Abstract: A controlling computing device connected to an interconnecting network can control specified functionality of a controllable computing device connected to the interconnecting network by using at least one unique identifier to establish a connection between the controllable computing device and the controlling computing device, with a user interface being provided to the controlling computing device to control the specified functionality of the controllable computing device may be created and modified by receiving control input comprising associations between actions on the controlling computing device and commands for controlling the specified functionality of the controlling computing device, and storing the actions, associations and commands as part of the user interface. User interfaces may be made available by storing a library of predefined user interfaces and, responsive to a request for a particular one of the user interfaces, transmitting that particular user interface to the requestor.
CONTROL OF COMPUTING DEVICES AND USER INTERFACES

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority to United States Provisional Patent Application No. 61/380,649 filed on September 7, 2010 and to United States Provisional Patent Application No. 61/415,204 filed on November 18, 2010, the teachings of each of which are hereby incorporated by reference.

FIELD OF INVENTION

[0002] The present invention relates to control of computing devices and user interfaces, and more particularly to control of one computing device by another computing device and to customization and socialization of user interfaces.

BACKGROUND OF THE INVENTION

[0003] Various methods are known for enabling a first network-connected computing device to control a second network-connected computing device. For example, in Virtual Network Computing the RFB protocol is used to send keyboard and mouse inputs from the first computing device to the second computing device and to return screen graphics from the second computing device to the first computing device.

[0004] U.S. Patent Application Publication No. 2009/0144629 in the name of Ferlitsch et al. describes the use of a mobile device such as a smartphone to control a networked multi-function peripheral ("MFP") such as a combined printer/scanner. Responsive to a request from the mobile device, a controlling application is loaded and executed for controlling the MFP, and a user interface is provided to the mobile device, which enables the mobile device to send control commands to the controlling application for controlling the MFP.

[0005] U.S. Patent No. 7,707,606 and U.S. Patent Application Publication No. 2010/0191833 in the name of Hofrichter et al. teach a home network system that provides for interoperability of the networked devices such as televisions, video players, video cameras and the like. A gateway device such as a personal computer with Internet access communicates with the
various devices to identify compatible applications, downloads the compatible applications from remote servers (e.g. the Internet) and provides a centralized GUI for enabling a user to select and execute a given application for controlling the devices.

[0006] U.S. Patent No. 7,783,702 to Liu et al. teaches a system and method for using a mobile phone to control a computing device such as a personal computer. The phone is connected wirelessly to the computing device, for example via Bluetooth, Wi-Fi, GPRS or CDMA1x. Interaction engines are installed on both the phone and the computing device to enable them to communicate with one another and to enable the mobile phone to control the computer.

[0007] Although the above references teach systems and methods for enabling one computing device to control another computing device, there remains a need for efficient systems and methods that enable a controlling computing device to identify computing devices that are available to be controlled, without the controlling computing device having to connect to those computing devices or to their local networks in order to identify them.

[0008] In addition, while it is known for software programs to enable their user interfaces to be customized on a local computing device, there remains a need for systems and methods that facilitate the creation, modification and sharing of user interfaces.

SUMMARY OF THE INVENTION

[0009] The present invention enables a computing device, such as an Internet-capable smartphone, to remotely control specified functionality of another computing device connected to the same interconnecting network. For example, a smartphone could be used to control a game console or a public video display system. The invention also enables a user community to develop, edit and share customized user interfaces for software applications and specified content.

[0010] In one aspect, the present invention is directed to a method for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network. The method comprises receiving at a connection management server, from the controlling computing device, at least one
unique identifier for the controllable computing device. The connection management server uses the unique identifier to facilitate a connection between the controllable computing device and the controlling computing device via the interconnecting network. The method may further comprise the connection management server delivering to the controlling computing device a user interface for controlling the controllable computing device. The connection may be a socket connection via the Internet, and the at least one unique identifier may be one or more of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address.

[0011] In another aspect, the present invention is directed to a method for enabling a controlling computing device connected to an interconnecting network to control specified functionality of a controllable computing device connected to the interconnecting network. The method comprises presenting, at the controllable computing device, at least one unique identifier for establishing a connection between the controllable computing device and the controlling computing device. The unique identifier is presented independently of any user interaction with the controllable computing device. The method further comprises receiving, at the controllable computing device, a connection request from the controlling computing device, with the connection request being based on the unique identifier. In response to the connection request, the method establishes a connection between the controllable computing device and the controlling computing device via the interconnecting network using the unique identifier, and the controllable computing device identifies and delivers to the controlling computing device a user interface for controlling the specified functionality of the controllable computing device. The at least one unique identifier may be at least one of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address. The connection may be a socket connection via the Internet, and the unique identifier may describe a URL from a web connection to the socket connection.

[0012] In one embodiment, the user interface is limited to a predefined set of commands associated with the specified functionality. The specified functionality may be an application or a file, or both.
In a further aspect, the present invention is directed to a method for defining a user interface for controlling, from a controlling computing device connected to a network, specified functionality of a controllable computing device connected to the network. The method comprises receiving control input comprising associations between actions on the controlling computing device and commands for controlling the specified functionality of the controllable computing device, and storing the actions, associations and commands as part of the user interface. In a preferred embodiment, the controlling computing device is a touch-screen device and the actions on the controlling computing device comprise actuation of touch-actuable widgets, and the method further comprises receiving positional input defining a positional layout of the touch-actuable widgets on a screen of the touch-screen device and storing the positional layout as part of the user interface.

In one embodiment, the user interface is a modified user interface and the step of receiving control input associating actions on the controlling computing device with commands associated with the specified functionality of the controllable computing device comprises modification of an original user interface. In another embodiment, the user interface is a new user interface.

In one embodiment, the user interface is stored on a server accessible by a plurality of parties and the user interface is made available to at least some of the plurality of parties.

In a still further aspect, the present invention is directed to a method of making user interfaces available. The method comprises storing a library of predefined user interfaces and, responsive to a request for a particular one of the user interfaces, transmitting that particular user interface to the requestor. The method may further comprise receiving additional user interfaces from third parties and adding the additional user interfaces to the library. Preferably, before adding the additional user interfaces to the library, the method validates the additional user interfaces and adds the additional user interfaces to the library only in response to successful validation of the additional user interfaces.
In one embodiment, the user interfaces enable control, from a controlling computing device connected to an interconnecting network, of specified functionality of a controllable computing device connected to the interconnecting network, and the particular user interface is transmitted to the controlling computing device in response to the request for the particular user interface.

The user interfaces may be associated with different software applications, and may include a subset of different user interfaces associated with a single software application.

The method may further comprise receiving a modified user interface comprising a modified version of the particular user interface transmitted in response to the request and adding the modified user interface to the plurality of predefined user interfaces.

In other aspects, the present invention is directed to computing devices and computer program products for implementing the above methods.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other features of the invention will become more apparent from the following description in which reference is made to the appended drawings wherein:

FIG. 1 shows schematically a first system for enabling a controlling computing device connected to an interconnecting network to control specified functionality of a controllable computing device connected to the interconnecting network, according to an aspect of the present invention;

FIG. 2 is a flow chart showing a first exemplary method for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network;

FIGS. 3A and 3B show schematically a first particular exemplary implementation of the system of Figure 1;
FIGS. 3C and FIG. 3D show schematically a second particular exemplary implementation of the system of Figure 1:

FIGS. 3E and FIG. 3F show schematically a third particular exemplary implementation of the system of Figure 1:

FIGS. 3G and FIG. 3H show schematically a first exemplary implementation of the system of Figure 1 using WiFi beacons as unique identifiers;

FIGS. 3I and 3J show schematically a second exemplary implementation of the system of Figure 1 using WiFi beacons as unique identifiers;

FIG. 4 shows schematically a first particular exemplary implementation of a second system for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network, according to an aspect of the present invention;

FIG. 4A shows schematically a second particular exemplary implementation of the second system for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network, according to an aspect of the present invention;

FIG. 4B shows schematically a third particular exemplary implementation of the second system for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network, according to an aspect of the present invention;

FIG. 5 is a flow chart showing a second exemplary method for enabling a controlling computing device connected to an interconnecting network to control specified functionality of a controllable computing device connected to the interconnecting network, according to an aspect of the present invention;
FIG. 6 shows schematically an exemplary system for creation, modification and distribution of user interfaces, according to an aspect of the present invention;

FIG. 7 is a flow chart showing an exemplary method for defining a user interface for controlling, from a controlling computing device connected to a network, specified functionality of a controllable computing device connected to the network, according to an aspect of the present invention;

FIG. 8 is a flow chart showing an exemplary method for making user interfaces available, according to an aspect of the present invention;

FIG. 9 is a schematic representation of an exemplary smartphone, which may be used in implementing aspects of the present invention; and

FIG. 10 is a schematic representation of an exemplary computer system, which may be used in implementing aspects of the present invention.

DETAILED DESCRIPTION

[0022] Referring now to Figure 1, an exemplary embodiment of a system for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network is shown generally at 100. The system 100 comprises a controlling computing device 102, a controllable computing device 104, and a connection management server 106. The controlling computing device 102 is connected to a first local network 105 and the controllable computing device 104 is connected to a second local network 107; the controlling computing device 102 and the controllable computing device 104 are not directly connected to the same local network and are not directly connected to each other. The controlling computing device 102 is connected, via the first local network 105, to an interconnecting network 108, and the controllable computing device 104 is also connected, via the second local network 107, to the interconnecting network 108. The connection management server 106 is also connected to the interconnecting network 108. As used herein, the term "interconnecting network" refers to a network that enables communication between remote computing devices that are not directly
connected to one another and are not connected to one another via a local network. An interconnecting network is typically a network of networks, and in the illustrated embodiment the interconnecting network 108 is the Internet.

