APPARATUS AND METHODS FOR DEFINITION OF PROPERTIES OF A WEBSITE, AND RETRIEVAL OF WEBSITE CONTENT

The invention relates to an apparatus and method for definition of properties of a website. The invention also relates to a web server, and a method of operation thereof. The invention also relates to a computer readable medium having stored thereon computer-readable instructions for executing the methods disclosed herein.

The invention has particular, but not exclusive, application and allowing website owners to create websites, including complex online e-commerce stores, using simple techniques which give the owner full control of the content and design without requiring the owner to have specialist programming skills.

Numerous efforts have been made towards simplification of the creation and editing process for websites, particularly complex websites such as online e-commerce stores. Conventional techniques do not provide an all-encompassing solution. One such effort is disclosed in United States Patent No. 6,263,352. This document discloses a computer-implemented system which is designed to assist a merchant in setting up an electronic online storefront that is customized to the merchant's business, without requiring the merchant to program. The system employs a store builder wizard to guide a merchant through a series of questionnaires designed to extract information pertaining to the merchant's business. The system further employs a page generator to create active server pages (ASPs) that form the customised storefront. The page generator creates the active server pages from a set of templates that are generic to formation of an online storefront. The templates are written as an extension to the active server page technology in a combination of hypertext language and scripting language. The active server templates specify an additional, higher order scripting level that distinguishes a second level of code by a new delimiter. During creation of the storefront, the page generator reads each active server template file and converts it to a scripting program having executable

lines of code derived from the higher-order level of instructions denoted by the new delimiters. The page generator then executes the scripting program using the merchant data as input to produce a customized set of active server pages. The resulting active server pages contain the hypertext language and the lower-order level of instructions in the scripting language denoted by the original delimiters. The active server pages are stored together to form an active server application customized to the merchant's storefront. Thus, the system disclosed in this document uses integrated scripting files.

Another effort is disclosed in United States Patent Publication No. 2006/0031147. This document discloses a system which invention creates a predesigned software product generally for e-commerce applications that is modifiable by a company-user in terms of not only look and feel, but also functionality and flow. Such a system includes a set of predesigned business modules where each business module performs a unique function. Each business module is accompanied by a respective set of templates, each of which models a web page. Within each template is embedded one or more commands. To customize its system, the company-user (1) selects which business modules it desires, (2) selects which templates it desires to use with each business module, and (3) customizes each template in terms of look and feel. By selecting the modules, the company-user determines the system's overall functionality, and by selecting the templates, the company-user determines the look and feel and flow of the system. Once customized, the company-user populates a database for use of the system with company-specific data.

The invention is defined in the independent claims. Some optional features of the invention are defined in the dependent claims.

Implementation of the techniques disclosed herein may provide significant technical benefits to a website owner or creator in comparison with conventional techniques such as those described above. For instance, generation and use of a new unique

identifier for the website means that content relating to multiple websites can be stored in a single database, with content stored in the database associated with the website using the identifier. Thus, when an Internet user request access to or content from the website, the techniques disclosed herein allow for only content relevant to the requested site to be pulled from the database. A template chosen by the website owner depending on the type of website which has been created is then populated with the relevant content as pulled from the database, and then made available to a user for download. The unique identifier will be discussed in greater detail below, but it is to be noted that this is a new unique identifier and is not a uniform resource locator (URL), a domain name or an IP address; the identifier is unique for the website/online store.

The website content and/or website properties (such as the type of website which may be defined with the template, the URL, and other properties relating to the database) may be defined using a "wizard" process, optionally in a dedicated apparatus (such as an "e-commerce server") which means the content can be defined, and the website created, by website owners not having the requisite programming skills; only basic Internet browser and computer knowledge may be necessary. Further, instead of complex website such as an online store taking weeks or months to create, implementation of the techniques disclosed herein may allow creation of a fully functional complex website in a very short space of time, perhaps in mere hours.

The apparatus for defining the website properties/content may also allocate a unique identifier for the website in the properties and content definition process. Thus, and as noted above, the website properties and/or content can be associated with the website using the identifier, thereby allowing only relevant content to be pulled from the common database responsive to a user request for content. Moreover, any changes made can be updated dynamically and without having to create complex webpage files in their entirety or having to upload these complete

webpage files to a web server prior to these being available for download and/or viewing.

Allowing a website owner to define the website content using the techniques disclosed herein may dramatically reduce the time required to create and publish a new website, such as an online shopping store. The user may select a template for the layout of the website from a plurality of templates, each of which defines basic needs for a different type of website or store such as a retail store, a school, a factory, a clinic, a restaurant, etc. Styles and style options to be applied to the template may be defined separately, thereby allowing a website owner to apply their own personal touches to the template, according to their own tastes or the needs of the business.

In exemplary techniques described below, each template may be a fully functional system back-end "store", with the templates having highly similar or identical functionality when compared with each other. This means that each template can define very different types of organisations, with just minimal common data being used between them. A significant drawback with, for example, the techniques disclosed in US 2006/031147 is that the stores do not share data, there is just some shared data between the individual template modules. Moreover, in this document, the templates are template modules performing a unique function within the overall functionality of the website.

