web application
- Why is HATS a good first step?
 - ▶ HATS Web services can be built very quickly and without impact to code
 - ▶ HATS tooling and EGL tooling run on the same Rational tooling platform
- Videos available on IBM developerWorks
 - ▶ http://www.ibm.com/developerworks/offers/lp/demos/summary/eglhats1.html?S_TACT=105AGX01&S_CMP=HP
 - ▶ http://www.ibm.com/developerworks/offers/lp/demos/summary/eglhats2.html?S_TACT=105AGX01&S_CMP=HP



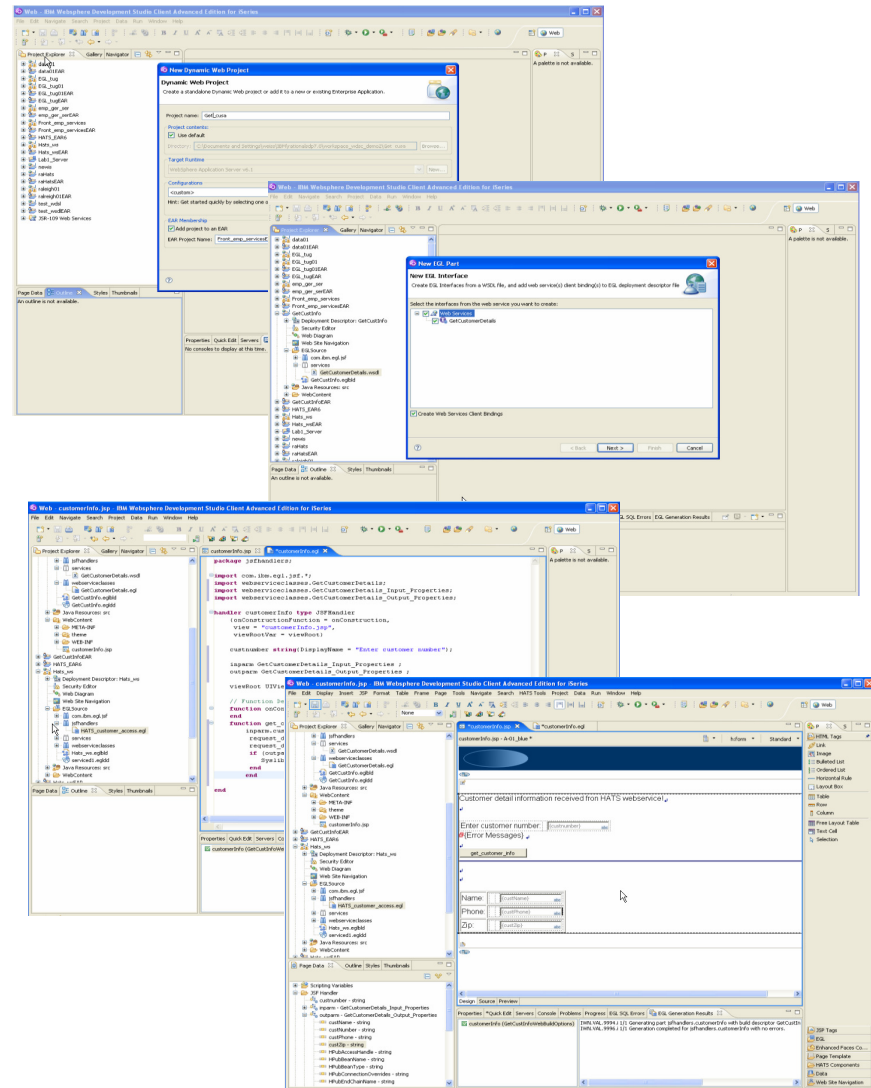
HATS Developer Tasks

1. Create a HATS Web project using the HATS toolkit.
2. Create a macro to navigate the screens, supply input, and extract output.
3. Use the HATS toolkit to generate a Java bean wrapper for the macro.
4. Use the Rational tooling to generate a Web service (and WSDL file) to drive the Java bean.
5. Deploy to WebSphere Application Server.



EGL Developer Tasks

1. Create a new Dynamic Web project, and add EGL support to it.
2. Import HATS Web services WSDL into EGL project and use the tooling to generate the EGL interface code.
3. Create new JSP page and code the page handler code to call the service.
4. Add new UI components to the page and wire to the page handler.
5. Deploy and run.



