(54) Titre : PROCEDE ET DISPOSITIF DE MISE EN OEUVRE D'UN SYSTEME DE LIVRAISON DE LOGICIEL TELECHARGEABLE
(54) Title: METHOD AND DEVICE FOR IMPLEMENTING A DOWNLOADABLE SOFTWARE DELIVERY SYSTEM

(57) Abrégé/Abstract:
A method and device implementing a downloadable operatively connected to a plurality of gaming machines either by a direct communications link or through the use of one or more site controllers or PCs. Each gaming machine and site controller or PC contains two executable spaces, one of which typically contains the software image currently being implemented by the device.
(57) Abrégé(suite)/Abstract(continued):
The other executable space is designated to receive from the central system a new software image that will be utilized by the device. The central system transfers a new image to be executed, via packet encrypted communications, to a networked device which stores the image in an executable space, while continuing to run the currently designated image. Upon instruction from the central system, the device switches over to the new image, allowing one or more machines to implement a software upgrade on a continuous basis.
(54) Title: METHOD AND DEVICE FOR IMPLEMENTING A DOWNLOADABLE SOFTWARE DELIVERY SYSTEM

Three Forks, MT 59752 (US), WALL, Robert, Lyle; 222 Westridge Drive, Bozeman, MT 59715 (US), SWENSON, Charles, Glen; 101 Churn Road, Bozeman, MT 59715 (US).

Agents: KIRCHER, William, B. et al.; Shook, Hardy & Bacon L.L.P., One Kansas City Place, 1200 Main Street, Kansas City, MO 64105-2118 (US).

(81) Designated States (national): AU, CA, PL.

(84) Designated States (regional): European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published:
— Without international search report and to be republished upon receipt of that report.

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(57) Abstract: A method and device implementing a downloadable operatively connected to a plurality of gaming machines either by a direct communications link or through the use of one or more site controllers or PCs. Each gaming machine and site controller or PC contains two executable spaces, one of which typically contains the software image currently being implemented by the device. The other executable space is designated to receive from the central system a new software image that will be utilized by the device. The central system transfers a new image to be executed, via packet encrypted communications, to a networked device which stores the image in an executable space, while continuing to run the currently designated image. Upon instruction from the central system, the device switches over to the new image, allowing one or more machines to implement a software upgrade on a continuous basis.
METHOD AND DEVICE FOR IMPLEMENTING A
DOWNLOADABLE SOFTWARE DELIVERY SYSTEM

TECHNICAL FIELD

In general, this invention relates to a downloadable software system, and

more particularly, to a method and device implementing a downloadable software system

for an electronic gaming machine communications network.

BACKGROUND OF THE INVENTION

In general, conventional gaming machine networks typically include a

central system operatively connected to one or more individual gaming machines via

intermediate communication site controllers. Although the gaming machines

communicate with the central system, each gaming machine or site controller contains

a central chipset which locally stores the computer code to be executed by the device to

perform gaming related functions. These chipsets typically consist of electronic

programmable read only memory (EPROM) which permanently store the computer code.

EPROM chipsets are conventionally preferred because the electronic memory can be

controlled in a secured manner without giving unauthorized access to the gaming

machine code. For example, in the event the computer code needs to be upgraded,

service personnel are required to manually change the chipset for each gaming machine

and/or site controller.

Because a service technician must perform the same operation for each

machine or controller, the current method of upgrading gaming machine/site controller

or PC software typically takes a long time to accomplish at a substantial cost, including

the cost of the technician time and the cost of a new chipset for each machine.

Accordingly, there is a need for a system which can upgrade computer

code within a networked device without requiring a manual change in the device

components or requiring a high cost of implementation.

SUMMARY OF THE INVENTION

Generally described, a gaming machine system is provided. The gaming

machine system includes a central system and one or more gaming devices having at least

one storage component operable to receive an executable software image and in
communication with the central system. Additionally, the gaming device receives the software image from the central system.

