CROSS-PLATFORM GAMING BETWEEN MULTIPLE DEVICES OF MULTIPLE TYPES

Inventors: Keren Kang, Austin, TX (US); Jeff Lujan, Austin, TX (US); Hoan Vu, Austin, TX (US)

Assignee: DIGITAL HARMONY GAMES, INC., Austin, TX (US)

Appl. No.: 13/214,196
Filed: Aug. 21, 2011

Publication Classification

Int. Cl. A63F 9/24 (2006.01)

ABSTRACT

Cross-platform gaming between multiple devices of multiple types, including: determining, by a cross-platform gaming server, the device type of a first gaming device; sending, in dependence upon the device type of the first gaming device, a gaming application from the cross-platform gaming server to the first gaming device; determining, by the cross-platform gaming server, the device type of the second gaming device; sending, in dependence upon the device type of the second gaming device, a gaming application from the cross-platform gaming server to the second gaming device; and communicating, in real-time, between the gaming application on the first gaming device and the gaming application on the second gaming device.
Determine The Device Type Of One Or More Gaming Devices 304

Send, To Each Of The One Or More Gaming Devices, A Gaming Application From The Cross-platform Gaming Server To The Gaming Device In Dependence Upon The Device Type Of The Gaming Device 306

Gaming Application 312
First Gaming Device 316

Gaming Application 314
Second Gaming Device 318

FIG. 2
Cross-Platform Gaming Server 302

Determine The Device Type Of One Or More Gaming Devices 304

Send, To Each Of The One Or More Gaming Devices, A Device Configuration Scanner 402

Receive, From Each Of The One Or More Gaming Devices, Scan Results 404

Send, To Each Of The One Or More Gaming Devices, A Gaming Application From The Cross-platform Gaming Server To The Gaming Device In Dependence Upon The Device Type Of The Gaming Device 306

Gaming Application 312

Gaming Application 314

First Gaming Device 316

Second Gaming Device 318

Dev. Config. Scanner 406

Scan Results 410

First Gaming Device 316

Second Gaming Device 318

Dev. Config. Scanner 408

Scan Results 412

FIG. 3
Cross-Platform Gaming Server 302

Determine The Device Type Of One Or More Gaming Devices 304

Send, To Each Of The One Or More Gaming Devices, A Gaming Application 306

Exchange Gaming Information Between Gaming Devices 502

Receive Gaming Information From A First Gaming Device In A First Data Format 504

Translate The Gaming Information From The First Gaming Device In The First Data Format Into Gaming Information In A Second Data Format In Dependence Upon The Device Type Of The First Gaming Device And The Device Type Of A Second Gaming Device 506

Send, To The Second Gaming Device, The Gaming Information In The Second Data Format 508

FIG. 4
Cross-Platform Gaming Server 302

Determine The Device Type Of One Or More Gaming Devices 602

Identify A Host Gaming Device 604

Transmit Gaming Device Connection Information 606

Gaming Device Connection Information 608

Gaming Device Connection Information 610

Host Gaming Device 612

Client Gaming Device 614

Gaming Application 616

Gaming Application 618

Establish Peer-To-Peer Connection 620

Host Gaming Engine 622

FIG. 6
Cross-Platform Gaming Server 302

Determine The Device Type Of One Or More Gaming Devices 602
Send, To Each Of The One Or More Gaming Devices, A Device Configuration Scanner 702

Receive, From Each Of The One Or More Gaming Devices, Scan Results 704

Send, To Each Of The One Or More Gaming Devices, A Gaming Application In Dependence Upon The Device Type Of The Gaming Device 706

Identify A Host Gaming Device 604

Transmit Gaming Device Connection Information 606

Gaming Device Connection Information 608
Host Gaming Device 612
Gaming Application 616
Host Gaming Engine 622

Establish Peer-To-Peer Connection 620

Gaming Device Connection Information 610
Gaming Application 618
Client Gaming Device 614

FIG. 7
Cross-Platform Gaming Server 302

Determine the device type of one or more gaming devices 602

Send, to each of the one or more gaming devices, a gaming application 706

Identify a host gaming device 604

Transmit gaming device connection information 606

Exchange gaming information between at least two gaming devices 801

Receive gaming information from a host gaming device in a first data format 802

Translate the gaming information from the host gaming device in the first data format into gaming information in a second data format 804

Send, to the client gaming device, the gaming information in the second data format 806
CROSS-PLATFORM GAMING BETWEEN MULTIPLE DEVICES OF MULTIPLE TYPES

BACKGROUND OF THE INVENTION

1. Field of the Invention
The field of the invention is data processing, or, more specifically, methods, apparatus, and products for cross-platform gaming between multiple devices of multiple types.

2. Description of Related Art
Computerized gaming is a rapidly developing use of modern computing resources. As computing resources have evolved, the number computing platforms capable of supporting computerized gaming applications has increased drastically. For example, mobile communications devices, mobile computing systems, dedicated gaming consoles, and a host of other technologies are currently available to support computerized gaming applications. Modern computerized gaming platforms, however, are frequently incompatible with each other and prohibit gaming users from interacting with each other in real-time over distinct gaming platforms.

SUMMARY OF THE INVENTION

Methods, apparatus, and products for cross-platform gaming between multiple devices of multiple types, including determining, by a cross-platform gaming server, the device type of a first gaming device; sending, in dependence upon the device type of the first gaming device, a gaming application from the cross-platform gaming server to the first gaming device; determining, by the cross-platform gaming server, the device type of the second gaming device; sending, in dependence upon the device type of the second gaming device, a gaming application from the cross-platform gaming server to the second gaming device; and communicating, in real-time, between the gaming application on the first gaming device and the gaming application on the second gaming device.

The foregoing and other objects, features and advantages of the invention will be apparent from the following more particular descriptions of example embodiments of the invention as illustrated in the accompanying drawings wherein like reference numbers generally represent like parts of example embodiments of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 sets forth a block diagram of automated computing machinery comprising an example computer useful in cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 2 sets forth a flow chart illustrating an example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 3 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 4 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 5 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 6 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 7 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 8 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 9 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.

FIG. 10 sets forth a block diagram illustrating an example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention.
(104), but many components of such software typically are stored in non-volatile memory such as, for example, on a disk drive.

[0019] The example cross-platform gaming server (102) of FIG. 1 includes a communications adapter (110) for data communications with other devices through data communications network (100). Such data communications may be carried out serially through RS-232 connections, through external buses such as a Universal Serial Bus (‘USB’), through data communications networks such as IP, data communications networks, and in other ways as will occur to those of skill in the art. Communications adapters implement the hardware level of data communications through which one computer sends data communications to another computer, directly or through a data communications network. Examples of communications adapters useful for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention include modems for wired dial-up communications, Ethernet (IEEE 802.3) adapters for wired data communications network communications, and 802.11 adapters for wireless data communications network communications.

[0020] The first gaming device (316) of FIG. 1 includes at least one computer processor (112) or ‘CPU’ as well as RAM (126) which is connected through a high speed memory bus (124) and bus adapter (122) to the processor (112) and to other components of the first gaming device (316). Stored in RAM (126) is an operating system (128). Operating systems useful in cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention include standard operating systems such as UNIX™, Linux™, Microsoft XP™, AIX™, IBM’s i5/OS™, as well as operating systems for mobile devices such as iOS™, Android™ OS, and others as will occur to those of skill in the art.

[0021] Also stored in RAM (126) is a gaming application (312), a module of computer program instructions that, when executed by the first gaming device (316), presents an electronic or computerized game to a user of the first gaming device (316) that is played by manipulating images on a video display (116), for example, through the use of a gaming controller such as a keyboard, touch screen display, or special purpose gaming controller.

