Title: ALLOCATION AND CACHING METHOD AND APPARATUS FOR SPEED-UP OF INTERACTIVITY FROM STORAGE MEDIA AND CORRESPONDING STORAGE MEDIA

Abstract: An allocation and caching method and apparatus for speed-up of interactivity from storage media. When running an interactive presentation, the user is presented with a choice of presentation paths. The respective initial parts of these possible paths are decoded and cached before the user chooses one subsequent path through interaction in a decision D. The subsequent path is immediately replayed from the fast cache memory. Jumping to the chosen path on the storage medium is slower and takes place while the cached data is being presented. When the path from the storage medium is accessed, playing is subsequently continued from the storage medium until the next interactive decision appears. Hence, no time is lost after making a decision D.

Declaration under Rule 4.17:

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.
Allocation and caching method and apparatus for speed-up of interactivity from storage media and corresponding storage media

FIELD OF THE INVENTION

This invention pertains in general to the field of playing back data from storage media and allocation of data on storage media, and more particularly to playing back interactive programmes from storage media and allocating the corresponding data on storage media, and even more particularly to avoidance of waiting time for users when choosing from several alternative paths when playing back interactive data allocated on storage media.

BACKGROUND OF THE INVENTION

Decisions are taken during the playback of interactive multimedia applications, such as interactive games or movies, played back from storage media, such as CDs. These decisions concern subsequent sequences within said games or movies. The decisions are based on multiple-choice decisions, i.e. a decision is taken by choosing one subsequent sequence from a multiple, limited number of choices at the end of a first sequence. The decision is often taken by means of a menu presented to a user or by means of an action executed by a cursor in a certain position. Data is stored in paths sequentially arranged on the storage medium. Access from one path to another is associated with a jump delay, which is inconvenient for the user. Time is lost during waiting for a subsequent path to be accessed and the contents to be displayed. This is annoying for the user.

WO03/036647 discloses a media player having a cache memory into which multimedia programs are loaded from a storage disk. Apparently, the media player is of the Apple iPOD® type and the storage disk is thus a hard disk. A method is disclosed for decreasing a time delay after a media play selection. A media program is reproduced before an entire media program is loaded into the cache memory in that an initial portion of a media title is loaded into memory at the beginning of said media program. However, in this case the above-described disadvantageous jump delay is still present. Furthermore, WO03/036647 aims at minimizing the additional time delay occurring after the physical jump of the drive to the beginning of the data sequence on the storage medium pertaining to a specific media program chosen for playing. Alternatively, the media program is already loaded into cache memory and played directly from the cache memory. The number of media titles stored in
such media devices is generally huge, up to several thousands of titles from which the user can choose. Thus, in order to enable this feature for any media program on the disc, i.e. to ensure a minimal time delay when accessing any media program, excessive memory usage is needed and in fact, the disc would be oblivious as all media programs would be stored in a memory. This is not feasible or cost-effective. Furthermore, it is mentioned in WO03/036647 that the latter alternative is typically not the case.

Thus, there is a need for a new way of allocating and caching interactive data from a storage medium, such as an optical disc, reducing the waiting time to zero until a subsequent section of data stored on the storage medium is played upon being chosen by a user.

SUMMARY OF THE INVENTION

The present invention overcomes the deficiencies in the art identified above and solves at least the problems identified above by providing a method, a storage medium, a device, and a computer-readable medium comprising a program code according to the appended patent claims.

The general solution according to the invention is given hereinafter. When running an interactive presentation, such as video or games, the user is presented with a choice of presentation paths, of which the user may choose only one path. The initial segments of these possible paths are cached in a solid state memory before the user chooses the subsequent path from this choice of paths. Upon user interaction, the subsequent path is chosen and immediately replayed from the fast cache memory. Jumping to the chosen path on the storage medium is slower and takes place while the cached data is being presented. The delay for accessing any path from the storage medium is among the determining factors for the cache memory size needed for the possible subsequent starting sequences. When the path from the storage medium is accessed, playing is subsequently continued from the storage medium until the next interactive decision appears. Hence, no time is lost after a decision.

