The Latest Web Developments

Brian Kelly, UKOLN, University of Bath, Bath, UK
b.kelly@ukoln.ac.uk

ABSTRACT

This paper outlines some of the latest World Wide Web developments, in particular standards which are emerging from W3C, the World Wide Web Consortium. The paper gives an overview of the architectural components of the Web, summarise their deficiencies and describe how these deficiencies are being addressed.

The paper should be of interest to people involved in developing applications and services on the Web and anyone who has a general interest in future developments of the Web.

BACKGROUND

The World Wide Web (often referred to as the web) is a distributed hypermedia system which is based on three key architectural components:

1. Data format
2. Addressing
3. Transport

The native file format for resources on the web is the *HyperText Markup Language* (HTML).

The address for resources on the web is given by a *Uniform Resource Locator* (URL).

Resources on the web are transported from a server to the user's client system using the *Hypertext Transport Protocol* (HTTP).

We will look at these three architectural components in more detail.

DATA FORMAT

HTML (Hypertext Markup Language) is an application of SGML (Standard Generalised Markup Language). The first release, HTML 1.0, provided the hypertext linking which Web users today will be familiar with. HTML 1.0, in keeping with the spirit of SGML of defining the structural elements in documents, included the basic structural elements still in use today, such as paragraphs (the `<P>` element) and headings (`<H1>` to `<H6>`) as well as a small number of formatting elements, such as italic `<I>` and bold `<B>`.

HTML 2.0 introduced a number of innovations which were incorporated in NCSA's Mosaic web browser, including inline images and forms. Yes the initial implementation of the web did not include inline images!

At the first international WWW conference held in CERN, Switzerland in May 1994 David Raggett outlined a roadmap for future developments of HTML. HTML 3.0 (which was initially known as HTML+) would include a range of new features such as tables, richer forms and support for mathematical equations.

HTML 3.0 was submitted to the Internet Engineering Task Force (IETF). Unfortunately it failed to be standardised, due to a failure to reach consensus. This failure was due partly to the size and complexity of the proposal and also due to the lack of interest from the commercial web browser vendors.

In October 1994, the first version of the Netscape browser was released. Although Netscape proved tremendously popular, it also, controversially, announced support for a number of HTML elements which have not featured in discussions of developments to HTML such as the infamous `<BLINK>` element.

By 1995 Microsoft had become aware of the importance of the web. Initially their browser, Internet Explorer, was based on a licensed version of the original Mosaic browser. By the time Internet Explorer 3.0 was released (which was developed in-house), Microsoft were beginning to compete with Netscape for browser market share. This competition resulted in both companies announcing a variety of new HTML elements, with, for example, Microsoft responding to `<BLINK>` with their `<MARQUEE>` element for displaying scrolling text.

The browser wars resulted in confusion within the marketplace. Large companies, who were beginning to invest large sums of money in corporate Intranets, found the lack of interworking across browser and platforms placed a barrier on further growth.

At the same time as large corporations began to express their concerns over the browser wars, developers of web standards began to raise doubts as to the long term effectiveness of what became known as the HTML "tag soup". Pressures from large corporate users on one side and the web standards community on the other helped to force Microsoft and Netscape to work together, within W3C working groups responsible for coordinating the development of new HTML proposals. By January 1997 the HTML 3.2 proposal was accepted as a W3C recommendation [1]. HTML 3.2 was based on current established working practices. During 1997 work began on a new version of HTML, which had the codename *Cougar*. In December 1997 W3C announced [2] that HTML 4.0 (as Cougar became known as) had been accepted as a W3C recommendation.

HTML 4.0 included enhancements in a number of areas, such as more sophisticated forms and tables. HTML 4.0 added features to make web resources more accessible by providing support for people with disabilities and for non-English speaking users. Although HTML 4.0 gave recognition to the widespread deployment of frames, it did not introduce a wide range of new features. It primarily provided hooks for embedding other resources within HTML documents, such as multimedia objects and scripting languages. In addition HTML 4.0 provided support for style sheets.