Provision of a web server which is configured to define the display website content by a plurality of separate content files comprising a plurality of files selected from a group of file types comprising: HTML files, cascading style sheets, scripting files and image files. This is particularly beneficial in comparison to the prior art, particularly United States Patent No. 6,263,352 mentioned above. In exemplary techniques described below, each of these separate content files have no direct dependent relation on one another, the files may just need to be in the correct format. The

styles or designs are made up of HTML (ASP or ASPX), CSS, scripts and images for the browser rendering. The code behind – the template files – provides the server processing, which is not connected to the styles but may be on the same shared hosting space. The template files may be in a binary language – for example, these may be .net files. Thus, these files can be considered to be "encapsulated" files offering significantly greater flexibility and simplicity of operation in comparison to the prior art.

Moreover, provision of this new website identifier facilitates domain migration, should a store owner wish to migrate his online store to a new domain. The correct website content will always be pulled by web server 102 from the database according to the website identifier, no matter what address or hosting space is used. This makes it very easy to place a newly-created store in a hosting space or to change the hosting space.

The invention will now be described by way of example only. The following description is provided in the context of the website being an online store/e-commerce site having items for sale through the store, but it will be understood that the techniques disclosed herein are equally applicable with practically any other type of website. Thus, the description of these exemplary techniques is provided with reference to the accompanying drawings in which:

Figure 1 is a series of block schematic diagrams illustrating the principal component parts of an exemplary system for implementing the techniques disclosed herein;

Figure 2 is a process flow diagram illustrating an algorithm for allowing a user to register with the exemplary system of Figure 1 for the purpose of creating a new online store;

Figure 3 is a process flow diagram illustrating an algorithm for allowing a user to define exemplary basic properties for the online store using the system of Figure 1;

Figure 4A is a process flow diagram illustrating an algorithm for allowing a user to define item properties for items for sale through the store;

Figure 4B is a schematic diagram illustrating multidimensional properties of items for sale through the store;

Figure 5 is a process flow diagram illustrating an algorithm for allowing a user to define items having the item properties defined using the process of Figure 4;

Figure 6 is a process flow diagram illustrating an algorithm for allowing a user to edit or to delete items defined using the process of Figure 5;

Figure 7 is a process flow diagram illustrating an algorithm for allowing a user to define categories for the store;

Figure 8 is a process flow diagram illustrating an algorithm for allowing a user to tag items defined using the process of Figure 5 to categories defined in the process of Figure 7;

Figure 9 is a process flow diagram illustrating an algorithm for allowing a user to edit screens of the online store for viewing by a user;

Figure 10 is a process flow diagram illustrating an algorithm for allowing a user – such as an owner or a "power user" – to define other users of the online store;

Figure 11 is a process flow diagram illustrating an algorithm for allowing a user to generate reports relating to the online store;

Figure 12 is a process flow diagram illustrating an algorithm for allowing a user to access the online store, to use it and/or edit it, where appropriate;

Figure 13 is a process flow diagram illustrating an algorithm for allowing a user (owner) to perform editing of a screen of the online store;

Figure 14 is a process flow diagram illustrating an algorithm for allowing a user to search for items in the online store;

Figure 15 is a process flow diagram illustrating an algorithm for allowing a user to select items in the online store for purchase;

Figure 16 is a process flow diagram illustrating an algorithm for allowing a user to purchase items selected using the process of Figure 15;

Figure 17 is a process flow diagram illustrating an overall flow of use of the techniques disclosed herein.

Referring now to the series of block diagrams of Figure 1, Figure 1A illustrates in overview an exemplary system 100 for implementing the techniques described herein. Thus, system 100 comprises a web server 102 configured to communicate through a network 104, such as the Internet, through communications link 106. An apparatus 108 for defining properties of the website (such as the "e-commerce server" mentioned above) communicates through network 104 through communications link 110. Database 112 may be utilised to store content defined by apparatus 108, the content being retrievable through web server 102. In the example described herein, database 112 is provided remotely from both web server 102 and apparatus 108, but it will be appreciated that the database 112 could be provided integrally with one or the other of web server 102 or apparatus 108. Further, the storage of the database may also be distributed across multiple locations. But in this example, database 112 is configured to communicate through network 104 by means of communications link 114. Database 112 may therefore be able to receive website content and properties from apparatus 108, and web server 102 may be able to retrieve relevant content, responsive to a user request for access to a particular website or website page, from database 112.

Referring now to Figure 1B, this provides a block schematic diagram illustrating in more detail the apparatus 108, in this example embodied in a server. Apparatus 108 comprises several hardware and software components as will now be described. Apparatus 108 comprises a microprocessor 116 and a memory 118 which is able to load, at least temporarily, instructions 120 therein. Apparatus 108 is configured to perform the functionality described herein, under control of the processor 116 executing the instructions 120 stored in the memory 118. Apparatus 108 comprises input-output module 122 at least to interface with network 104 through communications link 110. User interface 124 is provided to allow a user to define website properties as described below, although it will be appreciated that, additionally or alternatively, a user interface for this, or another, purpose could be

implemented in a remote user device (not illustrated) which in turn communicates with apparatus 108 through network 104, or by other conventional means.

Apparatus 108 may provide a display 126 for use by a user. It will also be appreciated that the user display may be provided in the remote user device mentioned above, if provided.

