

FORMATTING AND COMPRESSION OF CONTENT DATA

BACKGROUND OF THE INVENTION

Field of the invention

[0001] The technical field of the invention is formatting and compression of data. An example of an application of the invention is formatting and compression of data to optimize the data for transmission using a wireless network.

Description of the Related Art

[0002] On the Internet, most designers currently do not consider the size of the content data, such as graphics, text, animation, video and audio data, on their web pages believing that most users access the web pages via high speed broadband Internet connections, for example DSL, cable modem, T1, E1, or other high-speed connections, and believing low speed dialup connections are increasingly rare.

[0003] However, in the case of devices such as mobile phones, connections are typically slower than broadband internet connections. This is due to a number of factors, such as bandwidth restrictions, network resource limitations, device limitations and incumbent wireless network protocols. For example, using a GSM (Global System for Mobile communications) 2.5 GPRS (General Packet Radio Service) connection the data transfer rate is similar to that of a 9,600 baud modem. This slow data channel limits the user in quickly accessing data.

[0004] Currently the cellular industry is attempting to meet their customers' demands for faster data transfer by installing higher speed data connection network capabilities such as 3G, EDGE (Enhanced Data rates for GSM Evolution), UMTS (Universal Mobile Telecommunications System), HSDPA (High-Speed Downlink Packet Access) and other systems that provide higher data access speeds. This upgrade of the networks requires substantial expenditure on network infrastructure by the cellular network carriers and also mobile users as it is necessary for users to purchase new mobile handsets and devices that are compatible with the new higher speed networks. These newer network internet connections,

which allow up to 64 Kbps data transfer channels, are still very slow compared to DSL or Cable modem data transfer speeds.

[0005] Another problem for cellular users is that the development of the data infrastructure between the Internet and cellular base stations (cell towers) currently lags behind the network infrastructure development such that even with a high speed connection between the handset and the tower the connection to the Internet is lacking.

[0006] A second problem for the majority of mobile devices is that once data is delivered to the mobile device, even in formats designed for mobile devices such as the existing MIDP (Mobile Information Device Profile), WAP (Wireless Application Protocol), XML (Extensible Markup Language) and other data standards, these formats are too large to be easily processed to fit on small screens using slow processors with limited memory space.

[0007] There is currently a lack of compatibility between browsing and web page translators used on different mobile devices, which inhibits the ability for a web designer to control or even test how the content will be displayed on the mobile device. Further, not all browsers or translators are compatible with all the existing formats. The result of this is that most times a HTML web page or WAP page containing XML-formatted content will not appear correctly formatted on the screen of mobile devices.

[0008] The third problem in current mobile devices is that unlike the PC industry where only a few different screen sizes are used, there is great variation in the screen sizes, aspect ratio, proportions, pixel counts and resolutions of screens used for mobile devices. Today when a web page designer designs a page they use an 800 x 640 or a 1024 x 768 pixel count as the base page. When displayed on larger formatted screens the data is surrounded by white space. Currently, the problem of the great variation in mobile device screen size, aspect ratio, pixel count, resolution and proportions is ignored by website designers with the result that displays can not handle the data that is to be displayed. Also, displaying web pages on different size screens can cause the aspect ratio of the web page and elements on the page to change. Images that are displayed on non-computer screens can be distorted when the screen is smaller or larger or with different ratios of horizontal and vertical dimensions from the original image. This distortion effect typically occurs when resizing images for small

cellular phone screens or for large HDTV screens that are designed for TV programs and not web site displays.

SUMMARY OF THE INVENTION

[0009] One embodiment includes a method of optimizing content for reproduction on a target device, comprising obtaining at least one content element from a content source, each content element comprising text-based formatting instructions and visual content data; translating the text-based formatting instructions of each content element into numerical-based formatting instructions interpretable by the target device; resizing at least a portion of the visual content data based on a capability profile of the target device; and transmitting the numerical-based formatting instructions and visual content data to the target device to enable the content source to be rendered on the target device.

[0010] Another embodiment includes a formatting and conversion system comprising: an interpreter configured to read a content source file to obtain text-based formatting instructions and visual content data for at least one content element; a translator configured to translate the text-based formatting instructions of each content element into numerical-based formatting instructions interpretable by a target device; a data file converter configured to resize at least a portion of the visual content data based on at least a dimension of a display of the target device; and a transmitter configured to transmit the numerical-based formatting instructions and visual content data to the target device.

[0011] Another embodiment includes a system for optimizing rich content for transmission and reproduction on a target device, comprising means for obtaining at least one content element from a content source, each content element comprising text-based formatting instructions and visual content data; means for translating the text-based formatting instructions into numerical-based formatting instructions interpretable by the target device; means for resizing at least a portion of the visual content data based on at least a dimension of a display of the target device, and means for transmitting the numerical-based formatting instructions and visual content data to the target device to enable the content source to be rendered on the target device.

[0012] Another embodiment includes a computer readable medium comprising programming instructions that upon executing cause a machine to obtain at least one content element from a content source, each content element comprising text-based formatting instructions and visual content data; translate the text-based formatting instructions into numerical-based formatting instructions interpretable by the target device; resize at least a portion of the visual content data based on at least a dimension of a display of the target device, and transmit the numerical-based formatting instructions and the visual content data to the target device to enable the content source to be rendered on the target device.

[0013] Another embodiment includes a method, comprising parsing a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document; retrieving the second document; generating a third document integrating the first document and the second document, wherein the generating comprises modifying the at least one tag in the first document to include the referenced second document; and transmitting the generated third document via a wireless network to a target device.

[0014] Another embodiment includes a system, comprising an interpreter configured to parse a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document, and retrieve the second document; a data file converter configured to generate a third document integrating the first document and the second document by modifying the at least one tag in the first document to include the referenced second document; and a transmitter configured to transmit the generated third document via a wireless network to a target device.

[0015] Another embodiment includes a system, comprising means for parsing a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document; means for retrieving the second document; means for generating a third document integrating the first document and the second document, wherein the generating comprises modifying the at least one tag in the first document to include the referenced second document; and means for transmitting the generated third document via a wireless network to a target device.

[0016] Another embodiment includes a computer readable medium comprising programming instructions that upon executing cause a machine to parse a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document; retrieve the second document; generate a third document integrating the first document and the second document, wherein the generating comprises modifying the at least one tag in the first document to include the referenced second document; and transmit the generated third document via a wireless network to a target device.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] Figure 1 is a block diagram of a system according to an embodiment.

[0018] Figure 2 is a flowchart illustrating a method of formatting and compressing rich content in accordance with an embodiment.

[0019] Figure 3 is a flowchart illustrating a method of formatting instruction translation in accordance with an embodiment.

[0020] Figure 4 is a flowchart illustrating a method of data formatting in accordance with an embodiment.

[0021] Figure 5a is a flowchart illustrating steps performed in a first pass of a compilation method to provide an editable version of a converted source file in accordance with an embodiment.

[0022] Figure 5b is a flowchart illustrating steps performed in a second pass of a compilation method to produce a numeric data file optimized for a target device from an editable version of a converted source file in accordance with an embodiment.

[0023] Figure 6 is a diagram illustrating an example of adjusting positions of rich content elements in order to accommodate different target device display screen aspect ratios.

[0024] Figure 7 is a function block diagram of one embodiment of a data file formatter, such as illustrated in Figure 1.

[0025] Figure 8 is a flowchart illustrating a method of resizing visual content data.

DETAILED DESCRIPTION OF THE INVENTION

[0026] Disclosed herein are embodiments of a method and system for reducing a quantity of data used to reproduce rich content formatted for display, such as a web page, on a target device. Reducing the quantity of data may be by translating formatting instructions into a small footprint equivalent instruction format, resizing any visual content data appropriately for display on the target device screen, and/or compressing the translated and resized content data.

