Getronics

VS Computer Systems

VS Web Server

Open Source Freeware allowing Internet Access To Applications and Data resident on the VS

Highlights

- New product providing HTTP protocol services on a VS.
- Allows access to VS data from any browser.
- Can be used in an intranet, extranet or Internet environment
- Provided as Open Source Freeware and downloadable without restrictions
- Open source model will allow VS user community to control the evolution of this product.
- Allows users to control access to data on their VS.

Introduction

The Internet has become the de-facto standard for communication between computer systems. The nearly universal availability of local access points, combined with low access costs even for high bandwidth connections have made the internet the communications structure of choice.

The Internet, and specifically the World Wide Web, is the single greatest repository of information ever compiled. Comprised of hundreds of millions of pages of data on millions of servers and hosts, the Web is viewed as the gateway to all knowledge.

Of equal importance to corporate and government managers has been the widespread adoption of Intranets, private extensions of the Internet and WWW used to facilitate the internal availability of data needed by employees to perform their assignments.

Of equal importance, the Web is highly standards based. The use of well-defined protocols and standards in the presentation of data masks all of the intricacies of the server and the communication network from the user. The user rarely has any knowledge of the server that delivers his data.

The Internet and Intranets are no longer an experiment. They are no longer a competitive edge. They are mainstream technology whose adoption and use are critical to the success of the enterprise.

The Internet is evolving and rapidly moving beyond simple WWW access. A generation of client/server applications is evolving. These applications allow true worldwide distribution of end-users and allow these users to access and update server based data at any time. This shift of applications to the Internet is an indication of the true destiny of this technology as a mainstream business element.

Browsers and Servers

Each web server provides a home page and other pages of information written in the Hypertext Markup Language (HTML). Hypertext links on the pages link together information on different servers. By navigating these links, users can access the information they need on many different systems without actually logging on.

The user negotiates the web using a Web browser (client) which functions as a markup language interpreter. The web browser reads the HTML information and formats the data accordingly. The web browser can send data to the server. In some cases the server will launch a program to process the user request or data.

The VS Web Server

The VS Web Server, an application that runs on the Getronics VS system, allows an installation to use their VS as a repository of and delivery agent for Web pages. This means that information that is available on the Web enabled VS can be made available to anyone who needs it regardless of their location.

The VS Web Server is a HTTP (Hypertext Transport Protocol) server that operates on the Getronics VS computer system. The VS Web Server allows the VS to be a server member of the Internet, or a customer Intranet or Extranet. The VS Web Server uses the VS TCP/IP subsystem. Within the constraints of customer established security it allows any user on any computer, anywhere in the world to access the VS via a standard web browser such as Microsoft Internet Explorer or Netscape Navigator.

Free Distribution and Open Source

The rate of change and innovation in Internet technology is high and accelerating. Getronics has concluded that sharing this technology will be in the best interests of the VS community. The Web Server will be available at no charge to VS licensees.

Getronics is committed to creating a user community with open access to the VS Web Server Source code.

The open-source model has a lot to offer the VS community. It's a way that many companies and individuals can collaborate on a product that none of them could achieve alone. It's the rapid bug-fixes and the changes that the user asks for, done to the user's own schedule. Even more, it is a concrete

way to demonstrate the ongoing viability of the VS platform to the outside world.

The business model to be used is like that used with LINUX. The VS user community will be encouraged to provide enhancements to and maintenance of the source code. The only conditions will be that any fixes or enhancement to the WebServer product will be required to be freely available and in the public domain.

Gradient Levels of Application

From simple to complex, the VS Web Server will find levels of application that suit the customer's requirements and willingness to tailor the server environment. Notable steps in this gradient may be viewed to include, in increasing order of robustness:

Desktop access to Static Web pages

This is the simplest level, at which there is no involvement of VS application databases. The VS as a dependable primary computer system is used as the repository and server for externally created informational documents that are formatted in HTML.

HTML documents would be created and maintained using standard PC tools such as Microsoft Front Page, transferred to the VS using ftp and then posted to the VS Web Server database.

This usage may be most appropriate in an environment where the VS is connected to an intranet and used to host internal documents (such as policies, procedure manuals, employee info, training pages), particularly for occasional use.

Access to web pages generated on the VS.