[0022] Also stored in RAM (126) is a gaming engine (140). The gaming engine (140) of FIG. 1 may be embodied, for example, as a software package for supporting and developing gaming applications. The gaming engine (140) may include, for example, a rendering engine for displaying graphics, a physics engine for collision detection, as well as other interfaces for sound, animation, networking, and so on by exposing an Application Programming Interface (‘API’) to a game developer that presents pre-packaged gaming modules to the game developer for use by the gaming application (312). The operating system (128), gaming application (312), and gaming engine (140) of FIG. 1 are shown in RAM (126), but many components of such software typically are stored in non-volatile memory such as, for example, on a disk drive (138).

[0023] The first gaming device (316) of FIG. 1 includes a disk drive adapter (136) coupled through an expansion bus (130) and a bus adapter (122) to the processor (112) and to other components of the first gaming device (316). The disk drive adapter (136) connects non-volatile data storage to the first gaming device (316) in the form of a disk drive (138). Disk drive adapters useful in gaming devices (316, 318) for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention include Integrated Drive Electronics (‘IDE’) adapters, Small Computer System Interface (‘SCSI’) adapters, and others as will occur to those of skill in the art. Non-volatile computer memory also may be implemented for as an optical disk drive, electrically erasable programmable read-only memory (so-called ‘EEPROM’ or ‘Flash’ memory), RAM drives, and so on, as will occur to those of skill in the art.

[0024] The example first gaming device (316) of FIG. 1 includes one or more input/output (‘I/O’) adapters (134). I/O adapters implement user-oriented input/output through, for example, software drivers and computer hardware for controlling output to display devices such as computer display screens, touch screen displays, as well as user input from user input devices such as a keyboard, mouse, touch screen display, dedicated gaming controller, and so on. The example first gaming device (316) of FIG. 1 includes a video adapter (118), which is an example of an I/O adapter specially designed for graphic output to a display (116) device such as a touch screen display, television, computer monitor, and so on. The video adapter (118) is connected to the processor (112) through a high speed video bus (120), bus adapter (122), and the front side bus (114), which is also a high speed bus.

[0025] The example first gaming device (316) of FIG. 1 includes a communications adapter (132) for data communications with other gaming devices (318) and the cross-platform gaming server (302) through a data communications network (100). Such data communications may be carried out serially through RS-232 connections, through external buses such as a USB, through data communications networks such as IP data communications networks, and in other ways as will occur to those of skill in the art. Communications adapters implement the hardware level of data communications through which one computer sends data communications to another computer, directly or through a data communications network. Examples of communications adapters useful for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention include modems for wired dial-up communications, Ethernet (IEEE 802.3) adapters for wired data communications network communications, 802.11 adapters for wireless data communications network communications, adapters configured to communicate over third generation (‘3G’) and fourth generation (‘4G’) mobile telecommunications networks, and others as will occur to those of skill in the art.

[0026] The second gaming device (318) of FIG. 1 includes at least one computer processor (142) or ‘CPU’ as well as RAM (156) which is connected through a high speed memory bus (154) and bus adapter (152) to the processor (142) and to other components of the second gaming device (318). Stored in RAM (156) is an operating system (158). Operating systems useful in cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention include standard operating systems such as UNIX™, Linux™, Microsoft XP™, AIX™, IBM’s i5/OS™, as well as operating systems for mobile devices such as iOS™, Android™ OS, and others as will occur to those of skill in the art.

[0027] Also stored in RAM (156) is a gaming application (314), a module of computer program instructions that, when executed by the second gaming device (318), presents an
electronic or computerized game to a user of the second gaming device (318) that is played by manipulating images on a video display (146), for example, through the use of a gaming controller such as a keyboard, touch screen display, or special purpose gaming controller.

[0028] Also stored in RAM (156) is a gaming engine (170). The gaming engine (170) of FIG. 1 may be embodied, for example, as a software package for supporting and developing gaming applications. The gaming engine (170) may include, for example, a rendering engine for displaying graphics, a physics engine for collision detection, as well as other interfaces for sound, animation, networking, and so on by exposing an API to a game developer that presents prepackaged gaming modules to the game developer for use by the gaming application (314). The operating system (158), gaming application (314), and gaming engine (170) of FIG. 1 are shown in RAM (156), but many components of such software typically are stored in non-volatile memory such as, for example, on a disk drive (168).

[0029] The second gaming device (318) of FIG. 1 includes a disk drive adapter (166) coupled through an expansion bus (160) and a bus adapter (152) to the processor (142) and to other components of the second gaming device (318). The disk drive adapter (166) connects non-volatile data storage to the second gaming device (318) in the form of a disk drive (168). Disk drive adapters useful in gaming devices (316, 318) for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention include IDE adapters, SCSI adapters, and others as will occur to those of skill in the art. Non-volatile computer memory also may be implemented for as an optical disk drive, electrically erasable programmable read-only memory (so-called 'EEPROM' or 'Flash' memory), RAM drives, and so on, as will occur to those of skill in the art.

[0030] The example second gaming device (318) of FIG. 1 includes one or more input/output ('I/O') adapters (164). I/O adapters implement user-oriented input/output through, for example, software drivers and computer hardware for controlling output to display devices such as computer display screens, touch screen displays, as well as user input from user input devices such as a keyboard, mouse, touch screen display, dedicated gaming controller, and so on. The example second gaming device (318) of FIG. 1 includes a video adapter (148), which is an example of an I/O adapter specially designed for graphic output to a display (146) device such as a touch screen display, television, computer monitor, and so on. The video adapter (148) is connected to the processor (142) through a high speed video bus (150), bus adapter (152), and the front side bus (144), which is also a high speed bus.

[0031] The example second gaming device (318) of FIG. 1 includes a communications adapter (162) for data communications with other gaming devices (316) and the cross-platform gaming server (302) through a data communications network (100). Such data communications may be carried out serially through RS-232 connections, through external buses such as a USB, through data communications networks such as IP data communications networks, and in other ways as will occur to those of skill in the art. Communications adapters implement the hardware level of data communications through which one computer sends data communications to another computer, directly or through a data communications network. Examples of communications adapters useful for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention include modems for wired dial-up communications, Ethernet (IEEE 802.3) adapters for wired data communications network communications, 802.11 adapters for wireless data communications network communications, adapters configured to communicate over 3G and 4G mobile telecommunications networks, and others as will occur to those of skill in the art.

[0032] For further explanation, FIG. 2 sets forth a flow chart illustrating an example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example method of FIG. 2 includes determining (304), by a cross-platform gaming server (302), the device type of one or more gaming devices (316, 318). In the example of FIG. 2, the cross-platform gaming server (302) is a module of automated computing machinery configured to facilitate real-time gaming between gaming devices (316, 318) of differing types. Real-time gaming, as the term is used in the present application, refers to gaming applications in which each player may play at any point during the duration of the game. Real-time gaming is contrasted to turn-by-turn gaming in which only a single player is allowed to play at a particular point in time.

[0033] In the example of FIG. 2, a gaming device (316, 318) is any device configured to execute a gaming application (312, 314). Gaming devices (316, 318) may be embodied, for example, as a mobile communications device, a mobile computing device, a personal computing device, a dedicated gaming console, or any other computing device configured to execute computer program instructions that comprise a gaming application (312, 314).

