Title: CONTENT ACCESS PLATFORM AND METHODS AND APPARATUS PROVIDING ACCESS TO INTERNET CONTENT FOR HETEROGENEOUS DEVICES

Abstract: A content access platform and methods and apparatus are provided for improved access to internet content for users of heterogeneous devices, such as mobile telephones and PDAs. The platform provides a delivery mechanism for requested content that is adapted for the capabilities of the requestor device type, using device-type-specific mini pages that replace original Web pages and using templates implemented as XML configuration files for generating the mini pages. The platform also provides support for customized combinations of content to be provided in accordance with requirements specified in stored templates, such that Web site providers and others can provide a customized user experience for different users according to their preferences or device type. The mini pages are generated with reference to an analysis and categorization of the original Web page or site as well as the capabilities of requestor devices. Device-compatible directory pages are also provided, in response to directory requests including a device type identifier to ensure that an appropriate directory page is served.
CONTENT ACCESS PLATFORM AND METHODS AND APPARATUS PROVIDING ACCESS TO INTERNET CONTENT FOR HETEROGENEOUS DEVICES

FIELD OF INVENTION

The present invention relates to data access and communications, and in particular relates to improved access and customized access to Internet content from heterogeneous devices within a communications network, such as for improved access and customized access to Website content from mobile telephones and other wireless communication devices such as PDAs.

BACKGROUND

The Internet has made vast amounts of information available to computer users, as well as large numbers of applications such as games and other content. Although advances in telecommunications and data processing technologies have enabled Websites to be viewed on mobile devices, almost all of the available content is designed for the screen size and processing capability of desktop and laptop computers. Most Website providers only consider mobile Web content as an afterthought, rather than as a parallel development and distribution medium, and many consider it prohibitively expensive to customize all of their content for mobile device users.

It is very common for publicly accessible content such as Websites to include a range of different content types and embedded components or 'assets' including text, images, video and audio content, with feature-rich user interfaces including drop-down menus, scrolling news feeds, pop-up banner advertisements and so on. For some users and some user-devices, this rich content is desirable, but there are various reasons why it is often not practical to view such content on mobile devices:
The speed of viewing is affected by the method of access. This can be the result of any one factor or a combination of factors, such as: the resource constraints of the access device - such as limited working memory, limited processing power and/or limited communication bandwidth; the size of the data transfer; and the complexity of Websites. As a result, conventional Websites can take a long time to download and cannot be displayed correctly on many communication device display screens.

Many mobile devices possess display screens that are far too small for unstructured layouts, differ significantly in size from one screen to another, contain unconventional screen layouts, or do not support features that are designed for larger screens. It may therefore be impossible to display the original Web page in the way envisaged by the page creator, and known processes that scale down content for a small screen may render the content unreadable.

Some HTML mark-up is not appropriate with communication device user interfaces or supportable from the limited i/o interfaces of that device. Many communication devices still do not have touch-sensitive screens and most do not have a controller suitable for driving a PC Mouse-type pointer. Hence some Web page features such as drop-down menus, hover over pop-up windows, etc have no easy means of being controlled by the user of a communication device.

These issues remain relevant despite the increasing availability of mobile devices with larger display screens and increased processing power - since there remains a very large market for inexpensive mobile telephones. Despite the improvement of the mobile devices, their screen size and processing power are still not comparable to PCs, partly because the processing power of PCs is also improved over time. The complexity and resource requirements of new software and internet web pages tend to increase over time to take advantage of the improvement of the PC hardware.
Additionally, there is a high network communication overhead inherent in providing all of this content to large numbers of users, and the processing overhead in transcoding content for display on each of the different user device types is also very significant. These overheads are not justified for the large number of device users who either cannot display or do not wish to view much of this content.

Furthermore, device users attempting to access very popular Web sites at busy times often experience frustrating delays that result from communication network 'bottlenecks', further increasing latency to the extent that the user experience is not satisfactory.

There is a need to mitigate the various problems described above, both to improve the end user experience and to provide more efficient use of resources.

The inventors of the present invention have determined that it is possible to provide an enhanced user experience for mobile device users, while also making a number of improvements for Web site providers and other content providers, for telecommunication network providers, and for advanced users who wish to share their proposals for an optimised content selection and display.

A partial solution to some of the problems described above is to manually create alternative versions of Web pages in Wireless Markup Language (WML) and to provide access to these pages via the Wireless Application Protocol (WAP) and a WAP Browser running on the user device. However, a native WML page may still be presented differently on different end-user devices, due in part to variations in how WAP is implemented but primarily because of great differences between end-user devices. Developing a WML version of a Web site that enables complex site content to be displayed effectively on a wide range of wireless devices is a difficult and time consuming task. Many Website owners do not have the financial means or technical knowledge to be able to build a WAP site, and may be discouraged by the fact that building a WAP site does not guarantee the display of content on every mobile device.
Furthermore, WAP pages are designed to solve display problems on end-user devices and so their use does not automatically address the problem of network 'bottlenecks'.

Even when Website content is displayable to users, there is a high overhead inherent in modifying Web pages to create a mobile-friendly version for access by a large number of users of heterogeneous mobile devices, and in maintaining that modified set of pages.

Another solution is for mobile-friendly WML pages to be generated dynamically when requested by a mobile device user. Such run-time transcoding of site content can avoid the Website provider having to pre-generate and maintain separate WAP pages for mobile devices, but the run-time transcoding introduces other problems. The additional processing may contribute to unsatisfactory latency, regardless of whether this is implemented on the user's device or elsewhere in the network. Furthermore, not all Web page content can be converted to a format that can be displayed via a WAP browser, and not all devices are WAP-enabled.

A positive user experience is critical to the adoption of new services and providing truly open access to media content, and long upload and download times are a major source of frustration for typical mobile device users. Mobile access to Web content and services is discouraged by poor response times, and transcoded content currently fails to provide a really positive user experience. Thus, current solutions involving dynamic content transformations - whether from HTML to WML, or adapting HTML pages for smaller displays - are also not ideal solutions.

WO 2008/073207 discloses Web content adaptation for mobile devices. A mobile device transmits request data, that identifies a requested Web page and identifies the requestor device type, and awaits a response. The identification information in the request is compared with adaptation parameters to determine how response data obtained from a Web server should be adapted before being returned to the requestor mobile device. A main adapted sub-page and a set of subsequent adapted sub-pages are returned to the mobile device, each sub-page including adapted content that is in an order suited for viewing on a mobile device. The sub-pages may be cached for future reference.
WO 02/087135 discloses adapting information content for mobile electronic devices that have limited hardware or network communications capabilities. Data content is converted into a document object tree and this is adapted using content folders. The content folders of interest are then used to provide adapted content for the software, hardware and network characteristics of the device.

Although solutions are known which provide an automated transcoding mechanism, to help make the content of an original Website accessible from a mobile device, none of the known solutions can be relied upon to provide an optimal user experience. In particular, no known solution provides a complete solution to some of the user-experience problems when accessing Web content via a mobile device and its communications network.

SUMMARY

The inventors of the present invention have determined that an improved user experience is achievable for users of communication devices, by providing selected content in an optimized manner according to different device types, different content provider preferences, different user group preferences and/or for different categories of Web site and Web content. In one aspect, the invention provides a solution for customizing the displayed content and format and for creating new versions of Web content (including 'mini page' versions of Web pages) for a communications device that combines selected content from a number of different sources such as from multiple original Web pages.

Another aspect of the invention provides a solution for automated analysis of a Web page to identify content components, component layout patterns and component combinations. These can be recognized and matched with stored definitions or templates. Presentation rules or templates are stored that are specific to a recognizable pattern or combination of content components, and specific to a device type or user preferences. The Web page analyzer
according to an embodiment of this invention cooperates with a mini page generator to enable a reformatting of the content onto a mini page that is suitable for the requesting device type or preferences of the requesting user.

In another aspect of the invention, device-type-specific directory pages and content pages are hosted in a content access platform that greatly improves the end user experience when accessing web content from a resource-constrained device, while also reducing the burden on Website content providers and communication network providers. A customised directory definition and a customized set of mini pages can be generated and saved for the preferences of an individual user or user group. The required directory page for a given device type can be determined from received requests - either from device information in request header attributes (if available) or by use of predefined subdomain names that include a device type identifier. Similarly, a suitable version of a Web page or other Web contents can be identified based on the requestor device type.

A number of other advantages are also achievable for users, device providers, content providers and communication network providers, as described below.

The present invention provides a solution for use in a communications network, in which multimedia content such as Website content needs to be accessed from many heterogeneous devices such as mobile telephones and PDAs, as well as desktop and laptop computers. A content access platform and methods and apparatus according to various embodiments of the invention are provided to mitigate the problems identified above.