Success Story and Sales Aids

Business need

- A leading European automobile manufacturer sought to enhance the customer experience and streamline the sales process at automotive dealerships by modernizing its Vehicle Management System software. The company needed to move its back-office green screen dealer systems to the showroom as a thin-client, browser-based system that provided common entry points, simplified IT governance, complied with corporate image standards, and retained organizational knowledge.

Solution

- IBM Rational Business Partner ***Oxford International*** developed a Web-based system solution for global vehicle distribution, inventory, and order management. Using IBM WebSphere Host Access Transformation Services (HATS) and IBM Rational Application Developer, Oxford demonstrated and delivered a comprehensive modernization solution deployed on IBM System i servers running IBM WebSphere Application Server.

Benefits

- Rational Application Developer and HATS enabled Oxford to focus on business needs rather than the underlying technology to deliver a comprehensive solution. The time to achieve deployment readiness was three months, at less than 20% of the cost of building an entirely new application. The new system integrates an elegant interface and substantially improves the customer sales experience with a seamless ordering process.

▪ HATS Product Page

▶ <http://www-ibm.com/software/awdtools/hats/>

▪ Demos

▶ <http://rational.dfw.ibm.com/>

▪ Trial Code

▶ <http://www14.software.ibm.com/webapp/download/product.jsp?s=p&id=GMDL-5BJRPP>

▪ HATS Demo CD

▶ hatsdemo@us.ibm.com



Other Online Resources

- ▶ IBM WebSphere Host Integration Solution Product Page: <http://www-306.ibm.com/software/websevers/hostintegration/>
- ▶ Demos: http://rational.dfw.ibm.com/whidemo/atdemo_hats.html
- ▶ Library, Webcasts, Podcasts, Brochures, Presentations, Redbooks: <http://www-306.ibm.com/software/websevers/hats/library.html>
- ▶ Education: <http://www-306.ibm.com/software/websevers/hats/education.html>
- ▶ Services: <http://www-306.ibm.com/software/websevers/hats/services.html>
- ▶ Trial Code: <http://www14.software.ibm.com/webapp/download/search.jsp?go=y&rs=hatst>
- ▶ Customer Comments: http://www-306.ibm.com/software/websevers/hats/hats_customers.html
- ▶ HATS IBM Education Assistant content:
http://publib.boulder.ibm.com/infocenter/ieduasst/v1r1m0/index.jsp?topic=/com.ibm.iea.hats_v6/plugin_coverpage.html



Thank You for Joining Us today!

Go to www.ibm.com/software/systemz to:

- ▶ Replay this teleconference
- ▶ Replay previously broadcast teleconferences
- ▶ Register for upcoming events



Backup



Combining Screens

The screenshot illustrates the integration of different user interface elements. On the left, two overlapping 'Host Terminal' windows display a 'Display Report' with columns for 'Width', 'Column', 'Control', and 'Line'. The report lists various animal parts with their respective IDs and prices. On the right, a 'Microsoft Internet Explorer' window displays the 'JK Enterprises' website, which includes a search results table for animal products.

ID	Part	Name	Price	Category
000001	10	Bear, Teddy	26.00	Animals
000002	11	Bear, Panda	28.00	Animals
000003	12	Bear, Koala	25.00	Animals
000004	13	Dog, Beagle	15.00	Animals
000005	14	Dog, Dachshund	16.00	Animals
000006	15	Dog, Pug	17.00	Animals
000007	16	Cat, Manx	14.00	Animals
000008	17	Cat, Persian	21.00	Animals
000009	18	Cat, Siamese	22.00	Animals
000010	19	Kangaroo	19.00	Animals
000011	20	Monkey	20.00	Animals
000012	21	Wallaby	23.00	Animals
000013	22	Manatee	24.00	Animals
000014	23	Wombat	15.00	Animals
000015	24	Bird, Cockatoo	19.00	Animals
000016	25	Bird, Macaw	18.00	Animals
000017	26	Chinchilla	19.00	Animals
000018	27	Ferret	20.00	Animals
000019	28	Hampster	17.00	Animals
000020	29	Rabbit	23.00	Animals
000021	30	Alligator	21.00	Animals
000022	31	Crocodile	22.00	Animals
000023	32	Dinosaur	20.00	Animals
000024	33	Frog	12.00	Animals
000025	34	Bobcat	18.00	Animals
000026	35	Mountain Lion	19.00	Animals
000027	36	Jaguar	17.00	Animals

Combine and Integrate

The screenshot illustrates the integration of a web-based financial application with legacy mainframe systems. The browser window displays the 'JK Funds' homepage, which includes a market update section, a 'Data from 5250 Source' form for checking entitlements, and a 'Data from 3270 Host' table listing client information. Three terminal windows on the left show the underlying command-line interface, demonstrating how the web application interacts with the mainframe data sources.