According to aspects of the invention, a method, a storage medium, a device, and a computer-readable medium comprising a program code for speeding up interactive applications run from storage media are disclosed.

According to one aspect of the invention, a method for speeding up interactive applications run from storage media is provided, wherein a decision taken by a user determines a succession of interactive multimedia contents data sequences constituting said
interactive applications. The decision is a multiple-choice decision from a plurality of
subsequent interactive multimedia contents data sequences following a first interactive
multimedia contents data sequence. The multimedia contents data sequences are stored on
said storage media as data paths. The method comprises the following steps. Firstly, initial
sequences of each one of said plurality of subsequent interactive multimedia contents data
sequences are pre-loaded from the storage medium into a solid state memory at the end of
said first interactive multimedia contents data sequence and prior to the decision. Then the
chosen one of said plurality of subsequent interactive multimedia contents data sequences is
run, which comprises playing back a corresponding initial sequence from the memory and
simultaneously starting reading the remaining data sequence from the storage media with a
time delay. After the time delay, the chosen data sequence is played back from said storage
medium.

The present invention has the advantage over the prior art that it avoids
annoying time delays after decisions taken by users on how to continue a media application
after the decision.

BRIEF DESCRIPTION OF THE DRAWINGS

Further objects, features, and advantages of the invention will become
apparent from the following description of embodiments of the present invention, reference
being made to the accompanying drawings, in which

Fig. 1a is a schematic illustration of the optical disc system with user
interaction;

Fig. 1b is a schematic illustration of an interactive program comprising a
multiple path decision;

Fig. 2 is a schematic illustration of the physical arrangement of data stored on
a storage medium according to the prior art and corresponding to the decision illustrated in
Fig. 1;

Fig. 3 is a schematic timing diagram illustrating the time lost, according to the
prior art, between the taken decision and the display start of the next path;

Fig. 4 is a schematic illustration of the physical arrangement of data stored on
a storage medium according to an embodiment of the present invention and corresponding to
the decision illustrated in Fig. 1;
Fig. 5 is a schematic timing diagram illustrating that no time is lost between
the point of time when a decision is taken and the subsequent start of displaying the next
path;

Fig. 6 is a flowchart illustrating an embodiment of the method according to the
invention;

Fig. 7 is a schematic illustration of an embodiment of the apparatus according
to the invention; and

Fig. 8 is a schematic illustration of an embodiment of the computer-readable
medium according to the invention.

DESCRIPTION OF EMBODIMENTS

Fig. 1a is a schematic illustration of an optical disc system 100 with user
interaction. Data is read from a disc 101 and stored in a buffer 102. The data from the buffer
is decoded in a decoder 103 and presented on a display 104. The user watches the picture on
the display and makes a decision D when the picture displayed on display 104 asks for it. The
playback rate from the disc is generally higher than the leak rate from the buffer. A currently
typical value for the playback rate is 54 Mbps and the leak rate of the buffer is 10 Mbps, as
shown in Fig. 1a. Playback of the disc is interrupted when the buffer 102 is full and started
again when there is some empty space in the buffer 102.

Fig. 1b is a schematic illustration of an interactive program comprising a
multiple path decision. For instance, in an interactive gaming application 1 comprising a
sequence of alternative paths (P0, one of P1 to P4, ... Pn), a presentation of multimedia
contents (video/graphics) subsequent to a preceding path depends on a decision D. The
decision D has to be taken by the user of the application and is taken during playback of the
movie or game, as illustrated in Fig. 1, at the end of a path P0 and within a decision interval
d. d comprises information on which possible path may be followed after P0, i.e. any of P1 to
P4 in the current illustrative example. The decision is taken on how to continue the
interactive application after path P0, i.e. path P1, P2, P3, or P4. One of paths P1, P3, P4, and
P4 is chosen by decision D and the path chosen is played subsequently. The decision may be
taken, for example, by means of a menu or an appropriate action at the position of a cursor.

