

Desktop Client, Client Platform and Game entity in Desktop Multi-person Network Game System

Field of the Invention

The present invention relates to network communication fields, and more particularly to a desktop client, client platform and game entity in a desktop multi-person network game system.

Background of the Invention

With the development of Internet, more and more users communicate with each other on the Internet. Desktop multi-person network games are provided for the users, and in the desktop multi-person network game, the game is directly running on the desktop without any running window displayed on the desktop. In order to make users can play a multi-person game on the Internet, conventional desktop multi-person network games are usually configured based on a C/S structure including a desktop client and a game server, as shown in Figure 1. The game server is configured to decode data flows from the desktop client, verify whether the data flows is legal, process a game key logic, update data, such as game state and player information etc., and send new data to the desktop client through a Transmission Control Protocol (TCP) or a User Datagram Protocol (UDP). The desktop client is configured to respond to a user operation, execute a game client logic, decode a executing result into a binary flow, send the binary flow to the game server through the TCP or the UDP, update game output according to the data returned by the game server, and return updated game output to the user, referring to Figure 2.

Since the desktop client in the conventional technical solution is developed separately for each game, when the game changes, the desktop client can not be compatible with the changed game and needs to be developed again, thereby increasing the development period and development costs, which is difficult to meet the requirements of rapid development, publication and application of desktop multi-person network game products.

Summary of the Invention

The examples of the present invention provide a desktop multi-person network game system to reduce development period and development costs of the desktop multi-person network game system.

5 An example of the present invention provides a desktop client in a desktop multi-person network game system, and the desktop client includes: a client platform and a game entity; wherein

the client platform, configured to establish a data channel between a game server and the game entity, load and operate the game entity, forward an operation command of a user to the game entity, receive and execute game output data transmitted by the game entity, and provide an application interface for the game entity; and

10 the game entity, configured to configure a game logic and a game protocol, receive the operation command of the user forwarded by the client platform, generate the game output data according to the game logic, transmit the game output data to the client platform, and exchange information with the game server through the data channel.

15 Another example of the present invention provides a client platform in a desktop multi-person network game system, and the client platform includes:

a game engine, configured to load and operate a game entity, receive an operation command of a user, forward the operation command to the game entity, receive and execute game output data transmitted by the game entity, and output a game output result

20 to the user;
an application interface unit, configured to provide an application interface for the game entity; and

a network communication unit, configured to implement communication between the client platform and a game server, and communicate with the game entity through the application interface provided by the application interface unit.

25 Another example of the present invention provides a game entity in a desktop multi-person network game system, and the game entity includes:

a game logic unit, configured to configure a game processing logic;

a game protocol unit, configured to configure a game operating protocol;

30 a network protocol unit, configured to configure a communication protocol between a client platform and the game entity; and

a game logic/protocol processing unit, configured to receive an operation command of a user forwarded by the client platform, invoke the game processing logic in the game logic unit, generate game output data, transmit the game output data to the client platform, invoke an application interface provided by the client platform, trigger the client platform
5 to execute a corresponding operation, communicate with a game server through the application interface provided by the client platform.

In the examples of the present invention, the desktop client is divided into the client platform and the game entity; when a game changes, it is only needed to change the game logic and the game protocol in the game entity, and not needed to redesign the client
10 platform, thereby reducing the development period and development costs, and meeting the requirements of rapid development, publication and application of desktop game products.

Brief Description of the Drawings

Figure 1 is a schematic diagram illustrating a structure of a conventional desktop
15 multi-person game system.

Figure 2 is a schematic diagram illustrating an operation of a conventional desktop client.

Figure 3 is a schematic diagram illustrating a structure of a desktop multi-person game system according to an example of the present invention.

20 Figure 4 is a flowchart illustrating interaction between a client platform and a game entity according to an example of the present invention.

Figure 5 is a schematic diagram illustrating a structure of a client platform and a game entity according to an example of the present invention.

Detailed Description of the Invention

5 In order to make the object, technical solutions and merits of the present invention clearer, the present invention will be illustrated in detail hereinafter with reference to the accompanying drawings and examples. It should be understood that the examples described herein are only used to explain the present invention, but not used to limit the present invention.

10 In the examples of the present invention, a desktop client is divided into a client platform and a game entity; when a game changes, it is only needed to change a game logic and a game protocol in the game entity, and not needed to redesign the client platform, so as to reduce the development period and development costs.

15 Figure 3 is a schematic diagram illustrating a structure of a desktop multi-person game system according to an example of the present invention. In order to be convenient for description, only parts related to the examples of the present invention are shown.

a game server 12 processes a game rule, manages user information and parses a communication protocol between the game server 12 and a game entity 16;

20 a client platform 14 establishes a data channel between the game server 12 and the game entity 16, loads and operates the game entity 16, forwards an operation command of a user to the game entity 16, receives and executes game output data transmitted by the game entity 16, outputs a executing result to the user, and provide an application interface for the game entity 16 to implement functions which can not be implemented by the game entity 16 itself.