JK Funds Market Update:

Dow Jones	9959.71	37.85
NASDAQ	1918.26	13.61
S&P 500	1082.24	3.19
30 Yr Bond	5.191	0.04

Data from 5250 Source:

SC4103 Check Entitlement by Customer number 5/16/02 RALYSC1 10:13:31

Support Center/400 library : DEMOBOX20

Type choices, press Enter.

Customer number :

Location code :

Support Center/400 problem number :

Associated problem :

Data from 3270 Host:

Name	Phone	Type	Node	User	Mail
Carter-Willis, A. P. (Antoinette)	596-3739	Reg	IBMUSM25	ACARTERW	WEZA1
Willis Jr, Albert F. (Al)	799-7224	Reg	IBMUSM24	AFWILLIS	B-HAM
Willis, Al L.	293-2768	Mgr	IBMUSM08	ALW883	P524
Willis, Andre R. (Dre)	888-5081	Sup	IBMUSM22	AWILLIS	H07R1
Willis, Andy B.	419-1518	Reg	IBMUSM49	ANDYWILL	022D
Willis, Art *CONTRACTOR*	000-0000	Contr	IBMUSM38	AWILLIO	B676C
Willis, Arthur W. (Wayne)		Reg	IBMUS	WILLISW	na

Results from VT Data Source:

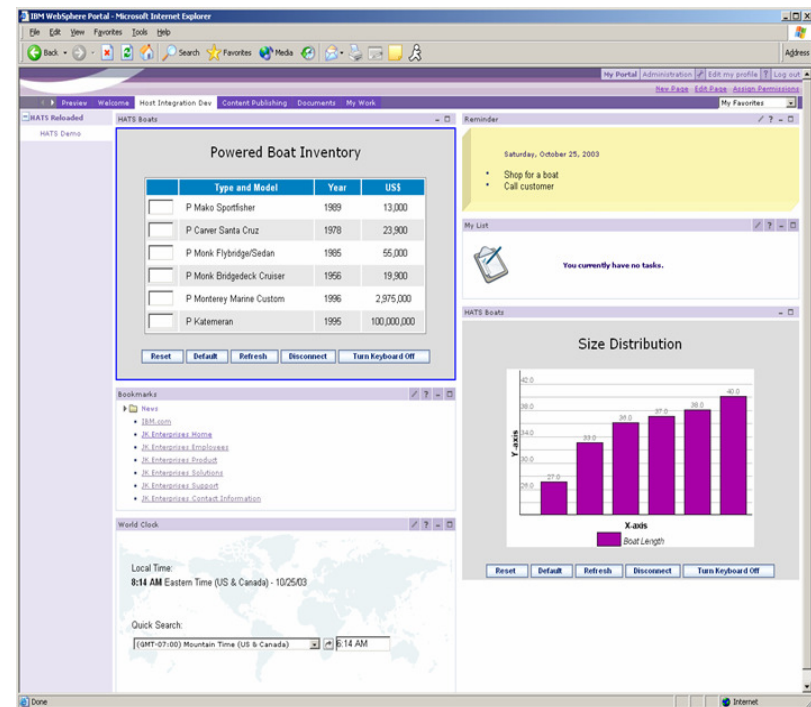
Query Directory: c:\

83 Dir(s) 22,888,353,792 bytes free

14 File(s) 4,175,691 bytes

Integration with WebSphere Portal

- Aggregate applications and content and deliver them as role-based applications
- Tailor your host application(s) to different user roles as portlets
- Present multiple host applications on the same page with business, productivity, and collaboration portlets
- HATS supports:
 - ▶ Click-to-Action
 - ▶ Credential Vault with
 - ▶ Single Sign-On



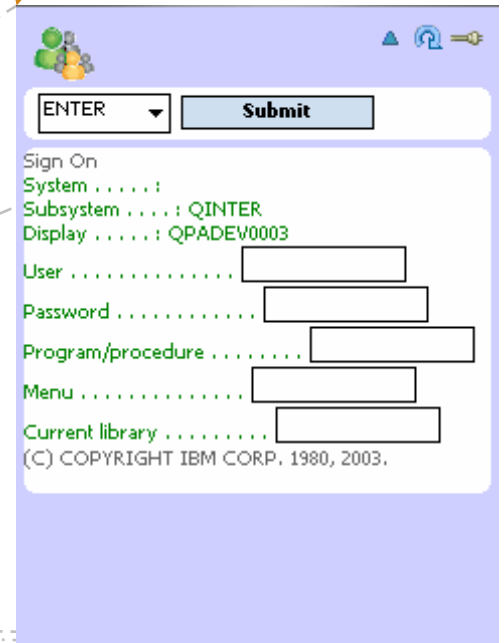
Desktop vs. PDA vs. Terminal

Desktop (1024x768, 50%)