[0027] Embodiments of the present invention are configured to work with a method and system for optimizing rich content for transmission and reproduction on the target device by obtaining content element data and formatting instructions for each rich content element and translating the formatting instructions of each rich content element into small footprint equivalent instructions interpretable by the target device. The content element data of each rich content element may be processed by resizing any content element data which is visual content data of the rich content element based on at least one screen dimension of the target device, and compressing the content element data. The translated instructions and processed content element data for each rich content element may then be stored in a data file that will enable the rich content to be appropriately rendered on the target device.

[0028] In some embodiments, formatting and control instructions for rich content are replaced by small footprint numeric format and control instructions readable by the target device, which reduces the total number of bits required for rendering the rich content on the target device. Where necessary, content is also re-sized for a target device screen and further compressed to minimize the total number of data bits to be transferred between the content provider and the mobile device. This can result in up to 90% of the total number of bits for a web page being removed before transmission of the rich content to a target device while still enabling the web page to be rendered in a usable form on the target device.

[0029] Further, resizing content such as images or video for the target device before transmitting the content to the target device reduces the amount of processing required of the mobile device to render the content for viewing on the device screen once the data is

downloaded as any content is already optimized for display on the mobile device. The resizing also allows resizing for large screen TV displays.

[0030] Content throughout this specification refers to information available as data from an electronic source and can include rich content such as available from an internet web site including graphically formatted textual information, images, animations, video and audio data. Content can also include information enabling accessing of further content such as links, URLs (Uniform Resource Locators), or forwarding address information. One web page or one rich content file may include a plurality of rich content elements.

[0031] “Rich content element” refers to a segment of rich content which includes both content data and formatting instructions associated with the content data of the rich content element to define how the content data is rendered to a user. For example a rich content element may be formatted text, a picture or photograph, an advertisement, a sound track, an animation, a video clip etc.

[0032] “Visual content data” refers to any content data which is displayed to a user as still or moving visual images. For example, visual content data includes pictures, drawings, graphs or charts, maps, photographs, video clips, animated images or animations, computer generated graphical moving images, and other such data.

[0033] “Formatting instructions” refers to instructions used to define how data will be rendered to a user. Instructions used may depend on the type of data. For example, where the content data is text, the formatting instructions may define color, font, italics, bold, and underline, among other characteristics. Where the content data is a visual image, formatting instructions may define whether the image is displayed in a border, whether the image is displayed in full color or black and white, or whether the image is animated so it appears as a moving image, for example. Where the data is audio data, formatting instructions can include whether the audio data is to be played simultaneously with the display of a particular image, when an action is performed, or whether the audio data is repeated a number of times, for example. Formatting instructions come in many forms and are often dependent on the editor used to prepare the content.

[0034] “Text based formatting instructions” refers to descriptive tag style instructions such as used in mark up languages, for example HTML (Hypertext Markup

Language), XML, SGML (Standard Generalized Markup Language). “Text based tag form” or “text based instruction tags” are also used in this specification to refer to text-based formatting instructions.

[0035] “Small footprint equivalent instructions” refers to a set of formatting instructions which have a numeric representation, also referred to as a bit level representation, which uses only a small number of bytes or bits for each instruction, such that the number of data bits used to represent these instructions is less than the equivalent instruction represented in a text-based formatting instruction form, such as HTML. The small footprint equivalent instructions can be represented using either an XML representation, for example to represent the instructions in an easily editable form to a user, or the numeric representation which is used in machine-readable files, or both.

[0036] Throughout the specification “target device” is used to refer to a plurality of different types of user devices that may be used to render rich content. For example a target device can be a mobile phone, PDA, personal computer, portable tablet computer, media player, game player or other device such as set top box and TV, or any other device capable of rendering rich content.

[0037] “Render” and “rendering” are used to refer to the reproducing or displaying content. Embodiments describes reproducing and displaying content data as the content provider intended to the content to be perceived by the user on the user’s equipment. For example, rendering the content includes, but is not limited to, reproduction of audio signals as sound output by device speakers, display of visual images on a display screen, and display of animated content as moving images on the display screen.

[0038] Figure 1 is a block diagram of a system 100 according to an embodiment. The system 100 comprises an interpreter 110, an instruction translator 120, and a data file formatter 130. The interpreter 110 determines for each rich content element the position, (e.g. where each content element should be located), context (e.g. what each content element is), and where appropriate, its relationship to other content elements. This information is derived from the original content source file 140 by reading and interpreting the XML or HTML tag commands that are contained within the content source. Rich content elements may additionally be resized, reformatted, and/or quantized for display on the target device

based on information in the target device capability profile 160. The interpreter 110 calls the functions of the instruction translator 120 and data file formatter 130 as appropriate, and provides necessary information to these functions to enable them to optimize the rich content for reproduction on the target device. The interpreter 110 may also output a numeric data file 150 based on the above functions.

[0039] For example, consider a rich content source such as an Internet web page. The page may contain a number of different content elements, such as text, pictures or photographs, video clips, advertisements, animations, data entry fields such as search fields or purchase fields, pull down menus, hot spots and links to other pages embedded in another element. The page will generally also have an overall formatting style within which each element appears in a particular location on the page as defined by the web page designer.

[0040] Rich content, such as is presented on Internet web pages, is typically formatted using text-based formatting instructions or tags known to persons skilled in the art as mark up languages. Some of the most common mark up languages currently being used are HTML, XML, and SGML. Web pages designed for devices such as mobile phones may use languages such as MIDP, WAP, XML and MHTML (Multipurpose Internet Mail Extensions (MIME) HTML). Mark up languages provide instructions regarding how the data is to appear on the page and can also include instructions for providing visual effects such as automatically scrolling text or image slide shows. The advantage of these text-based tag languages is they define what the format is to be rather than how the particular document editor or viewer achieves the formatting effect, which provides for wide editor compatibility and consistency of reproduction for documents prepared using such formatting languages.

[0041] However, a problem with all these mark up languages is that the text-based tags result in a large overall data file size for the web page source file. For example, each tag comprises a number of text characters. Typically, ASCII characters are used for text. Thus each formatting instruction tag will be a number of bytes long.

[0042] Rich content can also be defined using proprietary document formats having proprietary formatting instructions interpretable by a proprietary editor or viewer, such as used by Microsoft Word™ or Microsoft PowerPoint™. Proprietary formats use an instruction set generally developed and defined by the document editor, and these instruction

sets often change between editor versions. Documents developed using one editor can generally not be opened in another editor. Thus a target device may require a compatible document editor or viewer installed to read the proprietary instruction set to render the data for the user. Alternatively, the target device may require a document converter and viewer. Having to maintain multiple document editors, viewers and converters is cumbersome to a user and also the conversion of files from one format to another on the target device can cause delays or increased power consumption, which is particularly undesirable on a device such as a mobile device which typically has limited memory and processing capacity and runs on battery power.

[0043] To overcome such problems, proprietary document editors often include file format conversion functions to convert the file to an alternative format. For example, if the document is to be posted to a web site it will often be converted to a widely used text-based formatting instruction tag format such as HTML. However, other formats such as portable document format (PDF) or common document editor formats, such as Microsoft Word™ format are commonly used. Where a document has been saved in an editable format, such as Microsoft Word™ or PowerPoint™, some embodiments are further configured to convert the document format to a text-based formatting instruction format such as HTML for input to the optimization method as described herein.

[0044] A tool commonly use by web designers is to define a web page having an overall format, say defining a number of frames in which data can be displayed, and then link to data files to be display within each frame, resulting in a number of different data files being opened when displaying one web page. This has the advantageous effect that the content of a data file being linked to can be modified or the link changed to another data file without having to update the entire web page. Often the data files can be stored on a different source server to the web page server. Thus, a large quantity of data is transmitted when this source file and the data from linked files is accessed by a target device, such as a user's personal computer (PC), to reproduce the web page on the target device. The build method for reproducing the web page on the target device may require the target device to link to several different source files or servers to retrieve all the content data. This additional processing to obtain data from a number of files and servers is undesirable where the target

device is a mobile device and the user may pay for the amount of data transferred and also the length of time the device is connected to the wireless network to download the data.