[0034] In the example of FIG. 2, the gaming device (316, 318) may be configured for data communications with input peripherals such as a keyboard, touch screen display, dedicated gaming controller, motion sensor, or other input device configured to receive user input for a gaming application. A gaming device (316, 318) may also be configured for data communications with output peripherals such as, for example, a display screen, audio rendering device, and so on. Although the input and output peripherals may be external to the gaming device (316, 318), readers will appreciate that gaming devices (316, 318) according to embodiments of the present application may include embedded input and output mechanisms such that the gaming device (316, 318) operates as a stand alone device.

[0035] In the example of FIG. 2, the gaming device (316, 318) is also capable of data communications with the cross-platform gaming server (302) over a data communications network. The gaming device (316, 318) may be configured to communicate with the cross-platform gaming server (302), for example, over an IP based data communications network such as a WAN, a LAN, the Internet, or a dedicated gaming network. The gaming device (316, 318) may alternatively be configured to communicate with the cross-platform gaming server (302) over a mobile communications network such as, for example, a telephone network, a 3G telecommunications network, a 4G telecommunications network, and others as will occur to those of skill in the art. Readers will appreciate that the gaming device (316, 318) may be configured to communicate with the cross-platform gaming server (302) over many other data communications networks not explicitly listed in the present application.

[0036] In the example method of FIG. 2, each gaming device (316, 318) is characterized by a device type. The
device type of a gaming device (316, 318) may be determined (304) based on, for example, an operating system installed on the gaming device (316, 318), a type of data communications networks that the gaming device (316, 318) may communicate over, input mechanisms used by the gaming device (316, 318), output mechanisms used by the gaming device (316, 318), any combination thereof, and so on.

[0037] The example method of FIG. 2 also includes sending (306), to each of the one or more gaming devices (316, 318), a gaming application (312, 314) from the cross-platform gaming server (302) to the gaming device (316, 318) in dependence upon the device type of the gaming device (316, 318). In the example method of FIG. 2, the gaming application (312, 314) may be embodied as a module of computer program instructions that, when executed by a gaming device (316, 318), presents an electronic or computerized game to a user of the gaming device (316, 318) that is played by manipulating images on a video display, for example, through the use of a gaming controller such as a keyboard, touch screen display, or special purpose gaming controller.

[0038] In the example of FIG. 2, each gaming application (312, 314) may be supported by a gaming engine that is executing on the particular gaming device (316, 318) that the gaming application (312, 314) resides upon. A gaming engine may be embodied, for example, as a software package for supporting and developing gaming applications.

[0039] The gaming engine may include, for example, a rendering engine for displaying graphics, a physics engine for collision detection, as well as other interfaces for sound, animation, networking, and so on by exposing an API to a game developer that presents pre-packaged gaming modules to the game developer for use by the gaming application (312, 314).

[0040] In the example of FIG. 2, a gaming application (312) is sent to the first gaming device (316) in dependence upon the device type of the first gaming device (316). Consider an example in which the device type of the first gaming device (316) is based on the operating system installed on the first gaming device (316). In such an example, a gaming application configured to execute on a gaming device with an Android™ operating system may be sent to the first gaming device (316) if the first gaming device (316) includes an Android™ operating system. Alternatively, if the first gaming device (316) includes an iOS™ operating system, a gaming application configured to execute on a gaming device with an iOS™ operating system may be sent to the first gaming device (316). In such an example, the computer program instructions included in a gaming application (312) configured to execute on a gaming device with an iOS™ operating system may be different than the computer program instructions included in a gaming application (312) configured to execute on a gaming device with an Android™ operating system. In other examples, the particular gaming application (312) that is sent to the first gaming device (316) may be based on the input devices used by the first gaming device (316), the output device used by the first gaming device (316), the screen resolution of a display screen used by the first gaming device (316), and so on. Furthermore, the particular gaming application that is sent to the first gaming device (316) may be based on any combination of device type attributes that may occur to those of skill in the art.

[0041] In the example of FIG. 2, a gaming application (314) is sent to the second gaming device (318) in dependence upon the device type of the second gaming device (318). Consider an example in which the device type of the second gaming device (318) is based on the operating system installed on the second gaming device (318). In such an example, a gaming application configured to execute on a gaming device with an Android™ operating system may be sent to the second gaming device (318) if the second gaming device (318) includes an Android™ operating system. Alternatively, if the second gaming device (318) included an iOS™ operating system, a gaming application configured to execute on a gaming device with an iOS™ operating system may be sent to the second gaming device (318). In such an example, the computer program instructions included in a gaming application (314) configured to execute on a gaming device with an iOS™ operating system may be different than the computer program instructions included in a gaming application (314) configured to execute on a gaming device with an Android™ operating system. In other examples, the particular gaming application (314) that is sent to the second gaming device (318) may be based on the input devices used by the second gaming device (318), the output device used by the second gaming device (318), the screen resolution of a display screen used by the second gaming device (318), and so on. Furthermore, the particular gaming application that is sent to the second gaming device (318) may be based on any combination of device type attributes that may occur to those of skill in the art.

[0042] In the example of FIG. 2, the gaming applications (312, 314) may be sent to the gaming devices (316, 318) over many types of data communications connections that implement many data communications protocols. For example, the gaming applications (312, 314) may be sent to the gaming devices (316, 318) over an IP based network, over a mobile communications network, via a direct link between the cross-platform gaming server (302) and the gaming devices (316, 318), and in other ways as will occur to those of skill in the art.

[0043] For further explanation, FIG. 3 sets forth a flow chart illustrating an example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example method of FIG. 3 is similar to the example method of FIG. 2, as it also includes determining (304) the device type of one or more gaming devices (316, 318) and sending (306), to each of the one or more gaming devices (316, 318), a gaming application (312, 314) in dependence upon the device type of the gaming device (316, 318).

[0044] In the example of method of FIG. 3, determining (304) the device type of the one or more gaming devices (316, 318) includes sending (402), from the cross-platform gaming server (302) to each of the one or more gaming devices (316, 318), a device configuration scanner (406, 408). The device configuration scanner (406, 408) of FIG. 3 is a module of computer program instructions that, when executed, identifies the hardware and software characteristics of the device that is executing the device configuration scanner (406, 408). The device configuration scanner (406, 408) may identify, for example, details about the CPU included in the device, details about the graphics processing unit (‘GPU’) included in the device, details about the particular operating system that is executing on the device, details about the display screen that is part of or connected to the device, details about input components that are part of or connected to the device, and so on. After the device configuration scanner (406, 408) has been received by a gaming device (316, 318), the gaming device (316, 318) may begin executing the device configuration scanner (406, 408).
In the example method of FIG. 3, determining (304) the device type of the one or more gaming devices (316, 318) also includes receiving (404), by the cross-platform gaming server (302) from each of the one or more gaming devices (316, 318), scan results (410, 412). In the example method of FIG. 3, scan results (410, 412) are generated by each device configuration scanner (406, 408) and utilized by the cross-platform gaming server (302) to identify the processing capabilities of the device that executed the device configuration scanner. The scan results (410, 412) may be embodied, for example, as a data structure that includes information identifying the type of CPU that is in the device, the type of GPU that is in the device, the operating system that is executing on the device, the version of the operating system, and so on.

For further explanation, FIG. 4 sets forth a flowchart illustrating an example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example method of FIG. 4 is similar to the example method of FIG. 2, as it also includes determining (304) the device type of one or more gaming devices (316, 318) and sending (306), to each of the one or more gaming devices (316, 318), a gaming application (312, 314) in dependence upon the device type of the gaming device (316, 318).

The example method of FIG. 4 also includes exchanging (502), by the cross-platform gaming server (302), gaming information (510, 512) between at least two gaming devices (316, 318). In the example method of FIG. 4, gaming information (510, 512) is data that is used by one or more gaming applications (312, 314) to present the gaming application (312, 314) to a user. Gaming information (510, 512) may include, for example, scoring information for each player, graphical information to be displayed by a particular gaming device (316, 318), audio information to be rendered by each gaming device (316, 318), user input received through a gaming controller of a gaming device (316, 318), and so on.