The data belonging to paths P1-P4 is stored on a storage medium, such as an
optical storage medium, in sequential order. Fig. 2 is a schematic illustration of this physical
arrangement of data stored on a storage medium according to the prior art and corresponding
to the decision illustrated in Fig. 1b.
The timing of the operation described above is elucidated with reference to Fig. 3. A schematic timing diagram illustrates the time lost, according to the prior art, between the taken decision and the display start of the next path. The decision D is taken from feedback from the presentation of path P0. After the decision has been taken, first a jump to path P3 is performed, which takes a jump delay time \( t_J \) until reading from path P3 starts. The delay time \( t_J \) is caused by read-access delays due to the mechanical inertia of disc reading devices. Thereafter path P3 is read, and the decoding of the data read from path P3 starts after a decoding delay time \( t_d \). Subsequently, the decoded data from path P3 is presented after a presentation delay time \( t_p \). The time between the decision taken from the presentation of path P0 and the subsequent presentation of exemplary path P3 is lost time \( t_i = (t_J + t_d + t_p) \).

In an embodiment of the invention according to Fig. 6, the time lost \( t_i = 0 \). More precisely, a method 6 is provided for speeding up interactive applications run from a storage media, such as an optical disc. The method starts at step 60 with playing back path P0. At the end of P0, a decision interval d is entered in step 61. While waiting for the decision D in step 65, the first parts of paths P1 until path P4 are read already and cached in a memory, as illustrated in step 62. The initial part parts have a length, for example, of one GOP interval, wherein GOP is an abbreviation for the term "Group Of Pictures". A GOP is a collection of consecutive frames of video. Usually 0.5 seconds of video will be held in 1 GOP. Preferably, the first picture of each path is decoded in step 63. The decision D is based on information presented in step 64, informing the user of the possible choices for continuing after path P0, in this case one of exemplary paths P1-P4. As soon as the decision D is taken, the method 6 continues after stating positively that the decision has been taken in step 65. The first picture or sequence, stored in memory, is shown during one (or more) frame interval(s) in step 66. At the same time, redundant information may optionally be removed in step 602, so that the read data from the non-selected paths may be removed. In the meantime, the rest of the cached pictures from the chosen path are decoded, and in step 67 a jump is made to the location on the disc of the rest of the data from this path. This data is read in step 68, decoded in step 69, and presented without interruption in step 600. In this case, the lost time is reduced zero, and the presentation starts immediately and continues without interruption after the decision D has been taken.

For implementing the above embodiment, the sequence as illustrated in Fig. 4 is stored on a disc. The physical arrangement of data stored on the disc corresponds to the decision illustrated in Fig. 1b. At the end of path P0, the decision interval is stored on the
disc. Then follow the initial parts of paths P1-P4, and sequentially thereafter the remaining parts of paths P1-P4. Thus, when path P3 is chosen, the disc reader still has to jump to the location of path P3. However, the time $t_1$ needed for this action is used for displaying the initial part of path P3, thus bridging the time until the disc reader continuously reads data from path P3.

The process is illustrated on a time basis in Fig. 5, wherein a schematic timing diagram further shows that no time is lost between the moment when a decision is taken and the subsequent start of displaying the next path, which in the example is path P3 following path P0. Some realistic exemplary figures for illustrating this situation are given next.

Suppose the GOP length is 0.5 seconds and that one GOP is present in each of the paths P1-P4, and the leak rate of each of the paths P1-P4 is 10 Mbps, then the resulting size of $P1+P2+P3+P4$ is 2.5 Mbytes ($4\times0.5$ seconds $= 2$ seconds gives $20$ Mbit $= 2.5$ Mbytes). Thus, it takes 0.37 second (54 Mbps / 20 Mbytes) to read this data with a bit rate of 54 Mbps. The decision D to be taken is presented after the data from P1-P4 has been read. There is a delay in the buffer and the decoder. The data read from P3 is presented during 0.5 second (= one GOP). In the meantime, a jump is made which takes about 400 msec (0.4 second), leaving about 0.1 second for synchronizing and transferring playback from the memory to the player.

The example of paths P0-P4 described above and the jump to path P3 from the P0 following decision D, are illustrated as strictly sequential without gaps, i.e. the storage of the different parts is shown as strictly sequential without gaps. This is merely for illustrative purposes and not intended to limit the embodiment or invention to sequentially stored data without gaps. The invention also supports storage with gaps inside or between the path data fragments drawn, permutation of pieces of data, or storage of the same piece of data multiple times in different locations on a disc. In particular, the initial parts of paths P1 to P4 may be stored multiple times in different convenient locations.