[0023] The controlling computing device 102 may be any suitable device, such as a desktop computer, laptop computer, network-capable tablet computer, network-capable smartphone, or the like. Preferably, the controlling computing device 102 is a network-capable tablet computer or network-capable smartphone having a touch-screen display that can receive tactile input from a user. Connection of the controlling computing device 102, controllable computing device 104 and connection management server 106 to their respective local networks 105, 107 and thereby to the Internet 108 may be either wired or wireless, and in the case where the controlling computing device 102 is a network-capable tablet computer or network-capable smartphone, the connection between the controlling computing device 102 and the interconnecting network 108 is preferably wireless, for example by way of a WiFi connection or a 3G or 4G connection.

[0024] At least one unique identifier 110 is presented at the controllable computing device 104. Typically, the unique identifier 110 is assigned by the connection management server 106, and typically communicated from the connection management server 106 to the controllable computing device 104 via the Internet 108. As explained in greater detail below, the connection management server 106 uses the unique identifier 110 to facilitate a connection 112 between the controllable computing device 104 and the controlling computing device 102. The unique identifier 110 may be, for example, a URL an alphanumeric code, a WiFi beacon, a QR code, a bar code or an e-mail address, and is used to uniquely identify the controlled device 104. The unique identifier 110 is presented independently of any user interaction with the controllable computing device 104, that is, a user who wishes to control the controllable computing device 104 does not have to do anything to cause the unique identifier 110 to be presented. For example, where the unique identifier 110 is an alphanumeric code, a QR code, a bar code or an e-mail address, the unique identifier 110 may be presented on a display of the controllable computing device 104, or may be presented at a location in close proximity to the controllable computing device 104, such as on a wall.
adjacent to a display thereof, and may be a non-electronic display. Where the unique identifier 110 is a WiFi beacon, the WiFi beacon will broadcast without any action by the user who seeks to control the controllable computing device 104.

[0025] The unique identifier 110 is received by the controlling computing device 102, and then transmitted by the controlling computing device 102 to the connection management server 106. Where the unique identifier 110 is an alphanumeric code or e-mail address, it may be input into the controlling computing device 102 manually by way of a keyboard or touchscreen display. Where the unique identifier 110 is a QR code or a bar code, it may be scanned using a camera (built-in or peripheral) on the controlling computing device 102. In addition, where the unique identifier 110 is an alphanumeric code or e-mail address, it may be scanned using a camera and then interpreted using optical character recognition (OCR) software. Where the unique identifier 110 is a WiFi beacon, the controlling computing device 102 will automatically receive the WiFi beacon as long as the controlling computing device 102 has suitable wireless network capability.

[0026] When the connection management server 106 receives the unique identifier 110 from the controlling computing device 102, the connection management server 106 then uses the unique identifier 110 to facilitate a connection 112 between the controllable computing device 104 and the controlling computing device 102. For example, the connection management server 106 may use the unique identifier 110 to obtain the required network information about the controllable computing device 104 from a lookup table. Although the connection 112 between the controllable computing device 104 and the controlling computing device 102 may pass through the connection management server 106, preferably the connection 112 between the controllable computing device 104 and the controlling computing device 102 is one which, once established, bypasses the connection management server 106. In a particularly preferred embodiment, the connection 112 between the controllable computing device 104 and the controlling computing device 102 is a socket connection via the Internet. The socket connection may be, for example, a web socket connection, a scripted socket connection such as a socket, or other suitable socket connection. The connection 112 may use suitable encryption algorithms.
Generally speaking, the term "socket" refers to an endpoint of a bidirectional inter-process communication flow across an Internet Protocol-based computer network, such as the Internet 108, and a socket connection refers to such a communication flow. With an Internet socket, an application programming interface (API) for the TCP/IP protocol stack is provided, usually by the operating system, as a mechanism for delivering incoming data packets to the appropriate application process or thread, based on a combination of local and remote IP addresses and port numbers. Each socket is mapped by the operating system to a communicating application process or thread. A socket address is the combination of an IP address (the location of the computer) and a port (which is mapped to the application program process) into a single identity. A web socket (or "WebSocket") is a technology providing for bi-directional, full-duplex communications channels over a single Transmission Control Protocol (TCP) socket, and while designed to be implemented in web browsers and web servers, can be used by any client or server application.

In addition to facilitating the connection 112 between the controllable computing device 104 and the controlling computing device 102, the connection management server 106 also uses the unique identifier 110 to identify and deliver to the controlling computing device 102 a user interface 114 for controlling the controllable computing device 104. The connection management server 106 may deliver the user interface 114 either before or after facilitating the connection 112 between the controllable computing device 104 and the controlling computing device 102, but preferably does so after facilitating the connection 112. In an alternative embodiment, the user interface 114 may be sent to the controlling computing device 102 by the controllable computing device 104, rather than the connection management server 106, after the connection 112 between the controllable computing device 104 and the controlling computing device 102 has been established. The user interface 114 will typically be displayed on a screen of the controlling computing device 102, for example to receive input commands from a touch-screen display, and may also include audio features such as audio output and microphone input (e.g. voice commands) and hardware input devices such as keys or buttons. Optionally, the user interface 114 may be stored on the controlling computing device 102 for later reuse.
[0029] Now referring to Figure 2, an exemplary method for enabling a controlling computing device (e.g. controlling computing device 102) connected to an interconnecting network (e.g. the Internet 108) to control a controllable computing device (e.g. controllable computing device 104) is shown generally at 200. The method 200 may be carried out by a connection management server (e.g. connection management server 106).

[0030] At step 202, the connection management server receives, from the controlling computing device, a unique identifier for the controllable computing device, and at step 204 the connection management server uses the unique identifier to facilitate a connection between the controllable computing device and the controlling computing device. At step 206, which may be performed either before or after step 204, the connection management server uses the unique identifier to identify a user interface for controlling the controllable computing device and delivers the user interface to the controlling computing device.

[0031] Reference is now made to Figures 3A and 3B, which illustrate in more detail a particular exemplary implementation of the system shown generally in Figure 1, and corresponding reference numerals are used to refer to corresponding features in Figure 3, except with the prefix "3" instead of "1". The Internet continues to be denoted by reference numeral 108.

[0032] In the exemplary implementation shown in Figures 3A and 3B, a controllable computing device browser application 320 is running on the controllable computing device 304, and a controlling computing device browser application 321, which supports web sockets, is running on the controlling computing device 302.

[0033] The controllable computing device browser application 320 is specialized, and is configured, for example by the operator of the controllable computing device 304, to display a shell web page 322. The shell web page 322 acts as a frame that displays a target web page 323, which typically includes desired content, and the shell web page 322 also displays the unique identifier 310 associated with the controllable computing device 304. The unique identifier 310 will typically be obtained from the connection management server 306. The
unique identifier 310 is aligned to the controllable computing device 304, and does not change when the target web page 323 changes. Optionally, the connection management server 306 may be sensitive to the target web page 323, and may select the user interface 314 to send to the controlling computing device 302 from a plurality of user interfaces 314 based on the then-current target web page 323.

[0034] As shown in Figure 3A, to initiate the procedure for the controlling computing device 302 to obtain control over the specified functionality on the controllable computing device 304, a user of the controlling computing device 302 will cause the controlling computing device browser application 321 to send a request 324 to the connection management server 306. For example, the user may manually enter a URL pointing to the connection management server 306, scan a QR code or execute a link. The connection management server 306 will return a control initiation web page 326 to the controlling computing device browser application 321. The control initiation web page 326 receives the unique identifier 310. For example, the user may manually enter the unique identifier 310 into a form field on the control initiation web page 326, or the controlling computing device browser application 321 may support automatic filling of the form field from a scanned barcode, QR code, or other scanned unique identifier 310.

[0035] Referring now to Figure 3B, after the unique identifier 310 has been provided to the control initiation web page 326, the user would then cause the controlling computing device browser application 321 to send a further request 328 to the connection management server 306, which further request 328 contains the unique identifier 310 or information derived therefrom. In response to the further request 328, the connection management server 306 uses the unique identifier 310, either directly or by using information derived from the unique identifier 310, to identify the appropriate user interface web page 330 for the controllable computing device 304 corresponding to that unique identifier 310. The connection management server 306 then returns the identified user interface web page 330 to the controlling computing device 302. The user interface web page 330 contains the user interface 314, which in this embodiment is an HTML page encoded for receiving touch-screen input. Optionally, the connection management server 306 may detect whether the controlling
computing device 302 is a touch-screen device, for example by way of the screen resolution, and the user interface 314 may comprise an overlay of touch-actuable coordinates on the HTML page. Since the user interface 314 is predefined, the user interface 314 may be limited to a predefined set of commands associated with specified functionality on the controllable computing device 302. The specified functionality may be limited, for example, to a specific application, part of a specific application, or to a particular file or type of file. The user interface web page 330 also contains a socket identifier 332, which in this embodiment is HTML code for creating a web socket connection 312 with the controllable computing device 304.

[0036] In an alternative embodiment, the user may send a single request to the connection management server 306, which single request contains the unique identifier 310 or information derived therefrom, and the connection management server 306 may immediately return the user interface web page 330, without any need to use the control initiation web page 326. For example, a QR code may provide a URL that points directly to the user interface web page 330.