Apparatus 108 comprises storage 130, such as one or more disk drives, for storing various software modules, such as the content management system (CMS) 130, and the remaining modules illustrated. CMS 130 comprises sub-modules for allowing definition of store categories 132, allowing definition of properties 134 of items for sale through the store, allowing definition of tags 136 (described below), and allowing definition of items 138 themselves. Compiler 140 may be the compiler of an integrated development environment such as Microsoft Visual Studio, which may be utilised to receive user input, entered through wizard 141, to generate a configuration file 142 having an identifier 143, described in more detail below. It will be appreciated that although compiler 140 is illustrated as being stored in apparatus 108, this could also be stored at the remote user device mentioned above, if utilised. It will also be appreciated that website identifiers 142 need not be contained within a configuration file generated by a compiler, and the identifier may comprise, for example, a simple text file comprising the identifier with or without other data in the text file.

The identifier – whether in a configuration file or a different type of text file – may be published automatically or manually from the server apparatus 108 to the web server 102, for example, to a web application folder such as storage portion 142 (see Fig. 1D) stored therein.

Screens module 144 allows an owner to define which screens are available for display in his or her online store, the process for which being described in greater

detail below. Templates module 146, storing individual templates 148, allows a store owner to define one or more templates for the layout for the store, depending on the type of store required. Store creation and editing module 150 allows the user to create the store and to edit its properties, Styles module 152 and styles option module 160 allow an owner to define the styles to be applied to the chosen template, thereby allowing the owner to apply some personal touches to the pre-selected template. User module 154 allows the owner to define user privileges, such as to specify those who can have administrator privileges for the online store. The reports module 156 allows the store owner to select the reports he or she wishes to be available for any stores he or she creates using the system. Payment module 158 allows an owner to define the manners in which he or she can receive payment from an online customer.

As will be appreciated from the discussion below (particularly the discussion relating to Figures 1C, and 2 to 10), Figure 1B illustrates an apparatus 108 for definition of properties of the website. Apparatus 108 comprises a processor 116 and a memory 118, the apparatus 108 being configured, under control of processor 116 to execute instructions 120 stored in memory 118; to receive user definition of properties of a website; to define a website identifier 143 for the website; and to associate the properties of the website with the website using the website identifier 143.

As noted above, in order to cater for website owners not having the requisite programming skill, the website content may be defined using a wizard process, apparatus 108 generating for display, to the user (e.g. website owner/creator), a series of dialogue boxes prompting the user to define website content.

When a store owner is defining website properties and content in the apparatus 108, any content and/or properties thus defined are associated with the website using the website identifier 143. Apparatus 108 may do this in any one of a number of ways such as tagging (e.g. appending the content data with the identifier, or a

derivative thereof) or simply providing a logical link between the content and the identifier. Thus, apparatus 108 is configured to receive, from the user, definition of website content, the apparatus being configured to associate website content with the website using the website identifier.

In this exemplary implementation, website content is compiled using compiler 140, and the or each website identifier 143 inserted or appended into the compiler configuration file 142. Thus, apparatus 108 comprises a compiler 140, compiler 140 being configured to generate a configuration file 142 comprising the website identifier 143. However, and as noted above, in alternative implementations, any text file may be suitable for the purpose of storing the or each website identifier.

Figure 1C is a block diagram illustrating in some more detail database 112, storing database content 162. In this example, database 162 stores content (and other website properties) 164 received (e.g. copied) from server apparatus 108. Database 112 also stores the website identifiers relevant to the content, or otherwise associates the content with the website using the identifier. In this example, website content 164 is linked to a particular online store using the website identifier (or a derivative thereof) which is stored in an identifier column 166 adjacent (or appended to) the content the website is associated with. In an alternative arrangement, a reduced-number of instances (say, one) of the website identifier is stored, and the content logically linked to the instance of the website identifier thereby to provide the association. It will be appreciated that other arrangements are also envisaged.

The association may be effected by server apparatus 108, and/or by database 112.

Therefore, database 112 can store website properties and/or content for multiple websites (online stores) one or more of which having had their content defined in the wizard process described above, and associated with the website using the website identifier, so that when the website or a page thereof is requested by a user

(using, say, a web browser) only the relevant content is pulled from the database by the web server 102, now described in more detail below.

Figure 1D illustrates in more detail web server 102. Web server 102 comprises microprocessor 170 and memory 172 which is able to load, at least temporarily, instructions 174 therein. Web server 102 is configured to perform the functionality described herein, under control of the processor 170 executing the instructions 174 stored in the memory 172. Web server 102 comprises input-output module 176 to interface with network 104 through communications link 106. User interface 178 as provided to allow a user to control web server 102, although it will be appreciated that, additionally or alternatively, a user interface could be implemented in a remote user device (not illustrated) which in turn communicates with web server 102 through network 104, or by other conventional means.