[0045] One embodiment, described in more detail below, uses a tag language where each formatting instruction tag is translatable into a small footprint equivalent instruction tag having a numeric representation of only a few bytes or bits to define the instruction tags and associated tag attributes. This minimizes the number of bits used to provide formatting and control instructions for the rich content. A small footprint equivalent instruction tag and attribute set of this embodiment may be as illustrated in Example 1. An application, configured to read the numeric tags as formatting instructions for rendering the content, may be installed in the target device processor. The advantage of this small footprint equivalent instruction tag set is that the instructions are defined in a more data efficient manner using fewer bits, and thus a smaller data footprint, for the equivalent instruction compared to text-based tags. Translating the text-based formatting and control instructions into small footprint numeric equivalent instructions is a first step in an optimization method according to one embodiment of the invention.

[0046] One embodiment, described in more detail below, includes an optimization method for resizing any visual content data appropriately for display on the target device. Any data elements which comprise visual images, such as photographs or videos, may be a size inappropriate for display on the target device screen. For example, the photographs from an internet web page designed for display on a PC may be too large for display on the screen of a mobile phone, whereas a short video taken using a mobile phone camera may be too small to be displayed appropriately on the full screen of a television without significant image processing to enlarge and sharpen the image. The display capability of the target device is taken into consideration and any visual content data is resized appropriately for the target device before delivery of the rich content data to the target device.

[0047] In one embodiment, any visual content data is resized in order to optimize the visual content data for rendering on the target device based on the target device capability profile 160. The target device capability profile 160 can include information such as the display dimensions for the target device, which can be provided as a physical measurement or

as a number of pixels. The target device capability profile 160 can also include display resolution information which can include pixel size, number of colors able to be reproduced and luminosity ranges for each pixel. Other information which may be included in the target device capability profile 160 can include maximum buffer sizes and processor capacity, which may influence the output file size for visual content which can be handled for the target device or a data rate for streaming moving image visual content data to the target device.

[0048] In one embodiment each translated, resized and/or compressed rich content element of the content data is consolidated into a single data file for transmission and display on a target device. Preferably this data file is prepared such that the processor of the target device processes each instruction consecutively, for example reading the data file from start to finish and acting on each instruction as it is encountered in order to render the rich content for reproduction on the target device in a single processing pass. Thus, the rendering of the content on the target device is efficient, without requiring jumping between different addresses in the data file or opening other content data files. An advantage of this embodiment is that the processing time and power consumed by the target device is minimized by providing the content data file in an optimized form for reproduction on the target device. Further, the optimized data file minimizes the amount of data that is discarded during the rendering process on the target device. In one embodiment web or e-mail content may be contained in a first file and an advertisement may also be contained in the same file, appended to the web or e-mail content. In one embodiment the advertising content will be rendered from the file along with the web or e-mail content. In another embodiment the web or e-mail content comprises a first file and an advertisement that is associated with the web or e-mail content comprises a second file that is also loaded and displayed as part of the same function to display content, such that the content of both the first and second files is rendered for concurrent display on the device display screen.

[0049] When a web page is compiled, say to HTML source code, the overall formatting style and layout instructions to be used for the various content elements may be defined in an initialization or header portion of the source data file. The rich content source code 140 and target device capability profile 160 regarding the target device, such as the

target device screen size (e.g., in pixels) and resolution (e.g., pixels/inch), are obtained and input to the interpreter 110. The target device capability profile 160 may be transmitted by the target device directly to the interpreter or may be obtained by the interpreter in some other manner (e.g., from a database containing user profile information provided at a time of registration).

[0050] The interpreter 110 reads the rich content source code, as is illustrated at block 210 in Figure 2, and for each rich content element determines the content element context 215, which includes what the content element is and any defined relationship between the content element and other content elements, for example if two search fields should be displayed together. The interpreter 110 also determines the content element's position 220 within the web page layout, for example whether the content element is to appear at the top of the page and if necessary its location relative to another element for display, such as two related search fields to be displayed one above the other. Using this information the interpreter 110 determines the appropriate position 225 for the content element on the target device screen, based on the position within the web page layout.

[0051] The interpreter 110 passes the position information along with the formatting instructions for the content element to the tag translator 120. The translator translates 230 any formatting or instruction tags (HTML tags) along with the position information into equivalent small footprint instructions, in some embodiments of the system the translator converts the formatting and instruction tags directly into a numeric format.

[0052] Where the rich content element is a data file or link to a data file the interpreter calls the data file formatter 240 to appropriately re-format the data, which is then added 250 into the converted file with the translated tags for the content element and the interpreter continues to convert the next rich content element 260. The method is then repeated for the next rich content element 215 until the end of source file is reached 265. The rich content file is now in the form of a numeric data file 150. The numeric data file 150 is then output to the target device or stored 270 ready for download.

[0053] The functions performed by the formatter depend on the data file type, for example the formatter may perform any one or more of the following steps: resizing image content for the target device screen dimensions, fetching content data from a linked data file,

conversion of data file formats, or data file compression. The actions taken by the formatter are dependent on the type of data for the rich content element.

[0054] An example of the actions taken by the formatter 130 and translator 120 will be described in further detail with reference to an example of a weather update being formatted for delivery to a mobile phone. The weather update may be accessible through an internet web page. The mobile phone user may have the URL for the weather update stored on their mobile phone when the user subscribes to the weather update service. The user may also provide profile details about their mobile phone, such as the model type, screen size and resolution, to the weather update service when they subscribe. An application running on the web site server includes an embodiment of the interpreter, formatter and translator of the system. The weather update is prepared as part of the weather web site content in a HTML format. The posting of the new weather update in this embodiment triggers the application in the web server to prepare a copy of the content for downloading by the weather update service subscribers. The interpreter reads the HTML source for the web site and from within this source identifies the subscribed content elements, for example the “current conditions” and “forecast”. The interpreter identifies the two rich content elements as comprising formatted text data and also determines that maintaining of the position of these from the web site, for example at the top of a right side bar of the web page, is not necessary for the reproduction on the target device. This lack of a location restriction is based on the information from the user subscriptions and can be as simple as a flag indicating that only two content items are subscribed to and therefore the reproduction on the target device is independent of the web page screen location.

[0055] The interpreter reads 210 the source code for the web site and identifies the first element of the subscribed content 215, the content in the weather update frame of the web site. The interpreter recognizes the position in the web page layout as irrelevant in this instance and therefore indicates, as illustrated at block 310 in Figure 3, that the first element, the heading “Conditions for Sunnyvale,” will be drawn at the top of the screen. The interpreter also provides information regarding the general web page style formatting 315 such as color schemes for the tag translator to use for color formatting if necessary. For example, the tag translator uses this information to define the background color for the

weather screen. In this embodiment the tag translator determines 320 the equivalent small footprint instruction tags and attributes for the HTML tags using a look up table 325 of small footprint equivalent formatting instructions to determine the tag equivalents; the lookup table can include interpretation functions to recognize abbreviations or unusual HTML tags. Once the equivalent small footprint instruction tag and attributes have been determined the translator determines attribute values to be applied 330 for example, text color and size codes. The translated tags and attributes are then stored in the converted data file 340. This can be the numeric representation of the small footprint formatting instruction tags and attributes.

[0056] Where the data is plain text data, this may simply be inserted into the converted data file with the translated tags. However, where the data is an image file or a link to a data file, the data file formatter may be called to fetch the data file and determine whether any further formatting is necessary.

[0057] As illustrated in Figure 4, the input element data 410, for example the data file location or link to the data file, is input to the formatter. The formatter determines the data type 420. Where the data is a link to a data file stored on another source 430 the data file is fetched from that source and the type of data file is checked. If the data file is plain text data, determined at block 440, the text is optionally compressed 460 and inserted in line with the formatting and instruction tags 470.