Embodiments of the invention enable suitable versions of content such as Web page content to be provided to each end-user device that makes a request for the content, the provided versions being suitable for the capabilities of the requestor device and a desired user experience. For example, Web pages that are suitable for display on a particular mobile telephone's screen area via the telephone's Web browser can be generated and stored so as to be more easily available for retrieval. The invention can take account of the requirements and capabilities of users'
devices such as particular screen sizes, memory capacity, browser and operating system type, and other device-type characteristics. The invention can provide open access to Internet content from a very wide and extendable range of heterogeneous access devices in a seamless manner, potentially supporting the total device-type and browser base.

Additionally, embodiments of the invention enable elements of Web content to be combined in accordance with user-specific requirements and content-provider specifications, to provide a customized user experience.

The generation of suitable versions of Web pages ('mini pages') can be performed preemptively in advance of a request being received, or in response to a request being received. When implemented in advance of receipt of a request, suitable versions of Web pages ('mini pages') and/or suitable templates for generating 'mini pages' are generated and stored so as to be available for use when required. Any number of pre-formatted 'mini page' versions may be generated and stored, according to the number of different device types that may request access to this content.

The invention is also useful in providing improved access to suitable versions of downloadable application programs such as games, to ensure that a request for an application from a particular end user device is responded to by providing a version of the application that is suitable for the device type.

One aspect of the invention provides a content access platform comprising at least one content server for storing one or more versions of items of content (such as 'mini page' versions of Web pages), and a router for redirecting content requests (e.g. HTTP Web page requests) to a selected content server or repository in response to identifying the content request as having come from a particular requestor device type. The content server stores one or more versions of the content that are suitable for display on one or more specific types of requestor devices. The router identifies attributes of the request that are representative of the requestor device type, and redirects the request to an appropriate content server at which an appropriate version of the
content is stored, or redirects requests to a plurality of content servers to retrieve each of a plurality of elements of a desired mini page. The appropriate version of the content is retrieved (or is generated by combining a number of elements of content) and is sent to the requestor device, and the end user can then view the appropriate version of the media content.

The content access platform preferably includes a Website analyzer for scanning and analyzing the content of Web sites to identify high-level visual components of each Web page. The analysis can involve parsing the HTML within a Web page source file to identify static content component types, and generating HTML DOM structures corresponding to the pattern or layout of the Web page. The analysis can involve using a script engine or Browser engine to create the DOM tree of a page at runtime, to identify dynamic contents that are created by client-side script (such as Javascript or AJAX).

The analyzer according to one embodiment of the invention performs an analysis of the text on a page, such as a statistical analysis of high frequency words on the page or Website, to identify representative key words or to determine whether the site is a sports site, a retail site, a news site, and so on. Category information and/or identified keywords may then be stored in association with the Web page or Website. The analyzer also identifies the separate building blocks of the site and their layout (e.g. parsing the source HTML to determine the hierarchical HTML document structure of a page and the page hierarchy within a Website, and identifying major 'assets' that users of the site will require access to). The analyzer may perform its operations pre-emptively, for example when prompted by an operator of the content access platform, to generate mini pages in readiness for later retrieval by device users when users request access to the respective Web content.

New versions of Web contents referred to as 'mini pages' in the context of this patent specification include both minor changes to an original Web page (such as a scaled down and re-ordered pages including substantially the same content as the original page) as well as simplified pages that omit a proportion of the original Web page content for simplified display
on a resource-constrained device. As well as reduced-content versions of Web pages, a 'mini page' may also include components that were not included in a single original page, resulting in a new combination of content. The generation of device-type-specific mini pages is especially advantageous for users of resource-constrained devices that have either small screen sizes or constraints on download speeds or memory, but no limitation is implied by the phrase 'mini page' which is intended to encompass any replacement Web pages that are created for display on a particular type of device or created for a particular set of users.

In one embodiment of the invention, device-type-specific templates can be generated and stored for reconfiguring Web page content for display on devices of that type. The use of device-type-specific templates that encapsulate device characteristics and/or user requirements for generation of the mini-pages enables Web content to be presented in a manner that is optimised for the particular user, especially when combined with the intelligent modelling of a Website to categorize and model the site and to use that site model to guide the selection or generation of a suitable template.

In preferred embodiments of the invention, end users are unaware of redirection of requests by the router, and yet the user benefits from faster and more efficient downloads, and benefits from a device-type-suited version or a customized version of the requested content being displayed on their particular device. Thus, an improved user experience is achieved transparently. Users can continue making content requests in the conventional manner or via a new mechanism that is described below. The telecommunications network providers benefit from embodiments of the invention that enable requestor devices to be sent reduced-data-size versions of media content, as the invention can be used to provide an alternative access mechanism for very popular content that may otherwise suffer from communication bottlenecks and latency. By mitigating such problems, embodiments of the invention improve end user experiences and facilitate the delivery of additional services by the network providers.
In one preferred embodiment of the invention, a content server of the content access platform provides a repository into which many different Website providers and other content providers can save mobile-device-compatible 'mini page' versions of their Websites or selected content. Thus, Web content providers can benefit from the media access platform as a distribution channel for Web content, and content distribution may be a fee-based service to enable the media access platform provider to recover costs.

Firstly, the storage provided by the content access platform ensures that Website providers are not required to maintain additional storage for their different versions of Web content, as this and the labour-intensive processes of creating and maintaining a mobile-device-friendly version of a Website can be outsourced to the content access platform's template-based generation and content servers. This lowers the cost barrier for Web content providers who would like to provide a mobile-device-friendly version of their content. Secondly, the original Web servers can be shielded from the problems of potentially very high network traffic arising from mobile devices sending media content requests, as requests from such devices will be redirected to the respective content servers of the content access platform that hold suitable mini pages or hold content for building a mini page. In this way, responsibility for maintaining a satisfactory response to requests from mobile devices can be outsourced from the Website providers to the provider of the content access platform. This can be very beneficial to a Website provider, especially during periods of very high numbers of requests such as during popular sporting events. Many Web servers will be unable to manage the high workload that arises if requests from every different device type are competing for service from that server. Thirdly, the Website providers can rely on the router to achieve efficient routing of requests to the content server that holds the relevant mini pages or elements of mini pages, since the router has the requisite knowledge of where each version of the content is stored. This knowledge can help to avoid network bottlenecks.

A content access platform according to one embodiment of the invention provides a highly scalable mechanism for publication of customized Web pages and for publication of other
media content such as games and other application programs - enabling a bespoke user experience when this is desired but providing a solution that scales to millions of users of many different Websites. Website developers and other content providers that do not have sufficient resources to make their content widely available to mobile device users can take advantage of the content access platform to publish different versions of their content, each version being suitable for a different user device type and/or user group. Requests to retrieve the particular media content can be redirected by the router to retrieve the most suitable version, without the content provider having to participate in the identification of requestor device types or the routing of requests or the handling of potentially large numbers of requests from mobile device users. This aspect of the invention can be implemented as a fee-based service, providing commercial opportunities to the provider of the content access platform as well as the Web content providers.

Furthermore, particular desired combinations of media content may be specified within a new configuration file such that a customized combination of content is available for retrieval by some device users. The specification of a desired content combination may be by the original content provider or by another special interest group. End users may register their devices with the content access platform with an indication of which device-type and user-group they correspond to. This could involve users specifying language requirements, such that their content requests are routed to one or more content servers holding video and associated Mandarin text and audio, whereas another user may wish to access the same video with English text and audio.

In a preferred embodiment, the customized content combinations are achieved by the user specifying content requirements, which requirements data is then used (for example by a Website provider) to generate a template that captures the content requirements as an XML configuration file that includes one or more URLs for retrieval of desired content. Some users may desire to have a combination of applications such as a news feed component providing latest sports scores in combination with a small video screen area showing live action,
additional images and games. The content identifiers can be defined in a template or configuration file and can be used to combine the desired Web assets within a new mini page. The mini page can then be stored for access by the requesting user, and potentially by other users. Users will be able to design their own mini-pages combining content in a customised manner, and/or to design a customised mini-page directory, and then rely on the distribution mechanism of the content access platform to make such mini-pages and directories available to others. The media access platform thus provides a user-centric mechanism for experiencing content in a preferred and customized way, which differentiates from the mechanical content-transcoding-layer approach of known solutions.

If a particular user-specified selection of content becomes very popular, this could encourage wider access to some of the combined content than would otherwise be the case. This embodiment of the invention provides opportunities for placing advertisements with popular content for maximum visibility.

One embodiment of the invention includes a router that cooperates with a proxy service to enable more efficient route determination to be carried out for sending content requests across the network, using knowledge of the locations of device-type-specific versions of content to enable improved resolution of routing questions. In particular, the route determination will enable network bottlenecks to be bypassed in many cases. In a conventional system, it is common for bottlenecks to arise when requests from large numbers of mobile telecommunication devices are directed at the same Web server during a short period of time. By providing knowledge of content locations to a proxy service at an intermediate network location, an efficient route to the desired network location can be determined.