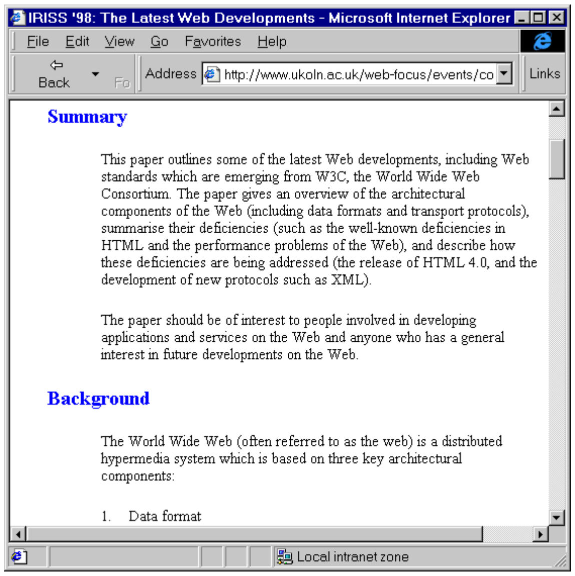
STYLE SHEETS

As mentioned earlier, HTML was originally intended to define the structure of a document. It has always been recognised that the appearance of a document was important. However it was felt that the appearance should be held separately from the content of a document.

The initial recommendation for style sheets, Cascading Style Sheets level 1 (CSS1), was announced in December 1996 [3]. However CSS1 was only partly supported in Microsoft's Internet Explorer 3.0 (which was available at the time) and was not supported in Netscape Navigator 3.0. Exerienced gained in the way in which CSS1 was used highlighted a number of backwards-compatibility issues.

In November 1997 a draft release of CSS level 2 was announced [4]. CSS2 provides a great deal of control over the appearance of a document. CSS can be included inline within an HTML element or within the `HEAD` of a document. However for maintenance purposes, it is better if the CSS is included as an external linked file. For example, all of the conference papers published in the conference proceedings could point to a single style sheet file. Changing the house style for the papers will simply require changing a single file.

An example of use of a simple style sheet is shown in Figure 1.


IRISS '98: The Latest Web Developments - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back
Address
http://www.ukoln.ac.uk/web-focus/events/co
Links
Local intranet zone

Summary

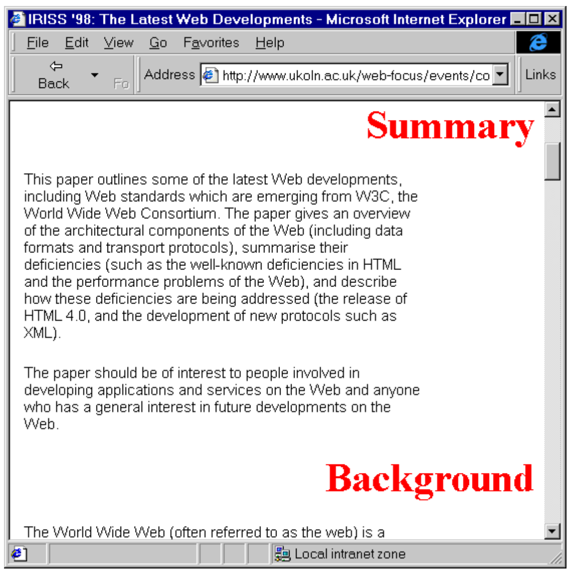
This paper outlines some of the latest Web developments, including Web standards which are emerging from W3C, the World Wide Web Consortium. The paper gives an overview of the architectural components of the Web (including data formats and transport protocols), summarise their deficiencies (such as the well-known deficiencies in HTML and the performance problems of the Web), and describe how these deficiencies are being addressed (the release of HTML 4.0, and the development of new protocols such as XML).

The paper should be of interest to people involved in developing applications and services on the Web and anyone who has a general interest in future developments on the Web.

Background

The World Wide Web (often referred to as the web) is a distributed hypermedia system which is based on three key architectural components:

1. Data format


IRISS '98: The Latest Web Developments - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back
Address
http://www.ukoln.ac.uk/web-focus/events/co
Links
Local intranet zone

Summary

This paper outlines some of the latest Web developments, including Web standards which are emerging from W3C, the World Wide Web Consortium. The paper gives an overview of the architectural components of the Web (including data formats and transport protocols), summarise their deficiencies (such as the well-known deficiencies in HTML and the performance problems of the Web), and describe how these deficiencies are being addressed (the release of HTML 4.0, and the development of new protocols such as XML).