In another aspect of the present invention, a method in a computer system for implementing a gaming machine system including a central system in communication with one or more gaming devices is provided. In accordance with the method, a software image to be downloaded to the one or more gaming devices is obtained. The software image is then downloaded to a selected group of the one or more gaming devices.

In a further aspect of the present invention, a method in a computer system for implementing a download of a software image is provided. In accordance with the method, a central processor obtains a software image to be downloaded and transfers the software image to one or more device processors. The one or more device processors receives and stores the software image. The at least one of the one or more device processors executes the software image.

In yet another aspect of the present invention, a gaming machine system is provided. The gaming machine system includes a central system and one or more gaming devices having storage means for receiving an executable software image. The one or more gaming devices are in communication with the central system. Additionally, the gaming machine system includes downloading means for transferring the software image from the central system to the one or more gaming devices.

A method and device implementing a downloadable software delivery system for an electronic gaming machine communications network is provided. A central system is operatively connected to a plurality of gaming machines either by a direct communications link or through the use of one or more site controllers. In this regard, it is contemplated that a PC or suitable computing device could be substituted for a site controller and that the downloadable software delivery still be effected. Each gaming machine and site controller or PC contains two executable spaces, one of which typically contains the software image currently being implemented by the device. The other executable space is designated to receive from the central system a new software image that will be utilized by the device. The central system transfers a new image to be executed, via packet encrypted communications, to a networked device which stores the image in an executable space, while continuing to run the currently designated image.
Upon instruction from the central system, the device switches over to the new image, allowing one or more machines to implement a software upgrade on a continuous basis.

**BRIEF DESCRIPTION OF THE DRAWING**

The present invention is described in detail below with reference to the attached figures, wherein:

FIG. 1 is a block diagram of a gaming machine network utilized in accordance with the present invention;

FIG. 2 is a block diagram illustrative of various device components utilized in accordance with the present invention;

FIGS. 3A, 3B & 3C are flow diagrams illustrative of a software image transfer method utilizing random key encryption in accordance with the present invention;

FIGS. 4A & 4B are flow diagrams illustrative of an image transfer error checking and bypass process in accordance with the present invention;

FIG. 5 is a flow diagram illustrative of a software image transfer method to a gaming machine in accordance with the present invention; and

FIG. 6 is a block diagram illustrative of a software image parsing embodiment in accordance with the present invention.

**DETAILED DESCRIPTION OF THE INVENTION**

In essence, the present invention enables a central system operatively connected to a plurality of gaming machines and site controllers (or PCs) to upgrade one or more software images via a communications link without requiring a manual change of the device chipset.

FIG. 1 is block diagram illustrative of a gaming machine network operable to be utilized by the present invention, designated generally by the reference numeral 10. Generally, the gaming machine network 10 includes a central system 12 operatively connected to a number of gaming machines 14 either by a direct communication link to each individual machine 14 or indirectly through the one or more site controllers or PC's 16. The connectivity of the central system 12 to the gaming machines 14 can include continuous, on-line communication systems, including local area networks and/or wide
may also be implemented and is considered to be within the scope of the present invention.

FIG. 2 is a block diagram illustrative of some of the components common to the gaming machines 14, site controllers 16 or other networked device (FIG. 1), generally referred to as a device 18, utilized in the present invention. Each device 18 preferably contains a processor 20, a memory 22, a communications input/output 24, such as a modem or network card, and at least two executable spaces 26. As would be readily understood by one skilled in the relevant art, the processor 20, memory 22 and communications input/output 24 includes any variety of component generally utilized in the implementation of the device. Moreover, in one embodiment, one or more of the executable spaces 26 are FLASH ROM. However, as would be readily understood, the executable spaces 26 may include DVD, CD-ROM, battery backed RAM or any other nonvolatile memory storage device.