One aspect of the invention provides apparatus for use in a communications network, comprising:
a template repository storing a set of template files that each specify information for determining how Web content is to be displayed on a particular type of user device that is associated with the respective template file;

a request analyzer, arranged within the network for analyzing requests for access to Web content, wherein the content requests include information enabling identification of the device type of the respective requestor device, and wherein the request analyzer is adapted to analyze requests from requestor devices to identify the requestor device type;

a router that is responsive to identification of a requestor device type to retrieve a respective template file from the template repository and, in response to the template specifying a plurality of separate sources of Web content, to send a plurality of Web content requests to the separate sources, thereby to retrieve required Web content from the plurality of separate sources; and

a content generator for combining the retrieved Web content and providing the combined content to the requestor device.

A further aspect of the invention provides a method for accessing Web content via a communications network, the method comprising:

responsive to receipt at a router of a content request from a content requestor device, which content request includes information enabling identification of the requestor device type, identifying the requestor device type from the information within the content request;

routing a first content request from the router to a first content server that is known by the router to have access to a first version of the requested content, wherein the first version of the requested content is predefined as suitable for display on devices of the identified requestor device type;
routing a second content request from the router to a second content server that is known by the router to have access to additional content that is predefined as suitable for display on devices of the identified requestor device type;

retrieving the first version of the requested content from the first content server and retrieving the additional content from the second content server;

combining the first version of the requested content and the additional content; and delivering the combined content to the content requestor device.

A further aspect of the invention provides a router system, for use in a communications network, wherein the router system is arranged for receipt of content requests from content requestor devices, which content requests include information enabling identification of the requestor device type, the router system comprising:

a repository storing network location information for a plurality of sources of content, the repository also including an identification of respective device types with which each of the plurality of content sources is associated;

a request analyzer for analyzing received content requests from requestor devices to identify the respective requestor device type; and

a request redirection component that is responsive to identification of a requestor device type to redirect the request to a selected plurality of the content sources.

A further aspect of the invention provides a method for automated generation of a displayable version of Web content, the method comprising:
analyzing Web content to detect a set of identifiers of content types within the content;

comparing the set of identifiers of content types with known patterns of content components;

in response to identification of a known pattern of content components, generating at least one device-type-specific Web content presentation; and

saving the presentation for future access.

Thereafter, in response to a request for the Web content, which request specifies the requestor device type, the saved presentation can be identified and served to the requestor device.

A further aspect of the invention provides apparatus for use in a communications network, comprising:

a Web content analyzer for analyzing and categorizing Web content;

a first content server system comprising a repository for storing a first version of one or more items of Web content, the first version of the Web content being predefined as suitable for display of Web content of a first category on devices of a first device type, and wherein the first content server system is configured to provide access to the first version of the Web content in response to receipt of a request for the Web content; and

a request analyzer, arranged within the network for analyzing requests for access to Web content wherein the content requests include information enabling identification of the respective requestor device type, wherein the request analyzer is adapted to analyze requests from requestor devices to identify the requestor device type; and
a request redirection component, which is responsive to the requestor device being identified as a device of the first device type, to redirect the request to the first content server system and to retrieve the first version of the Web content.

A further aspect of the invention provides a method for automated generation of a new version of media content, the method comprising:

analyzing media content to detect identifiers of content types within the media content;

establishing a media content model that determines the significance of the identified content types within the analyzed media content;

comparing the identified content types with a content requirements definition that is associated with a particular device type; and

selecting the identified content that matches the content types within the content requirements definition, and combining the selected content consistently with the media content model to create a new media content-version.

In a preferred embodiment of the automated generation method, the requirements definition includes information defining processing operations to be performed on selected content and the method of automated generation comprises performing the defined processing operations on selected content before combining the selected content to create a new media content version. The content requirements definition preferably comprises an XML configuration file specifying screen dimension and/or layout information for a type of user device.

Another aspect of the invention provides a media content generator for carrying out a method as described above.
Various components of the invention described above, such as the request analyzer, Website analyzer, mini page generator, template generator and the request redirection component and route determination function of the router and proxy server, may be implemented as one or more computer program products. Such products comprise computer program code recorded on machine-readable recording medium and configured to control the operation of a data processing system on which the program code is executed. Apparatus such as the proxy server and content server according to preferred embodiments of the invention preferably comprises one or more data processing systems having one or more processors, data storage resources including random access memory and non-volatile storage providing a content repository and cache for mini pages, and network communication capabilities. These systems are capable of running the computer program implementations of the above-described components.

DESCRIPTION OF DRAWINGS

Embodiments of the invention are described below in more detail, by way of example, with reference to the accompanying drawings in which:

Figure 1 is a schematic representation of the components of a Web content delivery platform according to an embodiment of the invention, for use in a communications network;

Figure 2 represents a directory access request and mini page access via the directory, according to an embodiment of the invention, and shows example screen shots including an example mini page and example MiniSite directory pages that can be used to link to the mini page;

Figures 3A, 3B, 4A and 4B show the sequence of operations of a content access method, which operations are performed at various components of the communications network of Figure 1, according to an embodiment of the invention;

Figure 5 shows examples of list modules in a Web page;
Figure 6 shows an example article module;

Figure 7 shows a mini page representation of a list module;

Figure 8 shows an example mini page representation of an article module;

Figure 9 shows an example visual list module and its corresponding DOM structure;

Figure 10 shows an example visual component and the way it is specified in the example DOM structure;

Figure 11 shows an example article module and its subdomain tree;

Figure 12 shows an exemplary snippet of a list module definition;

Figure 13 shows an example of two modules with matching DOM structures;

Figure 14 shows the structure of a configuration file according to one embodiment of the invention; and

Figure 15 shows compression and other transformations for resource handling, according to an embodiment of the invention.

**DESCRIPTION OF EMBODIMENTS**

Embodiments of the invention are described below with reference to Figures 1 to 14. It will be appreciated by persons skilled in the art that the described embodiments are illustrative of various features and advantages of the invention, but the invention is not limited to the particular embodiments described in detail. References to the invention below are to be interpreted as referring to one or more embodiments of the invention, without limitation. The invention includes various aspects encompassing all of the subject matter within the scope of
the accompanying claims, description and drawings, including all combinations of the described features except technically impossible combinations, and equivalents to the functions and components of the illustrative embodiments.

Figure 1 is a schematic representation of the architecture of a Web content access platform 10 according to the invention, and operational steps for accessing Web content are described with reference to Figures 3A, 3B, 4A, 4B. The platform comprises a router 20 that may be running on a data processing system such as a network gateway server system (not shown) for receiving Web content requests from a plurality of different types of end-user devices 100, 110, 120, 130. In practice, there is a distributed network of clustered server computers that each provide the functions of the router 20, but a single system is shown in Figure 1 for simplicity. There may be a large number of different types of requestor devices 100, 110, 120, 130, including desktop and laptop personal computers and a range of PDAs and high-functionality mobile telephones ('smart phones') and low-end mobile telephones. In particular, there may be many thousands or even millions of users requesting Web content via many different types of mobile device that submit their requests to the content access platform 10, each of these different device types having different capabilities and constraints in terms of their screen sizes, processing power, memory capacity, communication bandwidth, Web browsers and/or operating systems, etc. The Web content requests may comprise conventional HTTP requests or an alternative form of Web content request such as described below.

The Web content access platform 10 of this embodiment hosts 'mini pages' in a content repository 60 and hosts a MiniSite directory in directory repository 140. The MiniSite directory comprises a list of available mini pages, and can be provided as a set of alternative versions of the directory pages, with each version being suitable for presentation on a different type of device. Example screen shots showing a MiniSite Directory screen display are included in Figure 2, but the displayed representation can vary greatly between devices. The router 20 within the content access platform 10 includes a mechanism for intercepting directory requests
and Web content requests. This mechanism is provided by the request analyzer 30, and its operations are described in detail below.

In a first embodiment of the invention, user agent header attributes (such as HTTP request headers) are analyzed on receipt of a Web content request, and are compared with a device database that can match the user agent (UA) header attributes to a set of characteristics and capabilities of the requestor device. This is described below. However, in an extension or replacement to this first embodiment, end users devices have pre-installed user agents that are configured to specify their device type explicitly within non-standard requests sent to the content access platform 10. This is especially advantageous if the communication network includes other communication gateway servers that do not pass on UA header attributes when routing requests.