The paper should be of interest to people involved in developing applications and services on the Web and anyone who has a general interest in future developments on the Web.

Background

The World Wide Web (often referred to as the web) is a

Figure 1a - Simple Style Sheet Example **Figure 1b - Simple Style Sheet Example**

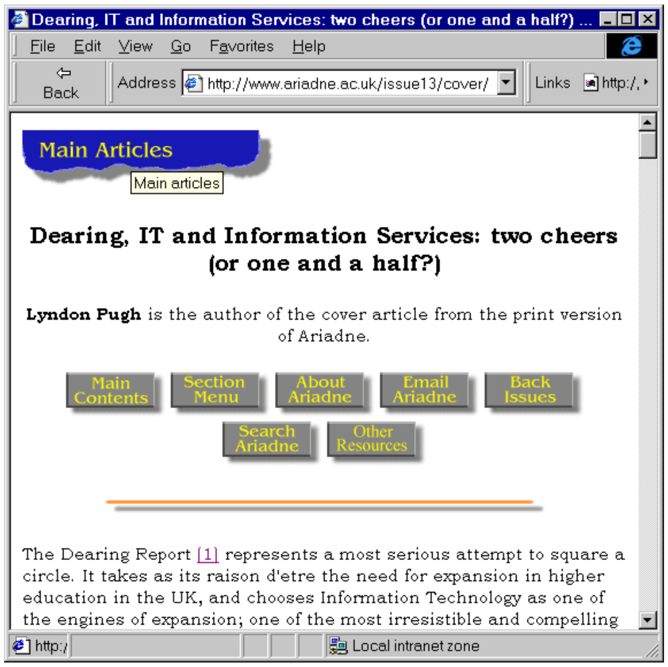
Figures 1a and 1b show the same document content, with slightly different style sheets. The corresponding style sheets are given below.

```
<STYLE>
<!--
H1, H2 {color: blue; margin-left: 5%}

H3 {margin-left: 10%}
P {margin-left: 15%}

OL {margin-left: 20%}
-->
</STYLE>
```

```
<STYLE>
<!--
H1, H2 {color: red; text-align: right;
        font-size: 24pt}
H3 {margin-left: 5%}
P {margin-left: 5%; margin-right: 20%;
   font-family: arial}
OL {margin-left: 20%}
-->
</STYLE>
```

Note that style sheets do not have to be supplied by an author. It is possible for an end user to define a style sheet to be used when accessing pages.

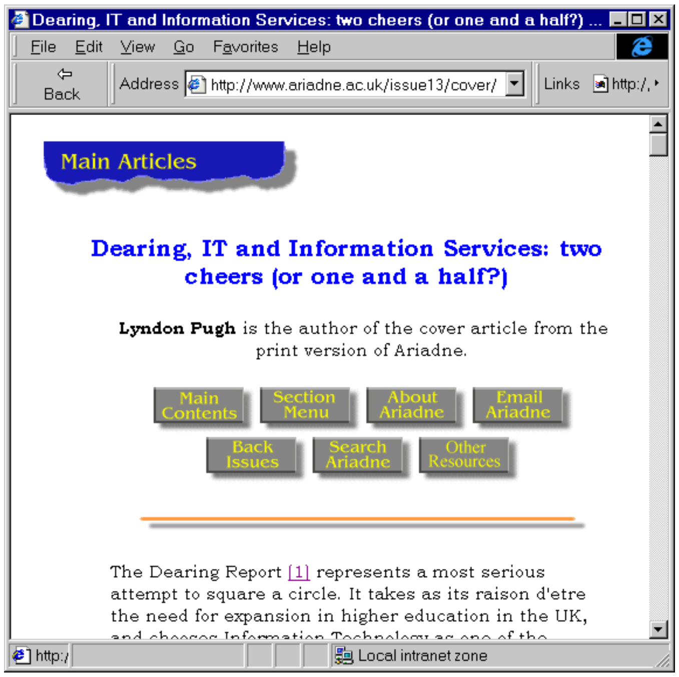
An example of user-supplied style sheet is shown in Figure 2.