Furthermore, the four shorter initial parts of paths P1-P4, illustrated in Figs. 4 and 5, may be multiplexed into the data stream of path P0 at a low data rate. The description with reference to Figs. 4 and 5 shows a method without any multiplexing, because it is easier to draw, but this is not to exclude the multiplexing case.

Moreover, the small initial fragments of the alternative paths P1-P4 need not be located directly after path P0. These initial segments of paths P1-P4 may alternatively be located somewhere else on the disc. In certain situations, it may be useful to have the 'decision' block contain an instruction that the initial fragment of stream PX should be read
from the start of the long path PX stream, not from a location very close to the end of path P0.

Instead of having a single 'decision' block in a stream with data about the possible alternative paths that can be taken in future, this data may also be coded in a less localized manner, or be derived from the mere presence of small initial parts in the multiplex. The information on what paths to expect could also be supplied by a higher-level interactivity program that is also controlling the playback of the stream.

The above description assumes that the player is not playing video while waiting for a decision. An alternative is possible where video is still being played, and while the video is being played the jumping decision can be made at any moment. In this alternative, the path P0 video stream continues after the four small initial segments of the alternative paths P1-P4. These initial small parts are cached in memory, which memory may be freed after the jump has been done, so that it is available for further application. Very often the stream P0 is part of the streams [P0-P4] to be selected. P0 may be the default stream if no selection is made within a certain time. Those skilled in the art will realize that the stream P0 is presented continuously until a selection is made or if no selection is made.

An alternative solution is proposed, using a caching model in which initial parts are not removed after a decision, but only replaced, all of them or only some of them, when one or more new initial fragments are encountered in the stream being played currently, or when instructions to fetch such fragments from other locations are encountered. This in effect leads to a model that supports a mode of interactive content authoring where, at any point during video playback, one or more alternative paths are available for immediate jumping access. The encoding of such a model with certain fixed parameters, e.g. to provide independent cache locations each with a size of e.g. Y, has the advantage of creating a lot of power and flexibility for authors, while the investment for supporting it is fairly low. As an example, a Movie is separated in several chapters. A Menu control enables jumps in the Movie to be made to a next or previous chapter. The first GOP of the next or previous chapter may be multiplexed in the stream from the chapter. So, with a relatively small memory (about 1 Mbyte) the reaction time (time interval between making a choice and the presentation of the choice) can be reduced considerably.

The cache memory described above may be filled, not as a result of encountering data or instructions in the stream, but based on instructions supplied by a higher-level interactivity program that is also controlling the playback of the stream.
Execution of such instructions leads most often to a temporarily interrupted reading of the current stream in order to read a small part from somewhere else.

An alternative choice needing less implementation cost is to not pre-decode the first frame(s) of the initial small segments, but to start decoding only after the jump. This implies that there will be a decoding delay, but the jumping delay is at least eliminated, and in practice the coding delay will in generally be much shorter than the jumping delay. The decoding delay is of the order of 1 frame interval (about 40 msec) for current practical implementation.

Instead of pre-decoding the first frame(s) of the initial small segments at full resolution, it is also feasible to store and/or pre-decode these segments at a lower resolution in order to save implementation cost. Storage and decoding may even use a different encoding format than the format used for the long paths.

The characteristics of the present invention are summarized below.

During a decision interval it is known what the next paths can be or what the next paths most probably will be.

The first parts of the next paths (e.g. One GOP interval) are read and cached. They may be read from a location immediately following or close to the previous data, or as a lower-bit-rate part of the multiplex being played currently, or in some cases from a part of the disc that is not close to the current playback position.

The first parts contain sufficient video information for decoding and presenting live video while the jump is being made to the chosen path. Typically, the first part is a closed GOP or the initial part of a closed GOP.

In a cost-effective alternative, the first video frames of these initial parts are pre-decoded and stored.

Jumping to the path takes place while the first part of the path is being presented.