[0058] Where the file type is visual content data, for example an image file or video file, the formatter checks 422 whether the visual content data is sized for the display of the target device, for example based on the image dimensions and the size of the target device screen. Optionally, the formatter can also take the aspect ratio of the image into consideration. In an embodiment where the target device is a mobile phone, the formatter only considers one dimension, the display screen width for determining whether the image is sized for display on the target device, this is because most mobile phone users are comfortable with scrolling content up and down to view it on the device screen. The formatter is provided with the pixel count 415 of the width of the target device display screen, the pixel count being the maximum number of pixels in one line across the display screen. The formatter determines based on the pixel count for the given dimension whether

the image is correctly sized for display on the target device. For example, if the pixel counts do not match, then the image will be resized either up or down to match the pixel count of the target device 450. An embodiment of a method for resizing is described below in reference to Figure 8. Before performing the resizing the formatter checks 425 whether it is necessary to first decode or decompress 455 the image from the format used in the image file, for example JPEG or MPG4, to a pixel image. If the visual content data file is in a compressed or encoded state the visual content data can be reconstructed 455 to a pixel image. The resizing to the target pixel count 450 is then performed. The resizing 450 maintains the aspect ratio of the original image. The resized image may then be compressed or encoded 458 using the original or another compression or encoding format. The image data may be either compressed or encoded 458 using a first process and then further compressed 460 using another process 460 and inserted in line in the output data file 470. The resizing function may also reformat the image and enhance the image for a screen larger than the original target, as will be described in more detail below. For example, in the case of a JPEG image designed for an 800 x 640 screen where the target device display is a 2048 HDTV screen, the image may be reformatted to enable additional data to be added to enable the aspect ratio to be adjusted and the image to be enhanced for this larger screen size.

[0059] Using embodiments, rich content can be converted directly to a reformatted and compressed numeric file for rendering on a target device. However some embodiments also provide a content editor which enables a source file to undergo formatting instruction translation conversion to convert the source document style formatting and instructions into an XML representation of the small footprint formatting and instruction tags, enabling a user to edit the translated file. Alternatively a file can be generated using the XML object representation of the small footprint instructions.

[0060] In this embodiment in a first conversion pass to produce an editable version of the source code with formatting instructions converted to XML objects, for example as shown in Example 1, the interpreter calls the translator to convert the formatting instructions only, and does not reformat any content data for a target device. Indeed for such conversion to or generation of an editable file format using an XML representation of the numeric instruction tags the target device may not be known. For example for rich content

which is targeted to mobile phone users, such as advertisements, it may be desirable to prepare the rich content directly using the XML object version of the small footprint instructions to minimize conversion time.

[0061] For example, as illustrated in Figure 5a, on the first pass the source file is read 500 and for each element the context is determined 510 and the relative position 515 for the element on the target device screen determined. The formatting and instruction tags are then translated 520 to the equivalent small footprint tags and represented as XML objects. Optionally any data from external sources, such as file servers or other web servers, is fetched 525 and visual content data files may optionally be decompressed or decoded to a pixel image format ready for resizing, if necessary. Any fetched data files or images can be appended as data files 530 linked in the XML object for that element. The next element 535 is then read and the method continues until the whole of the input source is converted.

[0062] An example of a section of the translated content source file in XML is give below:

Weather Alert

```
<gui color="255 255 255">
    <txt face="2" style="1" color="10 10 120">Conditions for Sunnyvale, CA at 8:56 am
    PST</txt>
    <n />
    
    <txt face="2" style="1" color="10 10 120">Current Conditions:</txt>
    <txt face="1" style="1" color="10 10 120">Fair, 17 C</txt>
    <txt face="2" style="1" color="10 10 120">Forecast:</txt>
    <txt face="1" style="1" color="10 10 120">Mon - Partly Cloudy. High: 21 Low: 12</txt>
    <txt face="1" style="1" color="10 10 120">Tue - Partly Cloudy. High: 20 Low: 12</txt>
</gui>
```

[0063] Once the target device display dimensions are specified a second conversion pass can be performed, to convert the XML objects to numeric instruction format and also resize, convert and compress any data files.

[0064] An example of the second pass is illustrated in Figure 5b. Each XML object 540 for each element is read and the tags converted to numeric format 545 and the data associated with the element formatted for the target device 550, this can include the resizing

of image content and compression of the data, and the element data placed in line with the numeric element data tags 555. The next element 560 is then similarly read and processed until the whole file is converted. The converted file is then stored or output to the target device 570.

[0065] As illustrated in Figure 6, when compiling the rich content source to numeric format, options may be provided to enable the position of content elements to be adjusted suitably for the target device. For example, menu 620 displayed as a bottom banner for a web page 610 may remain a bottom banner 622 where the target device is a mobile phone 630 but its position may be altered to become a side banner 625 where the target device is a wide screen digital TV 650 due to the different aspect ratios of the mobile phone 630 display 640 compared to the TV 650 display screen 660 and scrolling capabilities of the displays for the two target devices. Options may enable the interpreter to automatically identify one or more content elements to for which the relative position may be adjusted to suit the aspect ratio of the target device display. For example, the position adjustment may be based on element type, such as advertisement or menu, or based on element size and relationship between content elements. Alternatively, any adjustable content elements and their position options may be indicated by the content designer.

[0066] A content editor may be provided with a compiler and pre-viewer to enable the designer to enter information about the target device, such as a device profile or screen dimensions, for the rich content to be re-formatted and then rendered in the pre-viewer to demonstrate how the content will be rendered on the given target device. Such a pre-viewer may be provided as a formatter application, say to enable a web site designer to preview how the web site HTML source file will be rendered on a target device, however, all editing is done on the original HTML source. Alternatively, the pre-viewer may be included in a suite of editing or development tools which enable the designer to edit the content.

[0067] Where visual content data is to be resized, the resizing may be accomplished by reading the target device capability profile 160, such as illustrated in Figure 1, and determining a target rendering size for the visual content data based on the target device capability profile 160. It is then determined whether an original rendering size of the visual content data is equal to the target rendering size. Where the target rendering size is

smaller than the original rendering size, the visual content data is resized to reduce the rendering size for the visual content data to the target rendering size. Thus the visual content data is configured for display on the target device. The processing time and power for rendering the visual content data on the target device is minimized as no manipulation of the size of the image is required during rendering on the target device. Further, the visual content data can be optimized for reproduction on the target device to provide the best possible reproduction of the visual images possible within the limitations of the target device display. Such limitations may include display size, resolution, number of colors, and luminosity, for example.

[0068] The resizing of the visual content can also be formatted to fit into a scalable size that can be easily resized as a secondary step on the target device. An embodiment of this method is to format a image for display on a 320 pixel display device screen; this image can be used to display on a 320 pixel displayed device and by removing half of the data the same image can be also resized for a 160 pixel or half size screen providing that the image is formatted so that the image can quickly be divided for a one half 160 pixel or one quarter 80 pixel display screen size.

[0069] By grouping different target device screen sizes, of which there are over eighty different sizes, into groups of pixel display formats, a common resizing format can be used to reformat the visual data into a format that can be resized to allow for simple display on the target devices.

[0070] In one embodiment, resizing the visual content data is based on at least one of the height, width or aspect ratio dimensions of the target device screen, given in pixels. The target device has a screen having a fixed height and width and the resolution capability of the screen can be defined by the pixel density. Thus, a dimension of the screen can also be measured in number of pixels. In this embodiment, an image or video file in pixel format is resized in accordance with a given pixel count for one dimension while maintaining the aspect ratio, the ratio of height to width, of the image. The resizing of the image may either reduce or increase the number of pixels in the image depending on the target device. The resized pixel image can then be compressed using a compression process suitable for the target device to reduce the data file size. The advantage of resizing the image

prior to compression is that the image will be appropriately sized for the target device once it is decompressed, thus reducing the amount of processing performed by the target device to appropriately display the image.