Dearing, IT and Information Services: two cheers (or one and a half?) ...
File Edit View Go Favorites Help
Back
Address http://www.ariadne.ac.uk/issue13/cover/
Links
Main Articles
Main articles
Main Contents
Section Menu
About Ariadne
Email Ariadne
Back Issues
Search Ariadne
Other Resources
Local intranet zone

Dearing, IT and Information Services: two cheers (or one and a half?)

Lyndon Pugh is the author of the cover article from the print version of Ariadne.

The Dearing Report [1] represents a most serious attempt to square a circle. It takes as its raison d'etre the need for expansion in higher education in the UK, and chooses Information Technology as one of the engines of expansion; one of the most irresistible and compelling


Dearing, IT and Information Services: two cheers (or one and a half?) ...
File Edit View Go Favorites Help
Back
Address http://www.ariadne.ac.uk/issue13/cover/
Links
Main Articles
Main Contents
Section Menu
About Ariadne
Email Ariadne
Back Issues
Search Ariadne
Other Resources
Local intranet zone

Dearing, IT and Information Services: two cheers (or one and a half?)

Lyndon Pugh is the author of the cover article from the print version of Ariadne.

The Dearing Report [1] represents a most serious attempt to square a circle. It takes as its raison d'etre the need for expansion in higher education in the UK,

Figure 2a - Original Display

Figure 2b - Using A User-supplied Style Sheet

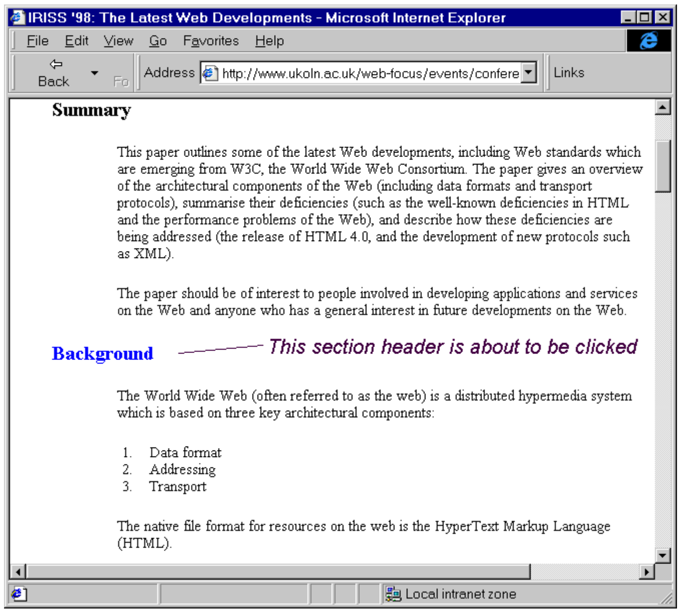
Figure 2a shows the original document, with the formatting defined by the author. Figure 2b shows the document when viewed using a user-supplied style sheet. In this example the style sheet indents the left-hand margin and specifies a colour for the headings.

DYNAMIC HTML

HTML 4.0 provides a means of defining the document structure, allowing CSS2 to define how the document appears. *Dynamic HTML* provides a way of enabling the content of HTML and CSS elements to be changed.

Dynamic HTML is based on a *Document Object Model* (DOM) [5] for HTML and CSS elements. The values of the HTML and CSS elements can be changed in response to a user action, such as clicking the mouse or moving the mouse over an object. The changes are initiated using a client-side scripting language, such as JavaScript.

Figure 3 gives an illustration of simple use of Dynamic HTML.


IRISS '98: The Latest Web Developments - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back
Address http://www.ukoln.ac.uk/web-focus/events/confere
Links
Local intranet zone

Summary

This paper outlines some of the latest Web developments, including Web standards which are emerging from W3C, the World Wide Web Consortium. The paper gives an overview of the architectural components of the Web (including data formats and transport protocols), summarise their deficiencies (such as the well-known deficiencies in HTML and the performance problems of the Web), and describe how these deficiencies are being addressed (the release of HTML 4.0, and the development of new protocols such as XML).

The paper should be of interest to people involved in developing applications and services on the Web and anyone who has a general interest in future developments on the Web.