Fig. 7 illustrates schematically an embodiment of another aspect of the invention, namely an apparatus 7 for speeding up interactive applications run from a storage medium 71. The apparatus 7 comprises a reader 72 for said storage medium 71, solid state cache memory 73, control means 74 such as a CPU, and decision input means 75 such as a mouse or a remote control, which all are operatively connected to each other. A decision input by means of said decision input means determines a succession of interactive multimedia contents data sequences constituting said interactive applications. The decision is a multiple-choice decision from a plurality of subsequent interactive multimedia contents
data sequences P1-P4 stored on said storage medium following a first interactive multimedia contents data sequence P0 stored on said medium, wherein the multimedia contents data sequences are stored on said storage media as data paths. Before said decision input, the control means pre-loads initial sequences of all of said plurality of subsequent interactive multimedia contents data sequences P1-P4 from the storage medium into said solid state memory by means of said reader at the end of said first interactive multimedia contents data sequence P0 and prior to said decision. The control means runs the chosen one P3 of said plurality of subsequent interactive multimedia contents data sequences P1-P4 by playing back a corresponding initial sequence from said memory. Simultaneously, the control means initiates the reader to start reading the remaining data sequence from the storage media with a time delay \( t \). After the time delay has elapsed, the player plays back the chosen data sequence from said storage medium without time delay. Thus, the apparatus enables immediate play-back from a storage medium, such as an optical disc, after a decision input.

Fig. 8 schematically illustrates an embodiment of the computer-readable medium according to another aspect of the invention. A computer-readable medium 8 accommodates a computer program 80 for speeding up interactive applications run from storage media 71 for processing by a computer 81, which computer program comprises a plurality of code segments 82-85. A decision D taken by a user determines a succession of interactive multimedia contents data sequences constituting said interactive applications, and the decision is a multiple-choice decision from a plurality of subsequent interactive multimedia contents data sequences following a first interactive multimedia contents data sequence, wherein the multimedia contents data sequences is stored on said storage media as data paths. The computer-program comprises a first code segment 82 for pre-loading initial sequences of all of said plurality of subsequent interactive multimedia contents data sequences from said storage medium into a solid state memory at the end of said first interactive multimedia contents data sequence and prior to said decision. Furthermore, the computer program comprises a second code segment 83 for running the chosen one of said plurality of subsequent interactive multimedia contents data sequences, such that said second code segment plays back a corresponding initial sequence from said memory. Moreover, a third code segment 84 of said computer program starts, simultaneously with said second code segment, reading the remaining data sequence from the storage media with a time delay. And finally, a fourth code segment 85 plays back the chosen data sequence from said storage medium after said time delay. Thus, the computer program enables immediate play-back from a storage medium, such as an optical disc, after a decision input.
Applications and use of the above described aspects of the invention are various and include exemplary fields such as interactive Multi-Story movies, where the decisions taken during playing of the movie determine the end of the movie (e.g. happy-ending or not), or such as interactive games, for example for children, where the sequence of scenes is selected by means of icons on the screen.

The present invention was described above with reference to specific embodiments. However, embodiments other than the preferred one above are equally possible within the scope of the appended claims, e.g. different ways of arranging data on a disc, different solutions of reading data from disc, different storage media formats, such as CD, CD-R, CD-RW, DVD, DVD-ROM, Blu-Ray ROM disc, etc., or characteristics and features other than those described above, performing the above method by hardware or software, etc.

Furthermore, the term "comprises/comprising" when used in this specification does not exclude other elements or steps, the terms "a" and "an" do not exclude a plurality and a single processor or other units may fulfil the functions of several of the units or circuits recited in the claims.
CLAIMS:

1. A method of speeding up interactive applications run from storage media, wherein a decision taken by a user determines a succession of interactive multimedia contents data sequences constituting said interactive applications, wherein said decision is a multiple-choice decision from a plurality of subsequent interactive multimedia contents data sequences following a first interactive multimedia contents data sequence, wherein said multimedia contents data sequences are stored on said storage media as data paths; said method comprising the steps of: pre-loading respective initial sequences of all of said plurality of subsequent interactive multimedia contents data sequences from said storage medium into a solid state memory at the end of said first interactive multimedia contents data sequence and prior to said decision, running the chosen one of said plurality of subsequent interactive multimedia contents data sequences, said running data sequences comprising playing back a corresponding initial sequence from said memory and simultaneously starting reading the remaining data sequence from the storage media with a time delay, and after the time delay playing back the chosen data sequence from said storage medium.

