

TEC 2002

*Unleash
the Power
of your iSeries*

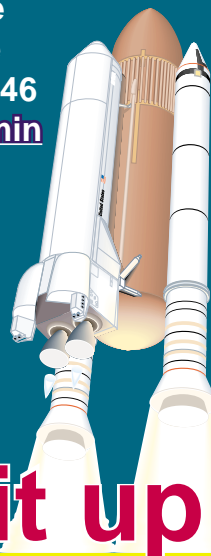


May 14 - 15, 2002
Sheraton Parkway Hotel

The Toronto Users Group for Midrange Systems (TUG) once again presents the annual Technical Education Conference and Showcase. We will feature the latest hot topics in the iSeries realm and the top-most expert speakers. This year the early-bird cut-off will be **March 22nd**. Members who register before this date qualify for the discounted rate of **\$695** for the full conference.

For more information contact the TUG Office
905-607-2546
e-mail: admin@tug.ca

Keynote
Speaker:
Al Barsa



Rev it up!

Using HTTP Beyond Your Browser

By Lana Ho

Using the power of HTTP, you can extend your systems to interact directly with your trading partners. This article offers an overview of how one can use socket applications to connect with external servers over the Internet. All you need is an Internet connection, and a basic understanding of socket programs.

Where would you use this? Some examples:

My business has set up a B2C shopping cart site, with dynamic access to my database to display my product catalog. Users can search my products, place an order which will connect with my back-end order processing system. How do I extend the benefits of my site to interface with 3rd party businesses for:

- shipment tracking
- on-line credit card processing
- supplier inventory checks, (I might not carry all my inventory), customer inquiry
- B2B procurement – e-marketplace or private exchange with trading partners, product inquiry, order placement

I want to have all these processes in place, without losing my customer to another site, in real-time...

What Is a Socket Program?

There are many variants of socket programs and they can be extremely complex, however, for my intent, I need to be able to run a client program from my server machine. There are two ends to a socket connection:

- The **server side** that has a socket that is bound to a specific port number. The server just waits, listening to the socket for a client to make a connection request.

- On the **client-side**: The client makes a request to a server via the appropriate URL. The service then returns the response to the client through the same socket connection.

It's like web browsing without a browser

My approach is to use a socket program to make a request to another server via the Internet. The socket application makes a request through an HTTP POST, and receives the result of the processing request as the response to the POST request. These requests are made through a TCP/IP connection over the Internet to the service. For sensitive data, an extra security layer, SSL protects the HTTP requests.

What information is sent via the sockets?

The client side: HTTP requests:

Data in the both the request and response is URL encoded using standard HTTP conventions. A typical request message is built up of a request header and request message.

1. Request headers are all of the form: Header-name: Header Value and they specify information and parameters that will help the server provide a suitable response.
2. Message Body: This is the actual data segment being sent. Fields are created as URL encoded name=value pairs.

The server side: HTTP responses:

The response to a request is returned as an HTTP header and then the URL encoded string containing a number of name=value pairs, which is the data segment being returned.

Some application coding will need to be done in order to parse the response.

How do I write the socket programs?

The best way to code a client/server program is to reuse code or examples from an existing service. There are many public domain examples to work from in C, Java, RPG, etc.

A server written to the sockets API for a connection-oriented protocol typically uses the following sequence of function calls: (These APIs are available for OS/400 implementation)

- A call to *socket()* to obtain a socket descriptor.
- Socket calls to activate a connection. It calls *bind()*, *listen()*, and *accept()* to activate a connection for a server program.
- The HTTP header and message request information is passed and interpreted using calls to the *read()* and *write()* APIs.
- A call to *close()* to destroy the connected sockets.

A client written to the sockets API for a connection-oriented protocol typically uses the following sequence of function calls:

- *socket()*
- A call to *connect()* to activate a connection for a client program.
- A call to *close()* to destroy the connected sockets.

Additional API calls are available for an application that requires Secure Sockets Layer (SSL) encryption.

Using socket programming may appear arduous at first, but once your first socket application is written and in place, you will be able to extend the same application again and again. [TUG](#)

Lana Ho, a Services Manager at LANSa Inc., has designed and developed multiple e-commerce-style Web sites. She can be reached at lana.ho@lansa.com

[Originally printed in iSeries Magazine]

Example 1: (shipment tracking request)

```
POST /application path/tracking.cgi HTTP/1.0<cr/lf>
Content-type: application/x-www-form-urlencoded<cr/lf>
Content-length: 150<cr/lf>
<cr/lf>
AppVersion=1.0&AcceptUPSLicenseAgreement=yes<cr/lf>
&ResponseType=application/x-tracking-full-response<cr/lf>
&InquiryNumber=1Z4523740341077981&TypeOfInquiryNumber=T
```

Example 2: An example of a response message would be:

```
HTTP/1.1 200 OK
Server: Netscape-Enterprise/3.5.1
Date: Mon, 04 Jan 1999 15:58:03 GMT
Content-type: text/plain

Status=E&erCode=131&erString=Operation+not+supported+%28131%29+
Please+check+request+parameters&subError=0
```

Things to do When Mid-Range is Your Business Partner: #18

JOIN A COMMITTEE

And after the meeting, call another meeting. Because, at Mid-Range, we're experts at keeping your iSeries 400 – AS/400 operating at peak performance. From CRM, BI, Lotus, Web Development, iNotes, ERP & Supply Chain solutions / hardware upgrades and performance tuning through logical partitioning, operational support / education and disaster recovery we have what you need. So you get more time to concentrate on your business. *And even more meetings.*

MID-RANGE 

Working For Your Peace of Mind

Call: 1-800-668-6470 www.midrange.ca