[0037] Continuing to refer to Figure 3B, the controllable computing device browser application 320 is configured to allow a socket connection to be received by the shell web page 322, and therefore loading the user interface web page 330 into the controlling computing device browser application 321 effects the socket connection 312 to the shell web page 322 by execution of the browser-executable code comprising the socket identifier 332. Thus, the connection management server 306 uses the unique identifier 310 to facilitate a connection between the controllable computing device 304 and the controlling computing device 302 by providing the user interface web page 330 containing the socket identifier 332 to the controlling computing device browser application 321.

[0038] With the socket connection 312 established, by touching elements of the user interface 314 defined in the user interface web page 330 displayed within the controlling computing device browser application 321, the user can send control signals 334 from the controlling computing device 302 through the socket connection 312 to the controllable computing device
In the embodiment illustrated in Figure 3, the controllable computing device browser application 320 is configured to allow the control signals 334 to pass through to the shell web page 322, which includes a communication layer 336, in the form of HTML code, which receives the control signals 334 from the controlling computing device 302, interprets them, and passes the commands indicated by the control signals 334 to the target web page 323.

Figures 3C and 3D show an alternative embodiment 300C of the system shown in Figures 3A and 3B, in which like elements are denoted by like reference numerals except with the suffix "C". The alternative embodiment 300C shown in Figures 3C and 3D is similar to the embodiment 300 shown in Figures 3A and 3B, except that controllable computing device browser application 320C is modified to enable the shell web page 322 (Figures 3A and 3B) to be eliminated, and the target web page 323C is displayed in the main frame of the controllable computing device browser application 320C. In such an embodiment, the controllable computing device browser application 320C is modified to display the unique identifier 3IOC, to directly receive the socket connection 312C from the controlling computing device 302C and to include the communication layer 336C that receives and interprets the control signals 334C and passes the corresponding commands to the target web page 323C.

Figures 3E and 3F show a further alternative embodiment of the system shown in Figures 3A and 3B, in which like elements are denoted by like reference numerals except with the suffix "E". In the alternative embodiment 300E shown in Figures 3E and 3F, rather than providing a shell web page 322 (Figures 3A and 3B) or configuring the controllable computing device browser application 320 (Figures 3A and 3B) to receive the socket connection 312 (Figures 3C and 3D), the controllable computing device 304E is provided with a separate socket application 340E for receiving the socket connection 312E from the controlling computing device 302E. The separate socket application 340E may display the unique identifier 310E, and includes an appropriate communication layer 336E that receives and interprets the control signals 334E and passes the corresponding commands. As best seen in Figure 3F, the socket application 340E may be given permission to pass commands to certain other applications 342E or parts thereof, thereby enabling the controlling computing
device 302E to have limited control over the controllable computing device 304E, or may be
given permission to pass commands to any application 342E, as well as to the operating
system 344E, so that the controlling computing device 302E will have virtually complete
control over the controllable computing device 304E. In such an embodiment, the socket
application 340E may be sensitive to the particular application 342E that is active, and the
user interface 314E sent to the controlling computing device 302E may depend on the active
application 342E, and a new user interface 314E may be automatically sent to the controlling
computing device 302E when the active application 342E changes.

[0041] Although the operating system 344E is shown only in Figures 3E and 3F, it will be
appreciated in the embodiments shown in Figures 3A to 3D as well, and is omitted from those
figures only for ease of illustration.

[0042] As noted above, in particular embodiments a WiFi beacon may be used as a unique
identifier. Two exemplary such embodiments will now be described.

[0043] A first exemplary embodiment 300G of a system in which a WiFi beacon is used as a
unique identifier is shown in Figures 3G and 3H, in which elements corresponding to those in
Figures 3A to 3F are denoted by corresponding reference numerals, except with the suffix
"G". Referring first to Figure 3G, the controlling computing device browser application 321G
is configured to provide location information 350G, such as from a GPS system or
triangulation, to the connection management server 306G. The connection management
server 306G checks the location information 350G against a master list 352G of controllable
computing devices 304G to determine which controllable computing devices 304G are within
a predetermined proximity to the controlling computing device 302G, based on the location
information 350G. The connection management server 306G then returns a proximity list
354G to the controlling computing device browser application 321G, as well as a control
initiation web page 326G. The proximity list 354G identifies the controllable computing
devices 304G are within a predetermined proximity to the controlling computing device
302G, and also provides information about the WiFi networks associated with those
controllable computing devices 304G. The information included in the proximity list 354G
may include information enabling hidden WiFi networks corresponding to the controllable computing devices 304G on the proximity list 354G to be detected. The controlling computing device browser application 321G then listens for the WiFi beacons 356G corresponding to the controllable computing devices 304G on the proximity list 354G, and presents the user with a selection list 357G of those controllable computing devices 304G on the proximity list 354G whose corresponding WiFi beacons were detected.

[0044] Referring now to Figure 3H, the user can then select one of the controllable computing devices 304G on the selection list 357G, and the controlling computing device browser application 321G will send the selection 360G to the connection management server 306G, which uses the WiFi beacon 356G indicated by the selection 360C as the unique identifier to determine and then return to the controlling computing device browser application 321G the user interface web page 330G to establish the socket connection 312G to the controllable computing device 304G corresponding to the WiFi beacon 356G indicated by the selection 360G. By restricting the proximity list 354G to controllable computing devices 304G that are within a predetermined proximity to the controlling computing device 302G, location-based limitations or "geo-gating" may be applied where desirable.

[0045] Figures 3I and 3J show a second exemplary embodiment 3001 of a system in which a WiFi beacon is used as a unique identifier. In Figures 3I and 3J, elements corresponding to those in Figures 3A to 3H are denoted by corresponding reference numerals, except with the suffix "I".

[0046] Referring first to Figure 3I, the controlling computing device browser application 321I listens for WiFi beacons 356I, and then transmits a beacon list 358I of detected WiFi beacons 356I to the connection management server 306I. The connection management server 306I checks the beacon list 358I against a master list 352I of controllable computing devices 304I to determine which of the detected WiFi beacons 356I correspond to controllable computing devices 304I available for control by the controlling computing device 302I. The connection management server 306I then returns a control initiation web page 326I to the controlling computing device browser application 321I. The control initiation web page 326I includes an
availability list 3531, which identifies the controllable computing devices 3041 whose WiFi
beacons 3561 were detected by the controlling computing device browser application 3211 and
were identified from the master list 3521 as being available for control by the controlling
computing device 3021, and the availability list 3531 is presented to the user. As shown in
Figure 3J, the user can select one of the controllable computing devices 3041 on the
availability list 3531, and the controlling computing device browser application 3211 will send
the selection 3601, which identifies the corresponding WiFi beacon 3561 serving as the unique
identifier, to the connection management server 3061. The connection management server
3061 then returns the user interface web page 3301 to the controlling computing device
browser application 3211 to establish the socket connection 3121.

[0047] The proximity list 354G (Figure 3G) and the availability list 3531 (Figure 3I) each
preferably include confirmation codes, such as alphanumeric codes, which confirmation codes
are also displayed on the controllable computing devices 304G, 3041, to enable the user to
verify that the user is obtaining control over the desired controllable computing device 304G,
3041 prior to sending the selection 360G, 3601 to the connection management server 306G,
3061.

[0048] It should be noted here that in the exemplary embodiments 300G, 3001 described
above, the controlling computing device 302G, 3021 does not actually connect to the wireless
network of the relevant controllable computing device 304G, 3041, and as such these wireless
networks may remain secured. The connection between the controlling computing device
302G, 3021 and the relevant controllable computing device 304G, 3041 is a socket connection
via the Internet 108. It will also be noted that the controlling computing device browser
application 321G, 3211 may of course cooperate with the operating system or other
applications on the controlling computing device 3021, 302G to obtain location information
350G and to detect the WiFi beacons 356G, 3561.

[0049] Additionally, it should be noted that the controlling computing device 302, 302C,
302E, 302G, 3021 may have a dedicated application for connecting to the respective
controllable computing device 304, 304C, 304E, 304G, 3041, rather than a browser
application 321, 321C, 321E, 321G, 321I. In such cases a socket utilizing the Internet would be provided, but need not necessarily comprise a complete web page.

[0050] Referring now to Figure 4, an alternative configuration of a system for enabling a controlling computing device connected to an interconnecting network to control specified functionality of a controllable computing device connected to the interconnecting network is shown generally at 400. The alternatively configured system 400 is similar to the system 300 shown in Figure 3, except that the alternatively configured system 400 does not include a connection management server, with the controllable computing device 404 being configured to act as a server to provide a user interface web page 430 to the controlling computing device 402. Corresponding reference numerals are used to refer to features in the alternatively configured system 400 that correspond to features in the system 300 shown in Figure 3, except with the prefix "4" instead of "3".

[0051] At least one unique identifier 410 is presented at the controllable computing device 404, independently of any user interaction with the controllable computing device 404. In the exemplary system 400 shown in Figure 4, the at least one unique identifier 410 contains sufficient information to enable the controlling computing device 402 to generate a request 470 to the controllable computing device 404, acting as a server, for the user interface web page 430 corresponding to the controllable computing device 404. The unique identifier 410 may be, for example, a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code or an e-mail address. Typically, the unique identifier 410 will be a URL from a web connection to the socket connection 412, via the Internet. Where the unique identifier is a WiFi beacon, the network name may be set to the relevant URL so that the URL can be captured by the controllable computing device browser application 422.

[0052] The user interface web page 430 contains a user interface 414, for example an HTML page encoded for receiving touch-screen input, and also contains a socket identifier 432, for example HTML code for creating a web socket connection with the controllable computing device 404. The controllable computing device browser application 420 is configured to allow a socket connection to be received by the shell web page 422. Thus, loading the user
interface web page 430 into the controlling computing device browser application 421 effects the socket connection 412 to the shell web page 422 by execution of the HTML code comprising the socket identifier 432.