In a particular embodiment, exemplified in Figure 2, a directory-reference URL is embedded in a user device for use by the device browser. The device type is specified by means of a subdomain name within the URL. Thus, a mobile telephone's Web browser is configured to send standard requests that include a unique resource identifier comprising a domain name prefixed by a device type identifier such as: 'e71.url23.com', where 'e71' identifies a specific device type such as a Nokia e71 mobile phone, and 'url23.com' identifies the domain name associated with a MiniSite directory. Although intuitive identifiers for device types (such as this example 'e71') are desirable, it is known in the art that there can be many levels within the hierarchy of labels within a domain name and each label can include up to 63 ASCII characters up to a total of 253 characters for the domain name, and so the simple example given above is just one simple example of use of a subdomain name as a device type identifier.

This use of a subdomain name is an alternative to using a more standard generic request format 'www.url23.com', and the content access platform of this embodiment allows for both request types to be used. Other implementations of the non-standard resource request can be used, as long as this is predefined for the requestor devices 100, 110, 120, 130 and the request analyzer 30.
of the router 20 such that any such non-standard requests can be interpreted correctly. Nevertheless, the use of device-type identifiers in subdomain names is advantageous because this can easily be extracted from requests and compared with a device type database.

On receipt of a request 200 by the router 20, the device type prefix defining a subdomain (e.g. the prefix 'e71' of subdomain 'e71.url23.com' of domain 'url23.com') can be extracted from the request and compared with a device database, to retrieve device characteristics or display requirements, and/or to determine which pre-stored MiniSite directory page 210,220,230 or mini pages 240 are suitable for the type of device.

In some cases, for example where the requested resource is a directory page, the device-type identifier within the request can be directly matched with a stored directory page that is associated with that device type. This process is described in more detail below with reference to Figures 3A, 3B, 4A and 4B.

The MiniSite directory support described above is advantageous for several reasons. Each type of mobile handset (e.g. each device series or model) can have its own version of directory pages, with different layout and look and feel. Customized directories can be provided and hosted within the content access server, with different content categories and different recommended URL links for different users. A user's personalised directory page can be held in storage of the content access platform and accessed by user requests specifying a sub-domain-name that identifies the directory page. This hosting of personalized resources enables users to share their directory pages (like music playlist sharing) with friends or work colleagues using either SMS or email to build a social network or increase workplace efficiency.

The embodiment described above takes advantage of a pre-installed directory-reference URL on mobile handsets 100,110,120,130 that is used by an installed browser 250,251 to send directory requests 200 via the router 20, but in an alternative embodiment a version of the MiniSite directory page 210,220,230 can be installed on the mobile handset itself. In either
embodiment, one major advantage of such a MiniSite directory page is the improved user experience that results from avoiding the need for users to key in Mini-page URLs.

Each request that is received by the content access platform 10 is received by the router 20, which forwards each received HTTP request to a respective content server. The content access platform 10 of this embodiment includes a plurality of content servers that each provide data storage suitable for storing content in an accessible form or for storing directories listing available content. For simplification, Figure 1 represents the content access platform 10 as a single system including a directory repository 140 and a content repository 60, but no limitation to a single system is required and it is well known that a single mainframe server computer can be replaced by a distributed network of servers, and references to a content server or a repository herein are intended to include the possibility of distributed architectures. Additionally, volatile cache storage is provided within the router 20 to provide fast access to recently used mini pages. The content access platform 10 has internet connectivity for accessing external Web servers 40 for accessing Web content that is not stored in these repositories, using known communication mechanisms.

Figures 3A, 3B, 4A and 4B show processes carried out within the content access platform 10 in response to receipt of a request 300 from an end user device. Referring firstly to Figure 3A, each request is received via the router 20 and analyzed by the router's request analyzer 30. The analyzer 30 initially checks 310 for a prefix other than 'www' within the subdomain name, identifying any such prefix as a potential device type identifier. Referring to Figure 4A, the analyzer compares 321 the subdomain name with known subdomain names representing device-type-specific resources (as a first mechanism for identifying device types or model numbers). If there is no identified match for the subdomain name, the analyzer also obtains 322 HTTP header attributes from the request and checks 323 for a device-identifier within HTTP header attributes, if such device attributes are received (some countries have communication infrastructures that do not pass on such header attributes, so it is advantageous that another mechanism, such as the use of subdomain names, is also implemented).
Having identified a device-type-identifying subdomain name or HTTP header attribute, the analyzer compares 320 the identifier with entries in a device database 50 that maps device type identifiers to device model numbers. Thus, if such a prefixed subdomain name is identified, the subdomain name is compared with the device database 50 to identify a device model number and this model number is compared 330 with the database to determine whether the characteristics or display requirements are known for that model. If the model is known, the subdomain name is also compared 340 with tables of user-specific subdomains to determine whether the request is for a directory page that is specific to an individual user or user group. If yes, the respective user-specific and device-type-compatible directory page is served 350 to the requestor device; if no, a device-type-compatible directory page is served 360.

However, if the comparison of a device model number (step 330) does not determine that the requestor is a known device type, a standard directory page is served 390 and an alternative mechanism is provided to determine an appropriate version of the directory page. This is achieved by sending 370 to the requestor device a request device information - preferably in the form of a user prompt, although this could be automated if communication devices are configured to automatically respond to requests for device information. The device model and retrieved information obtained in this way can then be added 380 to the device” database. Thereafter, devices of their type will be recognizable and can be matched to a respective device-compatible directory page. This provides a self-learning mechanism, such that the content access platform will support more devices over time.

The served directory pages enable easy user selection of mini pages that are available via the content access platform, via the known mechanism of browser activation when a user selects a hyperlink within a displayed Web page.

Described below is the analysis and processing of content requests, other than directory requests, when received by the router 20 of the content access platform 10. This includes
requests that are sent from a user's device when a user interacts with a displayed MiniSite directory page, as well as other requests for mini pages and conventional Web page requests.

When a specific mini page request is received by the router 20, the request analyzer 30 determines 310, from the domain name, that it is not a MiniSite directory request. When a mini page request is generated from a link within a MiniSite directory page, the requestor device type and a user preference token are included in the request. For other mini page requests, user agent information in HTTP request header attributes can be obtained 401 (when these attributes are available) and analyzed 402 and compared with a device profile database by the request analyzer 30 to determine 400 the device type. This can provide a specific device model number that has known screen size, processing power and browser capability. A check is carried out 410 of whether the requestor device is a full function personal computer (desktop or laptop PC).

If the requestor device is determined to be a conventional PC, the request is directed 420 to a standard Website on a Web server that includes content suitable for display or execution by the PC. The Web server provides access to the required resource in a conventional manner - retrieving and serving Web contents to the requestor device. This content can be served via the server system of the router, or via other network links.

The above-described routing of PC requests to conventional Web servers is not essential, and the Web access content server 10 includes features for hosting customized Web page versions that match user preferences, so some users may wish to exploit the capabilities of the content access platform even when there is no device-related motivation for special processing. In the present embodiment, conventional requests from PCs are routed to the target Web servers without serving mini pages but, in an alternative embodiment, PC users could register an interest in receiving customized content from the content access system.

If the requestor device is not a PC, the requestor device is assumed to be a device that would benefit from selection of a mini page or a version of Web content that is suitable for the requestor device. The router 20 thus determines 410 that the request should be processed further
within the content access platform 10. The request analyzer 30 analyzes the request and compares the requestor device model number with an index of device types in a device database 50 that is accessed by the router 20.

That is, although HTTP header attributes can include additional explicit device characteristics, in the present embodiment the device model number is extracted from the HTTP header for non-PC-originated requests and this number is used as a search key for accessing device capabilities from a device profile database 50. The device profile database 50 contains detailed characteristics for each of a large number of mobile device types, and is updateable as new devices come onto the market. Such device profile databases are known in the art (for example, open source lists of device capabilities are maintained in a publicly accessible manner). If user agent information is not available (e.g. is blocked within a communication network), a query page can be sent to the requestor device to invite users to specify device characteristics (screen size, manufacturer, supported media types, etc). The user can insert information into the query page and submit this to the content access platform. This device information can then be permanently stored (e.g. using cookies or a user account database).

The main additional information held in the device profile database 50 of the present embodiment is a device type identifier that can be used to represent the characteristics of the requestor device. In a simple example, all requestor devices having a particular range of screen dimensions (i.e. similar sized screens), memory within a defined range and communication bandwidth within a defined range may be identified as device type A. For devices having equivalent memory and bandwidth and browser support, devices with a smaller screen size may be identified as device type B and devices having an alternative screen layout may be device type C. A device with an identical screen size to device type A, but with less processing power, memory and colour depth may be device type D, and so on. There can be unique device-type identifiers associated with every mobile device model number, but grouping of device types
according to the present embodiment is desirable to enable greater reuse of different content versions and matching of new devices, as explained below.