25 In the examples of the present invention, the game output includes characters, images, sound and so on, and further includes physical output, e.g. game handle shake.

The game entity 16 configures a game logic and a game protocol, receives the operation command of the user forwarded by the client platform 14, generate the game output data according to the game logic, transmit the game output data to the client

platform 14, and exchange information with the game server 12 through the data channel established by the client platform 14.

The client platform 14 communicates with the game server 12 through the TCP/UDP.

5 In the examples of the present invention, in order to make different game carries can communicate with the game server 12 through the data channel established by the same client platform to reduce the development workload of the client platform 14, the client platform 14 communicates with different game entities through a uniform communication protocol. The uniform communication protocol may be self-designed, or a conventional
10 communication protocol is adopted, e.g. a JavaScript Object Notation (JSON) protocol.

 The JSON protocol adopts a flexible data exchange format, is easy to read and write, and is easy to be parsed and generated by a machine. The JSON protocol adopts a text format independent with languages, but adopts habits of C language family including C, C++, C#, Java, JavaScript, Perl and Python, and thus becomes an ideal data exchange
15 language.

 Figure 4 is a flowchart illustrating interaction between the client platform 14 and the game entity 16 according to an example of the present invention, which is described as follows.

 1) A user inputs an operation command through a mouse, a keyboard or a game
20 handle.

 2) The client platform receives the operation command of the user.

 3) The client platform forwards the operation command of the user to the game entity.

 4) The game entity processes the operation command of the user according to a
25 corresponding game logic, and generates game output data.

 5) The game entity transmits the game output data to the client platform.

 6) The client platform updates a game output according to the game output data of

the game entity, e.g. draws a game image, and issues a game music adjusting instruction and a game handle shake instruction.

7) The client platform transmits the updated game output to the user. In this way, the game client may display the drawn game image and play the adjusted music, or the game handle is triggered to shake.

Figure 5 is a schematic diagram illustrating a structure of a client platform and a game entity according to an example of the present invention. In order to be convenient for description, only parts related to the examples of the present invention are shown.

A game engine 142 loads and operates the game entity 16, receives an operation command of a user, and forwards the operation command to the game entity 16.

A game logic/protocol processing unit 162 receives the operation command of the user forwarded by the game engine 142, invokes a corresponding game processing logic in a game logic unit 164, generates game output data, and transmits the game output data to the client platform 14.

The game engine 142 executes the game output data after receiving the game output data transmitted by the game logic 162, e.g. draws a game interface and adjusts game sound, and outputs a game output result to the user.

An application interface unit 144 provides an application interface for the game entity 16 to implement functions which can not be implemented by the game entity 16.

As an example of the present invention, the application interface unit 144 may provide four types of application interfaces, which include:

a first type of application interface, which is used to directly operate a game desktop window, e.g. change the size of the window, stick the window on the top, hide or display the window and so on;

a second type of application interface, which is used to communicate with a related service logic, e.g. obtain information of a related service logic such as a QQ pet, open a chat window and so on;

a third type of application interface, which is a specific application interface specially used to a certain game or module, e.g. visit and invite a friend in a friend system;

5 a fourth type of application interface, which is specially used to make the game entity directly communicate with the game server 12.

The game logic/protocol processing unit 162 invokes a corresponding application interface provided by the application interface unit 144, triggers the game engine 142 to execute a corresponding operation, e.g. change the inside state of the game, or communicate with the game server 12 through the application interface provided by the application interface unit 144, and transmit related information.

The network communication unit 146 implements the communication between the client platform 14 and the game server 12 through the TCP or UDP, and communicates with the game entity 16 through the application interface provided by the application interface unit 144.

15 The game logic unit 164 configures a game processing logic, e.g. a motion mode, motion direction and position of a game role, drawing parameters of a graphical interface and so on.

The game processing logic generally includes following three types:

- (1) transmitting a message to the game server 12;
- 20 (2) updating game output, e.g. image output, sound output and physical output such as game handle shake;
- (3) changing the inside state of the game.

The game protocol unit 164 configures a game operation protocol.

25 In the examples of the present invention, the game operation protocol is a rule for decoding and encoding a command corresponding to the protocol. The information transmitted between the game server 12 and the game entity 16 is a series of original byte flows, and in the byte flows corresponding to different commands, the meanings of one

byte is different from that of another byte. Therefore, each command needs to correspond to a protocol to determine a rule for decoding and encoding the command. For example, in a game logging protocol, an account number and password for logging are encoded into byte flows, after the byte flows are processed through a network protocol between
5 the client platform 14 and the game entity 16 which is configured in a network protocol unit 168, e.g. the JSON protocol, the application interface provided by the application interface unit 144 is invoked, and the byte flows is transmitted to the game server 12 through the network communication unit 146.

