



# With Sockets, DISC Bridges Internet For Distributors

By Elizabeth Barnett

The stated vision of Distributor Information Systems Corporation (DISC) is to become an integral part of its Customers' businesses. With new I-Lumin.Net, its e-business solution, the company is positioned right where it wants to be: smack in the middle of the e-business conducted by DISC Customers.

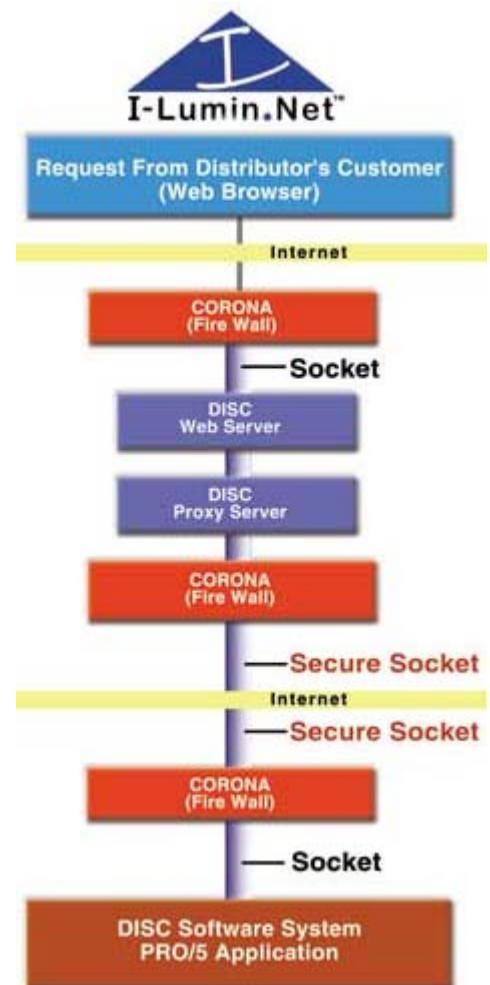
"We build bridges," says Ray Hrdy, Vice President of Technology for the Farmington, Conn., company that does about \$10 million in revenue annually. "DISC is right in the middle between our distributor-Customers and their Customers. Instead of competing with other companies, we build bridges to whatever and whomever our distributors and their Customers need."

Software sockets with PRO/5® are one of the ways DISC, makers of enterprise resource planning (ERP) software for the industrial distribution sector, has chosen to build those bridges. A socket is an end point of a two-way communication link between two programs running on a network. I-Lumin.Net uses sockets to connect PRO/5 programs to DISC's distributor-Customer Web sites in order to process requests from their end-user Customers. DISC hosts the I-Lumin.Net application for more than 500 of its distributor-Customers with about 20,000 end users among them. The average DISC distributor-Customer has about 5,000 to 8,000 end-user Customers.

I-Lumin.Net is automated customer relationship management (CRM) for distributors. The application seamlessly allows a distributor's end-user Customers to access the distributor's DISC software system to look up invoices and inventory items, place orders, track orders, etc. The distributor's Customer accesses information stored in the distributor's DISC software system through a login link on the distributor's Web site. The link connects to a Web server at DISC headquarters. The Web server at DISC sends the request through a socket connection to a proxy server at DISC, and then through other socket connections to the distributor's own DISC software system to process the request and retrieve information. I-Lumin.Net essentially works like this:

1. PRO/5 programs run as ghost tasks and create a persistent connection to the DISC proxy server the moment the distributor's DISC software system is booted up.

Figure 1. This diagram illustrates how DISC uses sockets in I-Lumin.Net. It shows the basic route a query takes in the application.



The tremendous benefit here in using sockets is that the socket connection remains open from that point on but consumes no CPU resources when it is idle. It is a completely event-driven process that produces an immediate response once a request is made (unlike using CGI, which must open a PRO/5 environment and load the PRO/5 program each time a request is made).

