

Using Websphere HATS on the iSeries

By Phil Coulthard, George Farr, and Danny Mace



In previous articles, we introduced the first of two IBM technologies for quickly Web-enabling an existing 5250 application. The **IBM WebFacing tool** transforms your display file DDS source members into Web page source at development time and then intercepts your application I/O data at runtime. Here, we introduce the second IBM offering in this space: **Host Access Transformation Server (HATS)**, which is based on the runtime transformation of the 5250 data stream.

Although WebFacing works well when you have the display file source, HATS is a great option when you don't have that source, or when you wish to get it to the Web as quickly as possible. The HATS default transformation is effectively a turnkey solution for instant Web enablement, and its customization support is easy and non-programmatic.

How It Works

HATS is an easy-to-use program designed to help you quickly modernize your 5250 and 3270 applications. Built on open standards, HATS provides HTML access to all of your

iSeries applications. By combining a powerful, on-the-fly transformation of iSeries application screens with a rich development environment, HATS lets you deploy applications immediately and then improve them at your own pace. HATS provides a developer studio that integrates with any of IBM's WebSphere Studio products, including WebSphere Developer Studio Client for iSeries. HATS applications are deployed to WebSphere Application Server Version 4.0 or later.

HATS uses the output of the Telnet 5250 server for all of its iSeries access. By relying only on the data provided by the Telnet server, HATS can transform any iSeries screen into HTML. No DDS or other source files are needed, and no changes are required to the existing application. As screens are received, HATS applications can immediately Web-enable applications and system screens with no development effort, using on-the-fly transformation technology. In cases where you want exacting control of the contents of each screen, the development studio for HATS lets you customize individual screens using standard HTML. You can have any mixture of automatically generated and customized screens.

screen you want to customize, click the Create Screen Customization option in the emulator window to launch the customization wizard. The wizard will prompt you

to define a way for HATS to "recognize" when to apply your customized screen. HATS can use many criteria to accurately identify when to display a customized HTML version of a screen and when to display the on-the-fly transformation. By default, HATS uses the text in the first few lines of the screen to uniquely identify the screen.

Next, HATS uses sophisticated mapping technology to transform host-based user interface elements into standard HTML elements. HTML is typically a much richer UI environment than that available with a 5250 terminal. Therefore, you have many options for improving your interface from its original presentation.

For example, the wizard in **Figure 1** shows an iSeries main system menu on the left, and on the right it shows the possible ways to display, or render, the same information in HTML. You can choose any of the available options, drop-down lists, buttons, HTML hotlinks, radio buttons, etc. The wizard allows you to quickly select and preview your options. You can also use powerful mappings, such as converting tabular data into graphical charts (**Figure 2**).

HATS creates a standard HTML/JSP page to represent your choices. Using WebSphere Studio's Page Designer, you can then add any custom HTML, JavaScript, or other Web technologies to achieve added functionality.

On-the-Fly Transformation

On its own, the on-the-fly dynamic transformation is a powerful feature. Without any development effort, you can deploy your applications and HATS will convert every screen automatically into HTML. You can use HATS to provide a basic function, browser-based 5250 emulator in this mode. Beyond the

Customizing Screens

HATS provides a wizard-based studio for creating customized HTML versions of your host applications. To create a customized screen, begin by running your application with a built-in basic 5250 emulator. When you navigate to a

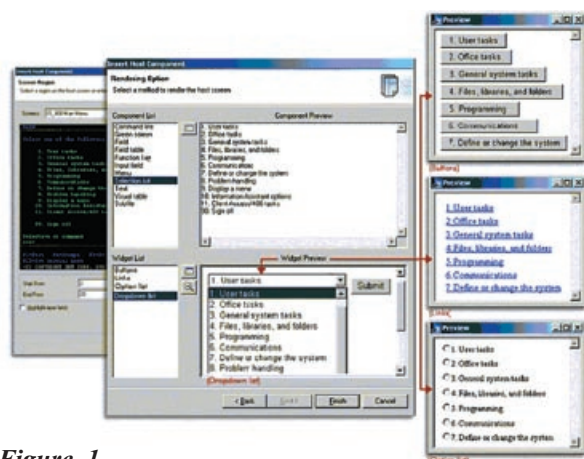


Figure 1. HATS wizard with options for displaying HTML data



Figure 2. Converting tabular data into graphical charts

typical 5250 emulator, HATS enhances the presentation to take advantage of common HTML features. For example, HATS converts subfiles into HTML tables, converts PF Keys to HTML links, changes selection lists (menus) to clickable HTML links, converts static text to buttons, and more.