[0053] Figures 4A and 4B show, respectively, alternative embodiments 400A and 400B of the alternatively configured system 400 which, aside from omission of the connection management server and the controlling computing device acting as a server, are similar to the alternative embodiments 300C and 300E shown in Figures 3C and 3D and 3E and 3F, respectively, and hence corresponding reference numerals are used to refer to corresponding features, only with the prefix "4" rather than "3" and the suffix "A" and "B" rather than "C" and "E", respectively.

[0054] Figure 5 is a flow chart showing an exemplary method 500 for enabling a controlling computing device (such as controlling computing device 402, 402A, 402B) connected to an interconnecting network such as the Internet to control specified functionality of a controllable computing device (such as controllable computing device 404, 404A, 404B) connected to the interconnecting network. At step 502, at least one unique identifier for establishing a connection between the controllable computing device and the controlling computing device is presented at the controllable computing device, independently of any user interaction with the controllable computing device. At step 504, the controllable computing device receives a connection request from the controlling computing device based on the unique identifier. At step 506, in response to the connection request, the unique identifier is used to establish a connection between the controllable computing device and the controlling computing device via the interconnecting network, and at step 506, the controllable computing device identifies and delivers a user interface to the controlling computing device for controlling the controllable computing device.

[0055] Systems according to aspects of the present invention for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the network facilitate a wide variety of practical applications.
The control and interactivity provided by the user interface may be layered over existing content.

[0056] In one embodiment, the controllable computing device may be a user's personal computer, and the controlling computing device may be that same user's mobile computing device, such as a tablet computer or smartphone. By obtaining a user interface in accordance with aspects of the present invention, the user's mobile device can become a specialized control peripheral, in addition to the keyboard and mouse, for controlling applications on the user's personal computer. The user interface may, as described above, be specialized for the particular application being controlled.

[0057] In another embodiment, a restaurant or a store may place a monitor inside their retail location, or in a window. The monitor may be coupled to a computer connected to an interconnecting network, such as the Internet, and appropriately configured as described above so that the computer can be a controllable computing device as described above. For example, the computer may cause the connected monitor to display a web browser displaying a shell web page that shows the unique identifier and frames, as the target web page, a web page for the store or restaurant. A customer or potential customer could then use an Internet-connected mobile device, such as a smartphone, to log onto the connection management server, enter the unique identifier, and then receive a user interface enabling the user to interact with the target web page displayed on the monitor. For example, in the case of a restaurant the user could interact with the target web page to obtain nutritional information about food sold by the restaurant, and in the case of a store the user could interact with the target web page to make inquiries into the store catalogue. In each case, the results would be displayed on the monitor, although additional results may be shown on the mobile device, along with the user interface. Similarly, a mall display may take the form of a large monitor coupled to an Internet-connected computer configured to be a controllable computing device, enabling a user to use an Internet-connected smartphone to obtain an interface for interacting with the mall display to obtain detailed directions, including GPS-based or WiFi triangulation based directions, and to call up web sites for a specific store in the mall. Again, the results of the interaction would be displayed on the large monitor in the mall display, although
additional results may be shown on the mobile device display. In certain embodiments, coupons or other promotional material can be delivered to the user's mobile device. Thus, a wide variety of advertising options are enabled by aspects of the present invention.

[0058] In addition, in certain embodiments the user interface on the controlling computing device can enable the user to provide content to the controllable computing device. For example, an Internet-connected computer whose monitor is displaying a poster for a movie, when configured as a controllable computing device according to an aspect of the present invention, may enable a user to take a photograph of him or herself, such as with a built-in camera on his or her smartphone, and insert the photograph into a predetermined location in the poster. A similar arrangement can be used in social venues such as nightclubs to enable a user to announce his or her arrival by displaying a photo of him or herself on a large screen above the dance floor or other mixing area.

[0059] Systems according to aspects of the present invention may be adapted to support control of specified functionality on the controllable computing device by more than one controlling computing device. Such adaptation is within the capability of one skilled in the art, now informed by the herein disclosure. For example, the communications from each controlling computing device may carry a controlling computing device identifier for uniquely identifying that controlling computing device.

[0060] In one exemplary implementation, the controllable computing device is running a video game, such as poker, and is coupled to a monitor which provides a main display of the game. Users can use their Internet-connected smartphones as controlling computing devices to play the game, and the users can view their cards on the local screen of their respective smartphones without the cards being visible on the monitor. Similarly, aspects of the present invention may be used to enable multi-player video gaming in a venue such as a movie theatre, with users' smartphones serving as game controllers. In such embodiments, use of the systems illustrated in Figures 4, 4A and 4B, in which the connection management server is omitted, may be more efficient than having each user's mobile device separately communicate with a connection management server.
[0061] In some embodiments, the controllable computing device may be an Internet-connected video game console, such as a Sony PlayStation 3, Nintendo Wii, Microsoft Xbox 360, or the like, or a conventional computer running a video game. In such embodiments, the required functionality may be provided by the console hardware or firmware, or by the software of a particular game, or may be specialized software. The controlling computing device may be, for example, a touch-screen device such as an Internet-connected tablet computer or smartphone, which may be used to replace an existing game controller, or as a supplement to an existing game controller, for example by physically attaching the tablet computer or smartphone to the controller using a specialized clip. The user interface on the controlling computing device may include an additional game display, and may provide for a single action on the user interface to send input to the controllable computing device that is equivalent to a complex series of inputs on the original game controller. Depending on the control options set for the user interface on the controlling computing device, specialized in-game features, such as special in-game equipment, may be provided via the user interface on the controlling computing device.

[0062] Aspects of the present invention may also support interactive polling, for example in movie theatres, or in a classroom setting to facilitate examinations. Feedback, for example rewards such as electronic coupons in the case of commercial polling, may be pushed to the controlling computing device.

[0063] Aspects of the present invention are also directed to enabling users to create, modify and socialize user interfaces for various applications.

[0064] Referring now to Figure 6, an exemplary system for creation, modification and distribution of user interfaces is shown generally at 600. The system 600 comprises a user interface distribution server 606 connected to an interconnecting network, in this case the Internet 108, and a plurality of computing devices 680, 682 also connected to the Internet 108. The computing devices 680, 682 may be connected to the Internet 108 via respective local networks 681.
The user interface distribution server 606 stores a library 684 of predefined user interfaces 686. The user interfaces 686 in the library 684 may be associated with various different software applications, and may include a subset of different user interfaces 686 associated with a single software application; that is, for a given software application such as photo editing software, spreadsheet software or video game software, the library 684 may include a plurality of different user interfaces 686 each associated with that particular software application.

The user interface distribution server 606 is accessible via the Internet 108 by the computing devices 680, 682 and hence accessible to a plurality of parties, and the user interfaces 686 may be made available for download to at least some of those computing devices 680, 682 and hence made available to at least some of those parties. In response to a request 688 from an authorized one of the computing devices 680 for a particular one of the user interfaces 686, the user interface distribution server 606 transmits that particular user interface 686 to the requesting computing device 680. The system 600 for creation, modification and distribution of user interfaces may be integrated with the above-described systems for enabling a controlling computing device to control a controllable computing device, and the interface distribution server 606 may also be a connection management server as described above. Thus, the user interfaces 686 in the library 684 may enable control, from a controlling computing device connected to the Internet 108, of specified functionality of a controllable computing device also connected to the Internet 108. Accordingly, the request 688 may be one of the requests 328, 328C, 328E or the selections 360G, 3601 described above in the context of the systems 300, 300C, 300E, 300G, 3001, and the user interface 686 transmitted to the controlling computing device 300, 300C, 300E, 300G, 3001 may be the user interface 314, 314C, 314E, 314G, 3141 contained within a user interface web page 330, 330C, 330E, 330G, 3301 transmitted to the controlling computing device 302, 302C, 302E, 302G, 3021 to enable the controlling computing device 302, 302C, 302E, 302G, 3021 to control the controllable computing device 304, 304C, 304E, 304G, 3041.

Various applications, such as commercial software applications like word processors, photo editors, spreadsheets and video games, and various files, such as a particular company's
web site, may have default user interfaces associated therewith, typically created by the provider of the application or file or by an associated party. Users may be permitted to modify these default user interfaces, or create their own user interfaces based on their personal preferences. Typically, even where users are permitted to create or modify user interfaces where a default user interface exists, the default user interface would retain the primary association with the relevant application or file.

[0068] In the exemplary system 600 shown in Figure 6, at least one of the computing devices 682 has user interface editing software 690 installed thereon. The user interface editing software 690 enables a user of the computing device 682 to create a new user interface or to modify an existing user interface. The user interface editing software 690 can receive one of the user interfaces 686 from the user interface distribution server 606, and can transmit a newly created or modified user interface 686 to the user interface distribution server 606 for storage in the library 684. Thus, the user interface distribution server 606 can receive additional user interfaces 686 from third parties, which may be new user interfaces 686 or modified user interfaces 686, and add those additional user interfaces 686 to the library 684. Preferably, the user interface distribution server 606 includes a validator 691, such as a javascript program, which, before adding an additional user interface 686 to the library 684, validates the additional user interface 686, and only adds the additional user interface 686 to the library 684 in response to successful validation of the additional user interface 686. The validator 691 may check for errors in the user interface 686 that would interfere with its effective functionality, or may check to see whether the author or submitter of the user interface 686 is authorized to create or modify user interfaces for the application or content with which that user interface 686 is associated. For example, a company may have created a specific user interface 686 for use with that company's web site so as to provide a customized, branded browsing experience, and therefore may wish to prevent third parties from placing new or modified user interfaces for that web site on the user interface distribution server 606. In such a case, where the validator 691 detects that a user interface 686 submitted for that company's web site was not submitted by a party authorized by that company, the validator 691 would prevent that user interface 686 from being stored on the user interface distribution
server 606 and return a suitable message to the submitter. Alternatively, the validator 691 could allow that user interface 686 to be stored as a non-default user interface. In other embodiments, all user interfaces 686 may be accepted into the library 684, but only those user interfaces 686 that were successfully validated would be made available to parties other than the submitter.