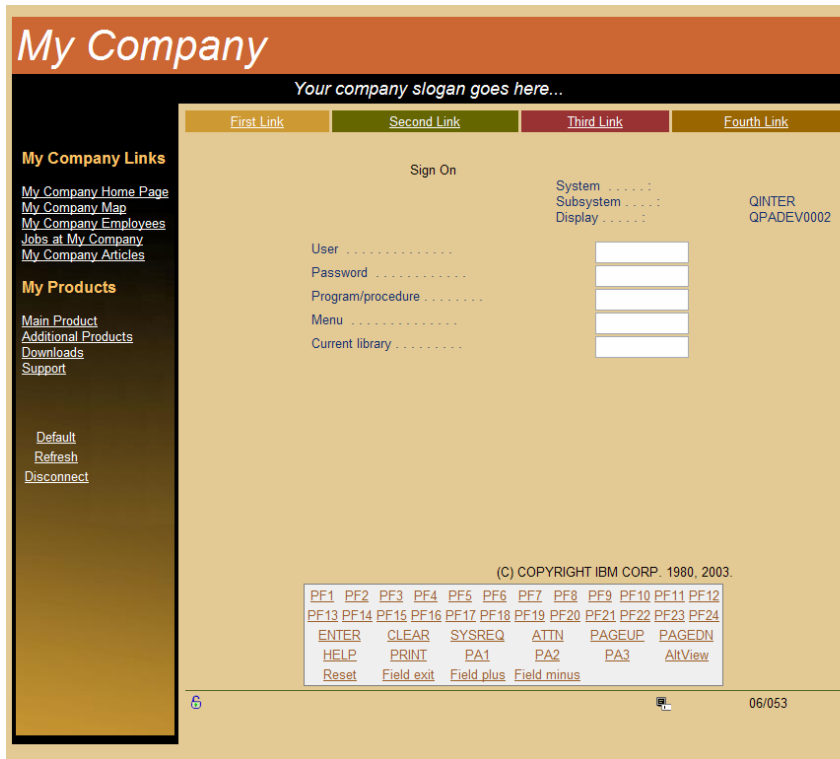
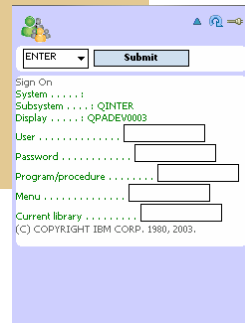


Terminal

PDA (320x240 100%)



PDA (320x240, 50%)



Improve Navigation & Productivity

- **Macros & Global Variables**
 - ▶ Skip unnecessary screens; combine screens; split screens, enter data on behalf of the end user
 - ▶ Store end user input as global variables to use again; prefill text entry fields
- **Improve User Interface with Screen Customization**
 - ▶ Add buttons, hot links, drop-down menus, valid value lists, etc.
 - ▶ Modify graphics, fonts, colors, and layout
- **Add business logic**
- **Local print support**
- **Native keyboard support**
- **Security – HTTPs, SSL**
- **Performance – HTTP Compression, Contention Resolution**
- **Supports iterative development**
 - ▶ Default Transformation rules - fully functional HATS application on the web quickly
 - ▶ Customize Default Transformation rules
 - ▶ Customize screens



Recommended Skills



- The HATS Toolkit is an integrated environment for building HATS applications – as with all tools, skills are developed over time. Ease-of-use has been a driving design goal since the first release (and was the theme for HATS version 6).
- HATS does not require Java skills, except for advanced integration (e.g. interacting with a database using JDBC) or when writing custom components and widgets.
- For Web application development: although HATS is integrated into the Page Designer (HTML editor) tool, HTML / Web skills are recommended when doing advanced screen customization.
- For rich client application development: Java and Eclipse/SWT skills are recommended for doing advanced screen customization and integration with other Eclipse and desktop-based applications.
- **The good news** : the skills required for advanced HATS development are reusable because HATS utilizes industry standard technologies (J2EE, Java, JSP, JavaScript, XML, HTML, CSS, Eclipse, SWT, etc).

