Features

Features Summary

Click on a feature for more information:

Records HTTP

HttpWatch captures a wide range of HTTP related data including:

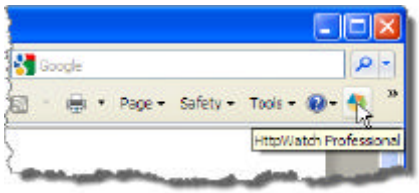
- Headers and Cookies
- URLs and method (e.g. GET, POST, etc)
- Parameters sent in a query strings and POST requests
- HTTP status codes and errors
- Redirections
- The network size of the HTTP response including headers
- Chunked encoding

Decrypts HTTPS Traffic

Secure browser sessions that use the HTTPS protocol are displayed in their unencrypted form in HttpWatch, making it easy to debug banking and finance applications.

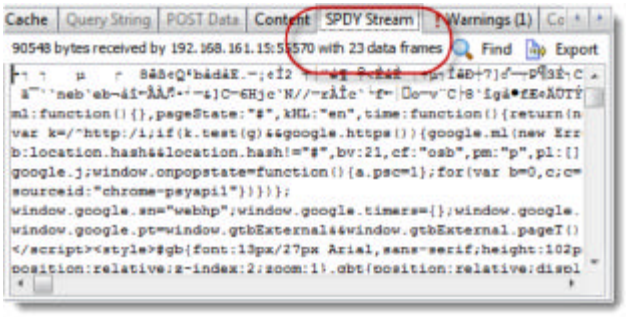
Integrates With Internet Explorer and Mozilla Firefox

Starting HttpWatch is simple and easy. An extra icon is added to Internet Explorer and Mozilla Firefox allowing HttpWatch to be opened and started with two mouse clicks:



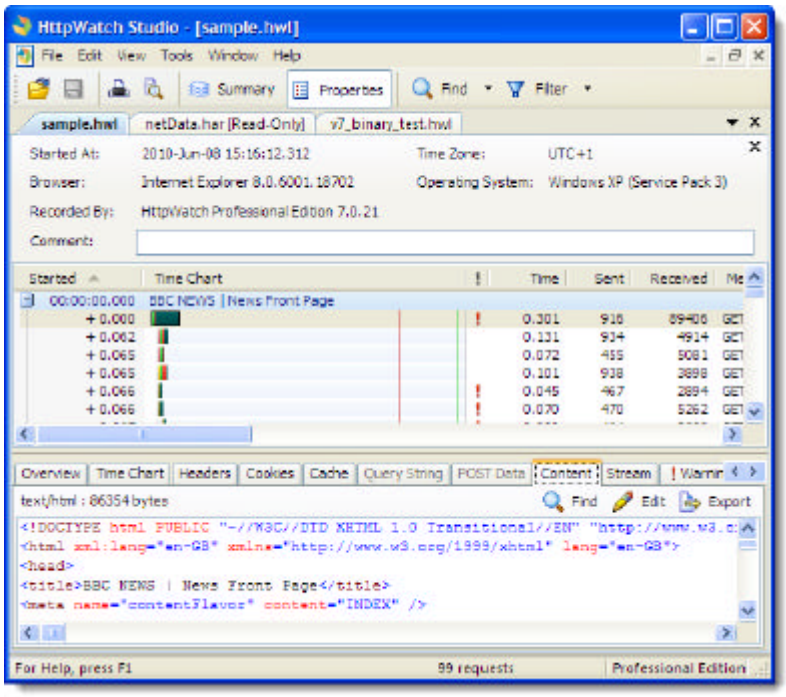
Supports the SPDY Protocol in Firefox

HttpWatch supports the SPDY protocol in Firefox 13+ showing information about the SPDY level stream:



Includes a Standalone Log File Viewer

A standalone log file viewer allows HttpWatch .hwl files to be viewed and modified outside the browser.

