Implementing a Web Server on OS/390: Part IV — Interconnecting Web Applications with DB2 Databases or CICS Transaction Servers

BY PATRICK RENARD

This concluding article examines how to interconnect web applications with "the heart" of your existing OS/390 system: DB2 databases or CICS transactions servers.

THE previous articles in this series (*Technical Support* September, October and November 1998) discussed how to implement a Web server on OS/390 using ICSS and Common Gateway Interfaces (CGIs) written in REXX, C/C++ or Java. This concluding article will examine how to interconnect web applications with "the heart" of your existing OS/390 system: DB2 databases or CICS transactions servers.

ACCESSING DB2 DATA WITH NET.DATA

For connecting applictions to DB2 databases, Net.Data is an especially useful tool. This free DB2 feature (program number 5655-DB2) allows you to easily create web dynamic documents accessing DB2 data. It is a complete development tool that replaces the DB2WWW utility. A Net.Data application is a set of macros that contain HTML statements, SQL calls and REXX procedures. These macros combine the simplicity of HTML language with the dynamic functionality of web CGI programs. Net.Data is installed using SMP/E (FMID: H24C110). The product is composed of several MVS datasets and one OpenEdition file system SDTWHFS that is mounted on /usr/lpp/netdata.

ICSS CONFIGURATION UPDATES

First, you need to update httpd.conf to add an exec directive to locate Net.Data CGI, as shown in Figure 1. You also need to modify httpd.envvars to add a LIBPATH entry to locate Net.Data modules, as shown in Figure 2. You should also add Net.Data loadlib to the IMWEBSRV STC steplib concatenation, as shown in Figure 3. Note that during the installation procedure you will have to bind a DB2 plan.

NET.DATA CUSTOMIZATION

Net.Data uses a configuration file, db2www.ini, as shown in Figure 4. This file is located using the standard ICSS search strategy, so this file must be located in the root directory of your "logical" server. In our case, it is /u/imwebsrv/www01 for "logical" server www01. This configuration file contains definitions to locate Net.Data objects (macros, included files, REXX external procedures, etc.) and to specify the DB2 target subsystem. We used the directories architecture shown in Figure 5 to store Net.Data objects. The directory "exec" contains REXX procedures and the directory "macros" is used to store Net.Data macros.

The sample shown in Figure 6 describes major functions of Net.Data with a Phone Directory application. This application presents an HTML form that prompts for a last name pattern and executes a DB2 query to return a phone list of users matching this pattern. Net.Data macro ctr01s001.d2w, as shown in Figure 7, consists of three major sections: DEFINE, FUNCTION and HTML.

◆ The DEFINE section allows you to define global variables that

> **IBM's S/390 family has successfully made the migration from being traditional mainframes to being very powerful servers.**

Figure 1: httpd.conf updates for Net.Data

```
Exec           /netdata-cgi/*    /us/lap/netdata/cgi-bin/*
```

Figure 2: httpd.envvars Updates for Net.Data

```
LIBPATH=.../usr/lpp/netdata/cgi-bin...
```

Figure 3: IMWEBSRV Steplib Concatenation for Net.Data

```
//STEPLIB  DD DISP=SHR,DSN=SYS1.SIMWMOD1
//         DD DISP=SHR,DSN=SPRODUIT.NETDATA.SDTWLOAD
```

Figure 4: Sample db2www.ini File

```
BROWSE — /u/imwebsrv/www01/db2www.ini ————- Line 00000000 Col 001 051
***************************** Top of Data *********************************
MACRO_PATH    /u/imwebsrv/www01/db2www/macro
INCLUDE_PATH  /u/imwebsrv/www01/html
EXEC_PATH     /u/imwebsrv/www01/db2www/exec
FFI_PATH      /usr/lpp/netdata/file-data;
DB2SSID DB2S
DB2PLAN DTWGA105
ENVIRONMENT(DTW_SQL) dtwsql ()
ENVIRONMENT(DTW_DEFAULT) defcdll (OUT RETURN_CODE)
ENVIRONMENT(DTW_SYSTEM) sysdll (OUT RETURN_CODE)
ENVIRONMENT(DTW_PERL) perldll(OUT RETURN_CODE)
ENVIRONMENT(DTW_REXX) rexxdll(OUT RETURN_CODE)
ENVIRONMENT(DTW_FILE) filedll(OUT RETURN_CODE)
ENVIRONMENT(DTW_APPLET) appldll (OUT RETURN_CODE)
ENVIRONMENT(DTW_ODBC) odbcdll(OUT RETURN_CODE)
***************************** Bottom of Data ******************************
```