2. The method according to claim 1, wherein said interactive application is an interactive video.

3. The method according to claims 1 or 2, said storage medium being a CD, DVD, Blu-Ray disc, or SFFO disc.

4. The method according to any of the preceding claims, said storage medium being a read-only medium.
5. The method according to any of the preceding claims, wherein at least a first image of each initial sequences of said plurality of subsequent data sequences from said storage medium stored into said memory is decoded and stored in a memory prior to playing back said chosen subsequent data sequence.

6. The method according to any of the preceding claims, wherein said initial sequence is at least one Group Of Pictures (GOP).

7. A storage medium (71) having written thereon interactive application data, wherein said data is stored sequentially distributed over said storage medium, said storage medium comprising a first interactive multimedia contents data sequence (P0) stored on said medium,

a plurality of subsequent interactive multimedia contents data sequences (P1-P4), and

initial sequences of all of said plurality of subsequent interactive multimedia contents data sequences stored in separate locations and derived from the respective corresponding subsequent interactive multimedia contents data sequence.

8. An apparatus (7) for speeding up interactive applications run from a storage medium (71), said apparatus comprising a reader (72) for said storage medium (71), a solid-state cache memory (73), control means (74), and decision input means (75), all of which are operatively connected to each other,

wherein a decision (D) input by means of said decision input means (75) determines a succession of interactive multimedia contents data sequences constituting said interactive applications,

which decision is a multiple-choice decision from a plurality of subsequent interactive multimedia contents data sequences (P1-P4) stored on said storage medium following a first interactive multimedia contents data sequence (P0) stored on said medium, said multimedia contents data sequences being stored on said storage medium as data paths,

and

before said decision (D) input, said control means (74) pre-loads respective initial sequences of all of said plurality of subsequent interactive multimedia contents data sequences (P1-P4) from said storage medium (71) by means of said reader (72) into said
solid state memory (73) at the end of said first interactive multimedia contents data sequence (P0) and prior to said decision (D), and

wherein said control-means (74) runs the chosen one (P3) of said plurality of subsequent interactive multimedia contents data sequences (P1-P4) by playing back a corresponding initial sequence from said memory (73) and simultaneously starting reading the remaining data sequence from the storage media with a time delay (t1) by means of said reader (72), and after the time delay the player plays back the chosen data sequence from said storage medium without time delay.

9. A computer-readable medium (8) having embodied thereon a computer program (80) for speeding up interactive applications run from a storage medium (71) for processing by a computer (81), the computer program (80) comprising a plurality of code segments (82-86),

wherein a decision (D) taken by a user determines a succession of interactive multimedia contents data sequences constituting said interactive applications,

wherein said decision (D) is a multiple-choice decision from a plurality of subsequent interactive multimedia contents data sequences following a first interactive multimedia contents data sequence,

wherein said multimedia contents data sequences are stored on said storage medium (71) as data paths,

said computer-program (80) comprising:

a first code segment (82) for pre-loading respective initial sequences of all of said plurality of subsequent interactive multimedia contents data sequences from said storage medium (71) into a solid state memory (73) at the end of said first interactive multimedia contents data sequence and prior to said decision,

a second code segment (83) for running the chosen one of said plurality of subsequent interactive multimedia contents data sequences, said second code segment playing back a corresponding initial sequence from said memory,

a third code segment (84) for starting reading, simultaneously with said second code segment, the remaining data sequence from the storage media with a time delay, and

a fourth code segment (85) for playing back the chosen data sequence from said storage medium after said time delay.
3/4
START

play back path P0

enter decision interval

read initial sequences of possible following paths P1-P4 into memory

decode initial sequences of possible following paths P1-P4 in memory

present choice

check if decision D is taken

NO

YES, P3

present initial sequence of chosen path P3

jump to path P3

start reading path P3

decode path P3

present path P3 from disc

remove redundant data from buffer

FIG. 6