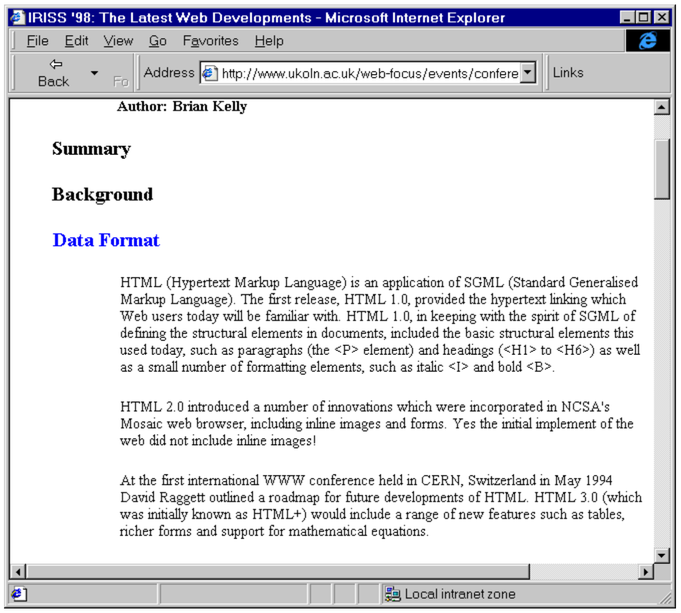
Background *This section header is about to be clicked*

The World Wide Web (often referred to as the web) is a distributed hypermedia system which is based on three key architectural components:

1. Data format
2. Addressing
3. Transport

The native file format for resources on the web is the HyperText Markup Language (HTML).

Figure 3a - Original Display


IRISS '98: The Latest Web Developments - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back
Address http://www.ukoln.ac.uk/web-focus/events/confere
Links
Local intranet zone

Author: Brian Kelly

Summary

Background

Data Format

HTML (Hypertext Markup Language) is an application of SGML (Standard Generalised Markup Language). The first release, HTML 1.0, provided the hypertext linking which Web users today will be familiar with. HTML 1.0, in keeping with the spirit of SGML of defining the structural elements in documents, included the basic structural elements this used today, such as paragraphs (the <P> element) and headings (<H1> to <H6>) as well as a small number of formatting elements, such as italic <I> and bold <B>.

HTML 2.0 introduced a number of innovations which were incorporated in NCSA's Mosaic web browser, including inline images and forms. Yes the initial implement of the web did not include inline images!

At the first international WWW conference held in CERN, Switzerland in May 1994 David Raggett outlined a roadmap for future developments of HTML. HTML 3.0 (which was initially known as HTML+) would include a range of new features such as tables, richer forms and support for mathematical equations.

Figure 3b - Display after Headings "Collapsed"

Figure 3a gives the original display. After clicking on the first two headings, the text underneath the headings is collapsed. This is achieved by setting the visibility of the section following the heading to "none" when the heading is clicked.

The code to achieve this effect is simply:

```
<H2 STYLE="cursor:hand" onclick="toggleDisplay(Background);"
onmouseover="this.style.color = 'blue'"
onmouseout="this.style.color = 'black'">Background</H2>
<DIV ID="Background" STYLE="display: visible">
...
</DIV>
```

Two Javascript routines, each of about 10 lines is executed when the heading is clicked which sets the display of the document identified by the name `<DIV ID="Background">` on or off. These Javascript routines are not included in this paper. However very simple Dynamic HTML can be seen in the HTML fragment above. The `onmouseover="this.style.color = 'blue'"` changes the colour of the heading to blue when the mouse is positioned over the heading.

XML

If HTML 4.0 can defines the document structure and provide the hooks for including multimedia objects, scripting languages and links to style sheets, CSS2 the appearance of the document and Dynamic HTML can provide a mechanism for dynamically altering HTML and CSS elements, does this mean that work on data formats is complete?

The answer to this is, perhaps not surprisingly, no.

The HTML standardisation process is too slow and time-consuming for new elements to be introduced. For example, if we wanted to introduce a new element called `<ABBREVIATION>`, even if we could achieve consensus within the HTML developers community, it would still take a long time for the recommendation to be agreed. Even then there would be no certainty that the browser vendors would provide support for the new element.