Web server 102 comprises storage 180 such as a disk drive for storing various software components such as the styles module 182, and the remaining modules illustrated. Styles module 182 implements the styles defined by the website owner using the styles module 152 of server apparatus 108. Styles module 182 comprises a styles option module 184 which, in turn, has style sub-modules, including module 186 for cascading stylesheets (CSS), images style option module 188 and script files style option module 190 which may be used for storing JavaScript files for use in website. The default style option or these style options 184 may be replaced by the CSS, scripts, and images already in the Style Options module 160 from apparatus 108 of Figure 1B to allow more choices. In at least one implementation, the CSS, scripts, and images are dependent on the Style Options and, as such, may not need ever to be changed. In some implementations, it is preferable to store some common images and styles features in web server 102, rather than at database 112. Thus, commonly-used content such as an "OK" button, which may be used time and time again across multiple websites owned by multiple owners may be stored locally at web server 102. Styles module 182 also comprises HTML module 192 for storing

HTML files such as ASPX pages which may be rendered in an Internet browser. Each of these is defined by the website owner in the website definition process using apparatus 108.

Storage 180 may also store configuration files 142 (or copies thereof) containing website identifiers 143 generated by and received from apparatus 108 of Figure 1B. Templates module 146 stores templates 148 defined by the website owner in the website creation process using apparatus 108. A more detailed description of the generation of website content to be made available for display at the user device will be given below, but, in summary, Figure 1D illustrates a web server 102 comprising a processor 170 and a memory 172. Web server 102 is configured, in response to a user request for access to a website (e.g. a user wishes to view a web page, or download website content therefrom), to execute, under control of processor 170 instructions 174 stored in memory 172 to determine the website identifier 143 for the website. Associated content for the website 164 is retrieved from a database – in this example remote database 112 – storing content for a plurality of websites, the associated content being associated with the website using the website identifier 143 (for example, being stored in identifier column 166 in database 112). Web server 102 is configured to make display website content available for display on a user device (not shown), the display website content comprising a template 148 populated with the associated content.

Web server 102 is configured to determine the website identifier 143 as a website identifier which has been allocated for the website in a process in which a website owner has defined properties for the website. Web server 102 is configured to identify the website identifier as a website identifier which has been allocated for the website by an apparatus for defining website content under control of a website owner

More detailed operation of the components of system 100 is now given with respect to Figures 2 to 17. In the following processes relating to apparatus 108, apparatus 108 generates for display (for example on display 126 or on a display of the user device), to the user, a series of dialogue boxes 127 prompting the user to define website content. The dialogue boxes 127 may be generated by wizard module 141.

Turning now to Figure 2, a process 200 implemented in apparatus 108 for allowing a user to register with the exemplary system of Figure 1 for the purpose of creating a new online store is illustrated. Process 200 begins at step 210 where the user is prompted to enter personal data such as a user name, a password, e-mail address and so forth. At step 220 apparatus 108 checks to determine whether the correct string patterns have been used and performs some basic error-checking. If errors are found at step 230, process 200 reverts to step 210 for the user to correct any errors. If no errors are found at step 230, verification is given to the user (for example, where the owner is remote from apparatus 108, communicating with apparatus 108 through network 104 from the owner's device), verification is sent to the owner by e-mail) and once the verification process has been completed at step 250 (for example, by the owner responding to the e-mail, or "clicking" a verification link in the e-mail) a table of users (not shown) stored in database 112 (or elsewhere) is updated with the verified new owner.

Of course, an owner may be able to create more than one store/website. In such situations unique identifiers are allocated to each of the stores/websites despite the fact they may be owned by a single store owner.

Figure 3 illustrates a process 300 for allowing a user to define exemplary basic properties for the online store using the server apparatus 108. At step 310, the user logs in with the log in credentials established in process 200. At step 320-a, the user is prompted through the wizard to enter criteria such as the store name and the type of store. The type of template 148 to be used for the online store may be defined by

the type of store the user selects in this step. For example, a School Template has screens for class schedules, calendars, photo albums etc., while a Factory Template may have very few images and may be used mainly for item cataloguing.

The user also selects store styles and style options, implemented in styles option module 152. The style describes the general page layouts (HTML), colours and control sizes/fonts/styles and the web scripting needed. Just some examples of the types of styles which can be used are "colourful balloons", "corporate", "vintage", "nautical" and so on. Some styles are "lightweight" meaning they will function better on handheld devices such as smart phones and tablets, devices suffering with slow Internet access, or on outdated web browsers. The style options can be variations on the style, with different colours, fonts and images. It may be that in some implementations at least, any style can work with any template because there is no direct link between the two. Thus, the files can be considered to be "encapsulated" files. This creates a large number of possible combinations.

At step 320-b, the owner is prompted by wizard 141 to enter other website properties such as the store URL/domain name, a URL for the store logo, Facebook link and Twitter links and the like. Although not explicitly shown in Figure 3, the unique identifier 143 for the online store may also be generated at this time. The data received and/or generated up until this point is then stored in one or more tables in database 112 or elsewhere at step 330. At step 340, the user is prompted to enter physical address parameters for the online store such as the store address, the zip code, telephone number etc. At step 350, this data is stored in a table in store address fields. The user is prompted at step 360 to define payment options and at step 370 a table is updated with the entered payment options.

Figure 4A is a process flow diagram illustrating a process 400 for allowing a user to define item properties for items for sale through the store. It may be desirable to define item properties before any items for sale are defined; that is, it may be

preferable to define the ways to describe the items before being able to define the items. At step 410, the user is prompted to add or edit a new possible size and weight, or “standard” colour for store items, which may be after valid size and weight types have been retrieved 420. At step 430, a table is updated with any new information. The multi-dimensional custom properties – discussed in more detail with reference to Figure 4B – may be added or edited at step 440, and the table updated with any new data relating to the properties at step 450, prior to the process ending at step 460.