In the example method of FIG. 4, exchanging (502) gaming information (510, 512) between at least two gaming devices (316, 318) can include exchanging real-time gaming information between at least two gaming devices (316, 318). Real-time gaming information is gaming information used by gaming applications (312, 314) in which each player may play at any point during the duration of the game. Examples of real-time gaming applications can include, for example, combat based games in which players battle against each other in real-time. Real-time gaming is contrasted to turn-by-turn gaming in which only a single player is allowed to play at a particular point in time.

In the example method of FIG. 4, exchanging (502) gaming information (510, 512) between at least two gaming devices (316, 318) can include receiving (504), by the cross-platform gaming server (302), gaming information (510) from a first gaming device (316) in a first data format. The first data format of FIG. 4 may be characterized by data communications protocols, data structures, message structures, and so on that are exchanged between the first gaming device (316) and the cross-platform gaming server (302). The first data format may include, for example, data structures for encapsulating user input from the first gaming device (316), data structures for encapsulating display information on the first gaming device (316), and so on. For example, a gaming device that includes a touch screen display through which a player interacts with the gaming application will produce different user input data than a gaming device that has a keyboard through which a player interacts with the gaming application. As such, the data format of user input will be different for each type of gaming device.

In the example method of FIG. 4, exchanging (502) gaming information (510, 512) between at least two gaming devices (316, 318) can also include translating (506), by the cross-platform gaming server (302), the gaming information (510) from the first gaming device (316) in the first data format into gaming information (512) in a second data format. In the example method of FIG. 4, the gaming information (510) from the first gaming device (316) in the first data format is translated (506) into gaming information (512) in a second data format in dependence upon the device type of the first gaming device (316) and the device type of the second gaming device (318).

Consider an example in which the first gaming device (316) is a mobile communications device that includes a touch screen display through which a first player interacts with a gaming application (312) executing on the first gaming device (316). Assume that in the same example, the second gaming device (318) is a personal computer that includes a keyboard through which a second player interacts with a gaming application (314) executing on the second gaming device (318). In such an example, the first player may touch a particular region of the touch screen on the first gaming device (316) to perform a certain action (e.g., firing a weapon in the gaming application) while the second player may press a particular key on the keyboard of the second gaming device (318) to perform the same action. In such an example, the cross-platform gaming server (302) may receive (504) gaming information (510) in a first data format from the first gaming device (316) indicating that the first player has touched a particular region of the touch screen on the first gaming device (316) to perform the action of firing a weapon in the gaming application (312) that is executing on the first gaming device (316). The cross-platform gaming server (312) may subsequently translate the gaming information (510) in the first data format into gaming information (512) in a second data format understood by the second gaming device (318), for example, by looking up the gaming information (510) in the first data format in a lookup table that associates gaming information (510) in the first data format with gaming information (512) in the second data format. In such an example, the cross-platform gaming server (312) may identify the keystroke on the second gaming device (318) that corresponds to firing a weapon and create gaming information (512) in the second data format that represents firing a weapon.

In the example method of FIG. 4, exchanging (502) gaming information (510, 512) between at least two gaming devices (316, 318) can also include sending (508), from the cross-platform gaming server (302) to the second gaming device (318), the gaming information (512) in the second data format. The gaming information (512) in the second data format may be sent (508) from the cross-platform gaming server (302) to the second gaming device (318), for example, over a telecommunications network, over an IP-based data communications network, and in other ways as will occur to those of skill in the art.

In the example method of FIG. 4, exchanging (502) gaming information between the gaming devices (316, 318) is discussed in terms of receiving (504) gaming information...
from a first gaming device (316), translating (506) the gaming information into a format understood by the second gaming device (318), and sending (508) the gaming information to the second gaming device (318) in a format understood by the second gaming device (318). Readers will appreciate that exchanging (502) gaming information between the gaming devices (316, 318) may also occur in the reverse direction. That is, gaming information may be received from the second gaming device (318), translated into a format understood by the first gaming device (316), and sent to the first gaming device (316).

[0054] For further explanation, FIG. 5 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example method of FIG. 5 includes starting (502) a game by the cross-platform gaming server (302). In the example method of FIG. 5, the cross-platform gaming server (302) may start (502) the game by making a game available to gaming devices (316, 318) that are connected to the cross-platform gaming server (302), for example, by including the game in a list of available games, by sending a notification to gaming devices (316, 318) that are connected to the cross-platform gaming server (302) indicating that a game is available, and so on.

[0055] The example method of FIG. 5 includes joining (504) the game by the first gaming device (316). The example method of FIG. 5 also includes joining (506) the game by the second gaming device (318). A gaming device (316, 318) may join (504, 506) a game, for example, by sending a request to the cross-platform gaming server (302) to include the device as a participant in a particular instance of a game. The request may include information such as an identifier for a player using the device, an identifier for the device, an identification of the operating system that is running on the device, an identification of the network connection and connection speed that the device is using to communicate with the cross-platform gaming server (302), authentication information such as a username and password, and so on.

[0056] In the example method of FIG. 5, the cross-platform gaming server (302) may send a response to a gaming device (316, 318) that is attempting to join the game. The response may indicate whether the gaming device (316, 318) has been added to a particular instance of a game. The response may also include, for example, timing information to synchronize all gaming devices that are participating in a particular instance of a game, a gaming token that identifies the gaming device, player, and particular instance of a game that the gaming device has joined, and so on.

[0057] In the example of FIG. 5, multiple players may join a particular game through the use of a gaming lobby. Such a gaming lobby may present users with different games that are available for a player to join, as well as different instances of the same game. The gaming lobby may include a verification mechanism through which players are identified and validated prior to any attempt by the player to join the game. Likewise, the lobby may include sorting mechanisms to refine the display of different games that are available for a player to join.

[0058] In the example method of FIG. 5, once the game has started, each player on each gaming device (316, 318) is free to perform a gaming action. Gaming actions may include moving a character, firing a weapon, and other gaming options that are available to players in a game. Such gaming actions may be initiated through a user input device such as a touch screen display, keyboard, gaming controller, or other input device that is configured to allow a player to interact with an instance of the game. In such an example, a gaming action is packaged into a gaming event that can be exchanged between multiple players and multiple devices to update the instance of a game that is being executed on each gaming device. Consider an example in which a player performs a gaming action of moving a character in the game to avoid a bullet being fired at the character. In such an example, the gaming action may be packed into a gaming event that includes, for example, the previous coordinate location of the character, the amount of coordinates to move the character, the direction to move the character, the time at which the movement of the character was initiated, and so on.

[0059] The example method of FIG. 5 also includes sending (510) a first gaming device (316) game event message (514) to the cross-platform gaming server (302). The example method of FIG. 5 also includes sending (508) a second gaming device (318) game event message (512) to the cross-platform gaming server (302). In such an example, the game event messages (512, 514) may be sent at any point during the duration of the game. Such messages may include information describing one or more gaming actions initiated on the gaming device (316, 318) that sent the game event message (512, 514).

