



(12) 发明专利

(10) 授权公告号 CN 111569426 B

(45) 授权公告日 2021.08.24

(21) 申请号 202010519399.X

审查员 谭辉桐

(22) 申请日 2020.06.09

(65) 同一申请的已公布的文献号

申请公布号 CN 111569426 A

(43) 申请公布日 2020.08.25

(73) 专利权人 网易(杭州)网络有限公司

地址 310000 浙江省杭州市滨江区长河街
道网商路599号4幢7层

(72) 发明人 陈家豪

(74) 专利代理机构 北京博浩百睿知识产权代理

有限责任公司 11134

代理人 赵昀彬

(51) Int. Cl.

A63F 13/60 (2014.01)

A63F 13/5252 (2014.01)

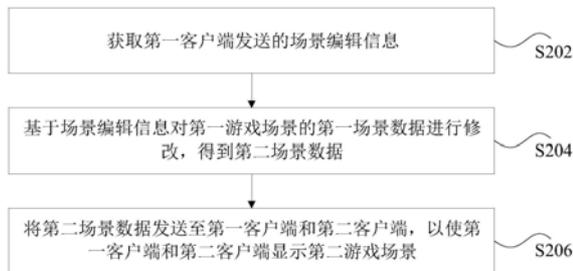
权利要求书4页 说明书16页 附图3页

(54) 发明名称

游戏场景的编辑方法、装置、存储介质和电子装置

(57) 摘要

本发明公开了一种游戏场景的编辑方法、装置、存储介质和电子装置。该编辑方法包括：获取第一客户端发送的场景编辑信息；基于场景编辑信息对第一游戏场景的第一场景数据进行修改，得到第二场景数据，其中，第二场景数据用于将第一游戏场景更新为第二游戏场景；将第二场景数据发送至第一客户端和第二客户端，以使第一客户端和第二客户端显示第二游戏场景。通过本发明，达到了提高了对游戏场景进行编辑的效率的效果。



1. 一种游戏场景的编辑方法,其特征在于,包括:

获取第一客户端发送的场景编辑信息;

基于所述场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,其中,所述第二场景数据用于将所述第一游戏场景更新为第二游戏场景;

将所述第二场景数据发送至所述第一客户端和第二客户端,以使所述第一客户端和所述第二客户端显示所述第二游戏场景;

所述场景编辑信息为由处于目标编辑模式下的所述第一客户端发送,所述目标编辑模式包括以下至少之一:

近景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;

远景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑的模式,所述预设的第二距离大于所述预设的第一距离;

全局参数编辑模式,用于针对所述第一游戏场景的全局参数进行编辑的模式。

2. 根据权利要求1所述的编辑方法,其特征在于,所述场景编辑信息为由所述第一客户端对所述第一游戏场景中的对象数据进行运算得到。

3. 根据权利要求1所述的编辑方法,其特征在于,在获取第一客户端发送的场景编辑信息之前,所述方法还包括:

获取所述第一客户端对应的虚拟摄像机的位置信息;

将所述第一客户端对应的虚拟摄像机的位置信息发送至所述第二客户端,使得所述第二客户端对应的虚拟摄像机的位置调整至与所述第一客户端对应的虚拟摄像机的位置一致;

其中,每个客户端呈现的游戏场景为每个客户端对应的所述虚拟摄像机捕获的游戏场景画面。

4. 根据权利要求1所述的编辑方法,其特征在于,在获取第一客户端发送的场景编辑信息之前,所述方法还包括:

获取所述第一客户端当前的游戏场景中的每个对象的状态信息;

将所述第一客户端当前的游戏场景中的每个对象的状态信息同步至所述第二客户端。

5. 根据权利要求1所述的编辑方法,其特征在于,在基于所述场景编辑信息对所述第一游戏场景的第一场景数据进行修改,得到第二场景数据之后,所述方法还包括:

对所述第二场景数据进行备份。

6. 根据权利要求5所述的编辑方法,其特征在于,在将所述第二场景数据发送至所述第一客户端和第二客户端之后,所述方法还包括:

在所述第二游戏场景更新为第三游戏场景的情况下,响应于场景恢复请求,将备份的所述第二场景数据发送至所述第一客户端和所述第二客户端,使得所述第一客户端和所述第二客户端将当前的第三游戏场景更新为所述第二游戏场景。

7. 根据权利要求1至6中任意一项所述的编辑方法,其特征在于,所述方法还包括:

获取所述第一客户端发送的语音信息;

将所述语音信息同步至所述第二客户端。

8. 一种游戏场景的编辑方法,其特征在于,包括:

第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器,其中,所述场景编辑信息用于使服务器对第一游戏场景的第一场景数据进行修改得到第二场景数据,并将所述第二场景数据发送至客户端,所述客户端包括所述第一客户端和第二客户端,所述第二场景数据用于将所述第一游戏场景更新为第二游戏场景;

