

PROBLEM TO BE SOLVED: To provide a game device and a communication game system, allowing a player to play a communication game as if the player faces an opponent player of a game machine disposed in a remote place without any trouble in the latency (communication delay) generated in a real-time communication game, in the time-lag, or in the communication quality.

SOLUTION: Replay data for replaying the development of the game played by the player are downloaded via the network to regenerate the replay data, and the game play is executed by the player at the same time. The game is played between players not by receiving the information on the process of the play in real time, but by communicating the data on the result of the game by the player and the replay data, by which the development of the game can be replayed, after the game play, storing these data in a terminal game machine, and regenerating these stored replay data.

COPYRIGHT: (C)2008,JPO&INPIT

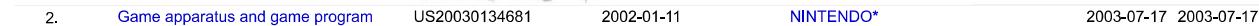
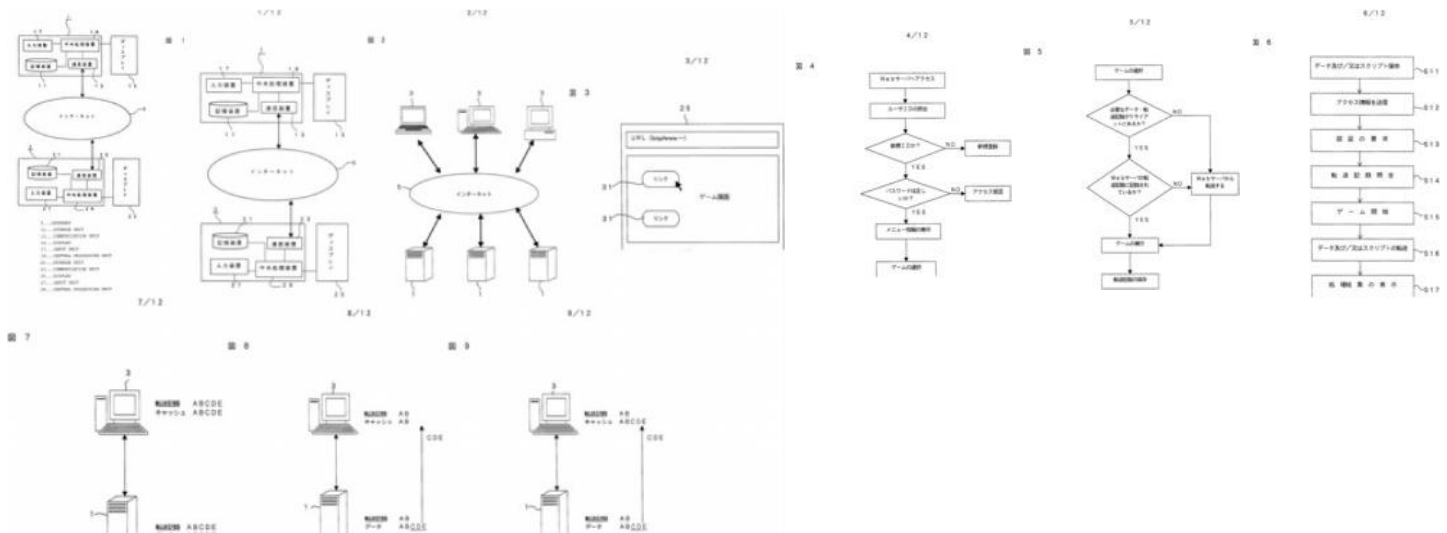
[illegible]

Figure 1 illustrates a multi-channel system architecture. The system is composed of several interconnected blocks and lines. At the top, a 'MULTI-MUX' block branches into four 'TO PROGRAM' lines (TO PROGRAM 1, TO PROGRAM 2, TO PROGRAM 3, TO PROGRAM 30). These lines connect to a 'CHANNEL SELECT' block, which then branches into four 'CHANNEL' lines (CHANNEL 1, CHANNEL 2, CHANNEL 3, CHANNEL 6). These channels connect to a 'MULTI-DEMUX' block at the bottom, which branches into four 'FROM AREA' lines (DC AREA, CPU AREA, SRAM AREA, CONTROLLER AREA). These lines then connect to four 'FROM' blocks (FROM 1, FROM 2, FROM 3, FROM 4). The 'FROM' blocks are connected to a 'MULTI-DEMUX' block at the bottom, which branches into four 'TO AREA' lines (TO AREA, TO AREA, TO AREA, TO AREA). These lines connect to a 'MULTI-MUX' block at the top, which branches into four 'TO PROGRAM' lines (TO PROGRAM 1, TO PROGRAM 2, TO PROGRAM 3, TO PROGRAM 30). The 'TO PROGRAM' lines are connected to a 'MULTI-DEMUX' block at the bottom, which branches into four 'FROM AREA' lines (DC AREA, CPU AREA, SRAM AREA, CONTROLLER AREA). These lines connect to four 'FROM' blocks (FROM 1, FROM 2, FROM 3, FROM 4). The 'FROM' blocks are connected to a 'MULTI-DEMUX' block at the bottom, which branches into four 'TO AREA' lines (TO AREA, TO AREA, TO AREA, TO AREA). These lines connect to a 'MULTI-MUX' block at the top, which branches into four 'TO PROGRAM' lines (TO PROGRAM 1, TO PROGRAM 2, TO PROGRAM 3, TO PROGRAM 30).



A method for providing a game using the Internet of a mode in which the charge for the play of a game is paid while provoking a possibility that the user purchases a game soft played past and preventing the user from making an unauthorized access. The method comprises the step of storing the data and/or script (hereinafter referred to as data) necessary to progress a game in storage means (11) of a Web server or file server (1) (hereinafter referred to as the Web server), the step of setting up connection with the Web server (1) by a client (3), the step of displaying the data as one scene of the game on a screen (15) of the client (3), the step of selecting a specific piece of the data stored in the Web server on the screen, the step of transferring the data from the Web server (1) to the client (3), and the step of displaying the transferred data again on the screen (15) by a browser incorporated in the client (3) thereby to progress the game.



PROBLEM TO BE SOLVED: To allow a player to enjoy a game by selecting a favorable piece among music pieces widely distributed through a distribution medium for music appreciation, and to use the distribution medium for music appreciation in a composite manner. SOLUTION: A CPU reads MIDI data from a CD-ROM containing the MIDI data before the starting of a game (S4). Then, from the read MIDI data, the CPU extracts sound quality track data specified by a player (S6), and generates the rhythm pattern of a music piece based on the extracted sound quality track data (S7). Then, the CPU evaluates a timing for the entry of an input operation signal from a controller or the like based on the generated rhythm pattern (S11) while the music piece is played and outputted based on the read MIDI data (S9).