Preferably, one executable space 26 is typically designated to store the software code, or image, currently being executed by the device 18. The other executable space is typically designated to receive a new image transferred by the central system. As would be understood, although the two executable spaces are preferably separate, the same effect is accomplished through the use of a single, larger executable space. In this embodiment, each device uses a portion of the executable space 26 to assist in receiving and storing incoming images from the central system.

As an alternative embodiment, the present invention may also be implemented with one executable space and sufficient other memory, which can include memory 22, to temporarily store a downloaded image. In this embodiment, the image would be downloaded to the temporary memory and then transferred to the more permanent executable space 26.

Generally, the present invention facilitates the implementation and replacement of a software image on a device in a gaming machine network by allowing the transmittal of a new image to a device while the device continues to execute and/or process a previous software image. Additionally, because the present invention may utilize one or more existing communication lines, the transfer of a new image can include various security and error checking features to ensure and preserve the secured character of the executable code.
FIGS. 3A, 3B & 3C are flow diagrams of an image downloading process utilizing a random key encryption in accordance with the present invention. With reference to FIG. 3A, at S28, the desired image to be downloaded is created, and loaded into the central system. Preferably, the operating system of the central system provides a user interface, such as a graphical user interface, that allows a user to download the image to the central system's memory. Additionally, the user interface can include prompts for a user to enter additional information needed for the downloading process including download time information, download windows and version numbers. As would be understood, depending on the function of the image being downloaded, the additional information needed to complete the download will vary.

Once the image has been downloaded to the central system, the user selects which devices are to receive the image. The user selection can include all of the devices or subsets of devices. Preferably, the central system includes some form of error checking that ensures that the designated device is compatible with the image to be downloaded. At S30, the central system generates a random encryption key for each device designated to receive the image and encrypts the image with each of the random keys at S32. The random keys and encrypted images are stored in the central system memory. Additionally, the central system stores a completed, unencrypted version of the image in memory to use a signature for verification that the download is complete.

Generally, the function of a site controller (or PC) download differs from the function of the gaming machine download. Accordingly, at S34 a determination of whether the download is for a site controller is made. With reference to FIGS. 3A & 3B, if at S34 the desired image is designated to be downloaded to a site controller or PC, the random keys used to encrypt the image are themselves encrypted with a general encryption key and sent to the site controller at S36. At S38, the site controller or PC decrypts the random keys and stores the keys in a memory, such as memory 22 (FIG. 2). The central system then sends the random key encrypted message to the site controller at S40. Once the download is complete, the central system sends additional instructions to the site controller such as to decrypt the image with the stored random keys or to store the image into its second executable space.

With reference to FIGS. 3A & 3C, if at S34, the desired image is designated to be downloaded to a gaming machine or other device, the central system sends the encrypted message to the site controller (or PC) associated with the particular
gaming machine at S44, preferably in a manner as described above in steps S36-S42. At S46, the central system sends the site controller a list of the gaming machines to receive the image and their preassigned general encryption keys, which are encrypted with a key known to the gaming machine. At S48, the site controller transfers the encryption keys to the gaming machine, which decrypts and stores the random keys in memory. The site controller then sends the random key encrypted image to the gaming machine at S50. Once the download is complete, the central system instructs the gaming machine, via the site controller, to prepare and store the image into its second executable space at S54.

With reference to FIGS. 4A & 4B, the present invention implements a bypass and error checking function between the central system and the site controller or PC. Because the site controller can be associated with a number of gaming machines or other devices, once the site controller stores the image into its executable space, it does not need to reexecute the downloading step for each subsequent transfer to a gaming machine. With reference to FIG. 4A, the central system begins the download process each time an image is to be transferred to a device as illustrated at S56. At S58, the central system checks whether a downloaded image has already been stored in the site controller’s executable space. If so, at S60, the central system verifies that the signature of the image loaded on the site controller is correct and the transfer is complete at S72. With reference to FIGS. 4A & 4B if an image is not present in the site controller’s executable space at S58 or if the signature does not match at S60, the central system sends the image via packets to the site controller or PC at S62.

