

Service Oriented Architecture: Example Client Implementation on NonStop

Rick Hunter

Vice President
CAIL
Denver, Colorado

Service Oriented Architecture (SOA) is taking different forms as it evolves into the enterprise business of today. The server-only approach of yesterday is becoming the combination of many mission-critical servers and even “appliances” that provide data that can be configured, channeled into new business services and customer-oriented solutions, or managed by new interfaces.

For NonStop environments, this can present challenges as existing applications are often based on NonStop facilities such as Pathway, Screen Cobol (SCOBOL), Open Systems Services (OSS), and other technologies. These technologies bring the unique NonStop capabilities to customer applications, but they are not easily decomposed into business logic and data components.

Many NonStop applications were built with session control and business logic integrally included in the Screen Cobol “requestors.” This business logic can be comprehensive and hard to remove from the original requestor environments. CAIL Studio is a new technology that allows this Pathway and SCOBOL business logic to be leveraged and reshaped into new and exciting business solutions without disturbing the integrity of NonStop applications.

This has now been applied to the needs of several customers and recently was done at a large international transportation company using NonStop servers. In this example

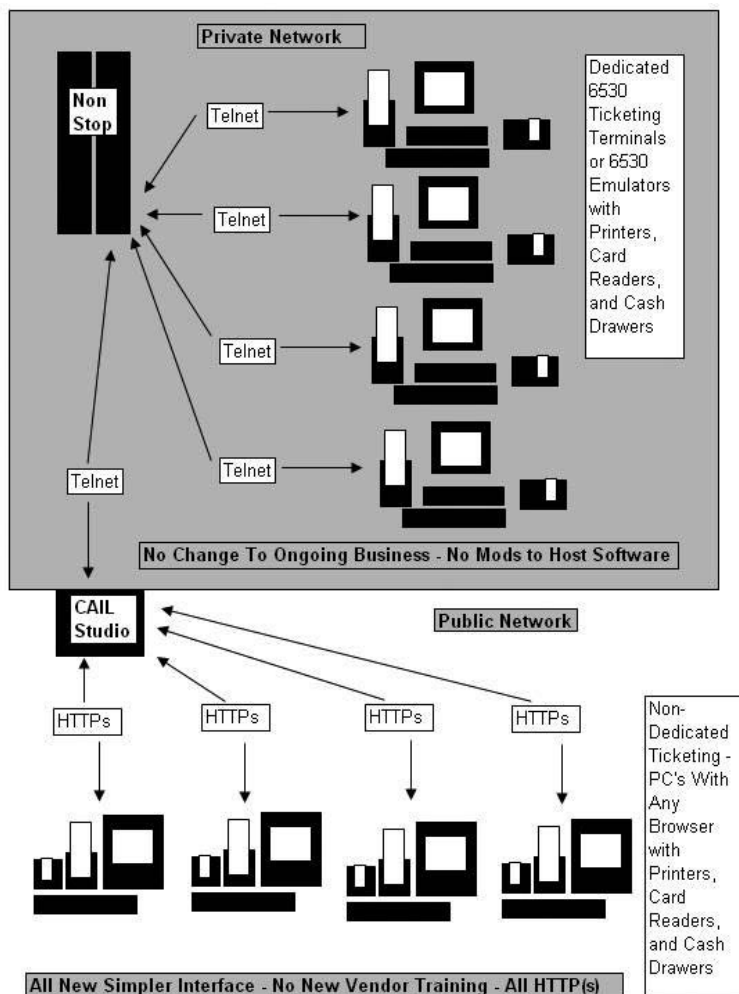
implementation, the customer wanted to expand its gross sales without more training of staff or financing new sales centers. The following story is true but the names are changed to protect the innocent.

This customer currently has a large Pathway and SCOBOL environment. Previous to this work, all booking agents accessed only the SCOBOL “front end” via the standard 6530 protocol. This was not optimal for the customer as it felt training and staffing new booking agents in the 6530 format was tedious and imposed a burden on its resources going forward. Secure Telnet communications was an issue, and the reality of reprogramming the 6530 environment was not convenient for developing new business solutions.

This customer required a more standard, easier-to-learn interface, so that it could integrate booking agents on an ad hoc basis and easily expand its gross sales and revenue.

CAIL was able to help with the new and open standards solution set called CAIL Studio. CAIL also provided some custom work to integrate additional hardware at the new remote service “kiosks” allowing secure and seamless operation at the new, easy-to-operate sales venues.