The game server 12 transmits a message to the game entity 16 through the network
10 communication unit 146, the network protocol unit 168 invokes the application interface provided by the application interface unit 144, receives the message transmitted by the network communication unit 146, and decodes the protocol through the game protocol unit 166. The game logic/protocol processing unit 162 triggers the game engine 142 according to the decoded message to execute corresponding game output, or
15 communicates with the game server 12.

Those skilled in the art can understand that all or part of steps in the method provided by the examples of the present invention can be implemented by instructing related hardware by a program, the program may be stored in a readable memory of a computer, and above method steps are included when the program is operated. The
20 memory includes a ROM/RAM, a disk, a Compact Disc (CD) and so on.

In the examples of the present invention, the desktop client is divided into the client platform and the game entity, when the game changes, it is only needed to change the game logic and the game protocol in the game entity, but not needed to redesign the client platform, thereby reducing the development period and development costs, and meeting
25 the requirements of rapid development, publication and application of desktop game products.

The foregoing are only preferred examples of the present invention and are not for use in limiting the protection scope of the present invention. Any modification, equivalent replacement and improvement made within the scope of the present invention should be
30 covered under the protection scope of the present invention.

Claims

What is claimed is:

1. A desktop client in a desktop multi-person network game system, comprising a client platform and a game entity; wherein

5 the client platform, configured to establish a data channel between a game server and the game entity, load and operate the game entity, forward an operation command of a user to the game entity, receive and execute game output data transmitted by the game entity, and provide an application interface for the game entity; and

10 the game entity, configured to configure a game logic and a game protocol, receive the operation command of the user forwarded by the client platform, generate the game output data according to the game logic, transmit the game output data to the client platform, and exchange information with the game server through the data channel.

2. The desktop client of claim 1, wherein the client platform comprises:

15 a game engine, configured to load and operate the game entity, receive the operation command of the user, forward the operation command to the game entity, receive and execute the game output data transmitted by the game entity, and output a game output result to the user;

20 an application interface unit, configured to provide the application interface for the game entity; and

 a network communication unit, configured to implement communication between the client platform and the game server, and communicate with the game entity through the application interface provided by the application interface unit.

25 3. The desktop client of claim 1, wherein the game entity comprises:

 a game logic unit, configured to configure a game processing logic;

 a game protocol unit, configured to configure a game operating protocol;

 a network protocol unit, configured to configure a communication protocol between the client platform and the game entity; and

30 a game logic/protocol processing unit, configured to receive the operation command of the user forwarded by the client platform, invoke the game processing logic in the

game logic unit, generate the game output data, transmit the game output data to the client platform, invoke the application interface provided by the client platform, trigger the client platform to execute a corresponding operation, communicate with the game server through the application interface provided by the client platform.

5

4. The desktop client of claim 1, wherein the client platform is configured to communicate with different game entities through a uniform communication protocol.

5. The desktop client of claim 4, wherein the client platform is configured to
10 communicate with different game entities through a JSON protocol.

6. The desktop client of any of claims 1 to 5, wherein the application interface comprises:

an interface for directly operating a game desktop window;
15 an interface for communicating with a service logic;
a special interface for a game or a module; and
an application interface for direct communication between the game entity and the game server.

20 7. A client platform in a desktop multi-person network game system, comprising:
a game engine, configured to load and operate a game entity, receive an operation command of a user, forward the operation command to the game entity, receive and execute game output data transmitted by the game entity, and output a game output result to the user;

25 an application interface unit, configured to provide an application interface for the game entity; and

a network communication unit, configured to implement communication between the client platform and a game server, and communicate with the game entity through the application interface provided by the application interface unit.

30

8. The client platform of claim 7, wherein the application interface comprises:

an interface for directly operating a game desktop window;
an interface for communicating with a service logic;

a special interface for a game or a module; and
an application interface for direct communication between the game entity and the game server.

5 9. The client platform of claim 7, wherein the client platform is configured to communicate with different game entities through a uniform communication protocol.

10 10. The client platform of claim 9, wherein the client platform is configured to communicate with different game entities through a JSON protocol.

10

11. A game entity in a desktop multi-person network game system, comprising:
a game logic unit, configured to configure a game processing logic;
a game protocol unit, configured to configure a game operating protocol;
a network protocol unit, configured to configure a communication protocol between

15 a client platform and the game entity; and

a game logic/protocol processing unit, configured to receive an operation command of a user forwarded by the client platform, invoke the game processing logic in the game logic unit, generate game output data, transmit the game output data to the client platform, invoke an application interface provided by the client platform, trigger the client platform
20 to execute a corresponding operation, communicate with a game server through the application interface provided by the client platform.

25 12. The game entity of claim 11, wherein the game entity and other game entity are configured to communicate with the client platform through a uniform communication protocol.

13. The game entity of claim 12, wherein the uniform communication protocol is a JSON protocol.