This is the next level of complexity. Simple VS programs in virtually any language capable of file manipulation can create HTML documents either directly or with the help of a "merge" utility, to provide information of a changing nature. Note that the information still needn't have anything directly to do with VS databases or applications, although it certainly can.

Information of a changing nature could be anything from the company's Snow or Severe Storm Emergency status ("We're closed" or "We're opening late") to advisory pages on the status of networks and system availability to periodic status or aging reports that do not need to reflect current live data in real time.

The most common use of this capability will be to run overnight, weekly or monthly reports using data on the VS and then publish the listings on the Web site as an alternative to hard copy distribution.

The same justifications apply here as in the case of static information pages, with the addition of the ease of writing small VS applications to generate ASCII files on whatever basis desired. A small PACE application, for instance, could provide for text entry that is formatted into presentable HTML pages by very simple HLI programs. The resulting HTML files are simply written to the file/library/volume locations that are the targets of links already contained in static Web pages that act as menus.

Desktop access to VS application data

The VS Web server supports CGI (Common Gateway Interface) within the VS environment. A browser request to the VS will cause the VS Web Server to launch a VS program, pass it parameters, and then format and send the resulting output back to the requesting browser. VS CGI programs may access any kind of native VS data, including PACE databases. VS CGI programs may generate HTML directly, or generate ASCII data that is merged with a pre-formatted HTML template by a facility that is built into the VS Web Server.

Some application programming is required to do this, but no TCP/IP or complex HTML coding is required. Template Web pages are constructed with any Page Design Tool on a PC. The applications executed by the server in response to specific URLs sent by desktop browsers make use of an HTML merge layer that merges application result fields with template HTML documents, producing presentable Web pages containing application data in response to user queries or requests.

The VS application code that does this is moderately portable, in the same sense that a customer's COBOL apps are portable, except that this application code is probably more portable because it operates within a well-defined set of layers and does not contain traditional user interface code.

This level of access might be used to allow a field representative to look up the status of a specific order, or to create an order for submission to your system.

Remote System Access to VS data.

At this point we depart from the usual, intuitive browser centric Web model. A non-VS system using HTTP (the Web server/client protocol) can make requests of the VS Web server without regard to displayable HTML content. That is, another system can use TCP/IP and the VS Web server to cause executables to run on the VS to return any kind of data in any format agreed on. The VS Web server is transparent.

This method of using a Web server is non-intuitive because it has nothing to do with displayable Web pages. It takes advantage of the fact that a certain class of HTTP requests causes the server to pass the request to the executable program addressed by the URL, together with optional parameters, and to pass back to the requestor any output generated by the executable. The content of the request and the content of the response can be any data in any format agreed to between the programmers of the applications on the two systems.

This capability allows another host computer on your network, especially one of a different type and make, to interact with a VS application and to get data from and

send data to the VS. File Transfer Protocol (ftp) is frequently used for such transfers, but leaves much to be desired, as ftp is a batch operation that transfers entire files.

HTTP provides something more akin to Remote Procedure Call in that a requestor can cause the execution of the program that generates the data the requestor wishes to receive or accepts the data the requestor wishes to send. Retrieved data can originate from VS files, be dynamically constructed, or any combination of the two.

An example of usage might be a situation where a company has applications distributed across multiple server platforms, including a VS. Some NT or Unix applications may need to retrieve or update VS resident data. Although the file structures and programming languages may differ dramatically, with HTTP, it is quite feasible to link non-VS applications to VS data.

PC Client Application Access to VS Data

This is a subset of the previous scenario, but in this case the "foreign" program is resident on a client PC rather than a remote host. Again, traditional "Web" pages are not necessarily involved. A custom PC program uses HTTP to run VS executables, passing back and forth whatever control and data messages the application designers have agreed upon. Typically the client program will be written in Visual Basic or a comparable language, and use commonly available internet controls to package the HTTP requests.

Using the VS Web server allows the developer to make the presence of the VS totally transparent and to avoid the need for custom TCP/IP programming on the VS. Forcing the dialogue into the HTTP standard envelope makes it both portable and reusable. This approach also allows the developer to avoid custom TCP/IP programming on both the client and the VS.

In this usage, the end user does not need to know that a VS is used. They interact with a graphical user interface on the PC. The program does the rest of the work. The significant potential of this approach is that it allows the user-visible data entry and query parts of an application to be rehosted using a screen oriented client application. The processor intensive work, the overnight and periodic closing processes and programs, and the maintenance component, can continue to reside on the VS.