Figure 4B is a schematic diagram illustrating multidimensional properties of items for sale through the store. The properties for the items can be defined in as many dimensions as can be conceived by the store owner. In the example of Figure 4B, the material properties are defined in three dimensions, but other numbers of dimensions are envisaged, such as two, four, or greater than five.

Firstly, in the first properties dimension 472, we have a number of different types of item properties (these custom properties can have any names that can mean anything which may be conceived by the store owner; the following are simply examples): materials, weight, (physical) dimensions of the items, complexity, frequency and so on. In a second properties dimension 474, the materials can be categorised as metal, plastic, wood, glass, gas, data and so on. In a third properties dimension 476, the wood material can be further defined according to the type, such as oak, pine, cedar etc. In another top level properties dimension 478, wood finishes to be applied to the wood material can be defined, such as paint colours, stain colours and so on. The colours themselves can be defined in the fifth properties dimension 480 (within the property dimension wood finishes) as black and so on. As noted, the number of dimensions is limited only by the imagination of the store owner. Thus, apparatus 108 is configured to receive user definition of properties of an item for display on the website, and to receive user definition of multidimensional

properties of the item, wherein a number of dimensions in the multidimensional properties is at least three, four, five or greater.

Provision of the multi-dimensional properties adds significant capability and flexibility to an online store. For example, many people may be able to make simple HTML pages, and then copy and paste them to shared hosting sites while just changing the titles. However, such simple sites have very limited functionality and may have no content management, no searching ability, nor any reporting ability. So, the multi-dimensional properties, which may be provided in combination with other features disclosed herein, create powerful and flexible online stores that can also be created quickly using, for example, the wizard described above and live on page editing.

Different custom properties can also have a different number of dimensions at the same time; that is not all properties must have the same number of dimensions. For example, materials can have three dimensions: the first dimension being Material; the second dimension being types of material (wood, metal, glass, etc); and the third dimension being types of wood (oak, pine, etc). In another example, the "Finish" can have more than three dimensions: the first dimension being finishes; the second dimension being types of finishes (wood, metal, etc); the third dimension being types of wood finish (paint, stain, etc); the fourth dimension being types of paint (premium, standard); the fifth dimension being sub properties of premium paint (paint colours, paint reflection, etc); the sixth dimension being types of paint colour (red, green, blue etc.); an alternative sixth dimension (within premium paint) being types of paint reflection (matte, glossy, metallic), and so forth.

This may be useful for items that require complex description. There can be some custom properties without added dimensions, which can also be useful for items that do not require complex description. Use of custom properties can greatly improve item searching within a store. For example, a search in in the category

Vehicles All Car BMW Brands (and sub-models) with the Colour Red AND has the material OAK interior AND has never been in an Accident AND has a 16 valve-4 cylinder engine AND costs between 100,000 – 150,000 AND made between 2002-2008 AND NOT pre-owned. The process of adding Custom Properties may be done visually by the Owner like adding branches to a tree and branches on branches.

Figure 5 illustrates a process 500 for creation of store items using the properties defined in process 400. The process begins at step 501, where the owner is prompted at step 505 to select the store, or one of the stores, registered to him or her. At step 510, an item may be added, the owner being prompted for a name, an item ID, and for the owner to upload an image of the item. Saved sizes and weights are retrieved from their table at 520, for a user to select sizes and weights at step 530, along with colours for the item from a predefined palette. At 540, any custom properties which have been defined at step 440 of Figure 4 are retrieved for the owner to select from, in step 550, for the particular item being added. The owner is prompted for an item description to be added at step 560, in one or more appropriate languages, and the relevant table is updated at step 570, before the process ends at step 580.

Store items may be edited or deleted in process 600 illustrated in Figure 6. The process begins at step 601, with the owner being prompted at step 605 to select the store, or one of the stores, registered to him or her. Item properties are retrieved at step 610 and at step 620, apparatus 108 generates for display for the user, a table showing all items created by the store owner. The owner is prompted to filter items in the table according to a search string, and an item record is selected at step 640. Should the owner wish to edit the item record, he or she is prompted to edit the name, the identifier, the description and so on at step 650. At 670, the user is prompted to add or change properties of the item, such as the colour, size, weight etc. From predefined values. At step 680, the user is prompted to select any combination of properties defined at step 440 of Figure 4. At step 690, the relevant

table is updated with any new information and the user is prompted as to whether or not he or she wishes to do more. If not, the process ends at step 699. If the user wishes to do more, he or she is returned to step 630.

Returning to step 640, if the owner wishes to delete a stored item, he is prompted at step 660 to do so, and any deletion is updated in the relevant table at step 665.