[0071] The compression of the image or video may also be based on the target device capability profile 160 which can include device-specific decoders that may be embedded in the device such as a MPEG or JPG hardware decoder or a Real Player application that was designed and shipped as part of the device.

[0072] Prior to the resizing step, the visual content data may be converted from an encoded or compressed format to a pixel format. For example, for a video file stored in MPG4 format, the image reconstructed from the MPG4 format file will have a given pixel count, generally that of the original image before the MPG4 compression. Thus, an image directly reconstructed from the MPG4 file and displayed on the target device may not be correctly sized for the device screen, or may require further processing in the target device to resize the image before display. This further processing may cause a delay or increased power consumption in the target device. An example is that MPG4 is normally encoded as 1920 pixels per line displayed on the screen where the largest cellular phone may only have 640 pixels per line. This requires that the cell phone download three times the amount of information per line than it requires to display the video image.

[0073] In other embodiments, the number of bits per pixel may also be converted during this method as most mobile devices may not have advanced image display processing and with a battery powered device the displays may not be capable of displaying a high resolution or rich color that the original image or video may be encoded with.

[0074] In an embodiment where a content element includes an encoded or compressed visual content data file, the visual content data is first reconstructed from the compressed or encoded format to a pixel image format. The pixel image is then resized appropriately for a given pixel count or bit encoding for at least one image dimension. The resized image is then re-encoded or re-compressed using either the same or another encoding or compression scheme. This encoding can also be a mathematical conversion directly from the existing format to the same or a different format including a reduction in the pixel and bit

count or image display size. This process is a common conversion process and is known to someone skilled in the art of image and video data conversion and compression.

[0075] Another embodiment is that the image and video conversion is not from a known format to another known format, but from a known format to a profile device specific format so that the pixels do not need to be reformatted by the target device to be displayed. Standard compression known to the target device can be used after the pixel and bit conversion has been completed.

[0076] In one embodiment, this visual content data resizing and compression is performed by the data file formatter 130, illustrated in Figure 1. An example of an embodiment of the data file formatter 130 is illustrated in Figure 7, where visual content data 780 and the target device capability profile 160 are provided to the data file formatter 130 by the interpreter 110, illustrated in Figure 1. The data file formatter 130 includes a controller 710 and a resizer 720. The controller 710 is configured to determine the size of the visual images reproduced from the visual content data 780 if it is rendered directly from the visual content data without applying any resizing or other processing functions. This is referred to as the original size of the visual content data 780. The controller 710 may also be configured to determine from a target device capability profile 160 a target rendering size for an image to be displayed on the target device display.

[0077] An example of the target device capability profile 160 can include the height and width of the display, and the target rendering size may be based on one or both of these dimensions and/or the aspect ratio of the original size of the visual content data. Where the target rendering size is based on one dimension, this may be the display width for a target device such as a mobile phone, such that if the resulting image height is greater than the height of the display the user can use display controls to scroll vertically to view the entire image. Where both the height and width dimensions of the target device display are taken into consideration, the controller determines the maximum size to enable the visual content data to be rendered on the display while maintaining the aspect ratio of the original rendering size. For example, for a video clip or moving image it may be undesirable to scroll to view the rendered visual content in its entirety. The controller may be configured to determine the

target rendering size using different criteria based on whether the visual content is moving or static image data.

[0078] In another example, the data formatting resizer may use the profile screen size to change the dimension of the image or video on the screen and not require that the screen be scrolled to see the complete image or video.

[0079] The resizer 720 is configured to resize the visual content data 780 from the original size to a target rendering size as determined by the controller 710. The controller 710 may provide the original and target rendering sizes to the resizer for use in performing the resize or the controller 710 may determine a reduction ratio, original size to target rendering size, and provide this to the resizer for resizing the image to maintain the aspect ratio of the original image. In some embodiments, the aspect ratio of the image may be allowed to be altered through clipping or stretching of visual images. In embodiments where this is allowed, parameter values for clipping and stretching may also be provided to the resizer 720 by the controller 710.

[0080] In one embodiment, the resizer 720 comprises a first resizer 722 for resizing a large image to an intermediate size and a second resizer 725 for rendering the image to the target rendering size. Performing a two stage size reduction can be more efficient and provide better rendered image quality than performing a single resize for large images. For example, where the height or width of the original rendering size of an image is greater than one thousand pixels, performing a first resize to reduce the image to an intermediate size and a second resize to the target size may require less processing instructions to be executed than a direct resize from the original to the target size, and hence require less time and processing resources than a direct resize. Further, the number and magnitude of errors in the image data, which need to be compensated for or corrected when images are rendered, will generally increase with increasing differences between the original and target image sizes, the extent of which will depend on the resizing process used. Performing a first resize to an intermediate size and a second resize to a target size can reduce the total number and magnitude of errors in the image data and hence improve the rendered image quality compared to that of an image resized directly from the original size to the target size.

[0081] The first resizer 722 and second resizer 725 may use the same or different resizing processes. The intermediate size may be a designated size or a size determined based on the original and target rendering sizes. The intermediate size may be determined by the resizer 720 or the controller 710. In an exemplary embodiment, the resizer 720 assesses whether to use a two-stage or one-stage resize based on the original rendering size and a designated threshold size value, for example one thousand pixels. The intermediate size for the first resizer 722 may be a specified size or a size determined based on the original image size and the target rendering size. For example, the intermediate size may be determined using the process A below:

Process A

If ($OW > TW$ or $OH > TH$)

 If ($TW/OW \leq TH/OH$)

 Then

$CR = TW/OW$

 Else

$CR = TH/OH$

$IW = CR * OW$

$IH = CR * OH$

where OW is the original image width, OH is the original image height, TW is the target width, TH is the target height, IW is the intermediate width, IH is the intermediate height, and CR is the conversion ratio. As can be seen from process A, in this embodiment, the first resize only occurs if one of the width or height of the original image is greater than the target image size. The conversion ratio used for the image size reduction is based on the relative widths or heights of the original and target image sizes.

[0082] An example of an process employed by the second resizer is given below as Process B. For example the conversion ratio for the resize from the intermediate size, or

original size where the first resize was not required, may be performed using the process B below:

Process B

If (OW>TW)

 If (OW-TW> OH-TH)

 Then

$$CR = TW/OW$$

 Else

$$CR = TH/OW$$

where OW is the original image width, OH is the original image height, TW is the target width, TH is the target height, and CR is the conversion ratio. It should be appreciated that where a first resize has been performed OW and OH in Process B should be substituted respectively with IW, the intermediate width, and IH, the intermediate height. The effect of resizing a large image using process A prior to resizing using process B is a substantial reduction in the processing required by the second resizer 725 compared to performing a direct reduction from the original size to the target size using process B.

[0083] Optionally the data file formatter 130 also includes a quantizer 730. Output from the resizer 720 is a reduced size or “thumbnail” version of the original image. The pixels of the thumbnail version of the image can optionally be quantized based on the display capabilities of the target device to optimize the image data for rendering on the target device display. For example, the device display capabilities included in the target device capability profile 160 may include data relating to display limitations such as the pixel size and display resolution, bit count encoding for number of colors able to be reproduced on the display, ranges and increments of luminosity, or relative brightness levels for pixels. Based on this data from the target device capability profile 160, pixel attribute criteria for rendering images on the target device can be determined. Pixel attributes of the resized image are compared with the selected pixel attribute criteria to determine whether the resized image may be rendered on the device display. The resized image can be quantized to comply with

the selected target device pixel attribute criteria, if necessary, by the quantizer 730. For example, where the color attributes of the resized image pixels has a color range of 256 colors and the selected pixel color attribute criteria based on the target device capability profile 160 is a range of 64 colors, the quantizer 730 can quantize the image data to reduce the number of colors. Quantization parameters can be calculated based on differences between the selected pixel attribute criteria and the resized image pixel attributes; the calculation can be performed by the controller 710 and input to the quantizer or by the quantizer 730 depending on the embodiment implemented.