Rick Hunter is a long-time NonStop Product Manager and Consulting Analyst for Tandem. After Tandem/ Compaq/HP, Hunter was vice president of strategy for a leading NonStop Operations Software partner and moved to CAIL about a year ago to help the new NonStop SOA initiative with CAIL Studio. Hunter has a combination of business and technical experience that lends its self to understanding how to help customers simplify and organize their NonStop servers and applications into new capabilities and business needs. In combination with experts like Jerry Waese from CAIL, SOA can become very accessible without huge infrastructure costs and upgrades, as shown by the customer example in this article.



NonStop Server environments and Pathway/Scobol applications featured largely at this customer's location and to expand its business the customer had three primary requirements:

1. No changes could be done to the production NonStop servers or Pathway applications.
 - *This customer was determined to reuse all existing host software and somehow create a completely new and simple Web-oriented capability for its system on a manageable budget.*
2. Remote connectivity had to be via secured HTTPs to keep the telnet traffic to a minimum and protect against hackers and spoofing attacks.
3. Individual Kiosk sessions had to include bank-card readers with remote printers and computing devices without appearing any differently to the NonStop Host computer.

The customer wanted to use remote computers equipped with a cash drawer and a bank-card reader owned by

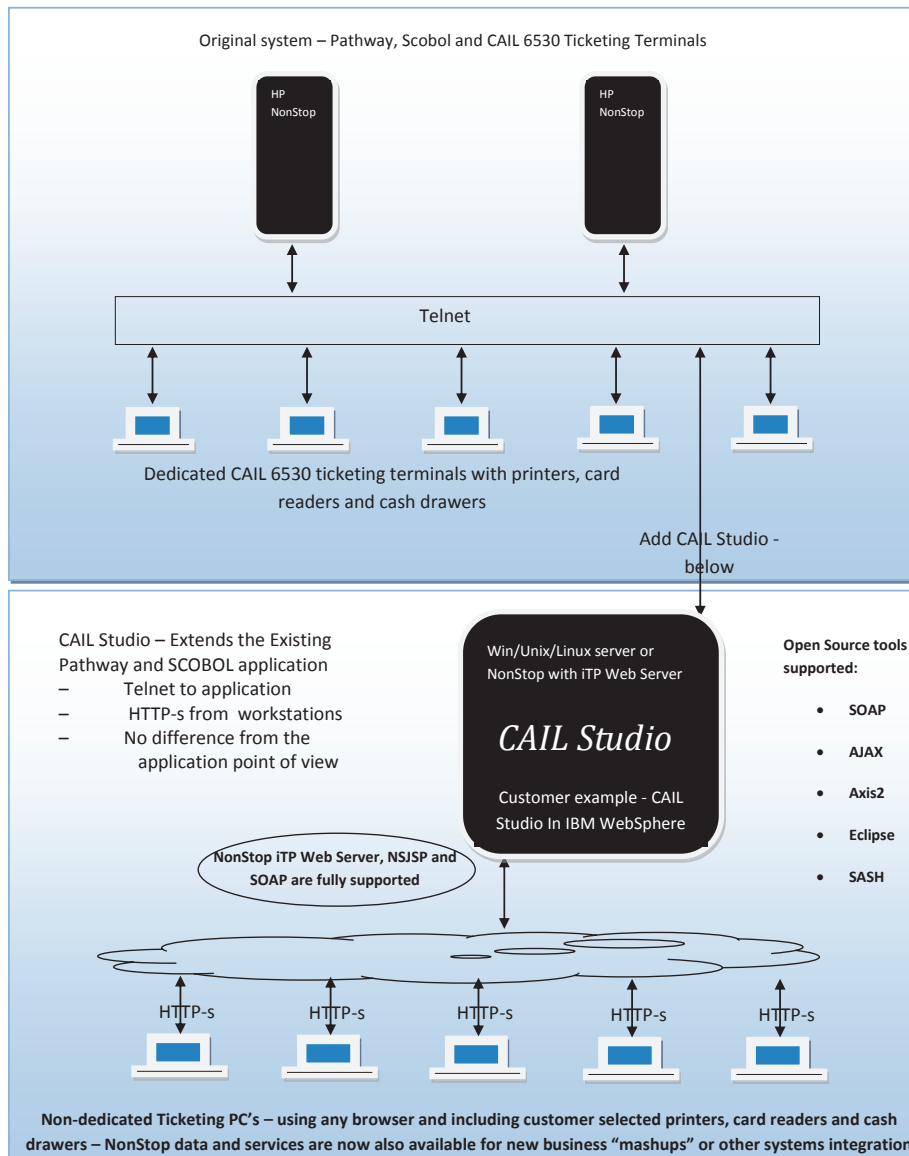
self-employed agents. These agents would sell tickets for transportation over a secure-HTTP session using only a standard browser and with very little or no training. In a manner of speaking, they needed to use the existing NonStop application as a "service" for the new business function of remote sales using a simplified and Web-oriented client method.