In Figure 7, a process allowing a user to define store categories (logical groupings of items, such as menswear, ladieswear, formal wear, trousers, jeans, etc), whether by addition, editing or deletion, is illustrated. Process 700 commences at step 701 with the owner being prompted at step 705 to select the store, or one of the stores, registered to him or her.). In the example of Figure 7, at this stage the store is empty and the owner may define categories for the store. As such, this allows stores to contain any kind of product to the preference of the owner. If the owner has defined any categories for the store previously, these are loaded from the categories table in the database at step 710. In other implementations, it may be preferable for common stores to have pre-defined categories which a store owner may select from. At step 720, multidimensional categories are displayed, with the store name (root category) being automatically selected. At 730, the user is prompted to select the root category, or another previously created category. If the owner wishes to create another category, the user is prompted to do that at step 740. If the owner wishes to rename the selected category, this is done at step 750. If the owner wishes to delete the selected category, this is done at step 760. Regardless of the option followed from 740, 750, 760, the relevant table for categories is updated at step 770. If the user wishes to perform more category definition, he is returned to step 730, or the process ends at step 799, if no more category definition is required.

Categories may be created in a multi-dimensional tree similar to the custom properties. It may be that the online store's menu structure will match the category structure. A user will be presented with a first level of categories, then inside that

first level, the user may be able to retrieve a second level of categories and so on. As with the properties, there is practically no limit to the category complexity. However, and in order to make a screen user-friendly, and to make menus easy to navigate, the store owner may choose to add limitations based on the screen size and the style chosen. The catalogue style can, for example, accommodate very large and complex category menu structures.

Figure 8 is a process flow diagram illustrating an algorithm 800 for allowing a user to tag items defined using the process of Figure 5 to categories defined in the process of Figure 7. The process begins at step 801, and at 805, the owner is prompted to select one of the stores registered to him or her. Previously-defined categories are retrieved at step 810 based on the selected store/template. At 820, apparatus 108 displays the multidimensional categories, with the store name (root category) automatically selected. The previously-defined items are retrieved at step 830, and apparatus 108 fills a table of items at step 840. Previously-defined tags, if any, are retrieved at step 850 and the table of these is filled at step 860 by apparatus 108. At step 865, the user is prompted to select a category from the list of categories, displayed in tree format. At 870, the user has the option to add or remove a tag. If tagging an item to a category, an item price is added at step 871, and a category is selected from the category tree and an item from the item table at step 872.

If the user wishes instead to remove a tag, the user is prompted to select a record from the tagging table at step 874 for deletion.

Whether a tag is being added or removed, the tag table is updated at step 873, before the user is asked whether or not they wish to do more. If yes, the process loops back to step 870. If the user does not wish to do anything, the process ends at step 899.

Pricing may not be defined at the time of creating new items. Tagging can be considered as the process of adding Items to a category. Or, to put it another way, tagging links an item to a category. Tagging an item to a category may be done by clicking on a tag with the selected item and selected category. Then the store owner may click on a place he or she wants the item to have in the selected category.

Tagging can be highly beneficial as an owner can list the same item (defined only once) for sale in different stores owned by the owner, but with different properties, for example, price, in each store. The item can be considered to belong to the store owner (rather than a specific store). Thus, apparatus 108 is configured to receive a user definition of a category of an item for retrieval from the website for user display, and to receive user definition of a link between the item and the category.

Figure 9 is a process flow diagram illustrating an algorithm 900 for allowing a user to edit screens of the online store for viewing by a user. Process 900 commences at step 901, and then the owner is prompted to select a store that he or she owns. A list of screens is retrieved at step 910. At 920, the user is prompted to add or delete a screen. Should the user wish to add a screen, screens can be added at step 930 by entering the screen name. An alternative language name may also be entered. Effectively, screens can relate to just about anything. For instance, the user may wish to have one or more screens relating to "about us" (the store provider), terms and conditions, privacy policy and so on. At step 931, the selected screen is added.

On the other hand, should the owner wish to delete a screen, apparatus 108 will remove the selected screen from the screen table.

Whether adding or deleting a screen, the table of screens is updated at step 960. The user is prompted to do more at step 970, and should further editing be required, the owner is returned to step 920, or the process ends at step 999. Screen editing may

be done using a full HTML WYSIWYG editor, and/or content pulled from the database to the online store.

In the example of Figure 9, the screens may have relatively simple functionality, such as adding screens for the store owner to provide information relating to “About us”, “Contact Us”, “Terms and Conditions”, requiring, for example, just text and graphics and formatting. The schedules, calendars, photo albums, etc have special functionality provided by the template and there are matching Styles to format these screens. Every screen has limited editing ability on the actual online web-site after logging in as the store owner.

Figure 10 is a process flow diagram illustrating an algorithm 1000 for allowing a user – such as an owner or a "power user" to define other users of the online store. An Owner can create Employees and Power Users (Employees and Power Users cannot create other Users). This allows other people to help maintain a Store. Employees and Power Users belong to a Store and cannot create a Store. If an Owner has multiple Stores, then Employees need to be created for each Store. To create an Employee or Power User, the owner invokes process 1000 starting at 1001.

The owner can select a Store to get a list of Users that belong to that Store. Then you owner can click on Add to add an Employee to that Store. Enter a username, password, user Type (cannot be Owner or Admin), Name, Full Address, Language, email, and National ID 1021. The Owner can select a different Store he/she wants to, then Update to create the User. Creating Users here does not require verification.

To Edit a User, select a User in the User list that belongs to the selected Store, then the Users info will appear. Edit the username, password, address, etc. 1042 and the store the User belongs to 1045, click Edit. If the Owner is editing himself/herself, then the Store the Owner belongs to Store “None” 1044.