[0060] The example method of FIG. 5 also includes receiving (516), by the cross-platform gaming server (302), one or more game events (512, 514) from one or more gaming devices (316, 318). In the example method of FIG. 5, the one or more game events (512, 514) that are received (516) by the cross-platform gaming server (302) may be translated into a format understood by other gaming devices participating in the game. Consider an example in which the first gaming device (316) is a mobile device running iOS™ with a display that is 300 pixels wide and the second device is a mobile device running Android™ OS with a display that is 600 pixels wide. In such an example, the cross-platform gaming server (302) may extract information from a first gaming device (316) game event message (514) that is received by the cross-platform gaming server (302) and convert the information into a format appropriate for the second gaming device (318). For example, if the first gaming device (316) game event message (514) including a gaming action to move a character 10 pixels to the left, it may be appropriate to move the character 20 pixels to the left on the second gaming device (318) in view of the fact that the display screen of the second gaming device includes twice the number of pixels as the display screen on the first gaming device. Furthermore, the differences in operating systems between the two devices may require that messages are formatted differently, in order to be understood by the operating system specific version of the gaming application that is executing on each gaming device (316, 318). As such, some level of translation may be required by the cross-platform gaming server (302) upon receipt of a game event message (512, 514). Alternatively, the cross-platform gaming server (302) may simply operate as a pass-through and may forward the game event message (512, 514) without performing any operations on the game event message (512, 514).

[0061] The example method of FIG. 5 also includes transmitting (518), by the cross-platform gaming server (302), the game event messages (520, 522) to one or more gaming devices (316, 318). In the example method of FIG. 5, the first gaming device (316) game event message (520) is sent to the
second gaming device (318) and the second gaming device (318) game event message (522) is sent to the first gaming device (316). A game event message initiated by the first gaming device (316) is sent to the second gaming device (318) so that the second gaming device (318) can process gaming actions taken by a player on the first gaming device (316). Likewise, a game event message initiated by the second gaming device (318) is sent to the first gaming device (316) so that the first gaming device (316) can process gaming actions taken by a player on the second gaming device (318).

In such a way, games presented on a particular gaming device (316, 318) may be updated to account for actions taken by other players using other gaming devices (316, 318).

[0062] In the example method of FIG. 5, the first gaming device (316) game event message (520) that is transmitted (518) from the cross-platform gaming server (302) to the second gaming device (318) is different than the first gaming device (316) game event message (514) that was sent (514) from the first gaming device (316) to the cross-platform gaming server (302). The game event messages (514, 520) may be different as the result of processing that has taken place on the cross-platform gaming server (302) as described above. Even in the event that no processing of the game event message (514) has taken place on the cross-platform gaming server (302), the game events messages (514, 520) may still be different in the sense that the messages are addressed to different recipients. In a similar manner, the second gaming device (318) game event message (522) that is sent to the first gaming device (316) may also be different than the second gaming device (318) game event message (512) that was sent from the first gaming device (318) to the cross-platform gaming server (302).

[0063] The example of FIG. 5 also includes processing (530), by the first gaming device (316), the second gaming device (318) game event message (522). In the example of FIG. 5, processing (530) the second gaming device (318) game event message (522) may be carried out, for example, by extracting information from the game event message (522) and passing the information to a gaming engine executing on the first gaming device (316). The gaming engine may subsequently update the current state of a gaming application executing on the first gaming device (316) in response to information contained in the game event message (522), thereby causing the presentation of the gaming application to a player using the first gaming device (316) to be updated.

[0064] The example of FIG. 5 also includes processing (523), by the second gaming device (318), the first gaming device (316) game event message (520). In the example of FIG. 5, processing (528) the first gaming device (316) game event message (520) may be carried out, for example, by extracting information from the game event message (520) and passing the information to a gaming engine executing on the second gaming device (318). The gaming engine may subsequently update the current state of a gaming application executing on the second gaming device (318) in response to information contained in the game event message (520), thereby causing the presentation of the gaming application to a player using the second gaming device (318) to be updated.

[0065] In the example of FIG. 5, the exchange of game event messages continues until some predetermined condition has been met signifying the end of the game. The predetermined condition has been met signifying the end of the game may include, for example, the expiration of a timer, the achievement of a certain score or goal by a particular player, the exhaustion of all energy of a particular character in the game, and so on.

[0066] For further explanation, FIG. 6 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example method of FIG. 6 includes determining (602), by a cross-platform gaming server (302), the device type of one or more gaming devices (612, 614). In the example of FIG. 6, the cross-platform gaming server (302) is a module of automated computing machinery configured to facilitate real-time gaming between gaming devices (612, 614) of differing types. Real-time gaming, as the term is used in the present application, refers to gaming applications in which each player may play at any point during the duration of the game. Real-time gaming is contrasted to turn-by-turn gaming in which only a single player is allowed to play at a particular point in time.

[0067] In the example of FIG. 6, a gaming device (612, 614) is any device configured to execute a gaming application (616, 618). Gaming devices (612, 614) may be embodied, for example, as a mobile communications device, a mobile computing device, a personal computing device, a dedicated gaming console, or any other computing device configured to execute computer program instructions that comprise a gaming application (616, 618).

[0068] In the example of FIG. 6, the gaming devices (612, 614) may be configured for data communications with input peripherals such as a keyboard, touch screen display, dedicated gaming controller, motion sensor, or other input device configured to receive user input for a gaming application. A gaming device (612, 614) may also be configured for data communications with output peripherals such as, for example, a display screen, audio rendering device, and so on. Although the input and output peripherals may be external to the gaming device (612, 614), readers will appreciate that gaming devices (612, 614) according to embodiments of the present application may include embedded input and output mechanisms such that the gaming device (612, 614) operates as a stand-alone device.

[0069] In the example of FIG. 6, the gaming device (612, 614) is also capable of data communications with the cross-platform gaming server (302) over a data communications network. The gaming device (612, 614) may be configured to communicate with the cross-platform gaming server (302), for example, over an IP based data communications network such as, for example, a WAN, a LAN, the Internet, or a dedicated gaming network. The gaming device (612, 614) may alternatively be configured to communicate with the cross-platform gaming server (302) over a mobile communications network such as, for example, a telephony network, a 3G telecommunications network, a 4G telecommunications network, and others as will occur to those of skill in the art. Readers will appreciate that the gaming device (612, 614) may be configured to communicate with the cross-platform gaming server (302) over many other data communications networks not explicitly listed in the present application.

[0070] In the example method of FIG. 6, each gaming device (612, 614) is characterized by a device type. The device type of a gaming device (612, 614) may be determined based on, for example, an operating system installed on the gaming device (612, 614), a type of data communications networks that the gaming device (612, 614) may communicate over, input mechanisms used by the gaming device (612,
In the example method of FIG. 6, one gaming device (612) is designated as a ‘host’ gaming device. The gaming device (612) may be a ‘host’ gaming device in the sense that the gaming device (612) includes a host gaming engine (622) or other module of computer program instructions for supporting peer-to-peer gaming between the host gaming device (612) and a client gaming device (614) that does not include a host gaming engine (622). In such a way, the host gaming device (612) may be responsible for a larger share of the processing that must occur to carry out peer-to-peer gaming between two gaming devices (612, 614). As such, the gaming device with the greatest processing capabilities may be designated as a ‘host’ gaming device to alleviate processing burdens from a ‘client’ gaming device with fewer available processing resources.

The example method of FIG. 6 also includes identifying (604), by the cross-platform gaming server (302), a host gaming device (612). In the example method of FIG. 6, the cross-platform gaming server (302) may identify (604) a host gaming device (612), for example, by determining the processing capabilities of each gaming device (612, 614) and designating the gaming device with the greatest amount of available computing resources (e.g., available processor cycles, available memory, etc.) as the host gaming device (612). The cross-platform gaming server (302) may alternatively identify (604) a host gaming device (612) by sending a message to each gaming device (612, 614) requesting that each gaming device (612, 614) identify itself as a ‘host’ or ‘client.’ Readers will appreciate that the cross-platform gaming server (302) may also identify (604) a host gaming device (612) in a number of other ways as will occur to those of skill in the art such as, for example, by randomly assigning a device as the host gaming device (612), by assigning the first gaming device that attempts to join a game as the host gaming device (612), by identifying particular types of devices as the host gaming device (612), and so on.