[0069] As noted above, in one embodiment the user interfaces 686 in the library 684 may enable control, from a controlling computing device connected to an interconnecting network such as the Internet 108, of specified functionality of a controllable computing device also connected to the interconnecting network. Accordingly, the user interface editing software 690 enables a user thereof to define such a user interface 686. The user interface editing software 690 receives control input 692 comprising associations 694 between actions 695 on the controlling computing device and commands 696 for controlling the specified functionality of the controllable computing device, and then stores the associations 694, typically along with the actions 695 and commands 696, as part of a user interface 686. The user interfaces 686 may be stored using a suitable markup language such as XML, and may be stored locally by the computing device 682 and then transmitted to the user interface distribution server 606, or may only be retained in non-persistent memory on the computing device 682, with the persistent storage of the user interface 186 being on the user interface distribution server 606.

[0070] The commands 696 with which the actions 695 on the controlling computing device are associated may comprise individual inputs such as keystrokes, button presses, joystick movements or mouse clicks, or combinations of such individual inputs. For example, in a user interface 686 for a video game, a single action 695 on the controlling computing device may be mapped to an ordered sequence of button presses on a game controller, enabling a single command on the controlling computing device to execute a complicated technique on the controllable computing device executing the video game software. In one embodiment, where a "special move" in a video game requires a precise series of several inputs on the game controller, a user may build a user interface 686 for that video game in which a single action 695, such as touching a single touch-actuable widget, maps to commands 696 corresponding
to that precise series of inputs on the game controller. As a result, a single action on the controlling computing device can execute the "special move" in the video game. Similar techniques can be used to allow a single action on the controlling computing device to execute more complicated, multi-command processes in other types of software, such as word processing software, spreadsheet software, photo editing software, and so on. The commands with which the actions on the controlling computing device are associated may also comprise individual functions of an application on the controllable computing device, such as "cut", "paste" or "increase font", or may comprise ordered combinations of individual functions. Where a user interface is to be used with an application that runs on an Apple Macintosh computer, the commands with which the actions on the controlling computing device are associated may include AppleScripts.

[0071] The actions on the controlling computing device may comprise button presses or combinations of button presses on the hardware of the controlling computing device, and where the controlling computing device includes an in-built accelerometer, the actions can also include the signals from the accelerometer. In a preferred embodiment, the user interfaces are adapted for use with controlling computing devices that are touch-screen devices. In such an embodiment, the actions on the controlling computing device that are associated with the commands on the controllable computing device will comprise actuation of touch-actuable widgets, and the user interface editing software also receives positional input establishing a positional layout of the touch-actuable widgets on a screen of the touch-screen device and stores the positional layout as part of the user interface. The touch-actuable widgets are individual touch controllers, such as buttons, sliders, rotators and virtual joysticks, and can be assigned various desired shapes, icons, colors, sizes, and so on. Individual touch-actuable widgets may be assembled into groups having a defined layout based on a common purpose or related functions, and the groups can be assembled (with or without additional individual touch-actuable widgets) according to a defined layout into "boards" occupying all or part of the screen on the touch-screen device. A user interface may consist of only a single board, or may comprise a plurality of boards, with the board displayed by the controlling computing device varying according to the context of the application or file.
being controlled. The positional layout of a given board may be mapped to the shape of a human hand, and may then be customized to a particular user's hand size and natural finger position. Thus, the look, feel and even the auditory feedback of a user interface 686 may be customized.

[0072] As noted above, various applications and files may have associated therewith default user interfaces 686 created by the provider of the application or file or by an associated party. These default user interfaces 686 may be distributed with the application or file, or may be distributed via the user interface distribution server 606. In some cases, the provider may wish to prevent any modification of that user interface 686, in which case that user interface 686 could be marked "no editing", and the user interface editing software 690 could be configured to prevent editing of such default user interfaces 686. In other cases, the provider may wish to permit users to edit the default user interface 686 for their own use but prevent distribution of such edited user interfaces 686. In such instances, the validator 691 would prevent any edited version of that default user interface 686 from being stored on the user interface distribution server 606.

[0073] In still other cases, the provider of an application or file may have no objection to modification of the default user interface 686 or to distribution of such modified user interfaces 686 or to the creation and distribution of new user interfaces for the provider's application or files. Thus, users who are not affiliated with the provider of an application or file may create or modify user interfaces 686 and upload them to the user interface distribution server 606 to make the user interfaces 686 available to other users. In order to do so, a user may be required to register with the user interface distribution server 606, which may be configured to charge a fee for registration or for the download of certain user interfaces 686, and users may optionally also be required to register with the user interface distribution server 606 in order to download user interfaces 686. The user interface distribution server 606 may track the consanguinity of the various user interfaces 686, such as that a particular user interface 686 is user X's modification of user Y's modification of a default user interface 686. Users may facilitate distribution of user interfaces by sending tags, URLs or other links to particular user interfaces 686 on the user interface distribution server 606 via e-mail or social
media. Additionally or alternatively, the user interface distribution server 606 may support browsing and searching of available user interfaces 686, which may be sorted by any suitable method, such as by associated application or file, alphabetically by name or creator, by user popularity or user ratings, or the like.

[0074] Reference is now made to Figure 7, which is a flow chart showing an exemplary method 700 for defining a user interface for controlling, from a controlling computing device connected to an interconnecting network, specified functionality of a controllable computing device connected to the interconnecting network. The method 700 may be carried out by a computing device (such as computing device 682) running user interface editing software (such as user interface editing software 690), either alone or in combination with a user interface distribution server (such as user interface distribution server 606). The method 700 may be applied to create a new user interface or to modify an existing user interface. At step 702, the method 700 receives control input associating actions on the controlling computing device with commands associated with the specified functionality of the controllable computing device, and at step 704 the method 700 stores the actions, associations and commands as part of a user interface. At optional step 706, where the controlling computing device is a touch-screen device, the method 700 receives positional input establishing a positional layout of touch-actuable widgets on a screen of the touch-screen device and at optional step 708, the method 700 stores the positional layout as part of the user interface. The steps 702 to 708 may be performed in any suitable order, and may be performed repeatedly as a user interface is created or modified. The resulting user interface may be stored on a server accessible by a plurality of parties, with the user interface being made available to some or all of the parties.

[0075] With reference now to Figure 8, an exemplary method 800 for making user interfaces available is shown in flow chart form. The method 800 may be used, for example, to make available user interfaces that enable control, from a controlling computing device connected to an interconnecting network, of specified functionality of a controllable computing device connected to the interconnecting network. The method 800 may be carried out by a user interface distribution server, such as the user interface server 606 described above.
At step 802, the method 800 stores a library of predefined user interfaces. At step 804, the method 800 receives an additional user interface from another computing device, typically from a third party. At optional step 806, before adding the additional user interface to the library (step 810), the method 800 validates the additional user interface, for example by checking functionality or by checking authorship or modification permission as described above. If the validation is unsuccessful ("no" at step 806), the method 800 sends an error message 808 and proceeds to step 812, and if the validation is successful ("yes" at step 806), at step 810 the method 800 adds the additional user interface to the library and typically indexes it for searching. At step 812, the method 800 receives a request for a particular one of the user interfaces, for example from a controlling computing device as described above, and at step 814, in response to the request received at step 810, the method 800 transmits the requested user interface to the requestor, for example to a controlling computing device. At step 816, the method 800 receives a modified user interface, which is a modified version of the requested user interface transmitted at step 814, and at step 818, before adding the modified user interface to the library (step 820), the method 800 validates the modified user interface. If the validation is unsuccessful ("no" at step 818), the method 800 sends an error message 822 and then ends. In response to successful validation ("yes" at step 818), at step 820 the method 800 adds the additional user interface to the library and typically indexes it. The method 800 shown in Figure 8 is merely an exemplary method for making user interfaces available, and other methods are also contemplated. For example, steps 804 to 810 may be omitted, and steps 816 to 820 may be omitted, or steps 804 to 810 may be carried out after step 814 and before step 816, or after step 820. Certain sets of steps may also be repeated.

One aspect of the present invention provides a management tool for sharing at least one user interface. The management tool monitors a computing device to determine an opened application and checks to see if the opened application has an associated user interface. If the management tool determines that the opened application has an associated user interface, the management tool then determines if the associated user interface is available locally. If the management tool determines that an associated user interface is not available locally, the management tool may prompt the user to allow download of an
associated user interface from a proprietary third party content aggregator, for example by way of a user interface distribution server, or prompt the user to create a new user interface for the opened application.