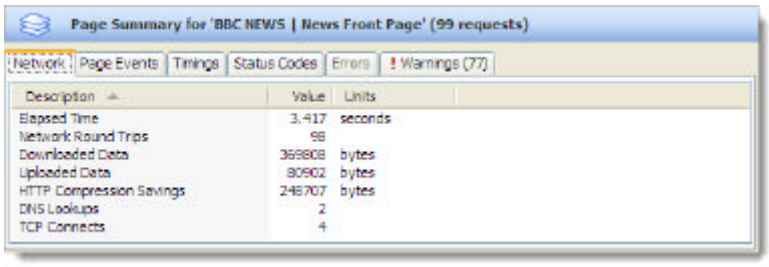


Summary of Recorded Traffic

The Summary view can be used at any time to quickly display data about the whole log, a single page or a number of selected items. The following types of data are shown:

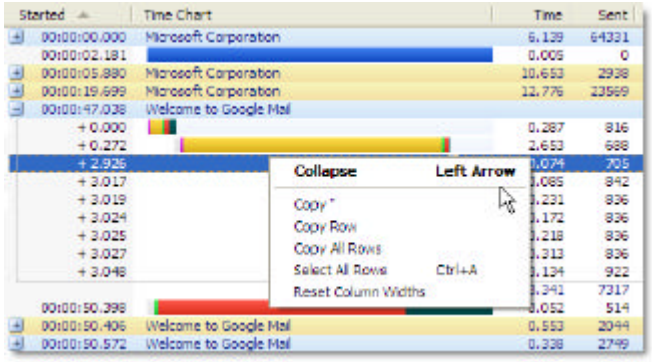
- **Performance** - this tab shows how long the requests took to complete and can be used to accurately measure page download time. It also shows how much network bandwidth was used and the savings achieved from using HTTP compression

- **Page Events** - this tab shows the events recorded during the loading of a single page
- **Timings** - this tab shows the minimum, maximum, average and total for each type of timing
- **Status Codes** - this tab records how many times each HTTP status code was received
- **Errors** - any network or HTTP level errors are displayed on this tab. Double clicking on an error finds the first request that caused the error. Pressing F3 then takes you to the next request that also failed in the same way.
- **Warnings** - any warnings that were detected by HttpWatch are summarized on this tab. Double clicking on a warning finds the first request that on which the warning was detected. Pressing F3 then takes you to the next request that also has the same warning.



Grouping Of Requests By Page

Requests are grouped by page as shown below. Each page group can be separately expanded or collapsed to aid navigation through large log files.

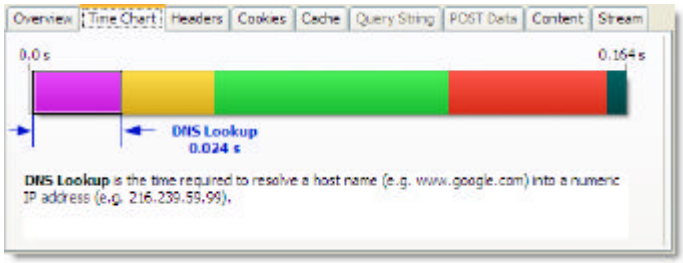


Real-Time Page Level Time Charts

Page level time charts are displayed and updated in real-time as you record requests in HttpWatch. This gives a direct, visual indication of how a site is performing - allowing common problems to be diagnosed at a glance:

Request Level Time Charts

The time chart displayed for each request is broken down into a number of colored sections to show network level timings such as DNS lookup and TCP connects:

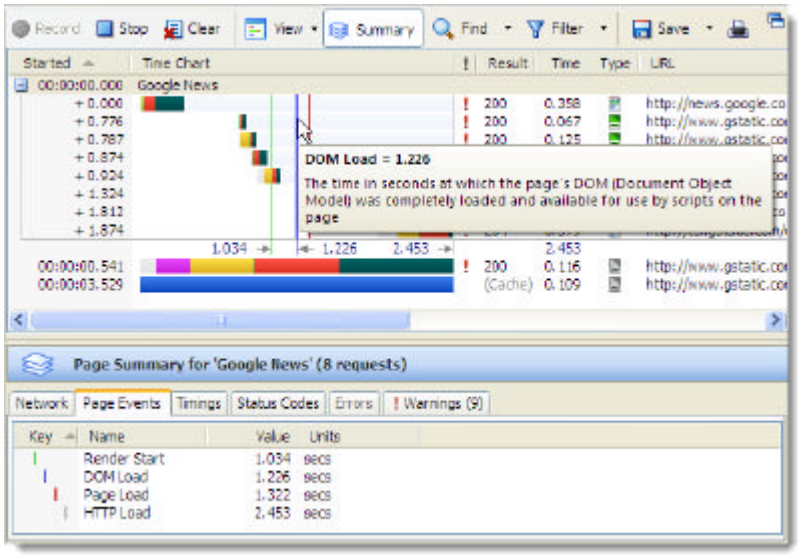


Page Level Events

Page level time charts include vertical lines to indicate these events:

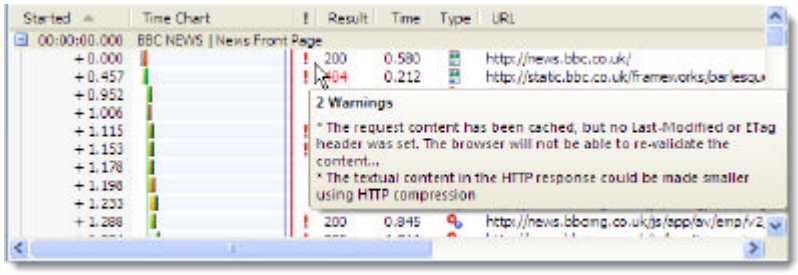
- **DOM Load** - the DOMContentLoaded javascript event fired indicating that the page's Document Object Model (DOM) was loaded and ready to be used (Firefox and IE 9+)
- **Page Load** - the page's onload event fired indicating the page and its images were loaded
- **Render Start** - the browser started to display the downloaded page
- **HTTP Load** - the end of network activity during the loading of the page

The timing of each event is available in the data tip or Page Events tab

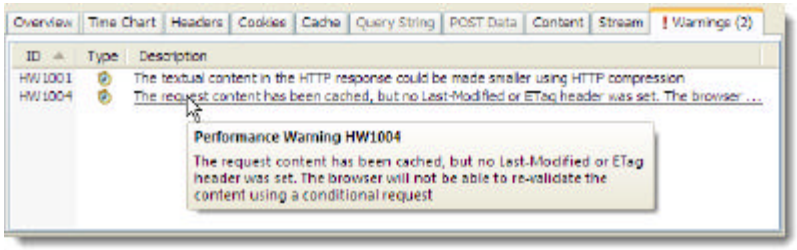


Detects Potential Problems

HttpWatch examines each request and issues warnings where problems relating to performance, security or functionality are detected. Requests that have warnings are highlighted with a Warning column marker:

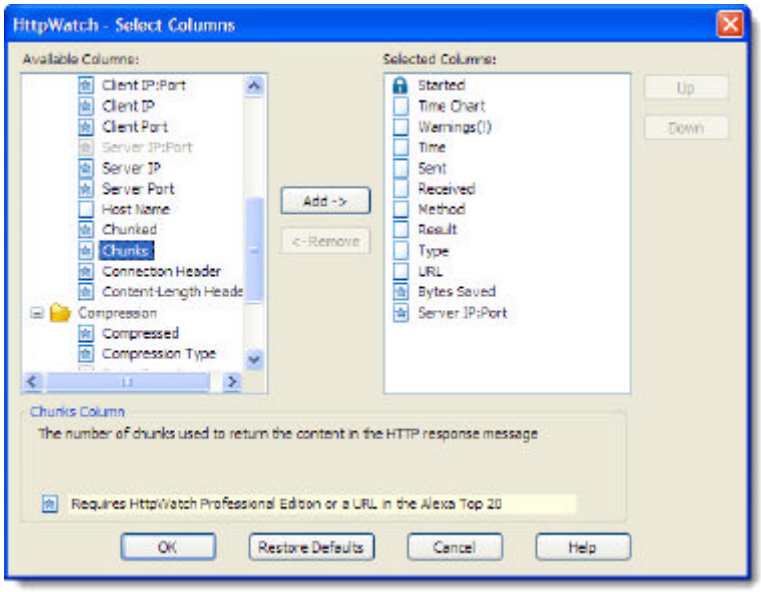


The request level Warnings tab shows the details of each potential problem:



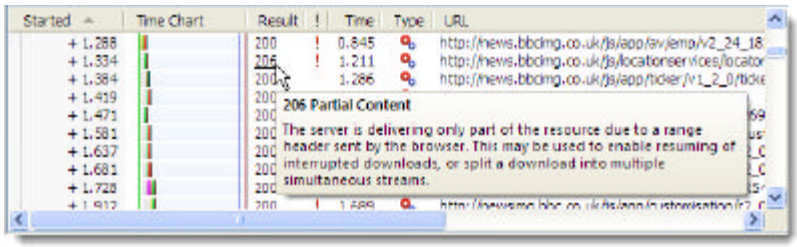
Customizable Data Columns

There's more than thirty columns to choose from in the main request grid; covering almost every data item that is displayed in HttpWatch

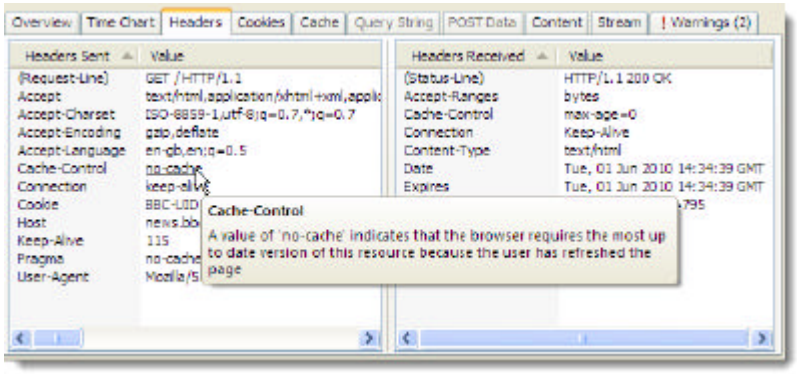


Data Tips

Data tips are now displayed when you hold the mouse pointer over an item such as an HTTP status code:



Or a header value saving you the trouble of looking up what the value means:



Collect Log Files from Customers and Users

The Basic Edition of HttpWatch saves exactly the same data as the Professional Edition in its .hwl files, If you open one of these files in the Professional Edition, you can view all the extended HTTP information (e.g. headers, cookies, timings, etc) without any restrictions.

This means that you, and your customers, can record and view log files without having to purchase extra HttpWatch licenses. Here are two scenarios where you could make use of the free Basic Edition of HttpWatch:

- **Send log files to your suppliers**

  If you have HttpWatch Professional Edition the log files you record can be viewed by anyone using HttpWatch Basic Edition. This can be useful if you want to send log files to a supplier to illustrate a problem you are experiencing with a web based service.

- **View log files from your customers or in-house users**

  You can view log files from your customers and in-house users by getting them to install the free Basic Edition of HttpWatch. They can then record a trace file and send it to you for analysis in HttpWatch Professional Edition.

Only one of the parties involved requires a license for HttpWatch Professional. The other party can use HttpWatch Basic Edition free of charge. Please see HttpWatch Basic Edition vs Professional Edition for more detail.

Automation Interface

HttpWatch has a comprehensive automation interface that can be used by most programming languages (e.g. C#, Javascript & Ruby). The interface can be used to control the HttpWatch plug-in for IE or Firefox and to access data in HttpWatch log files. If you are already running automated tests, you can integrate HttpWatch and record HTTP level information during your tests. The recorded data that then be checked for certain types of configuration and performance problems (e.g. HTTP compression is not enabled).