[0084] In another embodiment where the image resize is based on a size measured in millimeters or a percentage size reduction rather than a target pixel count for the resized image, it may be necessary to reduce the number of pixels for the resized image to optimize the image for display on the target device. For example, the resized image may be quantized to reduce the number of pixels based on the target display size and pixel size, which can also be referred to as display resolution. Those of skill in the art will appreciate that for image data optimized for display on the target device the number of pixels defining the image can be defined to match the number of pixels of the device display for at least one dimension, such as the width or height, and that the pixels can be defined to meet the pixel attribute criteria of the device display, thus minimizing the amount of processing required by the target device to render the visual content.

[0085] The target device capability profile 160 may also include data such as data file size limitations or buffer size and processing speed which may limit the ability of the target device to render visual content data. Based on data from the target device capability profile 160, it can be determined whether quantization is required to reduce the output image file size or to reduce the quantity of image data to in turn reduce the required bandwidth, buffer size or processing speed required for streaming visual content to the target device and rendering the visual image content by the target device. Quantization parameter values can be calculated based on the target file size or data rate to reduce the amount of data required for rendering the image, for example by reducing the number of colors or the image resolution. Optionally the target file size or data rate limitations may also be catered for by formatting and compressing the image data.

[0086] Quantization may cause a resizing of the image data, for example through color reduction causing a reduction in resolution and, in turn, a pixel count reduction. This may mean a further resize is required to adjust the image size to the target image size. In this instance the resizer 720 performs a further resize, for example based on the quantized thumbnail size and the target image size using process B as described above.

[0087] Embodiments of the data file formatter can also include a formatter 740 configured to encode and/or format the image data into a data format compatible with the target device. For example, a target device may only be capable of rendering moving image data provided in MPEG-4 format or still images in a JPEG format. Such target device file format compatibility data may be included in the target device capability profile 160. The data file formatter 740 is configured to format the thumbnail image data in accordance with a target file format selected based on the target device capability profile 160. The target device capability profile 160 may include preference data for the compatible file types to be used when selecting the target file format. For example, each compatible file format may be provided with a preference ranking, or a single most preferred or default format may be nominated by the target device user or manufacturer. Preference rankings may also be based on the data type, for example moving or still images. The target format selection may be performed by the formatter or the controller depending on the embodiment implemented. Alternatively a target format may be specified by a content provider. A content provider may also nominate a default data format to be used where no data format limitations are specified in the target device capability profile 160, or nominate a preferred format to be selected out of possible format options for a target device, for example preference for JPEG format for images where a target device is capable of rendering TIFF, JPEG, bitmap or PDF formats.

[0088] The data file formatter 130 may further comprise a compressor 750 for optional compression of image data. Whether or not data is compressed may depend on the target data file format. Further compression of the visual content data can be used to reduce the output data file size which, in turn, can reduce the time and bandwidth required to download the content data.

[0089] The formatter 740 may also be configured to segment image data into a plurality of linked files based on the target device capability profile 160 or content provider

preferences. For example, for moving image data the content provider may specify whether the resized data is to be streamed to a target device or delivered as a series of individually downloadable files, also referred to as "chunks." Alternatively whether to format the data as a series of chunks or as a single data file for streaming to the target device may be determined on a file by file basis. For example, whether to chunk a data file may be based on the target device capability profile 160 and target device limitations such as memory and/or buffer size or processor capacity, data transfer rates of the communication interface or loads on the network interface based on QOS (Quality Of Service) performance or user preferences included in the target device capability profile 160 such as a preference for chunking or streaming. Alternatively whether to chunk or stream a data file may be determined based on environmental factors such as network latency or available bandwidth. Where data is to be chunked for delivery to a target device the resized visual image data is divided into a plurality of sequential segments. Each segment is packaged into a separate data file along with link data to identify the next data file in the sequence, for example a file address or pointer to the data file on the content data server. The link data enables a target device to download the plurality of data files in sequential order. Downloading data files in a series of segments, rather than as a single data stream, may also be desirable where network security or network connection time is a concern. For example, using a mobile communication network a user may be charged for time on line rather than the amount of data transferred. In this scenario, the target device may be the user's mobile phone and the visual content data a music video. Streaming of data requires a network communication session to be opened for the entire time the content is being downloaded to the mobile phone, which may be a few seconds to a few minutes for the entire video depending on the data download speed and processing capacity of the target device which may be quite costly for the user if they are being charged by the second. Where the data is chunked, the target device can be configured to disconnect from the network while processing and rendering each section of data and reconnect to download the next section as required. From the user's point of view this disconnection and reconnection may be transparent so that it appears that the music video is being downloaded and played in a single action. Chunking of data files is also useful to enable downloading of content such as video clips to target devices, such as some mobile phone models, which are

not configured for streaming of content via the Internet. Chunking of data for downloading to a user device may also be performed independent of the steps of resizing, quantizing and formatting the visual image data. For example a data file may be chunked for downloading to a user device at the time download of the data is requested based on the device profile, content provider preference, or network factors.

[0090] The controller 710, resizer 720, quantizer 730, formatter 740 and compressor 750 of the data file formatter 130 may be implemented as software applications executable on computer processor hardware. Alternatively embodiments of the data file formatter may be implemented using any suitable software, hardware, firmware, or combination thereof.

[0091] An example of a resizing method is illustrated in Figure 8. Such resizing method may be used alone or in combination with the method illustrated in Figure 4, for example, such as by using the method of Figure 8 to effect the resize 450. When visual content data 780 is input to the data file formatter 130 the target device capability profile 160 is read 805. The original rendering size of the visual content data is determined 810. The image data file may be decompressed and decoded from a compressed and encoded format to a pixel format before or as part of the step to determine the original rendering size for the visual content. A target rendering size is also determined 815 based on display data provided in the target device capability profile 160. Whether a two stage or a one stage resize is performed in this embodiment depend on whether the image is determined to be a large image 820, based on defined threshold criteria. Whether to perform a first resize to an intermediate image size may be based on one or both the width and height dimensions exceeding the threshold value. The threshold value may be a given length in units such as centimeters or inches, or designated as a height or width pixel count. Alternatively, the threshold value may be a total image pixel count or other suitable measure for identifying a large image. The criteria for identifying a large image may be based on the image pixel density or resolution indicating a large image file size rather than simply on an absolute rendering size for the image. If the original rendering size of the image is larger than a threshold value 820 a first resize 822 to an intermediate size may be performed. If the original image size is below the threshold value then the first resize 822 is skipped and the

method proceeds directly to the second resize 825 to resize the image to the target size. In some embodiments resizing of the image data may not be required, for example if the original and target image sizes match. In this instance optimization of the image data by quantization, encoding/formatting and compression may still be performed by the data file formatter 130.

[0092] Whether or not quantization of the image data is required 830 is determined based on pixel attributes of the image data and selected pixel attribute criteria based on data from the target device capability profile 160. Quantization parameter values are determined based on the pixel attribute criteria for use in quantizing 835 the image data. The quantization step may be skipped if the image data meets the selected pixel attribute criteria.

[0093] The image data is formatted 840 into a target file format. The target file format may be selected based on data provided in the target device capability profile 160 or based on preferences specified to the data file formatter by an operator or content provider. Formatting the image data for output 840 may also include encoding of the image data using a selected encoding scheme or segmenting the data into a sequence of data files for delivery to the target device. Optionally the image data is compressed 855 before being output 860.

[0094] While the above detailed description has shown, described, and pointed out novel features of the invention as applied to various embodiments, it will be understood that various omissions, substitutions, and changes in the form and details of the devices or methods illustrated may be made by those skilled in the art without departing from the spirit of the invention. All changes which come within the meaning and range of equivalency of the appended claims are to be embraced within their scope.