[0078] Another aspect of the present invention provides a method for defining a touch-screen user interface for a touch-screen device. The method comprises associating the touch-screen user interface with at least one application and/or at least one document supported by the application, and then receiving user input establishing at least one positional layout of touch-actuable widgets, receiving user input establishing assignments of the touch-actuable widgets to respective application functions, operating system functions and/or device inputs such as keystrokes and mouse clicks, and bundling the widgets, the at least one positional layout, and the assignments as part of the touch-screen user interface. The touch-screen user interface may be published to a third party content aggregator server, such as a user interface distribution server, which may permit downloading of the touch-screen user interface. Upon uploading the touch-screen user interface to the third party content aggregator server, the third party content aggregator server may execute a validation routine on the uploaded touch-screen user interface, and only publish the touch-screen user interface in response to successful validation thereof.

[0079] A further aspect of the present invention provides a method of displaying a touch-screen user interface on a touch-screen device. The method detects the launch of an application or opening of a document or other content element on a computing device with which the touch-screen device can communicate, and checks whether either the computing device or the touch-screen device stores at least one touch-screen user interface that is associated with the application, document or other content element. If the method finds such a touch-screen user interface, the method automatically causes the stored touch-screen user interface to be launched on the touch-screen device. If the method does not find such a touch-screen user interface, the method checks to see if a remote publication server stores a touch-screen user interface that is associated with the application, document or other content element and, if one is found, facilitates downloading of that touch-screen user interface for loading onto the touch-screen device. The method may permit a user to select one of a
plurality of touch-screen user interfaces associated with the application, document or other content element. The method may also permit the user to provide input modifying the downloaded touch-screen user interface and store the modified touch-screen user interface, and may further permit the user to publish the modified touch-screen user interface and its association with the application, document or other content element, for example by upload to a publication server, and permit download of the modified touch-screen user interface, either by the user or by third parties, or both. The method may automatically store a parent-child relationship between the originally downloaded touch-screen user interface and the modified touch-screen user interface.

[0080] Another aspect of the present invention provides a method for editorial control of a user interface. According to the method, a first party creates a user interface and provides it to a third party content aggregator, which may be a proprietary third party content aggregator. The third party content aggregator may allow a second party to download the user interface for editing, thereby creating a modified user interface. Depending on the authorizations of the second party, the second party may provide the modified user interface to the third party content aggregator.

[0081] Another aspect of the present invention provides a method for controlling a controlled device using a controlling device, such as a touch-screen device. The method monitors a user interface on the touch device, and notifies a controller in the event a touch is detected on the touch device. The touch is mapped to an event, such as an application function or series of functions, or a hardware input or series of inputs such as keystrokes or mouse clicks, and the method executes the event mapped to the touch on the controlled device.

[0082] A further aspect of the present invention is directed to a method comprising executing an application on a server computer, sending application data wirelessly to a client device, executing a corresponding touch-screen user interface associated with the application on the client device, with the touch-screen user interface providing at least one application function for controlling the application remotely.
[0083] An additional aspect of the present invention provides a method for controlling a client device configured to display a touch-screen user interface. The method comprises monitoring a server to determine an application being executed by the server, detecting an open file associated with the application, where the open file may have a touch-screen user interface associated therewith, and checking the client device to determine if the touch-screen user interface associated with the open file is being displayed on the client device. If the method determines that no touch-screen user interface associated with the open file is being displayed on the client device, the method opens the touch-screen user interface associated with the open file on the client device or prompts the user to create a new touch-screen user interface for the open file.

[0084] Another aspect of the present invention is directed to a method for communicating between a client and a server. The method comprises sending a server message from the server to the client, with the server message indicating a server protocol version, and sending a client message from the client to the server, with the client message indicating a client protocol version. The method compares the client protocol version to the server protocol version, and if the client protocol version is different from the server protocol version, the method updates the lower of the client and server protocol versions.

[0085] Aspects of the present invention may be implemented on any suitable computer or microprocessor-based system, and the controlling computing device is preferably a network-capable tablet computer or smartphone.

[0086] Figure 9 shows an exemplary network-capable smartphone 900. The smartphone 900 includes a display 902, an input device in the form of keyboard 904 and an onboard computer system 906. The display 902 may be a touch-screen display and thereby serve as an additional input device, in which case the keyboard 904 may be omitted. The onboard computer system 906 comprises a central processing unit (CPU) 910 having one or more processors or microprocessors for performing arithmetic calculations and control functions to execute software stored in an internal memory 912, preferably random access memory (RAM) and/or read only memory (ROM) is coupled to additional memory 914 which will typically comprise
flash memory, which may be integrated into the smartphone 900 or may comprise a removable flash card, or both. The smartphone 900 also includes a communications interface 916 which allows software and data to be transferred between the smartphone 900 and external systems and networks. The communications interface 916 is coupled to one or more wireless communication modules 924, which will typically comprise a wireless radio for connecting to one or more of a cellular network, a wireless digital network or a Wi-Fi network. The communications interface 916 will also typically enable a wired connection of the smartphone 900 to an external computer system. A microphone 926 and speaker 928 are coupled to the onboard computer system 906 to support the telephone functions managed by the onboard computer system 906, and GPS receiver hardware 922 is also coupled to the communications interface 916 to support navigation operations by the onboard computer system 906. An accelerometer 930 for detecting tilting and movement of the smartphone 900 is also coupled to the onboard computer system 906. Input and output to and from the onboard computer system 906 is administered by the input/output (I/O) interface 918, which administers control of the display 902, keyboard 904, microphone 926 and speaker 928. The onboard computer system 906 may also include a separate graphical processing unit (GPU) 920. The various components are coupled to one another either directly or by coupling to suitable buses.

[0087] An illustrative computer system in respect of which aspects of the present invention may be implemented, for example as a controlling computing device or controllable computing device, is presented as a block diagram in Figure 10. The illustrative computer system is denoted generally by reference numeral 1000 and includes a display 1002, input devices in the form of keyboard 1004A and pointing device 1004B, computer 1006 and external devices 1008. While pointing device 1004B is depicted as a mouse, it will be appreciated that other types of pointing device may also be used.

[0088] The computer 1006 may contain one or more processors or microprocessors, such as a central processing unit (CPU) 1010. The CPU 1010 performs arithmetic calculations and control functions to execute software stored in an internal memory 1012, preferably random access memory (RAM) and/or read only memory (ROM), and possibly additional memory
1014. The additional memory 1014 may include, for example, mass memory storage, hard disk drives, optical disk drives (including CD and DVD drives), magnetic disk drives, magnetic tape drives (including LTO, DLT, DAT and DCC), flash drives, program cartridges and cartridge interfaces such as those found in video game devices, removable memory chips such as EPROM or PROM, emerging storage media, such as holographic storage, or similar storage media as known in the art. This additional memory 1014 may be physically internal to the computer 1006, or external as shown in Figure 10.

[0089] The computer system 1000 may also include other similar means for allowing computer programs or other instructions to be loaded. Such means can include, for example, a communications interface 1016 which allows software and data to be transferred between the computer system 1000 and external systems and networks. Examples of communications interface 1016 can include a modem, a network interface such as an Ethernet card, a wireless communication interface, or a serial or parallel communications port. Software and data transferred via communications interface 1016 are in the form of signals which can be electronic, acoustic, electromagnetic, optical or other signals capable of being received by communications interface 1016. Multiple interfaces, of course, can be provided on a single computer system 1000.

[0090] Input and output to and from the computer 1006 is administered by the input/output (I/O) interface 1018. This I/O interface 1018 administers control of the display 1002, keyboard 1004A, external devices 1008 and other such components of the computer system 1000. The computer 1006 also includes a graphical processing unit (GPU) 1020. The latter may also be used for computational purposes as an adjunct to, or instead of, the (CPU) 1010, for mathematical calculations.

[0091] The various components of the computer system 1000 are coupled to one another either directly or by coupling to suitable buses.

[0092] Embodiments of aspects of the present invention may be implemented entirely in hardware, entirely in software, or by way of a combination of hardware and software. In a
preferred embodiment, the invention is implemented in software, which includes but is not limited to firmware, resident software, microcode, and the like. Furthermore, the invention can take the form of a computer program product accessible from a non-transitory computer usable or computer readable medium providing program code for use by or in connection with a computer or any instruction execution system. In such embodiments, the computer program product may reside on a computer usable or computer readable medium in a computer such as the memory 912 of the onboard computer system 906 of the smartphone 900 or the memory 1012 of the computer 1006, or on a computer usable or computer readable medium external to the onboard computer system 906 of the smartphone 900 or the computer 1006, or on any combination thereof.

[0093] Several currently preferred embodiments have been described by way of example. It will be apparent to persons skilled in the art that a number of variations and modifications can be made without departing from the scope of the invention as defined in the claims.
WHAT IS CLAIMED IS:

1. A method for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network, the method comprising:

receiving at a connection management server, from the controlling computing device, at least one unique identifier for the controllable computing device; and

the connection management server using the unique identifier to facilitate a connection between the controllable computing device and the controlling computing device via the interconnecting network.

2. The method of claim 1, further comprising:

the connection management server delivering to the controlling computing device a user interface for controlling the controllable computing device.

3. The method of claim 1, wherein the connection is a socket connection via the Internet.

4. The method of claim 1, wherein the at least one unique identifier is at least one of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address.

5. A computer program product comprising a non-transitory computer usable medium embodying computer usable program code for enabling a controlling computing device connected to an interconnecting network to control a controllable computing device connected to the interconnecting network, the computer program product comprising:
computer usable program code for receiving at a connection management server, from the controlling computing device, at least one unique identifier for the controllable computing device; and

computer usable program code for the connection management server using the unique identifier to facilitate a connection between the controllable computing device and the controlling computing device via the interconnecting network.

6. The computer program product of claim 5, further comprising:

computer usable program code for delivering to the controlling computing device from the connection management server a user interface for controlling the controllable computing device.