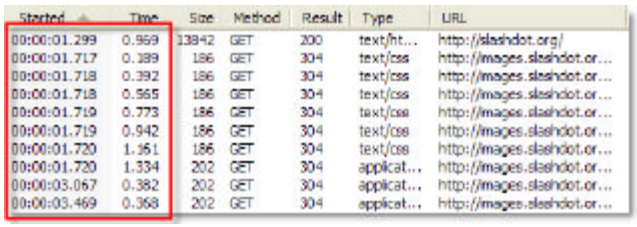
The ability to access data in an HttpWatch log file makes it possible to develop custom reports and automate repetitive tasks. The C# code, shown below, exports all javascript files from a log file.

```
// Start using HttpWatch Automation library
// and open a log file in c:\temp
Controller control = new Controller();
Log log = control.OpenLog(@"c:\temp\test.hwl");

//Check each HTTP/HTTPS request in log
uint count = 0;
foreach (Entry entry in log.Entries)
{
    // Export any javascript files
    if (entry.Content.MimeType == "text/javascript")
    {
        ++count;
        entry.Content.Export(@"c:\temp\test_" + count + ".js");
    }
}
```

Millisecond Level Timing

The timing information displayed in HttpWatch is accurate to a a single millisecond (0.001 sec).

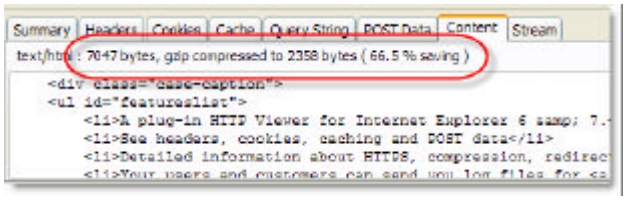


Supports Advanced Filtering

HttpWatch supports filtering of requests by wide variety of criteria such as content types, response codes, URLs, headers and content.

HTTP Compression

HttpWatch works with systems that have HTTP compression enabled, displaying the expanded content and providing information about the compression savings achieved.



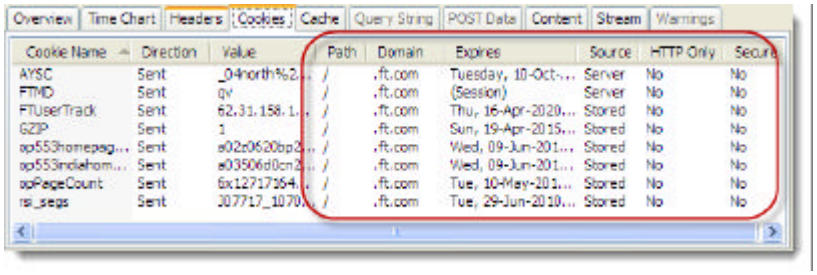
Dynamic Sorting

Data can be sorted in HttpWatch by clicking on a column heading. The sort order is applied to existing items and used to order new items as they appear.

Extended Cookie Information

Whenever a cookie is sent to a web server only its name and value appear in the HTTP request message. HttpWatch also displays the associated domain, path and expiration data making it easier to determine why a particular cookie value is being used. In version 7 it also shows the HttpOnly and Secure flags along with the source of the cookie value.

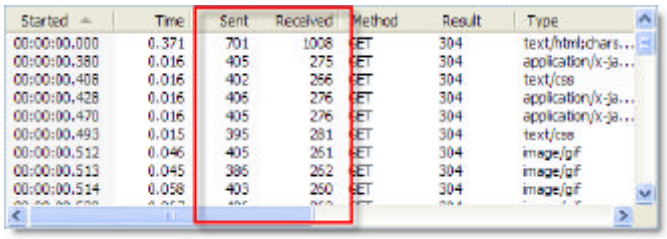
Other HTTP monitoring tools only display this information for cookies in the HTTP response message.