[0095] In the claims which follow and in the preceding description, except where the context requires otherwise due to express language or necessary implication, the word "comprise" or variations such as "comprises" or "comprising" is used in an inclusive sense, i.e. to specify the presence of the stated features but not to preclude the presence or addition of further features in various embodiments of the invention.

Example 1

TAGS:

TAG NAME	Numeric Tag Name	Description	XML Representation
Ads	0x00	<i>advertisement.</i> specifies an advertisement content and includes a pointer to its location	<ads pos="" delay=""> <frame src="" link=""/> </ads>
Anm	0x10	<i>animation.</i> specifies and points to the location of an animated image	<anm pos="" delay=""> <frame src="" /> </anm>
Btn	0x20	<i>button.</i> defines a "push button"	<btn type="" img="" label="" link=""> <ref id=""/> </btn>
Frame	0x30	<i>frame.</i> defines the frame of an element	<frame src="" link="" />
Grp	0x40	<i>group.</i> specifies the spatial rendering of a pair of elements. There are four possible elements but only two are allowed for each specified group	<grp color=""> <txt color="" face="" link="" style="" value=""/> <ads pos="" delay=""> <frame src="" link=""/> </ads> <anm pos="" delay=""> <frame src=""/> </anm> </grp>
Gui	0x50	<i>gui.</i> defines the container or field where the container where different components are added	<gui icon="" color="" label="" type="" to="" from=""/>
Img	0x60	<i>image.</i> specifies and points to the location of an image file	
N	0x70	<i>line.</i> indicates a new line	<n/>
S	0x80	<i>separator.</i> defines a horizontal border	<s width="" color=""/>
Ref	0x90	<i>reference.</i> specifies id values	<ref id="" />
Snd	0xA0	<i>audio.</i> specifies and points to the location of a sound file	<snd format="" loop="" src=""/>
Txt	0xB0	<i>text.</i> specifies text characteristics	<txt color="" face="" link="" style="" value=""/>
Txtbox	0xC0	<i>text box.</i> defines an input field where data can be entered	<txtbox label="" id="" constraint="" value="" maxsize=""/>
Vid	0xD0	<i>video.</i> specifies and points to the location of a video file	<vid src="" loop="" url=""/>

ATTRIBUTES:**Tag = ads**

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Delay	0x00	0 - 65535 (2bytes)	value in milliseconds
pos	0x01	left (default)	0x00
		center	0x01
		right	0x02

Child Tag	Numeric Equivalent
Frame	0x30

Child Tag Attribute	Numeric Equivalent	Possible Values	Numeric Equivalent
Link	0x00	url	
Src	NA	url	

TAG = ann

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Delay	0x00	0 - 65535 (2bytes)	value in milliseconds
pos	0x10	left (default)	0x00
		center	0x01
		right	0x02

Child Tag	Numeric Equivalent
Frame	0x30

Child Tag Attribute	Numeric Equivalent	Possible Values	Numeric Equivalent
---------------------	--------------------	-----------------	--------------------

Src	NA	url	
-----	----	-----	--

TAG = btn

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Img	0x00	url	
Label	0x10	string	
Link	0x20	url	
type	0x30	get	0x00
		post	0x01

Child Tag	Numeric Equivalent
Ref	0x90

Child Tag Attribute	Numeric Equivalent	Possible Values	Numeric Equivalent
Id	0x00		

TAG = grp

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Color	0x00	3bytes + extra 5b) RGB	

Child Tag	Numeric Equivalent
Ads	0x00
Anm	0x10
Img	0x60
Txt	0xB0

TAG = gui

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Color	0x00	3bytes+ extra 5 bytesRGB	
Icon	0x10	url	
From	0x20	text	
Label	0x30	text	
To	0x40	text	
Type	0x50	url	

TAG = img

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Link	0x00	url	
pos	0x10	left (default)	0x00
		center	0x01
		right	0x02
Src	NA	url	

TAG = s

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Color	0x00	3bytes+ extra 5 bytesRGB	
Width	0x10	% of screen width	

TAG = snd

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
format	0x00	amr	0x00
		midi	0x01
		mmf	0x02
		mp3	0x03
		ott	0x04
		wav	0x05
Loop	0x10	byte(0-255)	
Src	NA	url	

TAG = txt

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
Color	0x00	3bytes+ extra 5 bytesRGB	
face	0x10	proportional small	0x00
		proportional medium	0x01
		proportional large	0x02
		system small	0x03
		system medium	0x04
		system large	0x05
		monospace small	0x06
		monospace medium	0x07
		monospace large	0x08
link	0x20	url	
style	0x30	plain	0x00
		bold	0x01
		plain italic	0x02
		plain underlined	0x03
		plain italic underlined	0x04

		bold italic	0x05
		bold underlined	0x06
		bold italic underlined	0x07
value	0x40		

TAG = textbox

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
constraint	0x00	decimal	0x00
		emailaddr	0x01
		numeric	0x02
		password	0x03
		phone number	0x04
id	0x10		
label	0x20	text	
maxsize	0x30	numeric	
value	0x40		

TAG = vid

ATTRIBUTE NAME	Numeric Attribute Name	Possible Values	Numeric Equivalent
src	NA	url	
loop	0x00	byte(0-255)	
url	0x10	url	

WHAT IS CLAIMED IS:

1. A method of optimizing content for reproduction on a target device, comprising:
 - obtaining at least one content element from a content source, each content element comprising text-based formatting instructions and visual content data;
 - translating the text-based formatting instructions of each content element into numerical-based formatting instructions interpretable by the target device;
 - resizing at least a portion of the visual content data based on a capability profile of the target device; and
 - transmitting the numerical-based formatting instructions and visual content data to the target device to enable the content source to be rendered on the target device.
2. A method as claimed in claim 1, further comprising determining a relative location for rendering the at least one content element on a display of the target device.
3. A method as claimed in claim 2, wherein the translating comprises defining a position of each content element on the display in the numerical-based formatting instructions.
4. A method as claimed in claim 3, wherein the defined position for each content element is defined as a relative position.
5. A method as claimed in claim 1, further comprising acquiring at least one of a horizontal or a vertical dimension of a screen of the target device before the resizing.
6. A method as claimed in claim 5, wherein the screen dimension is measured in pixels.
7. A method as claimed in claim 6, wherein the resizing comprises converting the portion of visual content data from a compressed format into a non-compressed or non-encoded format, and reducing a pixel count of the non-compressed or non-encoded format in accordance with at least one of the horizontal or vertical pixel screen dimension while maintaining an original aspect ratio of the visual content data.

8. A method as claimed in claim 1, wherein the method comprises at least two passes, wherein a first pass comprises translating the text-based formatting instructions into small footprint formatting instructions to provide a content source file in an editable format, and a second pass comprises resizing and compressing the editable format file for the target device.

9. A method as claimed in claim 1, wherein the numerical-based formatting instructions are binary format instructions.

10. A method as claimed in claim 1, wherein the capability profile comprises a pixel count, a height, a width, or an aspect ratio of a display of the target device, and wherein the resizing is before the transmitting.

11. The method of claim 1, wherein the resizing comprises:
determining an original size of the portion of visual content data; and
determining a target rendering size of the portion of visual content data based on the capability profile of the target device, wherein the capability profile comprises at least a height or a width of a display of the target device,
wherein the resizing comprises resizing the portion of visual content data based on the determined target rendering size.

12. The method of claim 11, wherein the height or the width is a number of pixels.

13. The method of claim 11, wherein the resizing the portion of visual content data based on the determined target rendering size comprises resizing the portion of visual content data from the original size directly to the target rendering size.

14. The method of claim 11, wherein the resizing the portion of visual content data based on the determined target rendering size comprises performing an intermediate resizing of the portion of visual content data from the original size to an intermediate size, and then resizing the portion of visual content data from the intermediate size to the target rendering size.

15. The method of claim 14, further comprising determining that a height or width of the original size is less than a height or width of the intermediate size.