所述第一客户端接收所述服务器发送的所述第二场景数据,并基于所述第二场景数据将显示的所述第一游戏场景,更新为第二游戏场景;

第一客户端针对第一游戏场景进行编辑,包括:

响应于针对编辑模式的选择操作,确定目标编辑模式;

基于所述目标编辑模式,对所述第一游戏场景进行编辑,获得场景编辑信息;

所述目标编辑模式包括以下至少之一:

近景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;

远景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑的模式,所述预设的第二距离大于所述预设的第一距离;

全局参数编辑模式,用于针对所述第一游戏场景的全局参数进行编辑的模式;

浏览模式,用于禁止对所述第一游戏场景中的对象进行编辑的模式。

9. 根据权利要求8所述的编辑方法,其特征在于,所述第一客户端针对第一游戏场景进行编辑,包括:

所述第一客户端对所述第一游戏场景中的对象数据进行运算,得到所述场景编辑信息。

10. 根据权利要求8所述的编辑方法,其特征在于,在所述第一客户端向服务器发送场景编辑信息之前,所述方法还包括:

所述第一客户端将对应的虚拟摄像机的位置信息发送至所述服务器,使得所述服务器将所述第一客户端对应的虚拟摄像机的位置信息同步至所述第二客户端,以及使得所述第二客户端将对应的虚拟摄像机的位置调整至与所述第一客户端对应的虚拟摄像机的位置一致;

其中,每个客户端呈现的游戏场景为每个客户端对应的所述虚拟摄像机捕获的游戏场景画面。

11. 根据权利要求8所述的编辑方法,其特征在于,在所述第一客户端接收所述服务器发送的所述第二场景数据,并基于所述第二场景数据将显示的所述第一游戏场景,更新为第二游戏场景之后,所述方法还包括:

在所述第一客户端由所述第二游戏场景更新为第三游戏场景的情况下,向所述服务器发送场景恢复请求,其中,所述场景恢复请求用于请求获取所述服务器已备份的所述第二场景数据;

所述第一客户端接收所述服务器发送的第二场景数据,并基于所述第二场景数据将显示的所述第三游戏场景,更新为所述第二游戏场景。

12. 根据权利要求10所述的编辑方法,其特征在于,所述基于所述目标编辑模式,对所述第一游戏场景进行编辑时,所述方法还包括:

确定所述第一客户端当前编辑的所述第一游戏场景的目标对象,将所述目标对象设置

为锁定状态,其中,所述目标对象在所述锁定状态下不能被除所述第一客户端以外的客户端编辑。

13. 根据权利要求12所述的编辑方法,其特征在于,所述方法还包括:

响应于所述编辑模式的切换操作或者响应于当前游戏场景的更新,解除所述目标对象的锁定状态。

14. 根据权利要求11所述的编辑方法,其特征在于,所述确定目标编辑模式之后,所述方法还包括:

在所述第一客户端上呈现与所述目标编辑模式对应的标识信息。

15. 一种游戏场景的编辑装置,其特征在于,包括:

获取单元,用于获取第一客户端发送的场景编辑信息;

编辑单元,用于基于所述场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,其中,所述第二场景数据用于将所述第一游戏场景更新为第二游戏场景;

第一发送单元,用于将所述第二场景数据发送至所述第一客户端和第二客户端,以使所述第一客户端和所述第二客户端显示所述第二游戏场景;

所述场景编辑信息为由处于目标编辑模式下的所述第一客户端发送,所述目标编辑模式包括以下至少之一:

近景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;

远景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑的模式,所述预设的第二距离大于所述预设的第一距离;

全局参数编辑模式,用于针对所述第一游戏场景的全局参数进行编辑的模式。

16. 一种游戏场景的编辑装置,其特征在于,包括:

第二发送单元,用于使第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器,其中,所述场景编辑信息用于使服务器对第一游戏场景的第一场景数据进行修改得到第二场景数据,并将所述第二场景数据发送至客户端,所述客户端包括所述第一客户端和第二客户端,所述第二场景数据用于将所述第一游戏场景更新为第二游戏场景;

接收单元,用于使所述第一客户端接收所述服务器发送的所述第二场景数据,并基于所述第二场景数据将显示的所述第一游戏场景,更新为第二游戏场景;

所述第二发送单元还用于响应于针对编辑模式的选择操作,确定目标编辑模式;基于所述目标编辑模式,对所述第一游戏场景进行编辑,获得场景编辑信息;

所述目标编辑模式包括以下至少之一:

近景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;

远景编辑模式,用于针对所述第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑的模式,所述预设的第二距离大于所述预设的第一距离;

全局参数编辑模式,用于针对所述第一游戏场景的全局参数进行编辑的模式;