Let us assume in this first example that the model number of a current requestor device is held in the database 50 with device capability information and an associated device type identifier. The device type identifier is retrieved from the device profile database and combined with the Web content identifier (typically a URL from the HTTP request header that identifies a requested Web page), and this combination of a device type identifier and Web content identifier are then compared 500 with an index of cached versions of Web contents that are stored in association with the router 20. A cache 25 within the router system holds this index and holds copies of the most recently retrieved mini pages and other Web contents, to enable fast access. If a suitable mini page is held in the cache, it is served 510 to the requestor.

If a version of the requested Web contents suitable for the requestor is not held in volatile cache storage 25, the combined device type identifier and Web content identifier is compared 520 with an index of Web content versions stored in a content repository 60 of a content server. The router forwards 550 the content request to the content server holding the required version of the requested Web contents, and this version is served 530,560 to the requestor.

Although represented by a single element in Figure 1, the content server holding content repository 60 may comprise a highly distributed server network including a plurality of content server systems. With a distributed approach, network bottlenecks can be avoided and less expensive server systems can be implemented (with built in redundancy and peer-recovery if required - as is known in the art). However, although contents may be distributed, in the present embodiment of the invention the index of stored versions is stored at each system that provides routing functions.

In the present embodiment, each of the content versions stored within a repository of the access server 10 is a 'mini page' comprising an alternative version of the original target Web page, or is a device-type-specific version of a content component - such as an image, a video clip, a
game or another application program. In typical cases, the mini pages and stored content versions will have a smaller data size than the original Web page contents, and will be more suitable for display on a small display screen or execution within a mobile device.

As mentioned previously, 'mini pages' in the context of this patent specification include both minor changes to an original Web page (such as a scaled down and re-ordered pages including substantially the same content as the original page) as well as simplified pages that omit a proportion of the original Web page content for simplified display on a resource-constrained device. A 'mini page' may also include components that were not included in a single original page, resulting in a new combination of content, and no limitation is implied by the word 'mini' which is intended to encompass any replacement Web pages that are created for display on a particular type of device or created for a particular set of users.

If the index of stored contents has a matching entry, the corresponding mini page or content component is retrieved from the content repository 60 of the content server and transmitted 530 to the requestor device 100, 10, 120, 130. The content repository 60 may be located in close proximity to the server data processing system that is providing the router 20 functionality, but the content access platform of the present embodiment is not limited in this way, and a typical content request will be routed across the network to the relevant content server. For static content, this routing relies on the index of stored content held at the router system to route the request to the respective content server holding the respective repository.

Where no cached mini-page is available, a request that requires access to Web content as a mini-page is routed to a Web site (or to a plurality of content sources) that includes suitable content. The request may be routed to the same Web server as holds the original Web page, if the mini-page and original web page are held on the same server. However, the requests are typically re-routed to a different content server that holds mini-page versions of Web pages.

This is implemented by means of a proxy service 70 that re-routes requests (that request content from an original site) to an alternative network location at which is held an alternative version
of the requested content. The proxy service is implemented using the Domain Name System (DNS) and uses knowledge of the location of different versions of particular requested content ('mini page' versions).

A Web page analyzer 150 is provided by the content access server for analyzing the contents of selected Web sites. The Web page analyzer is used to identify content components and component combinations within a Web page, and to compare these with known component combinations and layout patterns. An identified combination or pattern can be matched with stored presentation definitions or templates. Presentation rules or templates are stored that are specific to a recognizable pattern or combination of content components, and specific to a device type or user preferences. The Web page analyzer according to this embodiment cooperates with a mini page generator to enable a reformatting of Web page content onto a mini page that is suitable for the requesting device type or preferences of the requesting user.

The first major function of the Web page analyzer 150 of this embodiment is to semantically identify the high-level components of a Web page and to extract their meta data. These are the components that would be visually identifiable as separate components by a typical PC user. The analysis involves parsing the HTML within a Web page source file to identify static content component types, but also involves generating HTML DOM structures corresponding to the pattern or layout of the Web page so that dynamic contents that are created by client-side scripts are also identifiable. A script engine is implemented within the Web page analyzer to create the DOM tree of the page so that this structure (representing the layout pattern of the components) is available for analysis. A combination of specific component types is referred to hereafter as a 'pattern', and a 'module' is the term used to describe a particular combination of actual components - i.e. a module is an instance or implementation of a pattern.

Having identified known component types and known combinations of component types ('patterns'), the components and modules are tagged using keywords extracted from their title and description fields in the component metadata. A module comprising a combination of Web
site content components is thus recognizable if it corresponds to a predefined pattern within the DOM structure of the page. The modules that correspond to known patterns can then be used by a mini page generator (described below) to generate a mini page based on a set of stored templates that define presentation requirements of a particular requestor device type. Templates can also be customized according to particular user or user-group requirements.

The above-described component recognition and tagging operations of the Web page analyzer, and the mini page generation operations, are described in more detail below.

Two different types of module have been defined by the inventors of the present invention: list modules and article modules. These different types differ in their purpose and in their contents, as follows:

A 'List module' is a reference to content that is available on a separate Web page, and mainly contains the following elements:

- Title
- Description
- Image
- Link

Figure 5 provides examples of list modules (represented as Module-1, Module-2, etc) in a conventional Web page. Figure 7 shows a mini page representation 710 of a list module.

An 'Article module' contains actual Web page content, and mainly contains:

- Title
- Article meta (author, date, etc)
- Text content
- Images (if any)
- One or more list modules (such as "related links")
An example article module is shown in Figure 6, with a set of separately identifiable components title 600, metadata 610, picture 620 and text 630, as well as some related links 640. A mini page article module 720 is shown in Figure 8.

Module content may be static or dynamic, with a different analysis procedure required for dynamic contents. For static content, HTML elements within the Web page source file can be analyzed using an html parser to extract the component-identifying HTML headers. For example, an article title and the body of a text article can be identified as separate elements. For dynamic content created by client side script (i.e. java-script, AJAX), the content components can be identified by a script engine that creates the Web page DOM tree at execution time. An example is user comments on a news Web site.

The module identification process includes a pattern recognition step. However, pattern recognition can be iteratively improved by means of an adaptive learning process as new pattern are identified and stored.

Initially, the system generates or is provided with a basic set of module-definition files (in XML files format). These can be obtained by analyzing a selected set of popular web sites (e.g. The Guardian, BBC, YouTube, etc) and identifying the static HTML components and dynamic components that are part of the DOM structure of pages on these sites. Having generated and saved a set of module-definitions, the content access platform will be able to automatically identify modules from unknown Web sites by using existing module-definition-files obtained from sites that have identical or similar HTML DOM structures. Additionally, each time the Web page analyzer within the content access platform encounters unknown patterns, a system administrator can be prompted to create new definitions for the new patterns. An example visual list module component 910 and its corresponding definition 920 in the DOM tree are shown in Figure 9. This can involve manual steps as set out below.
Based on a module definition file and other information, the system administrator can:

Specify URLs or URL-patterns applied to the module;

Specify location(s) of contents in the module, including:

- **List module**: specify the outer-most enclosing HTML element (i.e. `<DIV>`, `<TABLE>`, etc)

- **Article module**: specify HTML elements for each attribute in the module, including related links (which are list modules embedded in an article module)

Download the page source file using a traditional Web crawler;

Parse the HTML file into DOM tree using html parsers;

Locate the enclosing element according to module-definition information;

Extract data from DOM Tree;

- **List module**: extract all ANCHOR (`<A>` tag) within the enclosing element, then combine textual or imagery date with same "HREF" attribute

- **Article module**: locate and extract data from html elements of every attribute.

An example URL pattern is given below. This comprises a regular-expression (REGX) used as criteria to define a set of URLs. Any URL that matches the URL pattern REGX belongs to the set.

The following are examples of URL pattern and the corresponding URL set:

<table>
<thead>
<tr>
<th>URL pattern:</th>
<th>URL set:</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://nba.tom.com/index.html">http://nba.tom.com/index.html</a></td>
<td>Tom NBA front page</td>
</tr>
<tr>
<td><a href="http://nba.tom.com/news/*_htm">http://nba.tom.com/news/*_htm</a></td>
<td>All Tom NBA news pages under the URL</td>
</tr>
</tbody>
</table>
Figure 11 provides an example of an article module 1110, showing the association between the visual components within that module and their details in the subdomain tree. A snippet of a list module definition is provided in Figure 12.