Figure 5: Net.Data Directories Architecture

```
                          Directory List
/u/imwebsrv/www01/db2www/
Select one or more files with / or action codes.

  Type  Perm    Changed (GMT)     Owner      Size    Fil    Row 1 of 4
_ Dir   755     01/27/1998 09:33  I990557       0    .
_ Dir   755     02/19/1998 11:21  I990557       0    ..
_ Dir   755     01/27/1998 09:46  I990557       0    exec
_ Dir   755     01/26/1998 18:28  I990557       0    macro
```

Figure 6: Sample HTML Code to Call Net.Data

```
<FORM METHOD="POST"
  ACTION="/netdata-cgi/db2www/ctr01s001.d2w/report">
   Enter a search criteria: </H2> <BR>
   Last Name : <INPUT NAME="name"> <BR> <BR>
   <I> To get a generic list use % (ex: K% => all  K - names) </I> <P>
   <INPUT TYPE="SUBMIT" value="Enter">
</FORM>
```

can be referenced anywhere in the macro.

◆ The FUNCTION section allows you to define internal functions (SQL or REXX) that will be called from the HTML section. In the SQL function, a REPORT block is used to format and display data output results from DB2.

◆ The HTML section allows you to define what will be sent to the user as the result of his query. You can mix any HTML tag with internal function calls.

> **S/390 computers are now optimized for today's needs for securely storing, managing and quickly moving large volumes of information to the users over networks.**

ACCESSING CICS RESOURCES

Several solutions are available to connect CICS to the Internet. While it is not the intent of this article to describe in detail these complex solutions, it is interesting to note the architectures that are available to allow a web browser to access CICS/ESA applications.

EXCI CGI: This architecture uses the External Call Interface (EXCI) of CICS/ESA to allow a CGI program running under ICSS on OS/390 OpenEdition to pass distributed program link (DPL) requests to a CICS/ESA region. Using this facility, as shown in Figure 8, it is possible to include calls to CICS servers in CGI programs running under ICSS.

The fundamental flow of a CGI program is the same as other CGI programs, except that in the "Data Manipulation" phase the CGI has to prepare a Distributed Program Link (DPL) request to a CICS program using the API provided with EXCI. Figure 9 shows the main logic of a CGI program interfacing with a CICS server.

The other ways to use CICS transaction processing services from the Internet do not