浏览模式,用于禁止对所述第一游戏场景中的对象进行编辑的模式。

17. 一种存储介质,其特征在于,所述存储介质中存储有计算机程序,其中,在所述计算机程序被处理器运行时控制所述存储介质所在设备执行所述权利要求1至7中任一项中所

述的方法,或8至14中任一项中所述的方法。

18.一种电子装置,包括存储器和处理器,其特征在于,所述存储器中存储有计算机程序,所述处理器被设置为运行所述计算机程序以执行所述权利要求1至7中任一项中所述的方法,或8至14中任一项中所述的方法。

游戏场景的编辑方法、装置、存储介质和电子装置

技术领域

[0001] 本发明涉及计算机领域,具体而言,涉及一种游戏场景的编辑方法、装置、存储介质和电子装置。

背景技术

[0002] 目前,在游戏引擎中对游戏场景进行编辑时,如果多个客户端同时进行编辑,则会出现资源冲突的问题。因而,对于一个游戏场景的开发,同一时间仅允许一个客户端进行编辑。通常是设定一个先后顺序,串行地对游戏场景进行编辑,在一个客户端对游戏场景编辑完,另一个客户端才能对游戏场景继续进行编辑。

[0003] 上述方案在有多个游戏场景的情况下,能够避免其中一个人等候另一个人而浪费的时间。但是在只有一个游戏场景的情况下,这种由于等候而浪费时间的问题就变得比较明显和严重了,而在项目进度非常紧急的时候,这种开发方式是非常低效的,从而存在对游戏场景进行编辑的效率低的问题。

[0004] 针对现有技术中对游戏场景进行编辑的效率低的问题,目前尚未提出有效的解决方案。

发明内容

[0005] 本发明的主要目的在于提供一种游戏场景的编辑方法、装置、存储介质和电子装置,以至少解决游戏场景进行编辑的效率低的技术问题。

[0006] 为了实现上述目的,根据本发明的一个方面,提供了一种游戏场景的编辑方法。该编辑方法可以包括:获取第一客户端发送的场景编辑信息;基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,其中,第二场景数据用于将第一游戏场景更新为第二游戏场景;将第二场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第二游戏场景。

[0007] 可选地,场景编辑信息为由第一客户端对第一游戏场景中的对象数据进行运算得到。

[0008] 可选地,在获取第一客户端发送的场景编辑信息之前,该编辑方法还包括:获取第一客户端对应的虚拟摄像机的位置信息;将第一客户端对应的虚拟摄像机的位置信息发送至第二客户端,使得第二客户端对应的虚拟摄像机的位置调整至与第一客户端对应的虚拟摄像机的位置一致;其中,每个客户端呈现的游戏场景为每个客户端对应的虚拟摄像机捕获的游戏场景画面。

[0009] 可选地,在获取第一客户端发送的场景编辑信息之前,该编辑方法还包括:获取第一客户端当前的游戏场景中的每个对象的状态信息;将第一客户端当前的游戏场景中的每个对象的状态信息同步至第二客户端。

[0010] 可选地,在基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据之后,该编辑方法还包括:对第二场景数据进行备份。

[0011] 可选地,在将第二场景数据发送至第一客户端和第二客户端之后,该编辑方法还包括:在第二游戏场景更新为第三游戏场景的情况下,响应于场景恢复请求,将备份的第二场景数据发送至第一客户端和第二客户端,使得第一客户端和第二客户端将当前的第三游戏场景更新为第二游戏场景。

[0012] 可选地,场景编辑信息为由处于目标编辑模式下的第一客户端发送。

[0013] 可选地,目标编辑模式包括以下至少之一:近景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;远景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑的模式,预设的第二距离大于预设的第一距离;全局参数编辑模式,用于针对基于第一游戏场景的全局参数进行编辑的模式。

[0014] 可选地,该编辑方法还包括:获取第一客户端发送的语音信息;将语音信息同步至第二客户端。

[0015] 为了实现上述目的,根据本发明的另一方面,还提供了另一种游戏场景的编辑方法。该编辑方法可以包括:第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器,其中,场景编辑信息用于使服务器对第一游戏场景的第一场景数据进行修改得到第二场景数据,并将第二场景数据发送至客户端,客户端包括第一客户端和第二客户端,第二场景数据用于将第一游戏场景更新为第二游戏场景;第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景。

[0016] 可选地,第一客户端针对第一游戏场景进行编辑,该编辑方法还包括:第一客户端对第一游戏场景中的对象数据进行运算,得到场景编辑信息。

[0017] 可选地,在第一客户端向服务器发送场景编辑信息之前,该编辑方法还包括:第一客户端将对应的虚拟摄像机的位置信息发送至服务器,使得服务器将第一客户端对应的虚拟摄像机的位置信息同步至第二客户端,以及使得第二客户端将对应的虚拟摄像机的位置调整至与第一客户端对应的虚拟摄像机的位置一致,其中,每个客户端呈现的游戏场景为每个客户端对应的所述虚拟摄像机捕获的游戏场景画面。