Many installations find that 10% of the program code – the end user data entry parts – is all that most users see. This code may be 90% of the interactive CPU execution time.

VS Web Server Features

Supports Standard Web Server File types

The Web server can deliver text and binary files. Text files can be encoded in HTML or be plain text or VS print files. Binary files are PC files that are stored on the VS and delivered to the clients by the Web Server. The VS usually does not need to manipulate the content – only to deliver it. Typical formats include JPEG, GIF, MIDI, WAV, MPEG, PDF, as well as Microsoft word, Excel and PowerPoint Documents.

Web Server File Mapping

The Internet URL structure uses a familiar structure of /directory/subdirectory/subdirectory…./longfilename.ext. The Web Server provides a mechanism to map these lengthy file names to the VS “FILE in LIBRARY on VOLUME” structure.

Web Server Time Stamping

Most pages sent on the Internet are time and date stamped in order to facilitate orderly operation of the network. The VS Web Server supports full planetary Time Zone and GMT date/time translation and delivers documents with standard Internet time stamps.

Web Server IP Filtering

The VS Web Server provides a facility that optionally restricts access to specific networks and host machines. Traffic may be limited to the local Intranet in a non-firewall environment through the use of masking.

Any number of inclusions or exclusions is permitted.

This feature allows a Web Server administrator to interface a VS to the Internet and yet restrict access only to selected machines, networks or locations.

Common gateway Interface (CGI)

The VS Web Server supports CGI within the VS environment. This allows the VS web site to design pages that call VS executable programs, in any language, to access any kind of VS data including PACE databases. VS CGI programs may generate HTML directly or may make use of a merge capability to place the data into predefined Web pages.

Web Server API

The VS Web Server API is a straightforward adaptation of the CGI standard to the VS environment, with the addition of an optional HTML merge feature. VS CGI programs are invoked as subroutines or subprograms by VS Web Server, passed the arguments from the remote client, and pass back whatever output they generate. VS Web Server optionally merges the output with predefined HTML templates, and routes the output through TCP/IP back to the remote.The Web Server API masks all TCP/IP details from both user and application and thus simplifies development.

Implementing VS Web Server

VS Web Server utilizes services provided by VS TCP/IP. TCP/IP on the VS normally requires an 802.3 I/O controller and a physical connection to an 802.3 local area network. Connection to the Internet, or to other external networks will be accomplished by a router or other

dedicated communication hardware. In some instances it may be appropriate to run TCP/IP directly over an X.25 or other conventional telecommunication link.

TCP/IP requires the presence of OSN Netcore and an 802.3 transport in addition to the 802.3 controller.

VS Web Server is supported only in VS Operating System 7.53.00 and later. While it may be physically possible to install and run this software on earlier OS releases, these earlier releases are not Y2K compliant and will not be supported after December 31, 1999. No formal Quality Assurance testing has been done on earlier releases.

Web Server Compatibility

The VS Web Server application can coexist at the same time and on the same controller with other TCP/IP applications, such as Telnet, FTP and TCP API based applications.

Warranty

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

Ordering Information

The Getronics VS Web Server is distributed without charge, subject to the acceptance of license terms and conditions. The software, documentation and source code may be downloaded from www.VSWebCenter.com or other Internet sites.

Implementation Services

Although the physical installation of VS Web Server is not complex, the creation of an effective strategy for presenting VS based data to your clients and visitors may require considerable effort. Getronics' VS Professional Service Consulting Practice may be engaged by VS installations to provide assistance in implementing Web Access to the VS.

Getronics support center will also offer question and answer service for the web server on a fee basis. Call blocks allowing for five or more calls may be purchased from a Wang representative or the support center.

Customer Education classes will be available on a fee based basis, and may be delivered at Getronics facilities or customer

If TCP/IP has not been installed, Getronics can provide implementation and installation planning assistance needed to interface to the LAN and to integrate VS TCP/IP into your system in a minimum time.

Additional Site Management Tools

HTML Editor: Getronics can provide, as an unsupported VS Useraid, a HTML editor that can be used to fine tune HTML files when minor adjustments or changes are needed. This editor can be downloaded from a Getroninc s FTP site at no added charge. Instructions are included with the Web Server Product.

Getronics
Field Integrated Managed Services
836 North Street
Tewksbury, MA 01871 USA
1 800 639-9264