The example method of FIG. 6 also includes transmitting (606), by the cross-platform gaming server (302), gaming device connection information (608, 610). In the example of FIG. 6, gaming device connection information (608, 610) is information used by a particular gaming device to establish a direct connection with another gaming device. Gaming device connection information (608, 610) may be embodied, for example, as an IP address of a gaming device that a particular gaming device can use to establish a direct connection between the gaming devices. For example, the gaming device connection information (608) that is sent to the host gaming device (612) may include the IP address of the client gaming device (614), such that the host gaming device (612) can send data communications to the client gaming device (614). Likewise, the gaming device connection information (610) that is sent to the client gaming device (614) may include the IP address of the host gaming device (612), such that the client gaming device (614) can send data communications to the host gaming device (612).

In the example of FIG. 6, although gaming device connection information (608, 610) is transmitted (606) to both gaming devices (612, 614), readers will appreciate that the cross-platform gaming server (302) may transmit (606) gaming device connection information (608, 610) to only one of the gaming devices (612, 614) in embodiments of the present invention. For example, the cross-platform gaming server (302) may transmit (606) gaming device connection information (608) only to the host gaming device (612), at which point the host gaming device (612) may itself send gaming device connection information (610) to the client gaming device (614), such that the cross-platform gaming server (302) is not responsible for sending gaming device connection information (610) to both gaming devices (612, 614).

The example method of FIG. 6 also includes establishing (620) a peer-to-peer connection between the host gaming device (612) and the client gaming device (614). In the example of FIG. 6, establishing (620) a peer-to-peer connection between the host gaming device (612) and the client gaming device (614) may be carried out by each gaming device (612, 614) simply addressing data communications directly to the other gaming device (612, 614). Alternatively, establishing (620) a peer-to-peer connection between the host gaming device (612) and the client gaming device (614) may be carried out by establishing a data communications pipe between the two gaming devices (612, 614) for exclusive use by each gaming device (612, 614).

For further explanation, FIG. 7 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example of FIG. 7 is similar to the example of FIG. 6, as it also includes determining (602) the device type of one or more gaming devices (612, 614), identifying (604) a host gaming device (612), transmitting (606) gaming device connection information (608, 610), and establishing a peer-to-peer connection (620) between the host gaming device (612) and the client gaming device (614).

In the example method of FIG. 7, determining (602) the device type of one or more gaming devices (612, 614) can include sending (702), from the cross-platform gaming server (302) to each of the one or more gaming devices (612, 614), a device configuration scanner. The device configuration scanner of FIG. 7 is a module of computer program instructions that, when executed, identifies the hardware and software characteristics of the device that is executing the device configuration scanner. The device configuration scanner may identify, for example, details about the CPU included in the device, details about the GPU included in the device, details about the particular operating system that is executing on the device, details about the display screen that is part of or connected to the device, details about input components that are part of or connected to the device, and so on. After the device configuration scanner has been received by a gaming device (612, 614), the gaming device (612, 614) may begin executing the device configuration scanner.

In the example method of FIG. 7, determining (602) the device type of one or more gaming devices (612, 614) can also include receiving (704), by the cross-platform gaming server (302) from each of the one or more gaming devices (612, 614), scan results. In the example method of FIG. 7, scan results are generated by each device configuration scanner and utilized by the cross-platform gaming server (302) to identify the processing capabilities of the device that executed the device configuration scanner. The scan results may be embodied, for example, as a data structure that includes information identifying the type of CPU that is in the device, the type of GPU that is in the device, the operating system that is executing on the device, the version of the operating system, and so on.
The example method of FIG. 7 also includes sending (706) from the cross-platform gaming server (302) to each of the one or more gaming devices (612, 614), a gaming application (616, 618). In the example method of FIG. 7, the gaming application (616, 618) may be embodied as a module of computer program instructions that, when executed by a gaming device (612, 614), presents an electronic or computerized game to a user of the gaming device (612, 614) that is played by manipulating images on a video display, for example, through the use of a gaming controller such as a keyboard, touch screen display, or special purpose gaming controller.

In the example of FIG. 7, each gaming application (616, 618) may be supported by a host gaming engine (622) that is executing on the host gaming device (612). A gaming engine may be embodied, for example, as a software package for supporting and developing gaming applications. The gaming engine may include, for example, a rendering engine for displaying graphics, a physics engine for collision detection, as well as other interfaces for sound, animation, networking, and so on by exposing an API to a game developer that presents pre-packaged gaming modules to the game developer for use by the gaming applications (616, 618).

In the example of FIG. 7, a gaming application (616) is sent to the host gaming device (612) in dependence upon the device type of the host gaming device (612). Consider an example in which the device type of the host gaming device (612) is based on the operating system installed on the host gaming device (612). In such an example, a gaming application configured to execute on a gaming device with an Android™ operating system may be sent to the host gaming device (612) if the host gaming device (612) includes an Android™ operating system. Alternatively, if the host gaming device (612) includes an iOS™ operating system, a gaming application configured to execute on a gaming device with an iOS™ operating system may be sent to the host gaming device (612). In such an example, the computer program instructions included in a gaming application (616) configured to execute on a gaming device with an iOS™ operating system may be different than the computer program instructions included in a gaming application (616) configured to execute on a gaming device with an Android™ operating system.

In other examples, the particular gaming application (616) that is sent to the host gaming device (612) may be based on the input devices used by the host gaming device (612), the output device used by the host gaming device (612), the screen resolution of a display screen used by the host gaming device (612), and so on. Furthermore, the particular gaming application that is sent to the host gaming device (612) may be based on any combination of device type attributes that may occur to those of skill in the art.

In the example of FIG. 7, a gaming application (618) is sent to the client gaming device (614) in dependence upon the device type of the client gaming device (614). Consider an example in which the device type of the client gaming device (614) is based on the operating system installed on the client gaming device (614). In such an example, a gaming application configured to execute on a gaming device with an Android™ operating system may be sent to the client gaming device (614) if the client gaming device (614) includes an Android™ operating system. Alternatively, if the client gaming device (614) includes an iOS™ operating system, a gaming application configured to execute on a gaming device with an iOS™ operating system may be sent to the client gaming device (614). In such an example, the computer program instructions included in a gaming application (618) configured to execute on a gaming device with an iOS™ operating system may be different than the computer program instructions included in a gaming application (618) configured to execute on a gaming device with an Android™ operating system.

In other examples, the particular gaming application (618) that is sent to the client gaming device (614) may be based on the input devices used by the client gaming device (614), the output device used by the client gaming device (614), the screen resolution of a display screen used by the client gaming device (614), and so on. Furthermore, the particular gaming application that is sent to the client gaming device (614) may be based on any combination of device type attributes that may occur to those of skill in the art.

For further explanation, FIG. 8 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example of FIG. 8 is similar to the example of FIG. 6, as it also includes determining (602) the device type of one or more gaming devices (612, 614), identifying (604) a host gaming device (612), transmitting (606) gaming device connection information (608, 610), and establishing (620) a peer-to-peer connection between the host gaming device (612) and the client gaming device (614).