You can create an application that uses only the default transformation by completing a three-step wizard. Choose a project name, enter the iSeries TCP/IP address and connection information, and then select the display style (title and navigation banner). That's it! You can now create a J2EE application (EAR) file and deploy to your WebSphere Application Server. **Figure 3** shows a default transformation of an iSeries subfile system screen.

Advanced Features

HATS has several advanced features that let you really fine-tune your application. To help you increase end-user productivity, you can create macros that let you skip unnecessary screens, combine data from two or more screens into a single HTML screen, or automate complex tasks for the end user. Macros can also extract data from each screen and store it for later use.

Using global variables, you can pre-fill forms, perform data manipulation, or send and receive data to and from your custom code. Global variables can make your iSeries application data accessible via Java, JavaScript, or other Web languages. By providing a bridge between your Web applications and your 5250 applications, global variables provide an integration point for extending the built-in functionality.

Custom code also provides an extension point for integrating HATS applications with other J2EE, Web services, or legacy applications. Instead of displaying an HTML version of your 5250 application screen, HATS lets you execute your custom business logic as an alternative.

Support for Asynchronous Messages

The HATS user interface technology is built on standard J2EE and Web user interface models. Although these models offer ubiquitous access and familiar interfaces, there are some limitations when handling asynchronous data from host applications. Web applications are built on a synchronous communications model between the application and end-user's browser. That is, the end user first submits a request to the application, and then a response is received. There is no mechanism to update the Web browser's content until a new request is initiated from the browser.

5250 terminals, on the other hand, have the ability to update the screen data at will, and without requiring the end user to request a refresh. HATS can support these asynchronous applications by using an optional Java applet to maintain a separate communication path between the Web browser and a WebSphere server. By providing an applet to Java-enabled browsers, HATS makes it possible for the server to push content to the client whenever it becomes available. Users don't need to request a refresh or reload of the Web page to receive the new content. When new content becomes available after the browser receives a screen, HATS pushes the content to the browser when the applet feature is enabled. In cases where the applet is disabled, and there are asynchronous updates, the end user must refresh or reload the page to see any new content since the last request.

iSeries-specific Features

Although HATS supports both iSeries (5250) and zSeries (3270), a few features are unique to iSeries. Here's an iSeries-specific summary of HATS features for 5250 applications.

HATS Limited Edition. Bundled with iSeries Access for Web, HATS Limited Edition (HATS LE) provides a quick and easy path to modernizing your applications. HATS LE uses the same on-the-fly transformation technology as HATS to dynamically enable HTML access to all of your iSeries applications. HATS LE does not require WebSphere Studio and has a simple HTML administration interface to configure your applications.

Printing PDF files using iSeries Access for Web. HATS supports downloading PDF versions of host print files. For 3270 applications, HATS converts standard host print jobs into PDF and lets you view or print the print files from your Web browser. For 5250 applications, this support is provided by iSeries



Figure 3. Sample screen with a subfile

Access for Web. HATS provides integration with iSeries Access for Web when creating your project. When an end user of the HATS application issues a command to print files, iSeries Access for Web converts the host print jobs into PDF format and facilitates the download to the end user. The end user can click View Print Jobs on the HATS application keypad to display the iSeries Access for Web Printer Output window.

Workstation ID. HATS 4.0.2 expands the ability to select how the workstation ID session parameter is set. You have the ability to get the workstation ID from:

- the end user (by prompting the user before making the host connection)
- an HTTP session variable
- a fixed value that you select (it may contain wildcards)

These choices give the administrator flexibility when creating a HATS application where the workstation ID must be defined at application connection time.

DDS keyword support. DDS check keywords let application programmers improve the data validation handled by a 5250 emulator. For example, when a field is marked as “auto enter,” once a user supplies *x* number of characters into the field (*x* being the size of the field), the emulator automatically submits the page back to the host application for processing. Beginning in 4.0.4, HATS is aware of these attributes and makes them available to the client browser so that the end user sees the correct behavior for each field. Having support for these attributes will reduce the amount of interaction required with the HATS server and speed up field validation. The following 5250 check keywords are supported:

DDS Keywords related to validity checking:

- AB (allow blanks)
- ME (mandatory entry)
- MF (mandatory fill)
- MOD10 (IBM modulus 10 check)
- MOD11 (IBM modulus 11 check)
- generated text boxes marked with “mod11” attribute

DDS keywords related to keyboard control:

- RB (right fill w/blanks)
- RZ (right fill w/zeros)
- ER (auto enter on field completion)
- LC (allow lowercase characters)
- FE (field exit required)

(A field on a host screen can also be marked to only allow a certain type of input. For example, the user name and password fields on a typical iSeries Sign On screen allow only upper-case

letters and digits. With this additional data validation information, HATS can generate HTML text boxes that restrict key presses based on the field type. For example, if a host screen field is marked as “numeric only,” only numerals (and other characters treated as numeric) will be allowed to be entered into the generated Web page text box. This will reduce the chance of the screen becoming locked or being returned to the user with a message they may have never seen before (since most emulators restrict input.)