In addition, although HTML, in conjunction with CSS2, can be used as an output format it is not sufficiently rich to be used as a data storage format. For example we cannot develop a web application within our institution for storing records such as `<STUDENT-NUMBER>` or `<PART-NUMBER>`.

Finally even if communities such as mathematicians or chemists could agree on a set of elements for use within their community, adding them to a new version of HTML would result in a very large, complex language.

The *Extensible Markup Language* (XML) has been developed to address these issues. XML has been designed to be extensible, so that agreement on a set of standard elements does not necessarily have to be achieved. XML can be regarded as a light-weight version of SGML, designed for network use.

Although XML was only announced as a W3C recommendation in February 1998 [6], it is already becoming widely adopted in a number of areas. The Mathematical Markup Language (MathML), which is due to be submitted as a W3C Proposed Recommendation in February 1998, is an XML application, as is the Chemical Markup Language (CML).

In addition to use within the scientific communities, XML is also being used within the web community to develop new architectural components to the web, especially in the area of metadata, as discussed later.

ADDRESSING

The location of a resource on the web is given by its URL. For example the URL of W3C's HTML 4.0 recommendation is `http://www.w3.org/TR/REC-html40`

As can be seen in this example, URLs contain the domain name of the machine together with (in many cases) the location in the underlying directory structure. This can be regarded as equivalent to stating that the hard copy of the specification is located in the MIT library, and it's the sixth book along on the third shelf on the fourth floor.

URLs suffer from their dependency on the location. If an organisation reorganises its website, links to resources are likely to be broken. Similarly if an organisation changes its name, is taken over or sells part of the organisation, a reorganisation of its website to reflect the changes will also result in broken links.

There have been a number of proposals which attempt to provide a location-independent address for a resource including Uniform Resource Names (URNs) [7] and Persistent Uniform Resources (PURLs) [8].

A PURL acts as a URL which points to a resolution services rather than the resource itself. The PURL resolution service associates the PURL with the URL of the resource and uses a HTTP redirect to access the resource.

More recently the Digital Object Identifier (DOI) system has been developed [9]. The DOI system has three components: the identifier, the directory and the database. The system allows identifiers to be assigned at various levels. The directory is a distributed system based on CNRI's Handle system which provides a mapping from DOIs to URLs. DOIs have initially been aimed at the 'traditional' publishing industry, and there are plans to use the DOI as the basis of copyright management systems.

However none of the proposals for replacing URLs have been widely deployed. This is, in part, due to the need for an organisational infrastructure for registering location-independent resources.

TRANSPORT

HTTP, the *HyperText Transfer Protocol*, governs the transfer of resources between a web server and client. Typically clicking on a hypertext link in a web browser will send a HTTP `GET` request to the server. The web server will then send back a series of headers, together with the resource, if it exists.

It is possible to emulate a web client using `telnet`, as illustrated below:

```
% telnet www.w3.org 80             Telnet to port 80
GET / HTTP/1.0                     Request the default file
                                   Enter blank line
HTTP/1.1 200 OK                    Confirmation received
Date: Tue, 24 Feb 1998 09:14:31 GMT Misc headers displayed
Server: Apache/1.2.5
Last-Modified: Fri, 20 Feb 1998 17:55:12 GMT
ETag: "2c3136-23c1-34edc380"
Content-Length: 9153
Accept-Ranges: bytes
Connection: close
Content-Type: text/html; charset=ISO-8859-1
                                   HTML document displayed
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
..
```

In the initial implementation of HTTP, HTTP/0.9, the web browser could process the files based on the file suffix. So, for example, a PostScript file with a `.ps` could be passed to a PostScript viewer for displaying. This, however, was not a scaleable solution. In HTTP/1.0 [10] files are sent as MIME attachments, such as `text/html`, `image/gif`, `text/postscript`, etc.

Although HTTP/1.0 is now being widely used, there are a number of problems with it:

- HTTP/1.0 uses TCP inefficiently. Since most resources are small, and HTTP/1.0 opens and closes a new TCP connection for each operation, there is a large overhead.
- HTTP/1.0 does not have sufficient facilities for compression.
- HTTP/1.0's caching is very primitive.