During the customer's investigation and demystification stage, prior to contacting CAIL, it had consulting engineers locate several existing third-party connector software providers for 6530 access from Java. The idea was to create a Java Servlet that could engage in standard sessions and to price and sell tickets using ordinary computers and HTML Web pages. During the workups of the early software solutions, the consulting engineer encountered several insurmountable problems, with the non-CAIL alternatives.

This customer was a user of the existing CAIL Suite enabling it to emulate the 6530 protocol on Windows client systems and enabling it to do its work in normal 6530 mode, as it proceeded with this investigation. CAIL

had been working on a new product with feedback from this customer and others that fit this customer's requirements, but was an as-yet unadvertised product, now called the CAIL Studio and Library. The customer contacted us for assistance and received custom changes that would enable its new business solutions.

Using what is now the CAIL Studio Library and Servlet solution set, in short order the customer had its existing software functioning securely across the Web, and it was testing with real transactions through its existing Pathway and Scobol environment. CAIL helped this customer verify the proposed solution against very high loads and in a 24 x 7 production environment as it would encounter in full production. This involved analysis and troubleshooting to hosting Java environments on



Continued on page 41

HP NonStop Cluster Essentials: NonStop and Linux Cluster Management Using a Single User Interface

continued from page 21

perform historical analysis. Both high-level and detailed inventory reports can be created.

Integrated Performance Monitoring

For customers who wish to monitor cluster performance in the same integrated view as HP NonStop Cluster Essentials, HP provides another optional plug-in for cluster management called HP NonStop Cluster Performance Essentials. This product seamlessly integrates with HP NonStop Cluster Essentials to provide:

- Integrated view of all systems in the cluster to monitor real-time performance issues across the cluster

- Historical performance trends for the cluster as a whole and for individual systems on the cluster
- Drill-down capabilities to enable investigation of performance bottlenecks
- Alert notifications based on defined performance thresholds

For More Information

More information on HP NonStop Cluster Essentials and HP NonStop Cluster Performance Essentials can be found at:
www.hp.com/go/nonstop/clusteressentials
www.hp.com/go/nonstop/performanceessentials

Service Oriented Architecture: Example Client Implementation on NonStop

continued from page 34

various systems and OS's including BEA WebLogic, IBM WebSphere, and generalized Apache scenarios.

The very first testing environment for the CAIL Studio Library for this customer was the BEA WebLogic environment. We represented this to the client who confirmed CAIL's suitability not only for WebLogic, but also in any Java servlet container including WebSphere that it has as well and in which this solution is implemented and running today.

CAIL also added a simple Activex control that could articulate between the Banking Pin Pad Hardware and the Host Computer, across an ordinary P.C. bridge to the Web page over HTTPS to the Java Servlet.

As part of the transaction flow, the CAIL Library provides the interface to the NonStop Telserv and into the SCOBOL based Host software and back, all within critical time intervals, and under high load conditions using hundreds of simultaneous sessions provided by the customer in real-time.

The open Ajax environment was used extensively for the PinPad work, and after an example was given to the customer's software implementation staff, Ajax became part of the CAIL Studio design and is now part of the core thinking while going forward.

All of the Ajax code for this client is in Java Server Pages (JSP's i.e. host programmed script) but could be done directly in HTML pages as it behaves in exactly the same way in the browser.

The results to date: The new business functions are running in full production. Now, remote computers equipped with a cash drawer and a bank-card reader owned by self-employed agents are participating in expanded ticket sales over secure HTTPs using only a browser and virtually untrained operators and no new changes or software is required for the NonStop server.

To the existing NonStop Software, each CAIL Studio session appears identical to standard 6530 sessions, including the cash drawer and card-reading functions, so that no host NonStop software changes or additions were required to introduce the radically new business solution initiative and to deliver on the promise of a simple, and now proven profitable new business initiative.

In addition, SOA "mashups" of data collected from the NonStop Servers and other servers with critical data are now possible leaving open other new business alternatives to be considered.

This entire project was done using existing production S-series servers which are now being considered for upgrade to the new HP Integrity line of Itanium based NonStop servers. This kept the NonStop environment viable and upfront with the customer doing what it does best, keeping the business critical applications running 24 x 7 and providing new services and new business functionality to the field with minimal impact to the organization as a whole.

Check out www.cail.com/wsintro for more information. 