[0018] 可选地,在第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景之后,该编辑方法还包括:在第一客户端由第二游戏场景更新为第三游戏场景的情况下,向服务器发送场景恢复请求,其中,场景恢复请求用于请求获取服务器已备份的第二场景数据;第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第三游戏场景,更新为第二游戏场景。

[0019] 可选地,第一客户端针对第一游戏场景进行编辑,包括:响应于针对编辑模式的选择操作,确定目标编辑模式;基于目标编辑模式,对第一游戏场景进行编辑,获得场景编辑信息。

[0020] 可选地,目标编辑模式包括以下至少之一:近景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;远景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑的模式,预设的第二距离大于预设的第一距离;全局参数编辑模式,用于针对第一游戏场景的全局参数进行编辑的模式;浏览模式,用于禁止对第一游戏场景中的对象进行编辑的模式。

[0021] 可选地,基于目标编辑模式,对第一游戏场景进行编辑时,该编辑方法还包括:确定第一客户端当前编辑的第一游戏场景的目标对象,将目标对象设置为锁定状态,其中,目标对象在锁定状态下不能被除第一客户端以外的客户端编辑。

[0022] 可选地,该编辑方法还包括:响应于编辑模式的切换操作或者响应于当前游戏场景的更新,解除目标对象的锁定状态。

[0023] 可选地,在第一客户端上呈现与目标编辑模式对应的标识信息。

[0024] 为了实现上述目的,根据本发明的另一方面,提供了一种游戏场景的编辑装置。该装置可以包括:获取单元,用于获取第一客户端发送的场景编辑信息;编辑单元,用于基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,其中,第二场景数据用于将第一游戏场景更新为第二游戏场景;第一发送单元,用于将第二场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第二游戏场景。

[0025] 为了实现上述目的,根据本发明的另一方面,还提供了一种游戏场景的编辑装置。该装置可以包括:第二发送单元,用于使第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器,其中,场景编辑信息用于使服务器对第一游戏场景的第一场景数据进行修改得到第二场景数据,并将第二场景数据发送至客户端,客户端包括第一客户端和第二客户端,第二场景数据用于将第一游戏场景更新为第二游戏场景;接收单元,用于使第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景。

[0026] 为了实现上述目的,根据本发明的另一方面,提供了一种存储介质。该存储介质中存储有计算机程序,其中,在计算机程序被处理器运行时控制存储介质所在设备执行本发明实施例的游戏场景的编辑方法。

[0027] 为了实现上述目的,根据本发明的另一方面,提供了一种电子装置。该电子装置包括存储器和处理器,存储器中存储有计算机程序,处理器被设置为运行计算机程序以执行本发明实施例的游戏场景的编辑方法。

[0028] 该实施例通过获取第一客户端发送的场景编辑信息;基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,其中,第二场景数据用于将第一游戏场景更新为第二游戏场景;将第二场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第二游戏场景。也就是说,该实施例的服务器获取客户端同步过来的场景编辑信息,来修改游戏场景的场景数据,然后将修改后的场景数据分发给编辑该游戏场景的多个客户端,保证每个客户端呈现的都是目前最新的游戏场景,实现了多人同时编辑同一个场景的目的,避免了串行对游戏场景进行编辑所浪费的等候时间,也避免了将游戏场景划分多个子区域而容易导致合并出错和不能及时沟通的问题,从而解决了对游戏场景进行编辑的效率低的技术问题,达到了提高对游戏场景进行编辑的效率的技术效果。

附图说明

[0029] 构成本申请的一部分的附图用来提供对本发明的进一步理解,本发明的示意性实施例及其说明用于解释本发明,并不构成对本发明的不当限定。在附图中:

[0030] 图1是本发明实施例的一种游戏场景的编辑方法的移动终端的硬件结构框图;

[0031] 图2是根据本发明实施例的一种游戏场景的编辑方法的流程图;

[0032] 图3是根据本发明实施例的另一种游戏场景的编辑方法的流程图；

[0033] 图4是根据本发明实施例的一种每个客户端对应的虚拟摄像机存在的方式和呈现的内容的示意图；

[0034] 图5是根据本发明实施例的一种游戏场景的编辑装置的示意图；以及

[0035] 图6是根据本发明实施例的另一种游戏场景的编辑装置的示意图。

具体实施方式

[0036] 需要说明的是，在不冲突的情况下，本申请中的实施例及实施例中的特征可以相互组合。下面将参考附图并结合实施例来详细说明本发明。

[0037] 为了使本技术领域的人员更好地理解本申请方案，下面将结合本申请实施例中的附图，对本申请实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本申请一部分的实施例，而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都应当属于本申请保护的范畴。

[0038] 需要说明的是，本申请的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别类似的对象，而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换，以便这里描述的本申请的实施例。此外，术语“包括”和“具有”以及他们的任何变形，意图在于覆盖不排他的包含，例如，包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元，而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