HTTP/1.1 [11] was developed to address these deficiencies and to fix a number of bugs in HTTP/1.0. The HTTP/1.1 specification provides support for multiple TCP connections and more efficient support for caching.

A W3C Note on "*Network Performance Effects of HTTP/1.1, CSS1, and PNG*" [12] confirms the performance benefits of HTTP/1.1.

EXTENDING HTTP

Although HTTP/1.1 will provide performance benefits, the introduction of new facilities is still hindered by the standardisation process and the dangers of making HTTP more complex by the introduction of facilities which will be used by only small communities. HTTP faces similar development problems as does HTML.

Just as XML provides a extension mechanism for data formats, the Protocol Extension Protocol (PEP) [13] is designed to provide an extension mechanism for HTTP.

PEP examples which are given in the PEP draft specification include determining whether a server understands and supports the Distributed Authoring and Versioning (DAV) protocol extension and use of a micropayments scheme.

An example of the potential use of PEP is a micropayments system for accessing resources. The dialogue is illustrated below.

```
GET /Index HTTP/1.1
Host: some.host

420 Policy Not Fulfilled
PEP-Info: {{id "http://www.w3.org/PEP/MiniPayment"}
                {params {Price 0.02USD}} {strength must}}

PEP-GET /Index HTTP/1.1
Host: some.host
PEP: {{map "http://www.w3.org/PEP/MiniPayment" 12-}
         {strength must}} 12-Price: 0.02USD

HTTP/1.1 200 OK
```

In the example given above the client requests a resource. The server responds with an HTTP response code stating that the policy has not been fulfilled, and then uses the PEP extension mechanism to state a price which must be paid in order to access the resource, together with the address describing the minipayment protocol. A web client which does not understand the PEP request will treat the response as a file not found and display an appropriate error message. Otherwise the client can communicate with the server using the extension policy.

CONTENT NEGOTIATION

We have seen how PEP can be used to provide an extension mechanism for HTTP. Transparent Content Negotiation (TCN) [14] provides an extensible negotiation mechanism, layered on top of HTTP, for automatically selecting the "best" version when the resource is accessed. TCN enables new data formats and HTML elements to be smoothly deployed.

HTTP/NG

Although HTTP/1.1, together with PEP and TCN, are addressing a number of the deficiencies in the underlying transport protocol, we are still faced with a number of problem areas, including the complexity of HTTP, the poor scalability of HTTP when faced with today's network traffic load and the difficulty of introducing applications on the web, other than simple document retrieval applications.

HTTP/NG [15] will be a new architecture for the HTTP protocol based on a simple, extensible distributed object-oriented model. Work on HTTP/NG started recently. As yet there is little information publicly available.

OTHER AREAS

Metadata can be regarded as the missing architectural component of the web. Although HTML has allowed the basic elements of a document structure to be defined, it has, in general, not allowed information about the document to be defined in a structured, machine-parsable way.

The `<META>` HTML element was an initial attempt to provide a mechanism for storing document *metadata* in a standard way. The `<META>` element became popular for storing keywords to assist search engines, such as Alta Vista, in finding resources. Search engines would give a high priority to resources containing metadata as shown below:

```
<META NAME="description" VALUE="This is the HTTP specification">
<META NAME="keywords" VALUE="HTTP, web, transport protocol">
```

Dublin Core [16] is the name given to an initiative to agree a common, core set of metadata attributes to help with resource discovery. The Dublin Core now consists of 15 elements, such as Title, Creator, Date, etc. Initially attempts were made to embed Dublin Core metadata using the `<META>` element. However this was not sufficiently flexible to cater for more complex use of Dublin Core metadata, such as hierarchical structures, such as the creators name, postal address, email address, etc.

The development of a more general solution to the provision of metadata is being coordinated by the W3C. The Resource Description Framework (RDF) [17] is designed to provide an infrastructure to support metadata in a range of areas including resource discovery, sitemaps, rating schemes, and collections of resources.

WHAT'S ALL THIS MEAN FOR ME?

This document has described a number of developments to web protocols. But what are the implications for web administrators, support staff, software developers, information providers and end users?

The strict HTML philosophy has been to encourage authors to define document structure. With the release of CSS2 and support for CSS2 by the current versions of both the popular browsers, it is now possible for authors to provide a pleasing design for their resources, using technologies which will minimise future maintenance.