7. The computer program product of claim 5, wherein the connection is a socket connection via the Internet.

8. The computer program product of claim 5, wherein the at least one unique identifier is at least one of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address.

9. A connection management server connected to an interconnecting network for enabling a controlling computing device connected to the interconnecting network to control a controllable computing device connected to the interconnecting network, comprising:

a memory for storing instructions;
a processing unit coupled to the memory for executing the instructions stored in the memory, wherein the instructions, when executed by the processing unit, cause the connection management server to:

receive, from the controlling computing device, at least one unique identifier for the controllable computing device; and

use the unique identifier to facilitate a connection between the controllable computing device and the controlling computing device via the interconnecting network.

10. The connection management server of claim 9, wherein the instructions, when executed by the processing unit, further cause the connection management server to:

deliver to the controlling computing device a user interface for controlling the controllable computing device.

11. The connection management server of claim 9, wherein the connection is a socket connection via the Internet.

12. The connection management server of claim 9, wherein the at least one unique identifier is at least one of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address.

13. A method for enabling a controlling computing device connected to an interconnecting network to control specified functionality of a controllable computing device connected to the interconnecting network, the method comprising:
presenting, at the controllable computing device, at least one unique identifier for establishing a connection between the controllable computing device and the controlling computing device;

the unique identifier being presented independently of any user interaction with the controllable computing device;

receiving, at the controllable computing device, a connection request from the controlling computing device, the connection request being based on the unique identifier;

responsive to the connection request, establishing a connection between the controllable computing device and the controlling computing device via the interconnecting network using the unique identifier; and

the controllable computing device identifying and delivering to the controlling computing device a user interface for controlling the specified functionality of the controllable computing device.

14. The method of claim 13, wherein the at least one unique identifier is at least one of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address.

15. The method of claim 13, wherein the connection is a socket connection via the Internet.

16. The method of claim 15, wherein the unique identifier describes a URL from a web connection to the socket connection.
17. The method of claim 13, wherein the user interface is limited to a predefined set of commands associated with the specified functionality.

18. The method of claim 13, wherein the specified functionality is at least one of an application and a file.

19. A computer program product comprising a non-transitory computer usable medium embodying computer usable program code for enabling a controlling computing device connected to an interconnecting network to control specified functionality of a controllable computing device connected to the interconnecting network, the computer program product comprising:

   computer usable program code for presenting, at the controllable computing device, independently of any user interaction with the controllable computing device, at least one unique identifier for establishing a connection between the controllable computing device and the controlling computing device;

   computer usable program code for receiving, at the controllable computing device, a connection request from the controlling computing device, the connection request being based on the unique identifier;

   computer usable program code for, responsive to the connection request, establishing a connection between the controllable computing device and the controlling computing device via the interconnecting network using the unique identifier; and

   computer usable program code for identifying and delivering from the controllable computing device to the controlling computing device a user interface for controlling the specified functionality of the controllable computing device.
20. The computer program product of claim 19, wherein the at least one unique identifier is at least one of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address.

21. The computer program product of claim 19, wherein the connection is a socket connection via the Internet.

22. The computer program product of claim 21, wherein the unique identifier describes a URL from a web connection to the socket connection.

23. The computer program product of claim 19, wherein the user interface is limited to a predefined set of commands associated with the specified functionality.

24. The computer program product of claim 19, wherein the specified functionality is at least one of an application and a file.

25. A controllable computing device connected to an interconnecting network and having specified functionality controllable by a controlling computing device connected to the interconnecting network, comprising:

a memory for storing instructions;

a processing unit coupled to the memory for executing the instructions stored in the memory, wherein the instructions, when executed by the processing unit, cause the controllable computing device to:
present, at the controllable computing device, independently of any user interaction with the controllable computing device, at least one unique identifier for establishing a connection between the controllable computing device and the controlling computing device;

receive, at the controllable computing device, a connection request from the controlling computing device, the connection request being based on the unique identifier;

responsive to the connection request, establish a connection between the controllable computing device and the controlling computing device via the interconnecting network using the unique identifier; and

identify and deliver to the controlling computing device a user interface for controlling the specified functionality of the controllable computing device.

26. The controllable computing device of claim 25, wherein the at least one unique identifier is at least one of a URL, an alphanumeric code, a WiFi beacon, a QR code, a bar code and an e-mail address.

27. The controllable computing device of claim 25, wherein the connection is a socket connection via the Internet.

28. The controllable computing device of claim 27, wherein the unique identifier describes a URL from a web connection to the socket connection.
29. The controllable computing device of claim 25, wherein the user interface is limited to a predefined set of commands associated with the specified functionality.

30. The controllable computing device of claim 25, wherein the specified functionality is at least one of an application and a file.

31. A method for defining a user interface for controlling, from a controlling computing device connected to an interconnecting network, specified functionality of a controllable computing device connected to the interconnecting network, the method comprising:

receiving control input comprising associations between actions on the controlling computing device and commands for controlling the specified functionality of the controllable computing device; and

storing the actions, associations and commands as part of the user interface.

32. The method of claim 31, wherein:

the controlling computing device is a touch-screen device; and

the actions on the controlling computing device comprise actuation of touch-actuable widgets;

the method further comprising:

receiving positional input defining a positional layout of the touch-actuable widgets on a screen of the touch-screen device; and

storing the positional layout as part of the user interface.

33. The method of claim 31, wherein:
the user interface is a modified user interface; and

receiving control input associating actions on the controlling computing device with commands associated with the specified functionality of the controllable computing device comprises modification of an original user interface.

34. The method of claim 31, wherein the user interface is a new user interface.

35. The method of claim 31, wherein:

the user interface is stored on a server accessible by a plurality of parties; and

the user interface is made available to at least some of the plurality of parties.

36. A computer program product comprising a non-transitory computer usable medium embodying computer usable program code for defining a user interface for controlling, from a controlling computing device connected to an interconnecting network, specified functionality of a controllable computing device connected to the interconnecting network, the computer program product comprising:

computer usable program code for receiving control input comprising associations between actions on the controlling computing device and commands for controlling the specified functionality of the controllable computing device; and

computer usable program code for storing the actions, associations and commands as part of the user interface.
37. The computer program product of claim 36, wherein:

the controlling computing device is a touch-screen device; and

the actions on the controlling computing device comprise actuation of touch-actuable widgets;

the computer program product further comprising:

computer usable program code for receiving positional input defining a positional layout of
the touch-actuable widgets on a screen of the touch-screen device; and

computer usable program code for storing the positional layout as part of the user interface.

38. The computer program product of claim 36, wherein:

the user interface is a modified user interface; and

the computer usable program code for receiving control input associating actions on the
controlling computing device with commands associated with the specified functionality of
the controllable computing device comprises computer usable program code for modifying an
original user interface.

39. The computer program product of claim 36, wherein the computer usable program
code for receiving control input associating actions on the controlling computing device with
commands associated with the specified functionality of the controllable computing device
comprises computer usable program code for creating a new user interface.

40. The computer program product of claim 36, further comprising:
computer usable program code for storing the user interface on a server accessible by a plurality of parties; and

computer usable program code for making the user interface available to at least some of the plurality of parties.

41. A computing device for defining a user interface for controlling, from a controlling computing device connected to an interconnecting network, specified functionality of a controllable computing device connected to the interconnecting network, comprising:

a memory for storing instructions;

a processing unit coupled to the memory for executing the instructions stored in the memory, wherein the instructions, when executed by the processing unit, cause the computing device to:

receive control input comprising associations between actions on the controlling computing device and commands for controlling the specified functionality of the controllable computing device; and

store the actions, associations and commands as part of the user interface.

42. The computing device of claim 41, wherein:

the controlling computing device is a touch-screen device; and

the actions on the controlling computing device comprise actuation of touch-actuable widgets; and

wherein the instructions, when executed by the processing unit, further cause the computing device to:
receive positional input defining a positional layout of the touch-actuable widgets on a screen of the touch-screen device; and

store the positional layout as part of the user interface.

43. The computing device of claim 41, wherein:

the user interface is a modified user interface; and

receiving control input associating actions on the controlling computing device with commands associated with the specified functionality of the controllable computing device comprises modification of an original user interface.

44. The computing device of claim 41, wherein the user interface is a new user interface.

45. The computing device of claim 41, wherein the instructions, when executed by the processing unit, further cause the computing device to:

store the user interface on a server accessible by a plurality of parties; and

make the user interface available to at least some of the plurality of parties.

46. A method of making user interfaces available, comprising:

storing a library of predefined user interfaces; and

responsive to a request for a particular one of the user interfaces, transmitting that particular user interface to the requestor.
47. The method of claim 46, further comprising:

receiving additional user interfaces from third parties; and

adding the additional user interfaces to the library.

48. The method of claim 47, further comprising:

before adding the additional user interfaces to the library, validating the additional user interfaces; and

adding the additional user interfaces to the library only in response to successful validation of the additional user interfaces.

49. The method of claim 46, wherein the user interfaces enable control, from a controlling computing device connected to an interconnecting network, of specified functionality of a controllable computing device connected to the interconnecting network; and

responsive to the request for the particular user interface, the particular user interface is transmitted to the controlling computing device.

50. The method of claim 46, wherein the user interfaces are associated with different software applications.

51. The method of claim 46, wherein the user interfaces include a subset of different user interfaces associated with a single software application.
52. The method of claim 46, further comprising:

receiving a modified user interface;

the modified user interface comprising a modified version of the particular user interface transmitted in response to the request; and

adding the modified user interface to the plurality of predefined user interfaces.