[0039] 本申请实施例所提供的方法实施例可以在移动终端、计算机终端或者类似的运算装置中执行。以运行在移动终端上为例，图1是本发明实施例的一种游戏场景的编辑方法的移动终端的硬件结构框图。如图1所示，移动终端可以包括一个或多个(图1中仅示出一个)处理器102(处理器102可以包括但不限于微处理器MCU或可编程逻辑器件FPGA等的处理装置)和用于存储数据的存储器104，可选地，上述移动终端还可以包括用于通信功能的传输设备106以及输入输出设备108。本领域普通技术人员可以理解，图1所示的结构仅为示意，其并不对上述移动终端的结构造成限定。例如，移动终端还可包括比图1中所示更多或者更少的组件，或者具有与图1所示不同的配置。

[0040] 存储器104可用于存储计算机程序，例如，应用程序的软件程序以及模块，如本发明实施例中的一种数据处理的方法对应的计算机程序，处理器102通过运行存储在存储器104内的计算机程序，从而执行各种功能应用以及数据处理，即实现上述的方法。存储器104可包括高速随机存储器，还可包括非易失性存储器，如一个或者多个磁性存储装置、闪存、或者其他非易失性固态存储器。在一些实例中，存储器104可进一步包括相对于处理器102远程设置的存储器，这些远程存储器可以通过网络连接至移动终端。上述网络的实例包括但不限于互联网、企业内部网、局域网、移动通信网及其组合。

[0041] 传输设备106用于经由一个网络接收或者发送数据。上述的网络具体实例可包括移动终端的通信供应商提供的无线网络。在一个实例中，传输设备106包括一个网络适配器(Network Interface Controller, 简称为NIC)，其可通过基站与其他网络设备相连从而可与互联网进行通讯。在一个实例中，传输设备106可以为射频(Radio Frequency, 简称为RF)

模块,其用于通过无线方式与互联网进行通讯。

[0042] 下面从服务器一侧对在本实施例的游戏场景的编辑方法进行介绍。图2是根据本发明实施例的一种游戏场景的编辑方法的流程图。如图2所示,该编辑方法可以包括如下步骤:

[0043] 步骤S202,获取第一客户端发送的场景编辑信息。

[0044] 在本发明上述步骤S202提供的技术方案中,在游戏编辑场景下,第一客户端为在游戏场景开发过程中,正在对当前的第一游戏场景进行编辑的客户端,可以是正在进行第一游戏场景编辑工作的工作人员所使用的客户端,可以称为场景编辑器。其中,工作人员可以包括多个美术或策划人员。

[0045] 该实施例的服务器与上述第一客户端之间建立通信连接,可以获取第一客户端发送的场景编辑信息,该场景编辑信息用于对第一游戏场景进行编辑。可选地,场景编辑信息可以包括第一客户端在对第一游戏场景进行编辑操作时的操作信息和编辑内容。

[0046] 步骤S204,基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据。

[0047] 在本发明上述步骤S204提供的技术方案中,在获取第一客户端发送的场景编辑信息之后,基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,第二场景数据用于将第一游戏场景更新为第二游戏场景。

[0048] 在该实施例中,服务器获取第一游戏场景的第一场景数据,该第一场景数据是用于生成第一游戏场景的数据,可以包括但不限于第一游戏场景中的地形参数、灯光方向参数、灯光强度参数、模型参数、烘焙参数等,服务器通过场景编辑信息对第一场景数据进行修改,将修改后的第一场景数据确定为第二场景数据,以通过第二场景数据生成第二游戏场景。

[0049] 步骤S206,将第二场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第二游戏场景。

[0050] 在本发明上述步骤S206提供的技术方案中,在基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据之后,服务器可以将第二场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第二游戏场景。

[0051] 在该实施例中,服务器可以将第二场景数据分发至第一客户端和第二客户端,该第二客户端可以为其它用于对游戏场景进行编辑的客户端。可选地,该实施例的第一客户端和第二客户端在接收第二场景数据之前,可以统一显示第一游戏场景,而在接收到第二场景数据之后,通过第二场景数据生成第二游戏场景,并将原来显示的第一游戏场景更新为第二游戏场景,此时,第一客户端和第二客户端均显示第二游戏场景,从而使得第一客户端和第二客户端都可以显示游戏场景开发项目的最新编辑结果,也即,保证每个客户端的具体表现是一致的,从而保证每个工作人员均可以看到游戏场景开发项目最新的编辑进度,进而基于此调整自己对游戏场景进行编辑的工作内容。

[0052] 在该实施例中,用于对游戏场景进行编辑的任意客户端可以随时对当前的游戏场景进行编辑操作以产生场景编辑信息,服务器会在接收到场景编辑信息的情况下对当前的游戏场景的场景数据进行修改,得到修改后的场景数据,进而将修改后的场景数据分发至用于对游戏场景进行编辑的所有的客户端,使其均显示由修改后的场景数据生成的新的游