DDS keywords related to input restrictions (3270 and 5250):

- X (alphabetic only, lowercase are converted into uppercase)
- A (alphanumeric shift, everything permitted), 5250 only
- S (signed numeric, requires field exit), 5250 only
- N (numeric shift; 0-9, user must press field+,-,exit), 5250 only
- Y (numeric only, 0-9, plus, minus, period, comma, space)
- D (digits only, 0-9, field minus not allowed), 5250 only

(Beginning with HATS 4.0.4, the Web interface now has the same data validation, keyboard control, and input restrictions, as the original 5250 terminal-based application.)

Required Software

To use HATS, you should have the following software installed on your iSeries:

- OS/400 V5R1, V5R2 or V5R3 with the latest cumulative PTF fix package
- WebSphere Application Server:
 - V4.0.x, V5.0.x, or V6.0.x for iSeries
 - V4.0.x, V5.0.x, or V6.0.x Advanced Edition/Advanced Edition Single Server for iSeries
 - Network Deployment for iSeries V5.0.x or V6.0.x
 - Express V5.0.x or V6.0.x for iSeries
- IBM HTTP Server (powered by Apache) is recommended
- HATS Fix Pack 4.0.2 (APAR IC35970) or higher is recommended

More Information

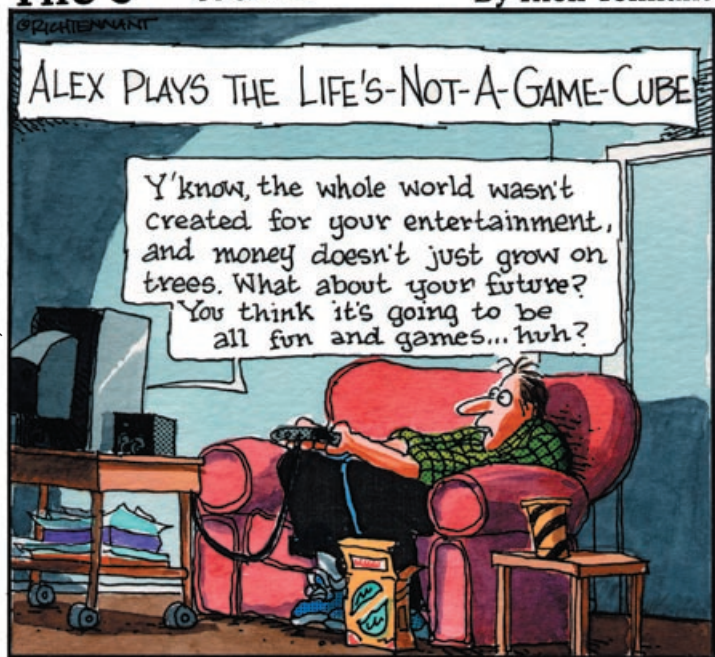
For more information about WebSphere HATS, visit www.ibm.com/software/webservers/hats. For more information about HATS LE, visit www.ibm.com/eserver/iseries/access/hatsle. You can access the public HATS discussion group at news.software.ibm.com; use the discussion group name ibm.software.websphere.hats. 

This article was first published in iSeries News magazine.

George Farr works at the IBM Toronto lab, where he is the technical development manager for the RPG and VisualAge for RPG languages, as well as the new RPG and Cobol tools in WDS. **Phil Coulthard** works at the IBM Toronto lab, where he is the lead architect for application development tools and languages on iSeries. Phil and George are frequent speakers at many conferences and user groups worldwide, and their books *Java for RPG Programmers, 2nd Edition* and *Java for S/390 and AS/400 COBOL Programmers*, are available from Penton and MC Press. **Danny Mace** is a software development manager for WebSphere Enterprise Modernization, also at the Toronto lab.

The 5th Wave

By Rich Tennant



© The 5th Wave, www.the5thwave.com