Additionally, module identification can be automated, as described below. When no module definition file and information is yet available, the analysis method involves downloading the page source file (e.g. using a traditional crawler) and parsing the HTML file to generate a DOM tree (using an html parser). The DOM tree is then traversed and, for every node $N$ traversed, the following steps are performed:

1. Check the sub-dom-tree rooted at $N$ against existing module-definitions in the system using a score-based matching algorithm
   - Match—score greater than threshold $T\text{-MAX}$
   - Not Match—score lower than threshold $T\text{-MIN}$

2. Select the matched module-definition $D$ that has the highest score, as the module definition for the sub-dom-tree, and identify the sub-tree as a module of type $D$

The specifying and locating of HTML elements involves use of a tree-path string of the element node in the DOM tree. For example, referring to Figure 10, to specify the location of text component 1010 "Ajax dynamic content ...", according to its element 1020 in the DOM tree, its
Locating a module or its attributes is then simply a tree path search. The tree path string comes in two forms:

- Absolute path—always relative to DOM tree root (like example above)
- Partial path—which does not start from the DOM tree root. For example, 

\[ \text{div[id:header]-div[class:title]-hl} \]

Having generated and stored a set of modules representing component patterns, modules within new Web sites that implement one of these patterns can be automatically recognized, as follows. Given a URL whose content is unknown to the system, the referenced page is retrieved and parsed into a DOM tree. The DOM tree is broken into Sub-trees and the system checks each sub-tree against existing module-definitions, trying to find a best match for each sub-tree. The best match module-definition is used to automatically extract data for the sub-tree (module). Note that if two HTML visual components have similar or identical DOM structures, the module definition of one module can be applied to another module. An example of two modules 1310 and 1320 having the same module definition (by virtue of having the same DOM structure) is shown in Figure 13. The "more news" and "latest multimedia" modules on an exampletnews Web page have different content but the same DOM structure.

Modules can be saved in categories according to keywords as well as HTML DOM structural information. Each of a set of categories such as news, finance, sports, is assigned with a set of representative tags or keywords. Keywords can be extracted from a module based on text word frequency or other measures, and the extracted keywords can be matched against a keyword set that is stored for each category. The keyword matching is evaluated by use of a weighted matching score, with the module identified as belonging to the category having the highest matching score.

A new mini page may be built in response to a user request at runtime, or as a pre-emptive operation when Web pages are identified that are expected to have a high demand for requests
from mobile devices. Newly created mini pages can be stored at a content server for reuse (for example caching in volatile storage, if this is in accordance with a mini page caching policy). Such mini pages can combine multiple components that are located on different content servers within a network. In this case, a configuration file is used to identify the components that are to be included in a mini page, the IP address of the respective server holding each component, and how the components are to be displayed on a particular type of requestor device. The structure and use of such configuration files is described in more detail below.

When a request is received by the content access platform from a mobile device, and the required mini page is not available in cache memory, a respective device-type-suitable configuration file is retrieved from a content server and information within this file is used to identify components of interest. The components can be retrieved from respective content servers and processed and combined as specified in the configuration file. The proxy server that re-routes requests to specific content servers can invoke the Web site analyzer program (described above) to retrieve required Web content over the Internet. The retrieved page content can be decomposed into its component parts using analysis functions within the Web site analyzer, and content components can be extracted as specified within the configuration file. The configuration file specifies how the components are to be combined and adapted to form a new mini page, including the component order and format determining how this mini page is to be displayed on relevant devices. The mini page is then returned to the requestor device. The mini page can be cached at this time.

The return of an alternative version of the originally requested Web page contents to the requestor device is transparent to the device user, who simply benefits from a faster response time and the ability to view a mini page that is suitable for the particular device screen, without needing to consider whether this is an alternative version or the originally requested content.

The content access platform 10 according to the present embodiment thus provides a distribution mechanism for mini pages, some of which may be manually created and some of
which may be generated automatically. The access and distribution mechanism described above functions in the same manner regardless of how the mini page was created, and regardless of whether it was added to the platform by the original Website provider or was generated automatically within the content access platform. Mini pages can also be provided by other interested parties (such as a mini page designer or an end user).

Manual creation of alternative Web pages for display on a specific type of device requires considerable effort. As well as the effort involved in actually implementing a new page, the creator has to decide on a desirable arrangement of page components on the display device, whether to maintain a page hierarchy or replace it, and which components of the original page to include and which to omit. The creation of alternative pages thus needs an assessment of the original page content and requires a lot of decisions about the presentation of the content on the display screen of the particular device. The work involved in creating a new version of a Website or Web page is therefore considerable, and requires careful consideration of the capabilities of the target devices and the requirements of users. This work is often considered to be too great to enable a different Website version to be provided for each of the many different user device types in existence.

The content access platform of one embodiment of the invention avoids this manual effort by providing a mini page generator 90 for automated mini page generation 590. A first function of the mini page generator is to apply pre-generated templates 800 (representing a suitable page layout and content display properties and requirements for a particular device type) to content obtained 560 from a Website of interest, selecting required components of the retrieved Web page and reformatting the content for display.

The templates 800 may be implemented as XML configuration files that define mini page presentation properties for an identified device type, and which can be applied to Web page contents to control the generation of a new version of the Web page contents. By generating a new version ('mini page') that is appropriate for a requestor device, and making that available
to the requestor device, optimized presentation of the requested Web content is enabled for the particular requestor device. The pre-generated templates can be stored in a combined template and content repository within the content servers, such that an initial search of the cache and mini page repository for a matching mini page can, if unsuccessful, be quickly followed by identification and retrieval of the relevant template. An appropriate mini page can then be created by applying the configuration instructions of the template to the original Web page contents, selecting components of the Web content and reformatting for display on the requestor device screen. Figure 14 shows an exemplary template structure.

The customized configuration file (template) structure of Figure 14 is implemented as an XML file. The tree structure illustrated is part of the Document Type Definition (DTD) of these XML templates and various nodes within the DTD tree structure are explained below.

- 'Template' - this is the root node of the structure.

- 'Identification information' - includes a unique template name (such as 'nba data' for a template associated with the National Basketball Association's statistical results data); and a template description (for holding a description such as 'nba statistic data', and may include other options).

- 'List of URLs of interest' - exists to hold URLs for content that is to be included in the mini pages created from this template. This includes a URL pattern (such as 'http://data.nba.tom.com/stats.htm' and 'http://nba.tom.com/news').

- 'Web page analyzer' - defines crawling properties, including a content server IP address at which the template may be applied to Web content. The analyzer node also specifies page content to be crawled (i.e., portions or parts of the target page to be retrieved and stored, such as specific HTML elements to be discarded and elements to be retained, and whether original style tags should be retained or not). Also specified is
an analyzer bean 810, which is a specific Java bean for extracting data from a source Web page.

- 'Presentation of mini pages' 820 - defines how to reformat Web page content based on the capabilities of the requestor device type that this template is associated with. This node includes a default view Java Bean 821 (which may be included if device-specific reformatting information is not included in the node 820) and a device rendering list 822 which lists characteristics for one or more device types. For example, a device may be specified by its device number 824 'nokia95' or 'nokia72' for example, its screen width 825 '240px' or '176px' and other capabilities such as whether there is JavaScript support ('yes' or 'no') and flash memory support ('yes' or 'no'). The device rendering list also includes a rendering style/format 828 and identification of a page component to be rendered 827.

- 'Accelerator Properties' 830 specifies whether to apply download acceleration or not ('yes' or 'no'). This can make use of a conventional dedicated download manager program (80 in Figure 1), which implements such techniques as searching 831 (i.e. crawling the Web) for mirror sites and performing parallel segmented downloading, variable bandwidth usage and limiting, resuming paused downloads, etc. Also specified, within a cache behaviour definition node 832, is a cache interval in seconds (e.g. 180 seconds).

The creation of a template for use to create and serve mini pages involves the following steps:

- Firstly, the contents included in the mini page must be specified within the XML configuration file, for example by specifying: URLs of original Web pages, whose contents will be used in the mini page; and, for each original Web page, a definition of which part(s) need to be included (the content components or "assets" of a Web page). A component can be specified using an HTML element ID, or using the position in the HTML document tree of the page such as, for example, the content contained in 3rd
division (DIV) of the page body. A component is uniquely identified by its ID. A component can be reused by other mini pages. For example, if a user wanted to create a mini page that shows live stock index chart from YAHOO and MSN money, the user needs to identify the source pages www.vahoo.com and www.msn.com as well as identifying the 2 components

- Secondly, how components are to be arranged and rendered on the mini page must be specified within the XML file. An operator specifies mini page metadata (such as a page title, CSS style-sheet, background, etc.). The IDs of components to be included in the mini page are specified. For each component, the order, size and style in which it should appear in the mini page is specified. Preferences for reformatting are specified (e.g. background, font).

Templates representing content display properties for a particular device type may be stored in the template and content repository for that particular device type without reference to a particular Web site or other Web content. However, a first embodiment of the invention selects a stored template that is appropriate for the particular type of Web site as well as the particular user-device type whenever such a template is available. For example, a site may be identified as a sports event site, a news site, a retail store site, and so on, and the way Website content is selected and presented on the requestor device can then be adapted for the type of site being considered. The achievement of this adaptation of display properties for different Website types or categories as well as different device types is described below.