Network Level Performance Data

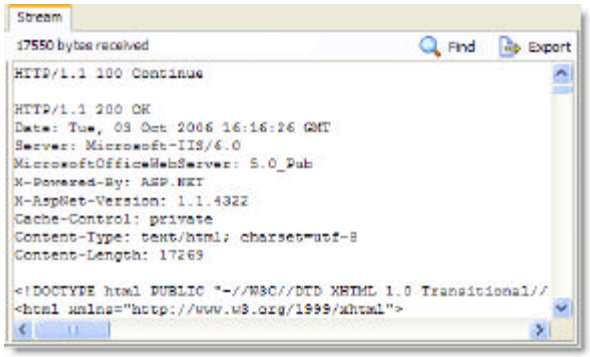
The Overview and Stream tabs show DNS lookups, TCP connects, IP addresses and ports used by an HTTP request. This can help locate network related problems and check that Keep-Alive connections are being used effectively.

The Send and Receive columns show the actual number of bytes that the browser had to send and receive when executing an HTTP request. Other tools just show you the content size, but it is the network level data sizes that really have an impact on performance:



Displays Raw HTTP streams

HttpWatch displays the raw HTTP streams sent to and received from a web site:



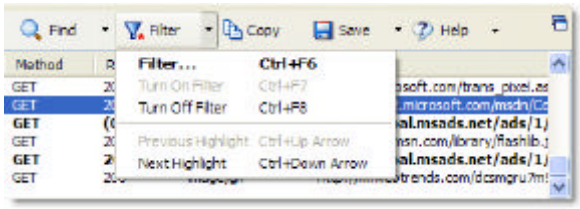
This low level view of the HTTP protocol helps to show the effect of using technologies such as chunked encoding or compression, and can be useful if you want to reproduce a request programmatically.

Shows Interaction with Browser Cache

HttpWatch shows the interaction between browser and its cache, not just network traffic between the browser and the web site. This is an important feature when a web site is being tuned for performance or to determine why pages are not updating correctly

Keyboard Accelerators

All commonly used actions in HttpWatch can be invoked with keyboard accelerators, even when the keyboard focus is in another part of the browser's user interface. For example, this menu shows the keyboard accelerators that can be used to control filtering:



Data can be exported to other formats

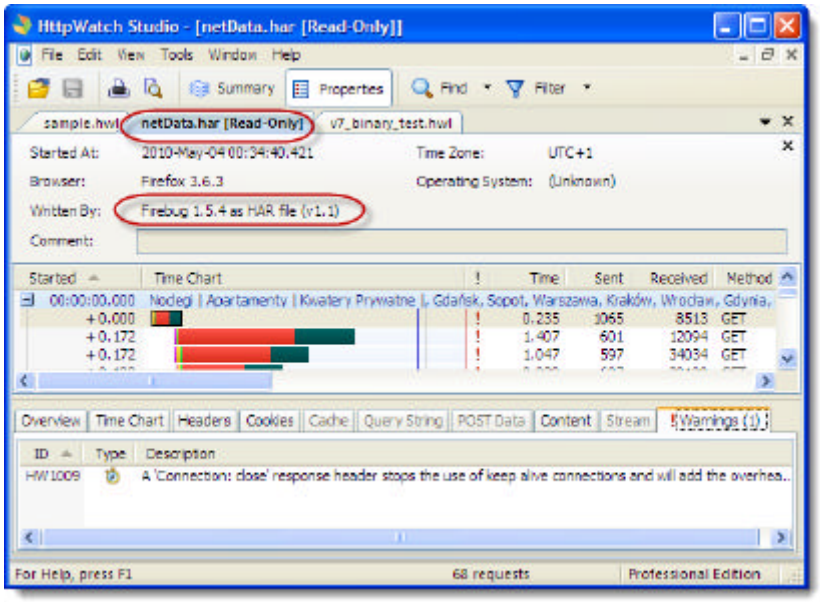
The data captured by HttpWatch can be exported in CSV (comma separated variable), HAR (HTTP Archive) or XML formats. Sample log files are available for download:

- Sample CSV (comma separated variable) log file
- Sample HAR (HTTP Archive) log file
- Sample XML format log file

Import HAR Files

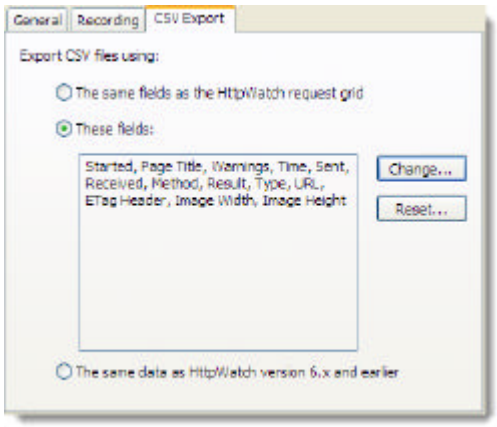
HTTP Archive (HAR) files can be opened in HttpWatch Studio and viewed in the same way as data recorded with

HttpWatch:



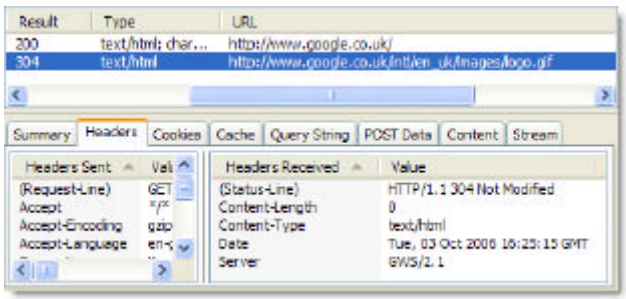
Customizable CSV Export

The CSV output now can now be customized to include the data fields that you need:



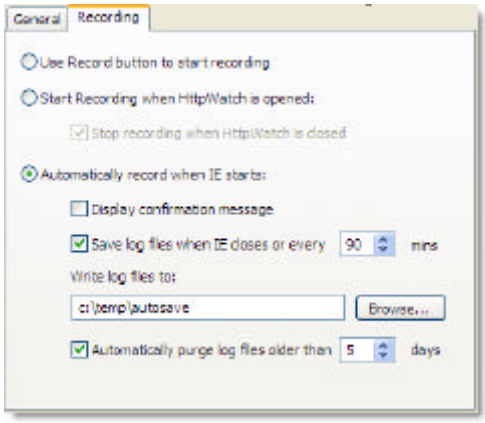
Accurately Records Requests and Responses

HttpWatch logs and displays all the intermediate responses caused by the use of techniques such as redirection, authentication and 1xx responses. Some tools just synthesize the response and may show invalid responses (e.g. non zero Content-Length headers on 304 responses). With HttpWatch you see the actual data returned by the web server, even over HTTPS connections.



Automatic Recording And Saving

HttpWatch can be configured to automatically record and save log files with no manual intervention or programming. Log files are written out to a specified directory at regular intervals or when the browser closes:

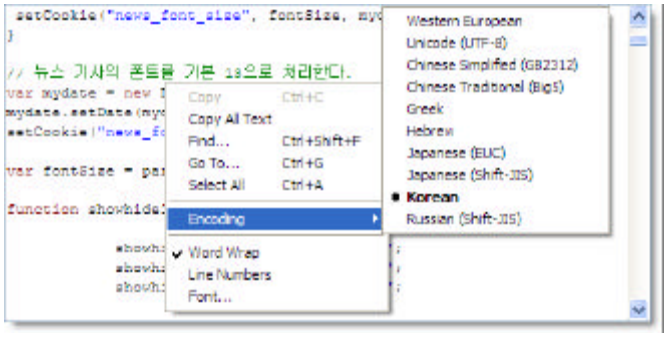


Access to Cached and Downloaded Content

Text, image and flash based content can be viewed within HttpWatch, exported to another application or saved to a file. The content window uses syntax highlighting on common web formats, such as XML, HTML and CSS:

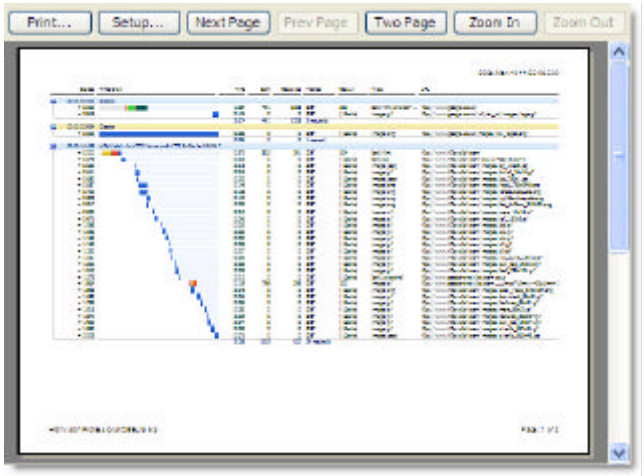
Support for International Characters

HttpWatch supports many commonly used character encoding schemes so that it can display the correct international characters:



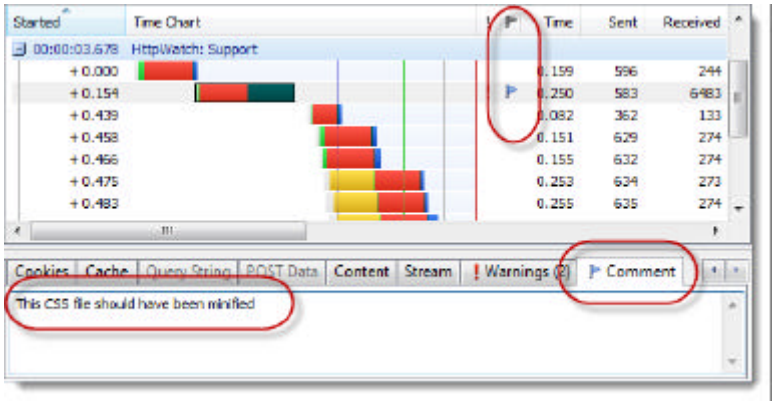
Printing Support

HttpWatch log files can be printed in HttpWatch Studio or from the browser plug-in:



Add Comments to Requests and Pages

Comments can be added to requests and pages to highlight areas of a web site's configuration that requires:



Compact & Comprehensive Log Files

The log file format used by HttpWatch results in much smaller files that other formats such as XML and contains ***everything*** that is displayed in HttpWatch. This includes binary format files and streams, compressed content and network information. If your customers send you HttpWatch log files you will get a full and accurate representation of the HTTP activity in their browser.

Easy, Simple And Powerful

HttpWatch works with Internet Explorer 6 - 9 and Mozilla Firefox 2.0 - 10.0 on Windows XP, Vista and Windows 7 (including IE protected mode) . It can be easily installed in a few minutes - No device drivers or proxies have to be configured.

The setup program is simple to run manually, and supports automated deployment by scripts or tools such as SMS.

HttpWatch was the first integrated HTTP sniffer for Internet Explorer and leads the way with a simple, yet powerful, user interface.

Change the way that you develop, debug and tune websites today!

HttpWatch

- Overview
- Features
- Compare Editions
- New in Version 9.x
- Awards & Reviews
- Download
- Pricing

Our Customers

- Who is using it?
- What are they saying?

Articles & Samples

- HttpWatch Blog
- HTTP Gallery
- Ruby, Watir & HttpWatch

Documentation

- HttpWatch Help
- HttpWatch Automation Reference

Company

- About Us
- Contact Us
- Support
- Blog
- Twitter

Search Sitemap Terms & Conditions Privacy Policy Copyright