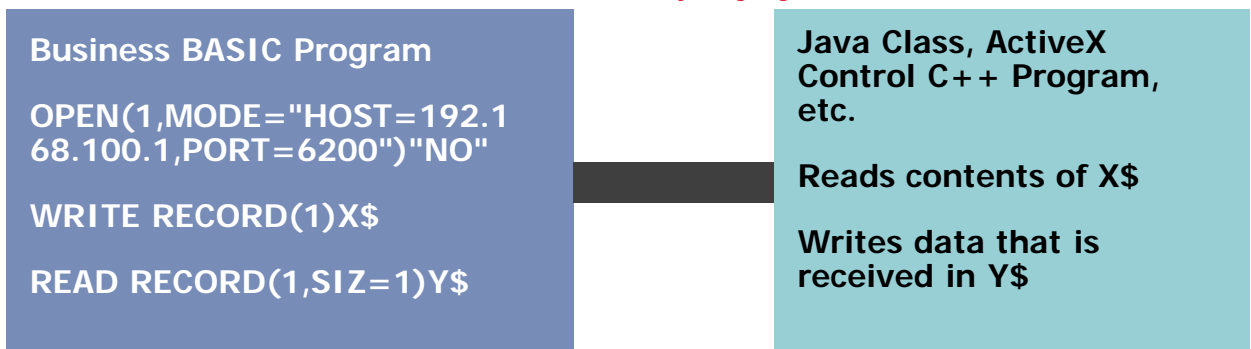
2. The application reads from the PRO/5 files.
3. The application writes to the socket connection using XML formatting.
4. On the Web server at DISC, the XML formatting is converted to HTML using Active Server Pages and XSL Transformations.
5. The application returns the HTML response back to the distributor's Customer on the distributor's Web site.

To the distributor's Customer, however, it appears that he never left the distributor's Web site. All the distributor's Customers see is the distributor's branding and Web look. DISC, the company, and its involvement in the transaction are invisible.

"The real beauty of sockets," Ray explains, "is that they allow any external program to access the middle of a running Business BASIC program. For any system that needs to talk to Business BASIC applications, sockets are one of the best methods to do that."

And unlike ODBC, sockets allow access to whole programs, not just data. "The investment DISC has is in the program, not the data itself," Ray says. "Sockets allow us to fully leverage the complex business logic that's wrapped up in our Business BASIC programs." The other end of the socket can be anything, any language, any program, any other application. Shown in Figure 2 is sample code that can connect any Business BASIC program to any external program.

Figure 2. The three basic steps to using sockets with a Business BASIC program: (1) open the connection, (2) write data to the remote program and (3) read data from the remote program. The receiving program can be written in any language.



"The trend in distribution is toward consolidation and the commodity brokering of products," Ray says. "Distributors don't just sell products; they really sell their unique knowledge and their services. That's how they add value to their Customers." Ray explains that to help DISC's distributor-Customers add value, he is "always having to connect to different languages and new applications." For example, he cites a UPS manifest system that many of his distributor-Customers use. By the end of the year, he says, DISC will have another socket application that will integrate the UPS system with the DISC software system, allowing a distributor to directly generate UPS shipping orders using Customer data from the DISC system.

But the UPS socket application is just one in a line of socket applications being developed to make it easier for DISC's distributor-Customers to serve their own end-

user Customers. Yet another implementation is the planned Distributor Alliance Network, which will connect distributors to each other through I-Lumin.Net.

DISC plans to eventually provide means to integrate all a distributor's suppliers, alliance partners and even other portals similar to I-Lumin.Net. Ray says, "DISC wants to adapt to whatever the other guy uses, whatever that may be. For DISC, that's not a bad thing. If there are 50 different things to connect to, no Customer of mine can afford to do all 50 on his own. But I can build the sockets for him and re-use the same methods of connection for other Customers. It's great for us."

## **Benefits of Sockets**

- **Use Existing BB<sup>x</sup> Call Programs**
- **Stay Running Between Transactions**
- **Don't Use Any CPU Cycles When Idle**