In the example method of FIG. 8, establishing (620) a peer-to-peer connection between the host gaming device (612) and the client gaming device (614) can include exchanging (801) gaming information between at least two gaming devices (612, 614). Exchanging (801) gaming information between at least two gaming devices (612, 614) can include exchanging real-time gaming information between at least two gaming devices (612, 614). Real-time gaming information is gaming information used by gaming applications (616, 618) in which each player may play at any point during the duration of the game. Examples of real-time gaming applications can be included, for example, combat based games in which players battle against each other in real-time. Real-time gaming is contrasted to turn-by-turn gaming in which only a single player is allowed to play at a particular point in time.

In the example method of FIG. 8, exchanging (801) gaming information between at least two gaming devices (612, 614) can include receiving (802) gaming information from the host gaming device (612) in a first data format. The first data format may include, for example, data structures for encapsulating a user input on the host gaming device (612), data types for encapsulating display information on the host gaming device (612), and so on. For example, a gaming device that includes a touch screen display through which a player interacts with the gaming application will produce different user input data than a gaming device that has a keyboard through which a player interacts with the gaming application. As such, the data format of user input will be different for each type of gaming device.

In the example method of FIG. 8, exchanging (801) gaming information between at least two gaming devices (612, 614) can also include translating (804) the gaming information from the host gaming device (612) in the first data format into gaming information in a second data format in dependence upon the device type of the host gaming device (612) and the device type of the client gaming device (614). Consider an example in which the host gaming device (612)
is a mobile communications device that includes a touch screen display through which a first player interacts with a gaming application (616) executing on the host gaming device (612). Assume that in the same example, the client gaming device (614) is a personal computer that includes a keyboard through which a second player interacts with a gaming application (618) executing on the client gaming device (614). In such an example, the first player may touch a particular region of the touch screen on the host gaming device (612) to perform a certain action (e.g., firing a weapon in the gaming application) while the second player may press a particular key on the keyboard of the client gaming device (614) to perform the same action. In such an example, the host gaming engine (622) may receive gaming information in a first data format from the host gaming device (612) indicating that the first player has touched a particular region of the touch screen on the host gaming device (612) to perform the action of firing a weapon in the gaming application (616) that is executing on the host gaming device (612). The host gaming engine (622) may subsequently translate the gaming information in the first data format into gaming information in a second data format understood by the client gaming device (614), for example, by looking up the gaming information in the first data format in a lookup table that associates gaming information in the first data format with gaming information in the second data format. In such an example, the host gaming engine (622) may identify the keystroke on the client gaming device (614) that corresponds to firing a weapon and create gaming information in the second data format that represents firing a weapon.

[0087] In the example method of FIG. 8, exchanging (801) gaming information between at least two gaming devices (612, 614) can also include sending (806) the gaming information in the second data format to the client gaming device (806). The gaming information in the second data format may be sent (806) from the host client device (612) to the client gaming device (614), for example, over the peer-to-peer connection established between the two gaming devices (612, 614) through the use of a telecommunications network, IP-based data communications network, and in other ways as will occur to those of skill in the art.

[0088] In the example method of FIG. 8, exchanging (801) gaming information between the gaming devices (612, 614) is discussed in terms of receiving (802) gaming information from a host gaming device (612), translating (804) the gaming information into a format understood by the client gaming device (612), and sending (806) the gaming information to the client gaming device (614) in a format understood by the client gaming device (614). Readers will appreciate that exchanging (801) gaming information between the gaming devices (612, 614) may also occur in the reverse direction. That is, gaming information may be received from the client gaming device (614), translated into a format understood by the host gaming device (612), and sent to the host gaming device (612).

[0089] For further explanation, FIG. 9 sets forth a flow chart illustrating a further example method for cross-platform gaming between multiple devices of multiple types according to embodiments of the present invention. The example of FIG. 9 includes sending (902) a host gaming device game event message (904) to a host gaming engine (622) executing on the host gaming device (612). In the example of FIG. 9, a host gaming device game event message (904) is a data structure that includes information related to one or more gaming actions (e.g., firing a weapon, moving a character, and so on) that was performed on the host gaming device (612). The example of FIG. 9 also includes sending (918), by a client gaming device (614), a client gaming device game event message (914) to the host gaming device (612). In the example of FIG. 9, a client gaming device game event message (914) is a data structure that includes information related to one or more gaming actions (e.g., firing a weapon, moving a character, and so on) that was performed on the client gaming device (614). In the example of FIG. 9, the client gaming device game event message (914) is sent to the host gaming device (612) because the host gaming device (612) carries out many of the same processing as the host gaming device (612) includes a host gaming engine (622). The example of FIG. 9 also includes receiving (910), by the host gaming device (612), a client gaming device game event message (914) from the client gaming device (612) and processing (906), by the host gaming engine (622) executing on the host gaming device (612), the client gaming device game event message (914) and the host gaming device game event message (904). In the example of FIG. 9, the host gaming engine (622) processes the client gaming device game event message (914) and the host gaming device game event message (904), thereby accounting for gaming actions taken by each player.

[0092] The example of FIG. 9 also includes sending (908), by host gaming engine (622) executing on the host gaming device (612), game data (916) for rendering (920) on the client gaming device (614) and game data (917) for rendering (901) on the host gaming device (612). In the example of FIG. 9, game data (916, 917) is the output generated by a host gaming engine (622) after processing (906) the game events (904, 914). The game data (916, 917) may be embodied as a data structure that includes information that defines the status of a particular game. For example, if a particular game event (904, 914) represents the firing of a weapon in a game, and the host gaming engine (622) determines that firing the weapon resulted in a hit on an opponent’s character, the game data (916, 917) may include an updated energy level for the character that was hit, along with a reduced count of available bullets for the player that fired the weapon. Such game data (916) is rendered (920) by the client gaming device (614) to present an updated version of the game on the client gaming device (614). Likewise, game data (917) is also rendered (901) by the host gaming device (612) to present an updated version of the game on the host gaming device (612).

[0093] Embodiments of the present invention are discussed primarily in the context of cross-platform gaming. Readers will appreciate, however, that the concepts described herein may be applied in areas other than gaming. That is, the concepts described herein may be utilized to support any software application that includes interactions between multiple people over devices of multiple types.

[0094] As will be appreciated by one skilled in the art, aspects of the present invention may be embodied as a system, method or computer program product. Accordingly, aspects of the present invention may take the form of an entirely hardware embodiment, an entirely software embodiment (including firmware, resident software, micro-code, etc.) or an embodiment combining software and hardware aspects that may all generally be referred to herein as a “circuitry,” “module” or “system.” Furthermore, aspects of the present invention may take the form of a computer program product
embodied in one or more computer readable medium(s) hav- ing computer readable program code embodied thereon.

Any combination of one or more computer readable medium(s) may be utilized. The computer readable medium may be a computer readable signal medium or a computer readable storage medium. A computer readable storage medium may be, for example, but not limited to, an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, or device, or any suitable combination of the foregoing. More specific examples (a non-exhaustive list) of the computer readable storage medium would include the following: an electrical connection having one or more wires, a portable computer diskette, a hard disk, RAM, read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), an optical fiber, a portable compact disc read-only memory (CD-ROM), an optical storage device, a magnetic storage device, or any suitable combination of the foregoing. In the context of this document, a computer readable storage medium may be any tangible medium that can contain, or store a program for use by or in connection with an instruction execution system, apparatus, or device.

A computer readable signal medium may include a propagated data signal with computer readable program code embodied therein, for example, in baseband or as part of a carrier wave. Such a propagated signal may take any of a variety of forms, including, but not limited to, electro-magnetic, optical, or any suitable combination thereof. A computer readable signal medium may be any computer readable medium that is not a computer readable storage medium and that can communicate, propagate, or transport a program for use by or in connection with an instruction execution system, apparatus, or device.