To Delete a user, select the User and click Delete 1031.

An Owner can also create a Guest or Customer – other users of the system – to allow them Store browsing or shopping. But this is normally done by registering at the Online Store.

Therefore, it will be appreciated that the foregoing describes a method for definition of properties of a website, the method being executed in an apparatus 108 and comprising, under control of the processor 116 of the apparatus 108: receiving user definition of properties of the website; defining a website identifier for the website; and associating the properties of the website with the website using the website identifier.

Turning to Figure 11, Owners have access to Reports 1100. Select Reports from the menu, select a Store, then select a Report 1120. A table will be filled with data the Report provides. Some reports are interactive updating Status 1140.

Figure 12 is a process flow diagram illustrating an algorithm 1200 implemented in web server 102 for allowing a user to access the online store, to use it and/or edit it, where appropriate.

To use the online store, a user, for example, enters the URL in an Internet Browser. The Store Home 1200 will load all the Categories into Menus which will contain the Store Items. On the facility for a user to register in the Home Screen at step 1245, the user can select a Category 1262, Select a Screen 1263, change the language 1210, or view the user's shopping cart. If the user is not registered to buy on the store 1240, he or she may register by clicking Register on the top of the screen 1245. This is the same process as registering to become a Store Owner.

After registering, you can now log in. If the person logging in is identified as the store Owner, 1250, the Owner can edit the screen using an HTML editor and save the screen to the Database 1255.

Process 1200 starts at step 1201. Should a user wish to access the online store, the user seeks to do so using any one of conventional means such as typing the website URL directly into the address bar of the Internet browser being used in the user's device. The user's request will be routed by conventional means to web server 102 through network 104. Web server 102 reads the URL and, from this, looks up the unique identifier 143 for that online store in the configuration files 142 stored in storage 180 of web server 102. Web server 102 then retrieves only the relevant content relating to that online store from database 112. For instance, and referring again to Figure 1C, web server 102 may be arranged to scan the table of database properties 162 for any content associated with that identifier. Thus, web server 102 comprises storage for storing a plurality of website identifier is for a plurality of websites, and is configured to determine the website identifier based on the user request to access the website.

The relevant content is then retrieved and communicated through network 104 to web server 102. The selected template 148 – defined by the store owner using apparatus 108 as described above – is determined, and the template populated with the retrieved content and made available for display on the user device, according to the style styles and styles options defined relating to the CSS module 186, the images module 188, the script files module 190 and the HTML pages 192.

That is, web server 102 is configured to store multiple template files, the multiple template files being for different categories of websites, and wherein the apparatus is configured to make the display website content available for display based on a user selection of a template.

As noted above, web server 102 is configured to define the display website content by a plurality of separate content files comprising a plurality of files selected from a group of file types comprising: HTML files, cascading style sheets, scripting files and image files. This is particularly beneficial in comparison to the prior art, particularly United States Patent No. 6,263,352 mentioned above. In this example, each of these separate content files have no direct dependent relation on one another, the files just need to be in the correct format. The styles or designs are made up of HTML (ASP or ASPX) CSS, scripts and images for the browser rendering. The code behind – the template files – provides the server processing, which is not connected to the styles but maybe on the same shared hosting space. The template files may be in a binary language – for example, these may be .net files. Thus, these files can be considered to be "encapsulated" files offering significantly greater flexibility and simplicity of operation in comparison to the prior art.

Thus, only the content relevant to the desired online store, defined in the simplified process as described above for apparatus 108, is retrieved for display from a database storing content and website properties for plural websites.

As such, it will be appreciated that there has been described a method for generating display website content for display on a user device in response to a user request for access to a website, the method being executed in a web server 102 and comprising, under control of a processor 170 of the web server 102: determining a website identifier 143 for the website; retrieving associated content 164 for the website from a database 112 configured to store content for a plurality of websites, the associated content being associated with the website using the website identifier; and generating display website content available for display on a user device, the website content comprising a template 148 populated with the associated content.

Additionally, the Store Home will load all the Categories into Menus which will contain the Store Items. On the Home Screen Register 1245, the user can select a Category 1262, Select a Screen 1263, change the language 1210, or view the user's shopping cart. If the user is not registered to buy on the store 1240, the user may Register by clicking Register on the top of the screen 1245. The process for doing this may be the same or similar as the process for registering to become a Store Owner. After registering, the user can now log in. If the person logging in is identified as the store Owner, 1250, the Owner can edit the screen using an HTML editor and save the scheme to the Database 112.

To get general information about the Store, a user can implement the process of Figure 13 and click on one of the Store Screens 1300 on the bottom of the Store Home screen. If the person logged in is the Owner, then the HTML editing capability will appear, and the Owner can make changes live, and then save the content. When finished reading the screen, the user may click on another screen or the Home button 1399 to get back to the Home page.

To search for an Item in the user may implement the process of Figure 14 and select a Category from the Menu to get all Items in that Category and all Items in the sub-Categories 1400. A search can also be done without selecting a Category 1410, but the search results may be less relevant. The user may enter a text Search String 1415 to get all Items with the same Name or Item ID. The user may select a Property to search for, such as a colour, size, weight, a price range 1440, or any combination of Custom Properties 1430. Web server 102 may then sort the search results by Name, ID, most Popular, or Price 1450. When finished, the user may select an item for Item Details 1499 or Click on another Category 1420 for a new search. The Price range, Colours, and Properties can be changed without reloading the Category.