Templates as set out above thus enable the creation of customized combinations and customized layouts and formats of content components, that may be retrieved from a plurality of different Web sites and other network-accessible sources. By providing a hosting mechanism and hosting service for mini pages and templates that are used to create mini pages, the content access platform of the present invention makes it possible for Website providers and others to provide these new combinations of content for access by a wide range of different types of
communication devices - including PCs, PDAs and high-end or low-end mobile telephones. By integrating this new publishing capability in a system that also handles routing and proxy service redirection functions, content adaptation functions and improved download speeds, embodiments of the invention can significantly improve access to Web content and enhance the user experience for people accessing Web content from a mobile device.

As noted above, a Website analyzer component is associated with the mini page generator, and implements an analysis of the contents of the selected Website in order to determine whether the Website can be categorized as one of a number of predefined Website types. In a first embodiment, the Website analyzer is implemented as a computer program that is configured to scan selected Websites (or other online content) to analyze their contents. A first function of the Website analyzer program is to determine whether the content matches any one of a set of predefined categories of content. The analyzer program scans the text of a Web page to identify high frequency words, and then compares the identified high frequency words with a knowledge base in which keywords are associated with specific categories of Website or Web content. This provides a first high-level categorization and indexing of crawled pages, enabling identification of sports-related Websites or pages, and separate identification of financial information Websites or pages, etc. Additional pattern matching algorithms and data mining techniques can be implemented, including image analysis and other content and context analysis techniques may be employed, but this is not essential. High frequency keywords and categorization information can then be stored in association with a Web page or Website.

Within a scanned Web page, various elements of content may be categorized separately. For example, images can be tagged in a Web page differently from video, differently from links to other pages, and differently from text. Even within a block of text, HTML tags can identify titles distinctly from other text paragraphs. The tags and word frequency statistics and other content-analysis can identify specific content within a Web page. The identification of different elements or "assets" within a Web page can thus make use of existing HTML tags and the HTML can be parsed to identify the separate elements. The HTML tags or other characteristics
of these elements can then be used to assign each element to one of a number of categories of content. Then, for each of a number of different user device types, a template can be automatically selected or a new template can be generated. The selected or generated template will indicate a subset of the identified categories of content that are to be included in a particular 'mini page' version of an original Web page or other content. The template also specifies their layout within a defined display screen area.

The templates in this embodiment are implemented as XML configuration files, and can specify which categories of content are to be included in a mini page without changes, and which categories of content are to be reformatted or transcoded in some other way when being included in the mini page. For example, an image, a video clip or an audio file could be replaced with reduced-data-size versions - providing the 'best fit' in terms of resolution/quality of video, image and audio while taking account of the resource constraints of the user's device (screen size, memory, bandwidth, etc). For example, a text column could be reformatted to fit within a predefined screen area. The configuration file that is defined for a particular device type can then be applied to the categorized set of building blocks of the Web page of interest to control generation of a mini page. The mini page can then be stored in a content server with other 'mini pages' and content that is suitable for devices of the 'particular type.

The above-mentioned templates or configuration files are generated with reference to end-user device characteristics, such as screen size data, and with reference to the requirements of the target demographic of device users and the requirements of the Web site provider. Some Web site components may be prioritized more highly than others to achieve particular emphasis on a display device. The generation of a template is described in detail above.

The above-described Website analysis can be performed pre-emptively by the Website analyzer in advance of specific requests being received for content from that site, with the Website analyzer running as a background task to identify Websites having content which matches a list of content types and predefined topics that are expected to be of interest to mobile device users.
and hence for which mini pages are likely to be required. However, in the present embodiment
a pre-emptive determination of which sites are to be analyzed and to have mini pages created
for them is based on a combination of pre-emptive Website provider decisions and measured
network traffic or measured Website hits corresponding to actual received requests. This
approach ensures that mini pages are generated for the most frequently accessed Websites.

Having identified the device type and a categorization of the Website, it is possible to assess
which components of the requested Web page are likely to be of greatest significance for users
as well as for devices of this type, and whether the significant components can be displayed
appropriately on devices of this type. In the present embodiment, this is implemented by storing
both a requestor-device-type identifier and a Website-content-type identifier with a stored
template. Taking account of both the characteristics of the device and the characteristics of the
Website enables selection of the best available stored template for generating a mini page.

Having generated a new mini page, an assessment is needed as to whether the new mini page
should itself be saved to the cache for ease of reuse, or discarded. The cache is organized in two
layers, comprising a first cache layer in volatile memory (e.g. RAM of the content server) for
storing frequently access mini pages and templates, and a second cache layer in non-volatile
storage (e.g. disk storage) for less-frequently accessed mini pages and templates. The
determination of whether to cache a mini page is based, in the present embodiment, on how
frequently the original Web page corresponding to the mini page is refreshed. However, the
caching policy can also take account of whether typical refreshes change significant
components of the data contents and how frequently significant contents change. The above-
described analysis of Website contents and Website categories enables this assessment of
whether significant content changes frequently. For example, in a sports Web page that
provides still images and match scores, a mini page should not be cached for such a long period
that the match scores become very outdated and yet retaining a still image on a mini page after
that image has been replaced on the main site is likely to be less important to users. The
analysis and categorization of a Website and its contents by the analyzer enables this Website-specific caching policy to be implemented.

The above description applies to situations in which either a suitable mini page exists for a requestor device or a suitable template exists for generation of new mini pages. However, the content access platform of the present embodiment also provides support for generation of mini pages using a default template, and support for the generation of new templates, in order to cater for new device types that become available in the future.

If a requestor device type cannot be identified from a device model number within the HTTP header attributes or a received request, two options exist. In a first embodiment, the HTTP header details are investigated more fully to identify characteristics of the requestor device type, and these characteristics are compared with the device capabilities information in the device profile database to identify similar device types. A device type identifier corresponding to the best fit device type is then used to select a template that is likely to be suitable for the new device type. In one embodiment, this template is used to generate a suitable mini page for the requested Web page, and the mini page is returned to the device user, but this mini page can be returned with a user prompt asking for further details of the device type. This can be implemented by sending the mini page with a pop-up window or a separate SMS text message inviting the user to reply if they are unable to view the mini page effectively on their device. The user can then be invited to download and execute a small footprint device analysis program, which obtains device capability information for use in generating a new device-type-specific template. The template structure shown in Figure 14 is used for the newly generated template, with device characteristics and other fields being populated with information obtained by the device analysis program. The new template is then added to the template and content repository of the content server.

Some users will decline to run the device analysis program, and an alternative approach is to ask the user a set of questions about their device, but if some users execute the program the
results can be used to update the device profile database, and one or more new templates can be generated.

An alternative embodiment selects a default template whenever a suitable template cannot be identified and this default template is used to generate a fist mini page for sending to that requestor device. As described above, the fist mini page can be returned with one or more user prompts to establish whether the mini page could be displayed or to obtain device information.

The content access platform according to an embodiment of the invention makes use of resource compression and other transformations, to ensure that the size and format requirements of different devices (e.g. different mobile handset models) are matched by the mini pages that are served to those devices. Figure 15 comprises an example table of multimedia resources and the compression and format transformations that are applied to them in one embodiment of the invention.
CLAIMS

1. Apparatus for use in a communications network, comprising:

   a template repository storing a set of template files that each specify information for determining how Web content is to be displayed on a particular type of user device that is associated with the respective template file;

   a request analyzer, arranged within the network for analyzing requests for access to Web content, wherein the content requests include information enabling identification of the device type of the respective requestor device, and wherein the request analyzer is adapted to analyze requests from requestor devices to identify the requestor device type;

   a router that is responsive to identification of a requestor device type to retrieve a respective template file from the template repository and, in response to the template specifying a plurality of separate sources of Web content, to send a plurality of Web content requests to the separate sources, thereby to retrieve required Web content from the plurality of separate sources; and

   a content version generator for combining the retrieved Web content and providing the combined content to the requestor device.

2. A method for accessing Web content via a communications network, the method comprising:

   responsive to receipt at a router of a content request from a content requestor device, which content request includes information enabling identification of the requestor device type, identifying the requestor device type from the information within the content request;
routing a first content request from the router to a first content server that is known by the router to have access to a first version of the requested content, wherein the first version of the requested content is predefined as suitable for display on devices of the identified requestor device type;

routing a second content request from the router to a second content server that is known by the router to have access to additional content that is predefined as suitable for display on devices of the identified requestor device type;

retrieving the first version of the requested content from the first content server and retrieving the additional content from the second content server;

combining the first version of the requested content and the additional content; and

delivering the combined content to the content requestor device.