Preferably, the central system relies on package acknowledge signals from the site controller to ensure that each individual packet is received by the site controller. Accordingly, at S64, the central system determines whether all the packets have been received. If one or more package acknowledge signals are not received, the transfer is incomplete at S70. At this point, the central system may resend the individual packets not received or may attempt to resend the entire image. Alternatively, the central system may just declare the transfer a failure.

If the packets are received and acknowledged at S64, the central system completes the transfer at S66. At S68, the central system requests a signature of the image from the site controller to verify a proper transmission and decryption. With reference to FIGS. 4A & 4B, if the signature is a match, the download is a success at S72.
and the site controller implements any downloading instruction. If the signature is not a match, the transfer is incomplete at S70.

With reference to FIG. 5, the present invention also implements an error transfer method for the downloading of an image from the site controller to the gaming machine. Upon receiving and storing the downloaded image in memory, the site controller (or PC) begins the download to the gaming machine at S74. Preferably as illustrated in FIG. 6, the software image 86 is organized into one or more frames 88 which are further organized into one or more blocks 92 per frame. Each of the blocks 92 can then be transferred as individual communication packets. During the download process, site controller transfers all packets that make up the frame with reference again to FIG. 5, at the end of the transfer frame the site controller requests an acknowledgment from the gaming machine at S70.

If the gaming machine did not receive some portion of the frame, the transfer is incomplete at S82. The site controller preferably resends only those packets which are incomplete. Alternatively, the entire image may be resent or the transfer may be declared a failure. Accordingly, the gaming machine does not need to acknowledge receipt of each packet. As would be understood, however, alternative methods of grouping and sending the software image would be considered within the scope of the present invention.

Upon the transfer of the entire image to the gaming machine at S78, the central system requests an image signature to verify the transfer was successful at S80. If the signature is a match, the transfer is successful at S84. If the image is not a match, the image is incomplete at S82.

The above-described transfer protocols have been incorporated with reference to two separate encryption methods. As would be understood, a system implementing only a portion, different or no encryption methods would be considered within the scope of the present invention.

Once the image has been successfully transferred to the device, the image can be executed. Preferably, the central system sends a command to the device to begin using the new image in the executable space. This command typically includes separate instructions for configuring the system to accommodate the new image and preventing the future play of the current image while the switch is occurring. Upon the completion of the command, the device begins executing the new image and the switch is complete.
Because the device contains at least two separate executable spaces, the old image previously being executed remains in the device executable space after the switch is complete. In the event that the new image is corrupt or not functioning properly, the central system can execute a command to revert to the old image if it is still available and intact.

Although the devices specifically referenced in the present application refer solely to gaming machines or site controllers or PCs, the present invention allows images to be transferred to any device that is configured to receive an image. Such devices could include peripheral devices such as printers and bill acceptors or other intermediate communications devices. As would be understood, the images associated with each device would vary with the type of device and its function in the system.

In the foregoing specification, the present invention has been described with reference to the specific exemplary embodiments thereof. It will be apparent to those skilled in the art that a person understanding this invention may conceive of changes or other embodiments or variations, which utilize the principals of this invention without departing from the broader scope of the invention.
What is claimed:

35. A gaming machine comprising: means for storing a software program; means for communicating with an outside source having a downloadable software program therein; and means for receiving and implementing said downloaded software program from said outside source without suspending the operation of the gaming machine.

36. The gaming machine as recited in claim 35 wherein said storage means comprises a first executable space for storing the software program currently being implemented by the gaming machine and a second executable space operable to receive the downloaded software program.

37. The gaming machine as recited in claim 36 wherein the first executable space and the second executable space are nonvolatile memory storage devices.

38. The gaming machine as recited in claim 37 wherein the first executable space and the second executable space are flash read only memory.