Figure 7: Sample Net.Data Macro ctr01s001.d2w

```
%,*** DEFINE section ****************************%_

%DEFINE ow="TSAMPLE"
%DEFINE tb="EMP"
%DEFINE where_clause=" WHERE LASTNAME LIKE '$(name)' ORDER BY LASTNAME"

%,*** FUNCTION section *************************%_

%,*** rexx cmd   ********************************%_

%FUNCTION(DTW_REXX) get_smfid() ,
   %EXEC , smfid.cmd %_
%_

%,*** sql query ********************************%_

%FUNCTION(DTW_SQL) query() ,
   SELECT EMPNO,FIRSTNME,LASTNAME,WORKDEPT,PHONENO,SEX
            FROM $(ow).$(tb) $(where_clause)

      %REPORT ,
            <TABLE BORDER=1>
               <TR>
                <TD><font size=-1>Row Number</font> </TD>
                <TD><font size=-1>Last Name</font> </TD>
                <TD><font size=-1>First Name</font></TD>
                <TD><font size=-1>Phone Number</font></TD>
                <TD><font size=-1>Employee Number</font></TD>
               </TR>
             %ROW ,
               <TR>
                <TD><font size=-1>$(ROW_NUM)</font></TD>
                <TD><font size=-1>$(V3)</font></TD>
                <TD><font size=-1>$(V2)</font></TD>
                <TD><font size=-1>$(V5)</font></TD>
                <TD><A href="/netdata-cgi/db2www/ctr01s002.d2w/report?emp=$(V1)"
                     <font size=-1>$(V1)</font></A></TD>
               </TR>
              %_
            </TABLE>
            <P>
            Total records retrieved: $(ROW_NUM)
      %_

      %message ,
        100: "No entries found for search argument for $(name)" : exit
      %_

%_

%,*** HTML section *****************************%_

%HTML(REPORT) ,
  <html>
      <head>
            <TITLE>PhoneBook Consultation</TITLE>
      </head>
      <body>
            %INCLUDE "header.htmls"
            <HR>
            <UL>
              <LI> <A href="/html/d8ct.htmls"> New Query </A>
            </UL>
            <HR>
            This request was running on partition  <b> ...get_smfid() </b> . <br>
            This Web application uses <b> $(DTW_MP_VERSION) </b> .
            <HR>
            <H2> Phone Query Results for request: $(name) </H2>
            <HR>
            <UL>
              <LI> Click on &quot Employee Number &quot for details.
            </UL>
            ...query()
            <hr>
      </BODY>
  </html>
%_
```

use ICSS but are mentioned to supply a complete view on the subject. They use two standard CICS interfaces: External Call Interface (ECI) or External Presentation Interface (EPI).

CICS Web Interface: This architecture provides a direct TCP/IP connection into CICS and is the most efficient way for a web browser to access CICS/ESA. However, it cannot interact with ICSS CGI programs. The CICS web interface, as Figure 10 illustrates, is a well-suited solution for Intranet applications.

CICS Internet Gateway: This interface allows a web browser to appear like a 3270 terminal and provides Internet access to existing 3270 applications. See Figure 11. This solution, based on an n-tier model, needs a web and a CICS client server on an intermediate system (e.g., OS/2, AIX, NT, etc.). This is also a solution for Intranet architectures.

CICS Gateway for Java: Shown in Figure 12, this is a Java application that allows a Java-enabled web browser to download a Java applet and then access a CICS/ESA application through an intermediate web server and a CICS client. The CICS Gateway for Java combines the portable and neutral architecture of the Java programming environment with the power, high integrity and robustness of CICS. It also enables workstations' Java applications to communicate with CICS server through a CICS client. It provides an API that allows conversations between an application or applet in Java and a transactional application running a CICS server.

The CICS Gateway for Java consists of two components: a supplied Java application that resides on the intermediate web server workstation and communicates with the CICS server through a CICS client, and a Java class library that includes three classes that provide APIs, enabling communication between a Java application or applet running on the web browser and the CICS Gateway for Java.

This n-tier model needs an intermediate web server on an OS/2, NT or AIX workstation. Also available is the CICS Gateway for Java (MVS). This gateway runs directly on OS/390 OpenEdition and can communicate with CICS Transaction Server 1.2.
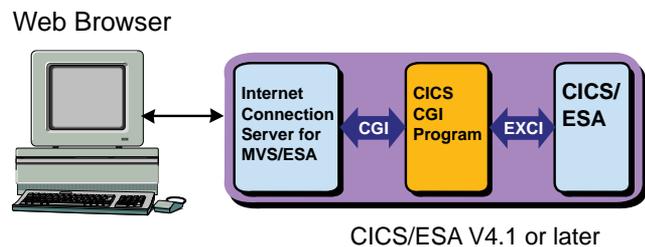
Figure 8: CICS EXCI CGI Architecture