Unfortunately Internet Explorer and Netscape have different implementation of style sheets, and so authors will have to make use of these new features with care. It is, possible, however, to layer new technologies, such as CSS and Dynamic HTML on to existing resources, provided the resources conform to standards.

Developers of computer aided learning software, who in the past have made use of proprietary, platform-specific authoring tools will appreciate the development of Dynamic HTML and the Document Object Model. This should enable rich interactive teaching systems to be developed based on open standards.

There are a number of implications for support staff responsible for providing and supporting web software, including web browsers and servers. For performance reasons servers should be upgraded to support HTTP/1.1. This should not prove too difficult as there are likely to be only a small numbers of web servers within an institution. Upgrading of web browsers to support new developments such as HTML 4.0 and CSS2, may be more difficult. However developments such as Transparent Content Negotiation may make it possible to deploy new features without disenfranchising large communities.

Developments to addressing have not progressed as rapidly as those to data formats. It appears unlikely that we will see in the near future the widespread use of location-independent identifiers. Authors will therefore have to continue to think long and hard about their directory naming conventions, to ensure that next year's reorganisation of a web site does not result in lots of broken links.

REFERENCES

1. *W3C Issues Recommendation for HTML 3.2*, W3C Press Release, <URL: http://www.w3.org/Press/HTML32-REC-PR.html>
2. *The World Wide Web Consortium Issues HTML 4.0 as a W3C Recommendation*, W3C Press Release, <URL: http://www.w3.org/Press/HTML4-REC>
3. *The World Wide Web Consortium Issues Recommendation for CSS1*, W3C Press Release, <URL: http://www.w3.org/Press/CSS1-REC-PR.html>
4. *The World Wide Web Consortium Publishes Public Draft for CSS2*, W3C Press Release, <URL: http://www.w3.org/Press/CSS2>
5. *Document Object Model*, W3C, <URL: http://www.w3.org/DOM/>
6. *The World Wide Web Consortium Issues XML 1.0 as a W3C Recommendation*, W3C Press Release, <URL: http://www.w3.org/Press/1998/XML10-REC>
7. *Naming and Addressing: URIs*, W3C, <URL: http://www.w3.org/Addressing/>
8. *PURLs*, OCLC, <URL: http://purl.oclc.org/>
9. *DOI System*, <URL: http://www.doi.org/>
10. *HTTP/1.0*, W3C, <URL: http://www.ics.uci.edu/pub/ietf/http/rfc1945>
11. *HTTP/1.1*, W3C, <URL: http://www.w3.org/Protocols/rfc2068/rfc2068>
12. *Network Performance Effects of HTTP/1.1, CSS1, and PNG*, W3C, <URL: http://www.w3.org/Protocols/HTTP/Performance/Pipeline.html>
13. *PEP*, W3C, <URL: http://www.w3.org/Protocols/PEP/>
14. *Transparent Content Negotiation in HTTP*, Koen Holtman, <URL: http://gewis.win.tue.nl/~koen/conneg/>
15. *HTTP/NG*, W3C, <URL: http://www.w3.org/Protocols/HTTP-NG/>
16. *Dublin Core*, OCLC, <URL: http://purl.org/metadata/dublin_core>
17. *RDF*, W3, <URL: http://www.w3.org/Metadata/RDF>

BIOGRAPHY

Brian Kelly is UK Web Focus - a national web coordination post for the UK Higher Education community. Brian is based at UKOLN (UK Office for Library and Information Networking), University of Bath. His responsibilities include monitoring web developments, keeping the UK HE community informed of web developments, coordinating various web activities within the community and representing JISC on W3C (the World Wide Web Consortium).

Brian has been involved with the Web since early 1993. He helped set up the Web service at Leeds University in January 1993 - the first institutional web service in the UK HE community. He was active in promoting the web throughout the community, giving numerous presentations. He attended the first WWW conference in Switzerland in May 1994, and gave a paper on *Providing Information On The World Wide Web* at the JENC 6 / INET 94 conference in Prague in June 1994.

From October 1995 to October 1996 Brian worked as the Senior Netskills Trainer for the Netskills project, based at Newcastle University. He moved to UKOLN in November 1996.