3. A router system, for use in a communications network, wherein the router system is arranged for receipt of content requests from content requestor devices, which content requests include information enabling identification of the requestor device type, the router system comprising:

a repository storing network location information for a plurality of sources of content, the repository also including an identification of respective device types with which each of the plurality of content sources is associated;

a request analyzer for analyzing received content requests from requestor devices to identify the respective requestor device type; and

a request redirection component that is responsive to identification of a requestor device type to redirect the request to a selected plurality of the content sources.
4. A method for automated generation of a displayable version of Web content, the method comprising:

   analyzing Web content to detect identifiers of content types within the content;

   comparing the Web content types with predefined categories of content to determine a content category;

   in response to a request for the Web content, which request specifies the requestor device type, identifying a template representing content display requirements for contents of the content category and devices of the specified device type; and

   providing Web content to the requestor device in accordance with the content display requirements represented by the identified template.

5. Apparatus for use in a communications network, comprising:

   a Web content analyzer for analyzing and categorizing Web content;

   a first content server system comprising a repository for storing a first version of one or more items of Web content, the first version of the Web content being predefined as suitable for display of Web content on a first category on devices of a first device type, and wherein the first content server system is configured to provide access to the first version of the Web content in response to receipt of a request for the Web content; and

   a request analyzer, arranged within the network for analyzing requests for access to Web content wherein the content requests include information enabling identification of the
respective requestor device type, wherein the request analyzer is adapted to analyze requests from requestor devices to identify the requestor device type; and

a request redirection component, which is responsive to the requestor device being identified as a device of the first device type, to redirect the request to the first content server system and to retrieve the first version of the Web content.

6. A Web contents analyzer, comprising:

an HTML parser for determining a DOM structure of a Web page; and

a pattern matcher for comparing the DOM structure with known patterns of Web page contents to determine whether a Web page includes one or more known patterns of Web page contents.

7. A computer program for use in a communications network, the computer program including program code for execution by at least one processor to control at least one data processing apparatus to analyze Web page contents, wherein the program code comprises:

an HTML parser for determining a DOM structure of a Web page; and

a pattern matcher for comparing the DOM structure with known patterns of Web page contents to determining whether a Web page includes one or more known patterns of Web page contents.
8. A computer program according to claim 7, further comprising a content version generator for generating a new version of said Web page contents in accordance with a predefined transformation for the matched pattern.

9. A computer program according to claim 8, wherein said predefined transformation comprises a device-type-specific transformation to match a requestor device type capability.

10. A computer program according to claim 9, wherein the content version generator employs a device-type-specific template representing presentation requirements for the matched Web page content pattern.

11. A computer program according to claim 8, wherein the generator is adapted to perform a transformation with reference to a defined user-preference.

12. A computer program according to any one of claims 7 to 11, further comprising a script engine for execution of active components of a Web page to determine the parts of the DOM structure corresponding to the active components.

13. A method for analyzing Web contents comprising: determining a DOM structure of a Web page; and comparing the DOM structure with known patterns of Web page contents to determine whether a Web page includes one or more modules implementing known patterns of Web page contents.
14. The method of claim 13, further comprising tagging said modules with meta data.

15. The method of claim 13 or 14, further comprising generating a transformed version of said modules in accordance with a known transformation requirement for said modules.

16. The method of claim 15, wherein said transformation is responsive to identification of a requestor device type to transform the modules to comply with presentation capabilities of the requestor device type.

17. The method of claim 15 or 16, wherein said transformation is responsive to identification of a requestor user preference to transform the modules to comply with the user preference.

18. The method of claim 13, further comprising:

   in response to a Web contents access request from a requestor device, analyzing the request to identify the requestor device type; and

   selecting one of a stored set of Web page transformation templates having a known association with the identified requestor device type.

19. The method of claim 18, further comprising:
transforming the requested Web contents in accordance with the selected transformation template, and serving the transformed Web contents.

20. A request handler, for analyzing requests from requestor devices for access to a Web contents directory, comprising:

means for analyzing a received request to identify a URL subdomain name within the request;

means for comparing the received request's subdomain name with a stored set of subdomain names that are representative of respective requestor device types within a set of requestor device types; and

in response to identifying a match between the received request's subdomain name and a first one of the stored subdomain names representative of a first requestor device type, initiating retrieval of a first directory page that is compatible with the requestor device type.

21. The request handler of claim 20, further comprising a repository of directory pages and means for serving the first directory page to the requestor device.
Glastonbury lineup: Lily Allen and Nick Cave to join Bruce Springsteen and Blur

Kasabian, the Specials and Metronomy to join veteran stars Tom Jones, Neil Young, Bruce Springsteen and Blur at this summer's festival, as lineup is revealed

Reade Swain
guardian.co.uk, Monday 25 May 2009 12.24 BST
Article history

Beauty of the Holy City
Ancient walls under occupation:
Jerusalem

Music
Glastonbury: Bruce Springsteen - Neil Young - Blur - Lily Allen - Kasabian - Jay-Z
Culture
Festivals
More news

More on this story:
Blog: What do you think of this year's lineup?
In full: Glastonbury festival lineup 2009
Win Glastonbury tickets!

Music from the Guardian shop
94.44 recording of all the Haydn Symphonies, compiled on 3CDs, including the 80-page No.45 Farewell, and of course the extraordinary No.44. "For ever."
Price: £96.00.

Related links

Guardian.co.uk/music news alert

It's a case of out with the new and in with the old at Glastonbury 2009, as organisers confirm the full lineup for this year's event. As well as headline slots from Blur, Bruce Springsteen and Neil Young, performances from Stone Roses, Pulp, Arcade Fire and Spiral TV have also been confirmed for the...
<html lang="en" xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  
  <head>
  </head>

  <body id="body">
    <div id="header">
      <div id="logo">
        <ul id="nav">
          <script type="text/javascript" src="http://www.google-analytics.com/urchin.js"></script>
          <script type="text/javascript"></script>
          <div class="title">
            <h1>Ajax dynamic content</h1>
          </div>
        </ul>
      </div>
    </div>
  </body>
</html>
David Cameron: I would reduce No 10’s power

David Cameron will tomorrow pledge to deliver the most dramatic redistribution of power in living memory.
## Format Transformation table

<table>
<thead>
<tr>
<th>Multi-media resource</th>
<th>Original formats</th>
<th>Converted formats</th>
<th>Original size</th>
<th>Converted size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Picture</td>
<td>jpg, bmp, gif, png</td>
<td>png or jpg, (imagery data are resized to fit into device screens)</td>
<td>160-300k</td>
<td>5-8k</td>
</tr>
<tr>
<td>Video</td>
<td>wmv, 3gp, mpg/mpeg</td>
<td>3gp (Video Codec: H.264/MPEG-4, Bitrate: 94/180kbps, Dimension: 176x144/320x240, Frame rate: 9/10/12fps)</td>
<td>1-3M</td>
<td>500k-1M</td>
</tr>
<tr>
<td>Audio</td>
<td>wma, mp3, rm</td>
<td>Audio Codec: AAC, Bitrate: 32kbps, Sampling Rate: 22.05kHz, Stereo</td>
<td>2-3M</td>
<td>500-1.5M</td>
</tr>
<tr>
<td>Flash</td>
<td>swf, flv, fla</td>
<td>swf, flv, fla (quality and frame rate may be lowered)</td>
<td>5-10M</td>
<td>2-5M</td>
</tr>
<tr>
<td>Java Applet</td>
<td>Java Applet</td>
<td>Java Applet (plug-in download required)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# INTERNATIONAL SEARCH REPORT

**International application No**

PCT/GB2010/000295

## A. CLASSIFICATION OF SUBJECT MATTER

**INV.** G06F17/30

ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database consulted during the international search (name of database and, where practical, search terms used)

EPO-Internal, INSPEC

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate of the relevant passages</th>
<th>Relevant to claim</th>
</tr>
</thead>
</table>

**D** Further documents are listed in the continuation of Box C

| X | See patent family annex |

- **"** Special categories of cited documents
- **"A"** document defining the general state of the art which is not considered to be of particular relevance
- **"E"** earlier document but published on or after the international filing date
- **"L"** document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- **"O"** document referring to an oral disclosure use exhibition or other means
- **"P"** document published prior to the international filing date but later than the priority date claimed
- **"T"** later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- **"X"** document of particular relevance, the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- **"Y"** document of particular relevance, the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents such combination being obvious to a person skilled in the art
- **"&"** document member of the same patent family

**Date of the actual completion of the international search**

21 April 2010

**Date of mailing of the international search report**

03/05/2010

**Name and mailing address of the ISA/Office**

European Patent Office, P B 5818 Patentlaan 2 NL - 2230 HV Rijswijk

Tel (+31-70) 340-2040,
Fax (+31-70) 340-3016

**Authorized officer**

Barieux, Marc

Form PCT/ISA/210 (second sheet) (April 2005)
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>WO 2008073207 A2</td>
<td>19-06-2008</td>
</tr>
</tbody>
</table>