

# Letters from the Trenches



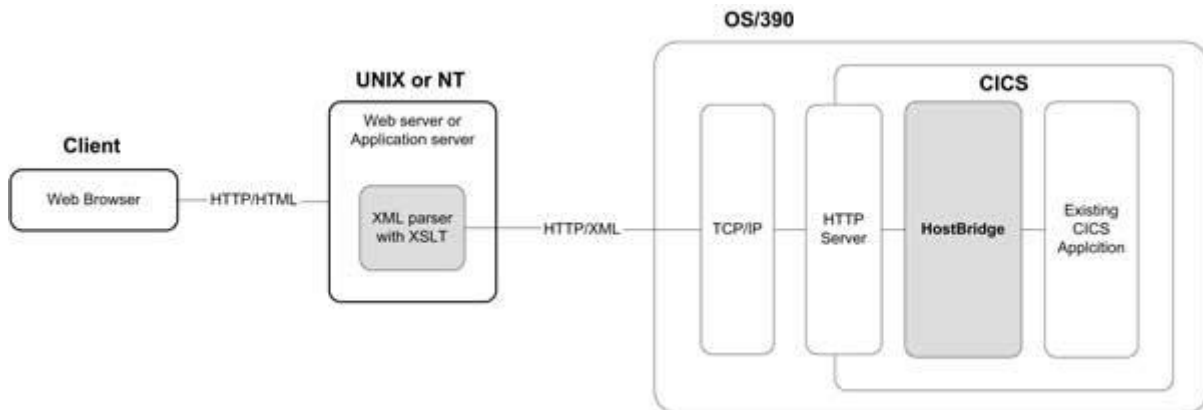
## Web-Enabling CICS Using XSLT and the Web Interface

### From the HostBridge Archive

HostBridge facilitates program-to-program interaction by allowing a non-CICS program to invoke a CICS transaction and receive its output as an XML document. By generating XML (which is independent of the user interface), HostBridge allows you to integrate CICS applications with other web-based applications and use their existing presentation tools to create the user interface.

One way to accomplish this is to use the industry standard Extensible Stylesheet Language Transformation (XSLT) to convert the XML output from HostBridge to HTML for consumption by a web browser. This method is most useful when you simply need to present transaction data to users without requiring them to interact directly with the CICS application.

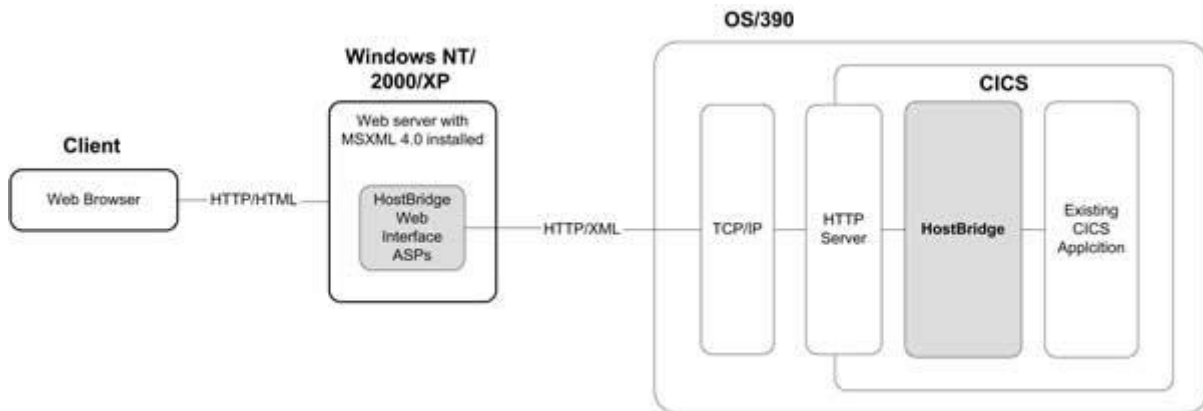
The basic configuration required for this approach is shown below.



**Figure 1. Basic configuration for XSLT web-enablement**

Several of our customers, however, need components that would allow an end-user to interact with a CICS application through a browser as though they were using a 3270 terminal. In other words, they need web-based “terminal emulation” without the emulator. Furthermore, they need a programmatic interface that would allow them to write an Active Server Page, or other server-based program, that could participate in the process.

To meet this requirement, we developed a software component to act as a HostBridge output post-processor, and an input pre-processor. This component runs on a server, not on the mainframe. The output post-processor accepts as input the XML generated by HostBridge, and returns as output an HTML representation of the screen (to be displayed by the browser). The input pre-processor accepts as input the HTML post data from the end user/browser, and returns as output the command string to be sent to HostBridge. Together, the output post-processor and input pre-processor define a server-based programming interface that can be called from within an Active Server Page, or any other server-based program. That basic configuration for using the Web Interface appears below.



**Figure 2. Basic configuration for the HostBridge Web Interface**

Built using Active Server Pages and JavaScript, the Web interface is a simple implementation of the programmatic interface. It presents users with an HTML-based presentation of a 3270 screen and allows users to submit data and interact with the back-end CICS applications using PF keys and other AID keys. The Web Interface illustrates how developers can build ASP-based applications that communicate directly with HostBridge. (See sample output from the Web Interface that shows the same information presented by the XML/XSLT/HTML samples.)

If you have any questions about web-enabling your applications using HostBridge with either XSLT or the Web Interface, let us know. See our web site for more information on HostBridge and more information on ROI. Or, call us at 1-866-XML-CICS (965-2427).

## Tech Notes

This section answers common technical questions.

### **Q. Can I use InterTest/CICS (from Computer Associates) to debug a transaction invoked through HostBridge?**

**A.** HostBridge changes nothing about the CICS transaction or its execution. With HostBridge, the only thing different is how the transaction is invoked and the how it communicates to the outside world. The transaction itself is absolutely unchanged.

One of the key features of products like InterTest or Xpeditor (from Compuware) is the ability to intercept and moderate a transaction's interaction with a terminal user. Thus, if you are having problems with a particular transaction and want to debug it with one of these tools, we suggest logging on to CICS and simply running the transaction from a terminal and use InterTest as you normally do.