



## Barracuda EventHandler<sup>TM</sup>

*Makes it possible to create real-time graphical user interfaces, which run inside a browser such as Internet Explorer.*

The Barracuda EventHandler plug-in allows a user to remotely control and supervise any kind of device, in real-time. No software needs to be installed on the client (browser) side, thus making it possible to use anything from a PC to the latest PDA's for the graphical user interface.

### Persistent Connection

The Hypertext Transfer Protocol (HTTP) is a stateless protocol, initiated from the client side. Ordinary web pages and web-services are limited by the constraints in the HTTP protocol. It is impossible to design true, *rich-client* interfaces using web-services like XML-RPC or SOAP since the server closes the communication link each time it sends a response message for a client request. In contrast to this, Barracuda EventHandler (Barracuda EH) keeps the communication link open, thus making it possible for the server or client to send messages, at any time.

Barracuda EH gives Web developers the ability to add server-based notification functions and two-way, real-time data exchange to browser interfaces. This improves the functionality of a wide range of Web applications, and extends their capabilities to such functions as alarm handling, live monitoring of devices, etc.

Barracuda EH is a protocol stack that enables real-time data exchange between a client browser and an embedded server. The protocol stack, which is implemented in JavaScript code for the client side and C code for the server side, maintains a persistent, bi-directional communication layer (over TCP/IP) between the server and the connected client, or clients. Because Barracuda EH is tunneled on top of HTTP it works through corporate firewalls.

### FEATURES

- Plugs into the Barracuda Web-Server.
- Provides real-time, remote, browser-based GUI's.
- Use with any browser-enabled device.
- Works through firewalls.
- Creates persistent connections.
- Bi-directional asynchronous communication.
- Good for alarms and control.
- No client-side installation required.
- Reduces client/server traffic.
- Supports distributed applications.

### Dynamic HTML

JavaScript (which should not be confused with Java) is an advanced, object-oriented, scripting language. Most browsers are capable of running programs written in JavaScript. The Document Object Model (DOM), as specified by the W3C consortium, is an interface for dynamically manipulating browser content. Most browsers make this API available to JavaScript code. Hence, it is possible to upload code to a browser that can interact with the browser and dynamically change the content of the browser window. JavaScript code, together with the Document Object Model, is sometimes referred to as *dynamic HTML* or *DHTML*.

### JavaScript GUI

DHTML makes it possible to implement a *rich-client* graphical user interface in JavaScript that runs within a browser. Many JavaScript widget libraries exist to make it easier to implement user interfaces.

One such library is Bindows. (See <http://www.bindows.net> for more information.)

Unlike client-installed applications that require significant support from network administrators, Barracuda EventHandler works with zero client administration or installation.

## EventHandler Operation

The rich-client interface, which contains JavaScript code for working with the GUI, is stored in the server as a regular HTML file. The HTML file, which contains HTML for the static parts of the GUI and JavaScript code for the dynamic parts of the GUI, is loaded into the browser by an ordinary HTTP GET. The HTML file also includes JavaScript library files that are loaded into the browser. One of the loaded JavaScript libraries is the EventHandler stack.

As soon as the HTML and all JavaScript are loaded, JavaScript code starts executing. This code initiates the EventHandler and sets up a persistent connection with the server. The server and the client EventHandler stack do some initial handshaking. When this is completed, user code in the server is notified of the persistent client connection. This persistent connection is now ready to transport messages from server to client and vice versa.

## Distributed System

When the rich-client interface is loaded and running, the web-page is no longer a regular web-page. Instead, it is the basis for a *distributed system* where one part of the system is running in the browser (the client) and the other part is running in the server. Other clients can be included in this distributed system, if desirable.

The communication model employed by Barracuda EH is much more efficient than using Remote Procedure Calls (RPC) because it can handle many concurrent messages. Also, because the connection is persistent, either end can initiate sending messages whenever desired (i.e. asynchronously). This greatly simplifies design for real-time interfaces.

## Barracuda EventHandler Demo

The Barracuda EH demo includes an MP3 player, which is embedded into a server. The server runs on a Windows PC. The rich-client interface, running in a browser, allows remotely starting, stopping, and controlling music on the MP3 player.



The rich-client is implemented in JavaScript, which is uploaded to the browser. The MP3 player that is embedded in the server contains logic for calculating the VU meters, the position in the song, and elapsed time. This information is sent, via the EventHandler, to the rich-client interface, whenever new data exists. The JavaScript code in the rich-client interface updates the VU meters, the slider and the elapsed time display whenever it receives new information. As the demo shows, this occurs in real time. You may download this demo from:

[www.smxinfo.com/rtos/tcpip/barracuda/BarracudaDemo.exe](http://www.smxinfo.com/rtos/tcpip/barracuda/BarracudaDemo.exe)

## How to Order

Contact us for the latest pricing:

Sales: (800) 366-2491  
Local: (714) 437-7333  
[Sales@smxinfo.com](mailto:Sales@smxinfo.com)  
[www.smxinfo.com](http://www.smxinfo.com)