戏场景,从而实现了多个客户端在一定时间内同时编辑一个游戏场景的目的,避免了由于串行对游戏场景进行编辑所浪费的等候时间。

[0053] 需要说明的是,该实施例的第一游戏场景并不是只针对第一客户端,用于对游戏场景进行编辑的每个客户端都是有第一游戏场景。

[0054] 在相关技术的游戏场景编辑方法中,通常将一个游戏场景划分成多个子区域,每个客户端编辑其中一个子区域,最后再将编辑好的子区域整合起来形成一个完整的游戏场景。但是,该编辑方法看似能够让多个客户端同时在一个游戏场景上进行编辑,但是实际上是将多个客户端需要编辑的子区域划分开来,一个客户端并不能实时显示另一个客户端的编辑结果,以调整自己的工作。这样的做法相当于将一个大游戏场景划分成了几个小场景,最后还需要将几个小场景合并起来,而在合并的过程中也容易出现,并且合并这个操作在不同的游戏引擎上的实现方式也不一样,从而影响了游戏场景的编辑效率,在反复修改当前游戏场景时也会遇到很多新的问题,从而进一步导致对游戏进行编辑的效率低的技术问题。

[0055] 然而,在本申请的游戏场景编辑方法中,通过上述步骤S202至步骤S206,获取第一客户端发送的场景编辑信息;基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,其中,第二场景数据用于将第一游戏场景更新为第二游戏场景;将第二场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第二游戏场景。也就是说,该实施例的客户端将场景编辑信息同步到服务器上来修改游戏场景的场景数据,然后编辑该游戏场景的多个客户端都可以接收到服务器分发的修改后的场景数据,保证每个客户端呈现的都是目前最新的游戏场景,实现了多人同时编辑同一个场景的目的,避免了串行对游戏场景进行编辑所浪费的等候时间,也避免了将游戏场景划分多个子区域而容易导致合并出错和不能及时沟通的问题,从而解决了对游戏场景进行编辑的效率低的技术问题,达到了提高对游戏场景进行编辑的效率的技术效果。

[0056] 下面对该实施例的上述方法进行进一步介绍。

[0057] 作为一种可选的实施方式,场景编辑信息为由第一客户端对第一游戏场景中的对象数据进行运算得到。

[0058] 在该实施例的游戏场景编辑中,可以采用帧同步的技术,在第一客户端上对第一游戏场景中的对象数据进行逻辑和计算,得到场景编辑信息,服务器获取第一客户端上传的计算好的场景编辑信息,以对第一游戏场景的第一场景数据进行修改,得到第二场景数据。其中,对象数据可以是在对第一游戏场景中的对象进行物理碰撞的计算时所用到的数据。

[0059] 需要说明的是,该实施例针对游戏场景编辑,采用帧同步而不采用状态同步的原因是,在对第一游戏场景进行编辑操作的过程中,会存在很多与物理碰撞相关的计算,这些计算的计算量是比较大的,如果将每个客户端的这些计算量直接放到服务器,则服务器的计算压力是非常大的,而且还会有比较大的延迟,所以将这些物理碰撞相关的计算放到客户端来计算会更好,以提高对游戏场景进行编辑的效率。

[0060] 作为一种可选的实施方式,在步骤S202,获取第一客户端发送的场景编辑信息之前,该编辑方法还包括:获取第一客户端对应的虚拟摄像机的位置信息;将第一客户端对应的虚拟摄像机的位置信息发送至第二客户端,使得第二客户端对应的虚拟摄像机的位置调

整至与第一客户端对应的虚拟摄像机的位置一致；其中，每个客户端呈现的游戏场景为每个客户端对应的虚拟摄像机捕获的游戏场景画面。

[0061] 在该实施例中，在第一客户端的编辑模式下，具有虚拟摄像机，服务器可以获得第一客户端对应的虚拟摄像机的位置信息，该位置信息决定了第一客户端通过虚拟摄像机所呈现的游戏场景画面，可以是不同视角下的游戏场景画面，包括位移旋转信息和视锥体信息，上述游戏场景画面包括了第一客户端需要编辑的内容。

[0062] 举例而言，第一客户端为客户端A，其用于呈现俯视角，该俯视角下的游戏场景画面包括了客户端A需要编辑的游戏场景；第一客户端为客户端B，其用于呈现左视角，该左视角下的游戏场景画面包括了客户端B需要编辑的游戏场景；第一客户端为客户端C，其用于呈现右视角，该右视角下的游戏场景画面包括了客户端C需要编辑的游戏场景。

[0063] 在服务器获取第一客户端对应的虚拟摄像机的位置信息之后，可以通过网络同步的功能，将该第一客户端对应的虚拟摄像机的位置信息同步至第二客户端，第二客户端将对应的虚拟摄像机的位置调整至与第一客户端对应的虚拟摄像机的位置一致，从而第二客户端可以通过该位置信息显示上述第一客户端在对应视角下的第一游戏场景，进而可以实时确定上述第一客户端对游戏场景开发项目的编辑进度，来调整第二客户端对游戏场景开发项目的工作内容，从而提高了对游戏场景的编辑效率。