Web Browser

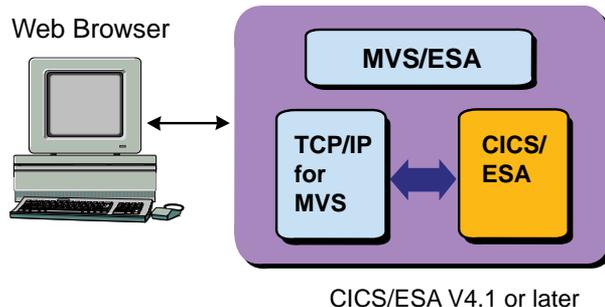Internet Connection Server for MVS/ESA — CGI — CICS CGI Program — EXCI — CICS/ESA

CICS/ESA V4.1 or later

Figure 9: CICS EXCI CGI Program Logic

Server

Client (Web Browser)	Internet Connection Server	Gateway Program

GET/HTTP/1.0 — Parse and process request

HTTP/1.0 200 (normal completion) send HTML document

```
<html>
<body>
<form
action=/BonusCGI2/gate
method=POST>
</form>
</body>
</html>
```

Search for gate using Exec directive
Spawn CGI space
Translate user data to EBCDIC
Pass environment data and user data

Accept user data from STDIN
Parse user data
Translate escaped characters

Fill out form
URL encode data
Send to server with POST

Prepare Distributed Program
Link request to CICS
Issue request, check response
Prepare response in HTML

Send response to server

Perform any required server processing — (STDOUT)

Translate to ASCII

Display resulting page — HTTP/1.0 200 (if normal completion)

Figure 10: CICS Web Interface Architecture

Web Browser

MVS/ESA

TCP/IP for MVS — CICS/ESA

CICS/ESA V4.1 or later

Figure 11: CICS Internet Gateway Architecture

Web Browser

Web Server — CGI — CICS Internet Gateway — EPI — CICS Client — CICS Server

OS/2, AIX, NT
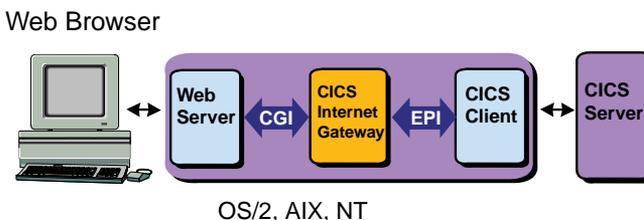
Figure 12: CICS Gateway for Java Architecture

Java enabled Web Browser

Web Server

CICS Gateway for Java — ECI / EPI — CICS Client — CICS Server

or Network Computer   OS/2, AIX, NT, Solaris
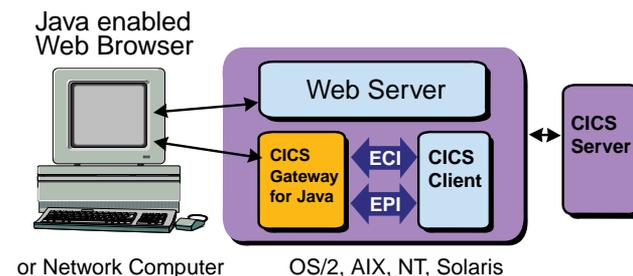
Using this gateway, MVS Java applications can create local gateway connections to CICS servers.

CONCLUSION

IBM's S/390 family has successfully made the migration from traditional mainframes to very powerful servers. S/390 computers are now optimized for today's needs for securely storing, managing and quickly moving large volumes of information to the users over networks.

OS/390 Release 5 includes a brand new TCP/IP stack that is a complete rewrite of the product. Enhancements in TCP/IP performance result from using specialized hardware functions (OSA) and exploiting new hardware instructions in G3 and G4 CMOS systems. The new TCP/IP stack and OS/390 R5 can handle up to four times the amount of work for the same CPU consumption as the previous release of OS/390. Additionally, IBM has announced their intentions to enable ICSS for OS/390 to participate in S/390 Parallel Sysplex environment. Given its degree of functionality, OS/390 offers some of the most advanced platform to develop your e-business applications. *ts*

NaSPA member Patrick Renard has been an MVS systems programmer for eight years. His experience includes DB2 Data Sharing and Parallel Sysplex implementation. He can be reached at renard@mygale.org.