CLAIMS

1. A method for updating tile maps in virtual maps, comprising:

determining, in a first area, a central area that a virtual object is located in (S410), wherein:

the first area is an area of a map that comprises tile maps loaded into a virtual map that the virtual object is located in,

a border of the central area is located in the first area,

a maximum distance between any tile map in the central area and the virtual object in an x-axis direction in a tile map coordinate system of the virtual map is less than X_1, wherein X_1 represents a predetermined positive integer,

a maximum distance between any tile map in the central area and the virtual object in a y-axis direction in the tile map coordinate system is less than Y_1, wherein Y_1 represents a predetermined positive integer,

a length of one tile map is used as a unit on the x-axis in the tile map coordinate system, and

a width of one tile map is used as a unit on the y-axis in the tile map coordinate system; and

determining a distance parameter after the virtual object moves out of the central area, and updating a loading status of a tile map not shared by the first area and a second area based on the distance parameter and the first area (S420), wherein:

the distance parameter indicates relative distances of a tile map that the virtual object is located in relative to the central area in the x-axis direction and the y-axis direction in the tile map coordinate system, and

the second area is a map area determined after moving the first area based on the distance parameter.

2. The method according to claim 1, wherein determining a distance parameter comprises:

determining that the distance parameter is (1, 0) when the virtual object moves to a tile map that is on the right of the central area and adjacent to the central area; or

determining that the distance parameter is (1, 1) when the virtual object moves to a tile map that is at the upper right of the central area and adjacent to the central area; or

determining that the distance parameter is (0, 1) when the virtual object moves to a tile map that is right above the central area and adjacent to the central area; or

determining that the distance parameter is (−1, 1) when the virtual object moves to a tile map that is at the upper left of the central area and adjacent to the central area; or

determining that the distance parameter is (−1, 0) when the virtual object moves to a tile map that is on the left of the central area and adjacent to the central area; or

determining that the distance parameter is (−1, −1) when the virtual object moves to a tile map that is at the lower left of the central area and adjacent to the central area; or

determining that the distance parameter is (0, −1) when the virtual object moves to a tile map that is right below the central area and adjacent to the central area; or

determining that the distance parameter is (1, −1) when the virtual object moves to a tile map that is at the lower right of the central area and adjacent to the central area, wherein

the tile map is a rectangular area, the distance parameter (a, b) indicates that the central area moves to the right by a tile maps and upward by b tile maps, and x and y are integers.

3. The method according to claim 1 or 2, wherein determining a distance parameter after the virtual object moves out of the central area, and updating a loading status of a tile map not shared by the first area and a second area based on the distance parameter and the first area comprises:

determining the distance parameter after the virtual object moves out of the central area for a time duration that exceeds a preset time, and updating the loading status of the tile map not shared by the first area and the second area based on the distance parameter and the first area.

4. The method according to any one of claims 1 to 3, wherein updating a loading status of a tile map not shared by the first area and a second area comprises:

loading added tile maps in the second area relative to tile maps in the first area, and removing reduced tile maps in the second area relative to tile maps in the first area.

5. The method according to any one of claims 1 to 4, wherein updating a loading status of a tile map not shared by the first area and a second area based on the distance parameter and the first area comprises:

determining an index of a tile map that needs to be updated based on the distance

parameter and indexes of tile maps in the first area; and

updating a loading status of a tile map of the virtual object in the virtual map based on the index of the tile map that needs to be updated.

6. The method according to any one of claims 1 to 5, wherein X_1 is equal to Y_1.

7. The method according to any one of claims 1 to 6, wherein both the first area and the central area have a same shape.

8. The method according to any one of claims 1 to 7, wherein before determining a distance parameter, and updating a loading status of a tile map not shared by the first area and a second area based on the distance parameter and the first area, the method further comprises:

determining corresponding map coordinates of the virtual object in the virtual map based on latitude and longitude of an actual object that corresponds to the virtual object in an actual environment, wherein the virtual map is generated based on an actual electronic map.

9. The method according to any one of claims 1 to 8, wherein before determining a distance parameter, and updating a loading status of a tile map not shared by the first area and a second area based on the distance parameter and the first area, the method further comprises:

determining a corresponding action parameter of the virtual object in the virtual map and operating the virtual object based on an action parameter of the actual object that corresponds to the virtual object in the actual environment.

10. The method according to claim 9, wherein the virtual map comprises a 3D game map and the virtual object comprises a game character in a 3D game, and wherein determining a corresponding action parameter of the virtual object in the virtual map and operating the virtual object based on an action parameter of the actual object that corresponds to the virtual object in the actual environment comprises:

determining a parameter related to movement of a part of a user corresponding to the game character, wherein the movement of the part of the user is captured using a sensor; and operating the game character based on the determined parameter.

11. The method according to any one of claims 1 to 10, further comprising determining that the virtual object moves out of the central area by determining a moving direction and moving distance of the virtual object in the virtual map based on a moving direction and a moving distance of an actual object that corresponds to the virtual object in an actual environment.

12. The method according to any one of claims 1 to 11, further comprising generating the tile maps in the virtual map, comprising:

determining a map area of a virtual map based on a scaling relationship between an actual electronic map and the virtual map;

dividing the map area into a plurality of equally sized tile areas; and

performing image processing on each tile area based on an area map that corresponds to the tile area in the actual electronic map, to generate a tile map of the tile area, wherein the image processing comprises rendering processing.

13. The method according to claim 12, wherein performing image processing on each tile area based on an area map that corresponds to the tile area in the actual electronic map, to generate a tile map of the tile area comprises:

determining a first actual area map that corresponds to a first tile area in the plurality of equally sized tile areas in the actual electronic map based on a size of the tile area and an index of the first tile area; and

performing image processing on the first tile area based on the first actual area map to obtain a first tile map that corresponds to the first tile area.

14. An apparatus for drawing a virtual map, comprising multiple units configured to perform operations according to the method of any one of claims 1 to 13.