[0064] 作为一种可选的实施方式，在步骤S202，获取第一客户端发送的场景编辑信息之前，该编辑方法还包括：获取第一客户端当前的游戏场景中的每个对象的状态信息；将第一客户端当前的游戏场景中的每个对象的状态信息同步至第二客户端。

[0065] 在该实施例中，第一客户端当前的游戏场景中包括多个对象，该对象可以是当前的游戏场景中的物体。在获取第一客户端发送的场景编辑信息之前，服务器可以获得第一客户端当前的游戏场景中的每个对象的状态信息，比如，获取每个对象在当前的游戏场景中的位移旋转缩放信息，将其同步到第二客户端，使得第二客户端与第一客户端呈现的第一游戏场景中的对象的状态保持一致。

[0066] 作为一种可选的实施方式，在步骤S204，基于场景编辑信息对第一游戏场景的第一场景数据进行修改，得到第二场景数据之后，该编辑方法还包括：对第二场景数据进行备份。

[0067] 在该实施例中，在服务器基于场景编辑信息对第一游戏场景的第一场景数据进行修改，得到第二场景数据之后，为了避免第二场景数据丢失，服务器还可以对第二场景数据进行备份。可选地，可以设置一定的定时时间，该实施例的服务器可以按照定时时间定时对第二场景数据进行备份，以便后期对其进行查询。

[0068] 需要说明的是，该实施例将第二场景数据备份在同一服务器上，以在每个客户端需要获取第二场景数据时，可以保证获取到的数据是一致的。

[0069] 作为一种可选的实施方式，在步骤S206，将第二场景数据发送至第一客户端和第二客户端之后，该编辑方法还包括：在第二游戏场景更新为第三游戏场景的情况下，响应于场景恢复请求，将备份的第二场景数据发送至第一客户端和第二客户端，使得第一客户端和第二客户端将当前的第三游戏场景更新为第二游戏场景。

[0070] 在该实施例中，在第二游戏场景更新为第三游戏场景的情况下，比如，服务器获取第一客户端(或者为其它客户端)发送的场景编辑信息，基于场景编辑信息对第二游戏场景

的第二场景数据进行修改,得到第三场景数据,该第三场景数据用于将第二游戏场景更新为第三游戏场景,服务器将第三场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第三游戏场景,但这可能是由于第一客户端的不慎操作而导致的将第二游戏场景更新为第三游戏场景的结果。由于该实施例的服务器已经对第二场景数据进行了备份,因而在服务器接收到客户端发送的场景恢复请求的情况下,服务器可以响应该场景恢复请求,将已经备份的第二场景数据发送至第一客户端和第二客户端,这样第一客户端和第二客户端就可以基于已经备份的第二场景数据恢复显示第二游戏场景,从而避免了因为某个其它客户端的不慎操作,将已编辑好的场景全部删掉,或者将别人的编辑好的场景不小心覆盖的情况出现,从而可以提高游戏场景的编辑效率。

[0071] 作为一种可选的实施方式,场景编辑信息为由处于目标编辑模式下的第一客户端发送。

[0072] 在该实施例的游戏场景的编辑方法中,第一客户端可以处于目标编辑模式,服务器接收到的场景编辑信息可以是由处于不同的目标编辑模式下的客户端发送过来的,其中,目标编辑模式可以用于指示每个客户端对游戏场景的功能进行编辑的权限。

[0073] 作为一种可选的实施方式,目标编辑模式包括以下至少之一:近景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;远景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑的模式,预设的第二距离大于预设的第一距离;全局参数编辑模式,用于针对第一游戏场景的全局参数进行编辑的模式。下面以第一客户端进行举例说明,

[0074] 在该实施例中,第一客户端可以设置的目标编辑模式包括近景编辑模式、远景编辑模式和全局参数编辑模式。其中,当第一客户端处于近景编辑模式下时,第一客户端可以针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式,其中,与对应的虚拟摄像机之间的距离小于预设的第一距离,也即,第一客户端对应的虚拟摄像机的视锥体内近处的距离,比如,第一客户端可以对视锥体内近处一定距离内的物体进行编辑,产生场景编辑信息;当第一客户端处于远景编辑模式时,第一客户端可以针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第二距离的对象进行编辑,从而生成场景编辑信息,其中,与对应的虚拟摄像机之间的距离小于预设的第二距离,也即,客户端对应的虚拟摄像机的视锥体内远处的距离,比如,第一客户端可以对能看到的远处场景的天空盒等远景物体进行编辑,从而生成场景编辑信息,需要说明的是,该实施例的远景编辑机模式并不是只能对远处的对象进行编辑,而是指除了近景的对象可以编辑之外,远景的对象也可以编辑;当第一客户端处于全局参数编辑模式时,第一客户端可以针对第一游戏场景的全局参数进行编辑,从而生成场景编辑信息,第一客户端可以在对游戏场景进行编辑的过程中修改游戏场景的全局参数的变量,比如,修改游戏场景的灯光方向、灯光的强度、全局的烘焙参数等。