53. A computer program product comprising a non-transitory computer usable medium embodying computer usable program code for making user interfaces available, the computer program product comprising:

computer usable program code for storing a library of predefined user interfaces; and

computer usable program code for, responsive to a request for a particular one of the user interfaces, transmitting that particular user interface to the requestor.

54. The computer program product of claim 53, further comprising:

computer usable program code for receiving additional user interfaces from third parties; and

computer usable program code for adding the additional user interfaces to the library.

55. The computer program product of claim 53, further comprising:

computer usable program code for validating the additional user interfaces before adding the additional user interfaces to the library; and
computer usable program code for adding the additional user interfaces to the library only in response to successful validation of the additional user interfaces.

56. The computer program product of claim 53, wherein the user interfaces enable control, from a controlling computing device connected to an interconnecting network, of specified functionality of a controllable computing device connected to the interconnecting network, further comprising:

computer usable program code for, responsive to the request for the particular user interface, transmitting the particular user interface to the controlling computing device.

57. The computer program product of claim 53, wherein the user interfaces are associated with different software applications.

58. The computer program product of claim 53, wherein the user interfaces include a subset of different user interfaces associated with a single software application.

59. The computer program product of claim 53, further comprising:

computer usable program code for receiving a modified user interface;

the modified user interface comprising a modified version of the particular user interface transmitted in response to the request; and

computer usable program code for adding the modified user interface to the plurality of predefined user interfaces.
60. A user interface distribution server for making user interfaces available, comprising:

- a memory for storing instructions;
- a processing unit coupled to the memory for executing the instructions stored in the memory, wherein the instructions, when executed by the processing unit, cause the user interface distribution server to:

  - store a library of predefined user interfaces; and
  - responsive to a request for a particular one of the user interfaces, transmit that particular user interface to the requestor.

61. The user interface distribution server of claim 60, wherein the instructions, when executed by the processing unit, further cause the computing device to:

- receive additional user interfaces from third parties; and
- add the additional user interfaces to the library.

62. The user interface distribution server of claim 61, wherein the instructions, when executed by the processing unit, further cause the computing device to:

- before adding the additional user interfaces to the library, validate the additional user interfaces; and
- add the additional user interfaces to the library only in response to successful validation of the additional user interfaces.
63. The user interface distribution server of claim 60, wherein the user interfaces enable control, from a controlling computing device connected to an interconnecting network, of specified functionality of a controllable computing device connected to the interconnecting network; and

wherein the instructions, when executed by the processing unit, further cause the computing device to, responsive to the request for the particular user interface, transmit the particular user interface to the controlling computing device.

64. The user interface distribution server of claim 60, wherein the user interfaces are associated with different software applications.

65. The user interface distribution server of claim 60, wherein the user interfaces include a subset of different user interfaces associated with a single software application.

66. The user interface distribution server of claim 60, wherein the instructions, when executed by the processing unit, further cause the computing device to:

   receive a modified user interface;

   the modified user interface comprising a modified version of the particular user interface transmitted in response to the request; and

   add the modified user interface to the plurality of predefined user interfaces.
START

Receive, from controlling computing device, unique identifier for controllable computing device

Use unique identifier to facilitate connection between controllable computing device and controlling computing device

Identify and deliver user interface for controlling controllable computing device to controlling computing device

END

FIG. 2
Present at least one unique identifier for establishing connection between controllable computing device and controlling computing device at controllable computing device, independently of any user interaction with controllable computing device.

Receive, at controllable computing device, connection request based on unique identifier from controlling computing device.

Use unique identifier to establish connection between controllable computing device and controlling computing device via interconnecting network.

Identify and deliver user interface for controlling controllable computing device to controlling computing device.

FIG. 5
Receive control input associating actions on controlling computing device with commands associated with specified functionality of controllable computing device

Store actions, associations and commands as part of user interface

Receive positional input establishing positional layout of touch-actuable widgets on screen of touch-screen device

Store positional layout as part of user interface

FIG. 7
START

Store library of predefined user interfaces

Receive additional user interface from another computing device

Validation successful?
Yes
Add additional user interface to library

Error

No

Receives request for particular user interface

Transmit requested user interface to requestor

Receive modified user interface

Validation successful?
Yes
Add modified user interface to library

END
FIG. 9 (PRIOR ART)
A. CLASSIFICATION OF SUBJECT MATTER

IPC: H04L 12/56 (2006.01), G06F 3/00 (2006.01), H04L 12/24 (2006.01)

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: H04L 12/56 (2006.01), G06F 3/00 (2006.01), H04L 12/24 (2006.01)

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

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

TotalPatent, Esp@cenet, Canadian Patent Database, USPTO Database, IEEE Xplore.

Keywords: control, device, unique, identifier, interconnect, network, user, interface, connection, management, server, socket, url, wifi.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>US2009/0074184 (BAUM ET AL.) 19 March 2009 -abstract; -see abstract; -see paragraphs [0043]-[0047], [0181]-[0183], [0204]-[0205], [0238] and [0262]-[0269]; -see figures 1-12.</td>
<td>1-30</td>
</tr>
<tr>
<td>A</td>
<td>US2006/0271695 (LAVIAN, Yoel) 30 November 2006 -see entire document.</td>
<td>1-30</td>
</tr>
</tbody>
</table>

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

Date of the actual completion of the international search

22 December 2011 (22-12-2011)

Date of mailing of the international search report

23 December 2011 (23-12-2011)

Name and mailing address of the ISA/CA

Authorized officer

Canadian Intellectual Property Office

Hassan Bayaa (819) 997-7810

Place du Portage I, C1 14 - 1st Floor, Box PCT

Gatineau, Quebec K1A 0C9

50 Victoria Street

Facsimile No.: (011)-819-953-2476

Form PCT/ISA/210 (second sheet) (My 2009)
**INTERNATIONAL SEARCH REPORT**

**Box No. II**  
**Observations where certain claims were found unsearchable (Continuation of item 2 of the first sheet)**

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1.  [ ] Claim Nos.:
   because they relate to subject matter not required to be searched by this Authority, namely:

2.  [ ] Claim Nos.:
   because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3.  [ ] Claim Nos.:
   because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

**Box No. III**  
**Observations where unity of invention is lacking (Continuation of item 3 of first sheet)**

This International Searching Authority found multiple inventions in this international application, as follows:

See supplemental sheet.

1.  [ ] As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2.  [ ] As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.

3.  [ ] As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claim Nos.:

4.  [X] No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claim Nos.:

   **Remark on Protest**  
   [ ] The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

   [ ] The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

   [ ] No protest accompanied the payment of additional search fees.
Lack of Unity

The claims are directed to a plurality of inventive concepts as follows:

Group A - Claims 1-30 are directed to a method and system for enabling control of a computing device connected to an interconnecting network wherein a connection management server uses a unique identifier for the device to be controlled.

Group B - Claims 31-45 are directed to a method and system for defining a user interface for controlling a device connected to the network and receiving control input.

Group C - Claims 46-66 are directed to a method for storing interfaces in a library and accessing a particular user interface being requested.

The claims must be limited to one inventive concept as set out in Rule 13 of the PCT.
<table>
<thead>
<tr>
<th>Patent Document</th>
<th>Publication Date</th>
<th>Patent Family</th>
<th>Publication Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>CA2559842A1</td>
<td>29 September 2005</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP1738540A2</td>
<td>03 January 2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP1738540A4</td>
<td>13 October 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP2188794A1</td>
<td>26 May 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP2191351A1</td>
<td>02 June 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP2327063A1</td>
<td>01 June 2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB0620362D0</td>
<td>29 November 2006</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2428621A</td>
<td>07 February 2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2428621B</td>
<td>04 June 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB0724248D0</td>
<td>23 January 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2442628A</td>
<td>09 April 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2442628B</td>
<td>24 September 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB0724760D0</td>
<td>30 January 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2442633A</td>
<td>09 April 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2442633B</td>
<td>24 September 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB8000400D0</td>
<td>13 February 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2442640A</td>
<td>09 April 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB2442640B</td>
<td>06 August 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US771179EB2</td>
<td>04 May 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2008180240A1</td>
<td>31 July 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US7911341B2</td>
<td>22 March 2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009070436A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US8073931B1</td>
<td>06 December 2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2005216302A1</td>
<td>29 September 2005</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2005216580A1</td>
<td>29 September 2005</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2007286210A1</td>
<td>13 December 2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2008183842A1</td>
<td>21 July 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009066788A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009066789A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009070473A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009070477A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009070681A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009070682A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009070692A1</td>
<td>12 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009077167A1</td>
<td>19 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009077622A1</td>
<td>19 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009077623A1</td>
<td>19 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009077624A1</td>
<td>19 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009134998A1</td>
<td>28 May 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009138600A1</td>
<td>28 May 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2009138958A1</td>
<td>28 May 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US200916514A1</td>
<td>25 June 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2010023885A1</td>
<td>28 January 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2010052612A1</td>
<td>04 March 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2010082744A1</td>
<td>01 April 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US201009511A1</td>
<td>15 April 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2010095369A1</td>
<td>15 April 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2010153853A1</td>
<td>17 June 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2010245107A1</td>
<td>30 September 2010</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US201102171A1</td>
<td>05 May 2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US2011128378A1</td>
<td>02 June 2011</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO2005091218A2</td>
<td>29 September 2005</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO2005091218A3</td>
<td>27 July 2006</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO2009023647A1</td>
<td>19 February 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO2009029590A1</td>
<td>05 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO2009029597A1</td>
<td>05 March 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO2009064795A1</td>
<td>22 May 2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO2010019624A1</td>
<td>18 February 2010</td>
</tr>
</tbody>
</table>