Program code embodied on a computer readable medium may be transmitted using any appropriate medium, including but not limited to wireless, wireline, optical fiber cable, RF, etc., or any suitable combination of the foregoing.

Computer program code for carrying out operations for aspects of the present invention may be written in any combination of one or more programming languages, including an object oriented programming language such as Java, Smalltalk, C++ or the like and conventional procedural programming languages, such as the "C" programming language or similar programming languages. The program code may execute entirely on the user's computer, partly on the user's computer, as a stand-alone software package, partly on the user's computer and partly on a remote computer or entirely on the remote computer or server. In the latter scenario, the remote computer may be connected to the user's computer through any type of network, including a LAN or a WAN, or the connection may be made to an external computer (for example, through the Internet using an Internet Service Provider).

Aspects of the present invention are described above with reference to flowchart illustrations and/or block diagrams of methods, apparatus (systems) and computer program products according to embodiments of the invention. It will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, can be implemented by computer program instructions. These computer program instructions may be provided to a processor of a general purpose computer, special purpose computer, or other programmable data processing apparatus to produce a machine, such that the instructions, which execute via the processor of the computer or other programmable data processing apparatus, create means for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks.

These computer program instructions may also be stored in a computer readable medium that can direct a computer, other programmable data processing apparatus, or other devices to function in a particular manner, such that the instructions stored in the computer readable medium produce an article of manufacture including instructions which implement the function/act specified in the flowchart and/or block diagram block or blocks.

The computer program instructions may also be loaded onto a computer, other programmable data processing apparatus, or other devices to cause a series of operational steps to be performed on the computer, other programmable apparatus or other devices to produce a computer implemented process such that the instructions which execute on the computer or other programmable apparatus provide processes for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks.

The flowchart and block diagrams in the Figures illustrate the architecture, functionality, and operation of possible implementations of systems, methods and computer program products according to various embodiments of the present invention. In this regard, each block in the flowchart or block diagrams may represent a module, segment, or portion of code, which comprises one or more executable instructions for implementing the specified logical function(s). It should also be noted that, in some alternative implementations, the functions noted in the block may occur out of the order noted in the figures. For example, two blocks shown in succession may, in fact, be executed substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved. It will also be noted that each block of the block diagrams and/or flowchart illustration, and combinations of blocks in the block diagrams and/or flowchart illustration, can be implemented by special purpose hardware-based systems that perform the specified functions or acts, or combinations of special purpose hardware and computer instructions.

It will be understood from the foregoing description that modifications and changes may be made in various embodiments of the present invention without departing from its true spirit. The descriptions in this specification are for purposes of illustration only and are not to be construed in a limiting sense. The scope of the present invention is limited only by the language of the following claims.

What is claimed is:

1. A method of cross-platform gaming between multiple devices of multiple types, the method comprising:
   determining, by a cross-platform gaming server, the device type of one or more gaming devices; and
   sending, to each of the one or more gaming devices, a gaming application from the cross-platform gaming server to the gaming device in dependence upon the device type of the gaming device.

2. The method of claim 1 wherein determining, by a cross-platform gaming server, the device type of one or more gaming devices further comprises:
   sending, from the cross-platform gaming server to each of the one or more gaming devices, a device configuration scanner; and
3. The method of claim 1 wherein determining the device type of one or more gaming devices further comprises identifying an operating system executing on each gaming device.

4. The method of claim 1 further comprising exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices.

5. The method of claim 4 wherein exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices further comprises exchanging real-time gaming information between at least two gaming devices.

6. The method of claim 4 wherein exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices further comprises:
   receiving, by the cross-platform gaming server, gaming information between at least two gaming devices; and
   translating, by the cross-platform gaming server, the gaming information from the first gaming device in the first data format into gaming information in a second data format in dependence upon the device type of the first gaming device; and
   notifying, by the cross-platform gaming server, each gaming device that a new version of a particular gaming application has been created; and
   sending, by the cross-platform gaming server to each gaming device, a gaming application update.

7. The method of claim 1 further comprising:
   notifying, by the cross-platform gaming server, each gaming device that a new version of a particular gaming application has been created; and
   sending, by the cross-platform gaming server to each gaming device, a gaming application update.

8. Apparatus for cross-platform gaming between multiple devices of multiple types, the apparatus comprising a computer processor operatively coupled to computer memory, the computer memory including computer program instructions that, when executed by the computer processor, cause the apparatus to carry out the steps of:
   determining, by a cross-platform gaming server, the device type of one or more gaming devices; and
   sending, to each of the one or more gaming devices, a gaming application from the cross-platform gaming server to the gaming device in dependence upon the device type of the gaming device.

9. The apparatus of claim 8 wherein determining, by a cross-platform gaming server, the device type of one or more gaming devices further comprises:
   sending, from the cross-platform gaming server to each of the one or more gaming devices, a device configuration scanner; and
   receiving, by the cross-platform gaming server from each of the one or more gaming devices, scan results.

10. The apparatus of claim 8 wherein determining the device type of one or more gaming devices further comprises identifying an operating system executing on each gaming device.

11. The apparatus of claim 8 further comprising computer program instructions that, when executed by the computer processor, cause the apparatus to carry out the step of exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices.

12. The apparatus of claim 11 wherein exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices further comprises exchanging real-time gaming information between at least two gaming devices.

13. The apparatus of claim 11 wherein exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices further comprises:
   receiving, by the cross-platform gaming server, gaming information from a first gaming device in a first data format;
   translating, by the cross-platform gaming server, the gaming information from the first gaming device in the first data format into gaming information in a second data format in dependence upon the device type of the first gaming device and the device type of a second gaming device; and
   sending, from the cross-platform gaming server to the second gaming device, the gaming information in the second data format.

14. The apparatus of claim 8 further comprising computer program instructions that, when executed by the computer processor, cause the apparatus to carry out the steps of:
   notifying, by the cross-platform gaming server, each gaming device that a new version of a particular gaming application has been created; and
   sending, by the cross-platform gaming server to each gaming device, a gaming application update.

15. A computer program product for cross-platform gaming between multiple devices of multiple types, the computer program product disposed upon a computer readable storage medium, the computer program product comprising computer program instructions that, when executed, cause a computer to carry out the steps of:
   determining, by a cross-platform gaming server, the device type of one or more gaming devices; and
   sending, to each of the one or more gaming devices, a gaming application from the cross-platform gaming server to the gaming device in dependence upon the device type of the gaming device.

16. The computer program product of claim 15 wherein determining, by a cross-platform gaming server, the device type of one or more gaming devices further comprises:
   sending, from the cross-platform gaming server to each of the one or more gaming devices, a device configuration scanner; and
   receiving, by the cross-platform gaming server from each of the one or more gaming devices, scan results.

17. The computer program product of claim 15 wherein determining the device type of one or more gaming devices further comprises identifying an operating system executing on each gaming device.

18. The computer program product of claim 15 further comprising computer program instructions that, when executed, cause a computer to carry out the step of exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices.

19. The computer program product of claim 18 wherein exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices further comprises exchanging real-time gaming information between at least two gaming devices.
20. The computer program product of claim 18 wherein exchanging, by the cross-platform gaming server, gaming information between at least two gaming devices further comprises:

receiving, by the cross-platform gaming server, gaming information from a first gaming device in a first data format;

translating, by the cross-platform gaming server, the gaming information from the first gaming device in the first data format into gaming information in a second data format in dependence upon the device type of the first gaming device and the device type of a second gaming device; and

sending, from the cross-platform gaming server to the second gaming device, the gaming information in the second data format.

* * * * *