39. The gaming machine as recited in claim 35 wherein said storage means comprises a first executable space for storing the software program currently being implemented by the gaming machine and a temporary memory operable to receive the downloaded software program.

40. The gaming machine as recited in claim 39 wherein said storage means is a nonvolatile memory storage device.

41. The gaming machine as recited in claim 40 wherein said storage means is a flash read only memory.
42. The gaming machine as recited in claim 35 wherein said storage means comprises a single executable space partitioned into two or more portions with one portion used for storing the software program currently being implemented by the gaming machine and a second portion used to receive the downloaded software program.

43. A method for implementing a downloaded software program in a gaming machine, the method comprising: receiving the downloaded software program; storing the downloaded software program; and operatively implementing the downloaded software program in said gaming machine without suspending operation of the gaming machine.

44. The method as recited in claim 43 further comprising decrypting the downloaded software program without suspending operation of the game machine.

45. The method as recited in claim 43 further comprising: determining whether the gaming machine is currently using a copy of the downloaded software program; and if the gaming machine is currently using a copy of the downloaded software program, cancel the download of the software program.

46. A gaming system having a central system and one or more gaming machines operatively connected therein, said gaming system comprising: means located in at least one gaming machine for a plurality of software images; means for said central system and said central system to communicate; and means for transferring a software images from said central system to said at least one gaming machine and implementing said downloaded software images without suspending the operation of said at least one gaming machine.

47. The gaming system recited in claim 46 wherein the central system is in communication with the one or more gaming machines via a dedicated, continuous communication network.
48. The gaming system recited in claim 46 wherein the central system is in communication with the one or more gaming machines via a nondedicated communication network.

49. The gaming system recited in claim 46 wherein the central system is remote from the one or more gaming machines.

50. The gaming system recited in claim 46 wherein the central system includes interface means for accepting the software image to be downloaded to the one or more gaming machines.

51. The gaming system recited in claim 46 wherein the gaming machines correspond to a different random key.

52. A gaming machine system comprising: a central system; one or more gaming machines; and a site controller, said site controller comprising means for communicating with said central system and with one or more gaming machines; wherein each gaming machine has means for storing a plurality of software images therein; means for communicating with said site controller; means for receiving a software image from said site controller; and means for implementing said downloaded software image therein without suspending operation of the gaming machine.

53. A method in a computer system for implementing a gaming system including a central system in communication with one or more gaming devices, the method comprising: transferring a software image from the central system to one or more gaming machines; receiving the downloaded software image within one or more gaming machines without suspending operation of the receiving gaming machine; storing the downloaded software program in said one or more gaming machines without suspending operation of the storing gaming machine; and implementing the downloaded software image into said one or more gaming machines without suspending operation of the implementing gaming machine.
54. The method recited in claim 53 further comprising: encrypting the software image with a random key before said software image is transferred; and decrypting the downloaded software image without suspending operation of the gaming machine.

55. The method recited in claim 54 further comprising: encrypting the random key with a general key encryption; and transferring the encrypted key to the selected group of the one or more gaming machines.

56. The method as recited in claim 53 further comprises: detecting whether the one or more gaming machines are currently using a copy of the downloaded software image; and if the gaming machine is currently using a copy of the downloaded software image, cancel the download of the software image.

57. The method as recited in claim 53 wherein the transfer step comprises: transferring a software image from the central system to one or more site controllers; receiving the software image within the one or more site controllers; and transferring the software image from the one or more site controllers to one or more gaming machines.
FIG. 1.

CENTRAL SYSTEM

SITE CONTROLLER

SITE CONTROLLER

SITE CONTROLLER

SUBSTITUTE SHEET (RULE 26)
A

Encrypted keys are sent to site controller

Is image for site controller?

Yes

Image is encrypted with random keys

Random keys are generated

Image is obtained

No

B

Site controller decrypts image

Encrypted image is sent to site controller

Site controller decrypts keys and stores them in memory

A