The Item Detail 1500 may have additional information the Search Screen does not. This screen has the full description in the language selected and a larger Image 1520.

A user can at this time enter the quantity he or she wants to buy or go back Home if the user changes his or her mind. If the user selects Add to Cart, this will add the Item with the quantity and take the Customer to the Cart screen. The Customer can then choose to continue shopping or make the purchase.

Store Shopping Cart 1600 is where the Customer goes to purchase what is in the shopping Cart. At this point, the Customer can remove Items in the Cart 1650 or Change the quantity. Next, the customer may select his or her shipping address if the Customer has more than one address saved. The customer may select the Payment type has Cash or Charge (one of many possible payment options) 1681. The Customer can at any time go back to the Home screen to continue shopping 1699.

If the Customer selects the Cash option (Cash or Bank Transfer), then web server 102 creates an invoice, emailed to the Customer & Store Owner, and the Status of the payment Report will remain Unpaid until Cash is received. If the Customer selects a Charge option (PayPal, VISA, MasterCard, etc.) the web server 102 will call the Online Payment company Web Service 1690 to effect the Payment. The Web Service will then return the result back to the Online Store. The Store will show the transaction Result and create an Invoice if the result is successful 1696. The shopping Cart will be cleared in any case. Then the Customer can continue shopping.

An overall view of the macro environment in which these techniques can be implemented is given in Figure 17.

It will also be appreciated that the invention extends to a computer readable medium – such as storage 128 of apparatus 108 or storage 180 of web server 102 – having stored thereon computer-readable instructions for executing under control of a respect of processing device, the methods described or claimed herein.

It will be appreciated that the invention has been described by way of example only. Various modifications may be made to the techniques described herein without departing from the spirit and scope of the appended claims. The disclosed techniques comprise techniques which may be provided in a stand-alone manner, or in combination with one another. Therefore, features described with respect to one technique may also be presented in combination with another technique.

Claims

1. Apparatus for definition of properties of a website, the apparatus comprising a processor and a memory, and being configured, under control of the processor, to execute instructions stored in the memory:

to receive user definition of properties of a website;

to define a website identifier for the website; and

to associate the properties of the website with the website using the website identifier.

2. The apparatus of claim 1, wherein the apparatus is configured to receive, from the user, definition of website content, the apparatus being configured to associate website content with the website using the website identifier.

3. The apparatus of claim 1 or claim 2 configured to generate for display, to the user, a series of dialogue boxes prompting the user to define website content.

4. The apparatus of any preceding claim, wherein the apparatus comprises a compiler, the compiler being configured to generate a configuration file comprising the website identifier.

5. The apparatus of any preceding claim, wherein the apparatus is configured to receiver user definition of properties of an item for display on the website, and to receive user definition of multidimensional properties of the item, wherein a number of dimensions in the multidimensional properties is at least three, four, five or greater.

6. The apparatus of any preceding claim, wherein the apparatus is configured to receive a user definition of a category of an item for retrieval from the website for

user display, and to receive user definition of a link between the item and the category.

7. A web server comprising a processor and a memory, the web server being configured, in response to a user request for access to a website, to execute, under control of the processor, instructions stored in the memory:

to determine a website identifier for the website;

to retrieve associated content for the website from a database configured to store content for a plurality of websites, the associated content being associated with the website using the website identifier; and

to make display website content available for display on a user device, the display website content comprising a template populated with the associated content.

8. The web server of claim 7 configured to determine the website identifier as a website identifier which has been allocated for the website in a process in which a website owner has defined properties for the website.

9. The web server of claim 7 or claim 8 configured to identify the website identifier as a website identifier which has been allocated for the website by an apparatus for defining website content under control of a website owner.

10. The web server of any of claims 7 to 9 comprising storage for storing a plurality of website identifiers for a plurality of websites, and is configured to determine the website identifier based on the user request to access the website.

11. The web server of any of claims 7 to 10 configured to define the display website content by a plurality of separate content files comprising a plurality of files selected from a group of file types comprising: HTML files, cascading style sheets, scripting files and image files.

12. The web server of any of claims 7 to 11 configured to store multiple template files, the multiple template files being for different categories of websites, and wherein the apparatus is configured to make the display website content available for display based on a user selection of a template.

13. A method for definition of properties of a website, the method being executed in an apparatus and comprising, under control of a processor of the apparatus:

receiving user definition of properties of the website;

defining a website identifier for the website; and

associating the properties of the website with the website using the website identifier.

14. A method for generating display website content for display on a user device in response to a user request for access to a website, the method being executed in a web server and comprising, under control of a processor of the web server:

determining a website identifier for the website;

retrieving associated content for the website from a database configured to store content for a plurality of websites, the associated content being associated with the website using the website identifier; and

generating display website content available for display on a user device, the website content comprising a template populated with the associated content.

15. A computer readable medium having stored thereon computer-readable instructions for executing, under control of the processing device, the method of claim 14 or claim 15.