16. The method of claim 14, further comprising determining that a ratio of the original size compared with the target rendering size is greater than a predetermined threshold.

17. The method of claim 1, further comprising quantizing the portion of visual content data based on the capability profile, wherein the capability profile comprises a maximum number of colors supported by a display of the target device, a luminosity range supported by a display of the target device, or a pixel size of a display of the target device, and wherein the quantizing is before the transmitting.

18. The method of claim 1, further comprising:

determining an original format of the portion of visual content data;

determining a target format of the portion of visual content data based on the capability profile of the target device, wherein the capability profile comprises at least a format or decoder supported by the target device; and

formatting the item of visual content into the format supported by the target device or into a format that can be decoded with the decoder supported by the target device,

wherein the formatting is before the transmitting.

19. The method of claim 1, wherein the transmitting the visual content data comprises:

determining to transmit the portion of visual content data as a plurality of data files based on the capability profile, wherein the capability profile comprises at least a buffer size or a processor capability of the target device; and

transmitting the portion of visual content data to the target device as a plurality of data files.

20. The method of claim 19, wherein the transmitting a plurality of data files comprises packaging the portion of visual content data into a plurality of sequential data files, wherein each data file comprises a segment of the portion of visual content data and a sequence identifier for enabling the segments to be downloaded in sequence, and wherein a size of each segment is based on the capability profile.

21. The method of claim 1, wherein the transmitting the visual content data comprises:

determining to transmit the portion of visual content data as a single stream of data based on the capability profile, wherein the capability profile comprises at least a buffer size or a processor capability of the target device; and

transmitting the portion of visual content data to the target device as a single stream of data.

22. The method of claim 1, further comprising receiving the capability profile from the target device or obtaining the capability profile from a database containing user profile information

23. The method of claim 1, further comprising compressing at least the portion of visual content data.

24. The method of claim 1, wherein the content source comprises a first document, and wherein the method further comprises:

obtaining the capability profile of the target device, wherein the capability profile indicates a size in pixels of a screen of the target device; and

generating a second document comprising the resized portion of visual content data,

wherein the resizing comprises resizing the portion of visual content data to a pixel size that is substantially equivalent to at least one dimension of the screen size of the target device, and

wherein the transmitting the visual content data to the target device comprises transmitting the second document comprising the resized portion of visual content data to the target device via a wireless network.

25. A formatting and conversion system comprising:

an interpreter configured to read a content source file to obtain text-based formatting instructions and visual content data for at least one content element;

a translator configured to translate the text-based formatting instructions of each content element into numerical-based formatting instructions interpretable by a target device;

a data file converter configured to resize at least a portion of the visual content data based on at least a dimension of a display of the target device; and

a transmitter configured to transmit the numerical-based formatting instructions and visual content data to the target device.

26. A system as claimed in claim 25, wherein the interpreter is further configured to determine a relative location for rendering the at least one content element on a display of the target device.

27. A system as claimed in claim 26, wherein the translator is further configured to define a position of each content element on the display in the numerical-based formatting instructions.

28. A system as claimed in claim 27, wherein the defined position for each content element is defined as a relative position.

29. A system as claimed in claim 25, wherein the interpreter is configured to provide at least one of a horizontal or a vertical dimension to the data file converter for use in resizing visual content.

30. A system as claimed in claim 29, wherein the dimension is measured in pixels.

31. A system as claimed in claim 30, wherein the data file converter is configured to convert the visual content from a compressed format into a non-compressed or non-encoded format, and reduce the pixel count for the non-compressed or non-encoded format in accordance with at least one of the measurements of horizontal or vertical pixels while maintaining an original aspect ratio of the visual content data.

32. A system as claimed in claim 25, wherein the system is configured to perform at least two passes, wherein the translator is configured to translate the text-based formatting instructions into small footprint formatting instructions to provide a content source file in an editable format in a first pass, and wherein the data file converter is configured to resize and compress the editable format file for the target device in a second pass.

33. A system as claimed in claim 25, wherein the numerical-based formatting instructions are binary format instructions.

34. A system as claimed in claim 25, wherein the data file converter is configured to compress at least the portion of visual content data.

35. A system as claimed in claim 25, wherein the interpreter is further configured to obtain a capability profile of the target device, wherein the capability profile indicates a dimension in pixels of the display of the target device, wherein the content source file comprises a first document, wherein the data file converter is configured to resize the visual content data to a pixel size that is substantially equivalent to the dimension of the display, wherein the data file converter is further configured to generate a second document comprising the resized visual content data, and wherein the transmitter is configured to transmit the visual content data to the target device by transmitting the second document to the target device via a wireless network.

36. A system for optimizing rich content for transmission and reproduction on a target device, comprising:

- means for obtaining at least one content element from a content source, each content element comprising text-based formatting instructions and visual content data;

- means for translating the text-based formatting instructions into numerical-based formatting instructions interpretable by the target device;

- means for resizing at least a portion of the visual content data based on at least a dimension of a display of the target device, and

- means for transmitting the numerical-based formatting instructions and visual content data to the target device to enable the content source to be rendered on the target device.

37. A system as claimed in claim 36, wherein the content source comprises a first document, and wherein the system further comprises:

- means for obtaining a capability profile of the target device, wherein the capability profile indicates a dimension in pixels of the display of the target device;
- and

- means for generating a second document comprising the resized portion of visual content data,

wherein the resizing means comprises means for resizing the portion of visual content data to a pixel size that is substantially equivalent to the dimension of the display of the target device, and

wherein the transmitting means comprises means for transmitting the second document comprising the resized portion of visual content data to the target device via a wireless network.

38. A computer readable medium comprising programming instructions that upon executing cause a machine to:

obtain at least one content element from a content source, each content element comprising text-based formatting instructions and visual content data;

translate the text-based formatting instructions into numerical-based formatting instructions interpretable by the target device;

resize at least a portion of the visual content data based on at least a dimension of a display of the target device, and

transmit the numerical-based formatting instructions and the visual content data to the target device to enable the content source to be rendered on the target device.

39. The computer readable medium of claim 38, wherein the content source comprises a first document, and wherein the medium further comprises programming instructions that upon executing cause a machine to:

obtain a capability profile of the target device, wherein the capability profile indicates a dimension in pixels of the display of the target device; and

generate a second document comprising the resized portion of visual content data,

wherein the resize comprises resizing the portion of visual content data to a pixel size that is substantially equivalent to the dimension of the display of the target device, and

wherein the transmit of the visual content data to the target device comprises transmitting the second document comprising the resized portion of visual content data to the target device via a wireless network.

40. A method, comprising:
parsing a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document;
retrieving the second document;
generating a third document integrating the first document and the second document, wherein the generating comprises modifying the at least one tag in the first document to include the referenced second document; and
transmitting the generated third document via a wireless network to a target device.

41. The method of claim 40, wherein the at least one tag is in one or more text-based formatting languages, and wherein the generating the third document further comprises reformatting the at least one tag to a binary-based formatting language.

42. A system, comprising:
an interpreter configured to parse a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document, and retrieve the second document;
a data file converter configured to generate a third document integrating the first document and the second document by modifying the at least one tag in the first document to include the referenced second document; and
a transmitter configured to transmit the generated third document via a wireless network to a target device.

43. A system, comprising:
means for parsing a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document;
means for retrieving the second document;
means for generating a third document integrating the first document and the second document, wherein the generating comprises modifying the at least one tag in the first document to include the referenced second document; and
means for transmitting the generated third document via a wireless network to a target device.

44. A computer readable medium comprising programming instructions that upon executing cause a machine to:

parse a first document, wherein the first document comprises a plurality of tags, and wherein at least one of the tags references a second document;

retrieve the second document;

generate a third document integrating the first document and the second document, wherein the generating comprises modifying the at least one tag in the first document to include the referenced second document; and

transmit the generated third document via a wireless network to a target device.