[0075] 可选地,该实施例的第一客户端可以随意自由地切换目标编辑模式,从而提高对游戏场景进行编辑的灵活度。

[0076] 可选地,该实施例的第一客户端还可以具有浏览模式,在浏览模式下,第一客户端不能编辑游戏场景中的任何对象,但第一客户端可以获取由服务器同步过来的其它客户端

的虚拟摄像机的位置信息,可以显示其它客户端在对应视角下的第一游戏场景,从而可以看到别的客户端的编辑操作是如何进行的,这样可以方便某个工作人员想要参考和学习其他人通过客户端编辑的游戏场景,或者是编辑游戏场景所使用的手法和技巧,为后续对游戏场景进行编辑而做铺垫,从而提高了对游戏场景进行编辑的效率。

[0077] 需要说明的是,在该实施例的游戏场景的编辑中,上述目标编辑模式具有不同类型的原因是,每个客户端对游戏场景进行编辑的内容和关注的点都是不一样的,每个客户端在对游戏场景进行编辑的时候使用不同的模式,可以在不影响开发效率的前提下,避免冲突操作。

[0078] 作为一种可选的实施方式,该编辑方法还包括:获取第一客户端发送的语音信息;将语音信息同步至第二客户端。

[0079] 在该实施例的游戏场景编辑中,还可以引入语音通话功能。可选地,服务器获取第一客户端发送的语音信息,该语音信息可以是与游戏场景的编辑相关的信息,服务器将语音信息同步至第二客户端,从而使得第二客户端接收到语音信息,这样每个客户端不仅可以同时对游戏场景进行编辑,并且还可以与其它客户端实时交流,从而进一步提高游戏场景的开发效率以及降低游戏场景在开发过程中的冲突的风险。

[0080] 下面从客户端一侧对本发明实施例的游戏场景的编辑方法进行介绍。

[0081] 图3是根据本发明实施例的另一种游戏场景的编辑方法的流程图。如图3所示,该编辑方法可以包括以下步骤:

[0082] 步骤S302,第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器。

[0083] 在本发明上述步骤S302提供的技术方案中,场景编辑信息用于使服务器对第一游戏场景的第一场景数据进行修改得到第二场景数据,并将第二场景数据发送至客户端,该客户端包括第一客户端和第二客户端,第二场景数据用于将第一游戏场景更新为第二游戏场景。

[0084] 该实施例的第一客户端与服务器之间建立通信连接,可以向服务器发送场景编辑信息。可选地,该场景编辑信息包括第一客户端在对第一游戏场景进行编辑操作时的操作信息和编辑内容,其用于使服务器对第一游戏场景进行编辑,可以使服务器对第一游戏场景的第一场景数据进行修改,得到第二场景数据。其中,第一场景数据是用于生成第一游戏场景的数据,可以包括但不限于地形参数、灯光方向参数、灯光强度参数、模型参数、烘焙参数等,第二场景数据可以用于生成第二游戏场景。上述第二场景数据可以由服务器发送中客户端,该客户端包括上述第一客户端和第二客户端。

[0085] 步骤S304,第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景。

[0086] 在本发明上述步骤S304提供的技术方案中,在第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器之后,第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景。

[0087] 第二场景数据由服务器发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示的第一游戏场景更新为第二游戏场景。

[0088] 在该实施例中,第二客户端可以为所有用于对游戏场景进行编辑的客户端中除第

一客户端之外的其它客户端,该第一客户端和第二客户端均可以获取服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景,此时,第一客户端和第二客户端均显示第二游戏场景,从而使得第一客户端和第二客户端都可以显示游戏场景开发项目的最新编辑结果,也即,保证每个客户端的具体表现是一致的,保证每个工作人员均可以看到游戏场景开发项目最新的编辑进度,进而基于此调整自己的工作内容。

[0089] 在该实施例中,第一客户端和第二客户端都可以随时对第一游戏场景进行编辑操作以产生场景编辑信息,该场景编辑信息会使服务器对第一游戏场景的第一场景数据进行修改,得到第二场景数据,第二客户端接收服务器分发的第二场景数据,并显示第二游戏场景,从而实现了多个客户端在一定时间内同时编辑一个游戏场景的目的,避免了由于串行对游戏场景进行编辑所浪费的等候时间。

[0090] 本申请通过上述步骤S302和步骤S304,第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器,其中,场景编辑信息用于使服务器对第一游戏场景的第一场景数据进行修改得到第二场景数据,并将第二场景数据发送至客户端,客户端包括第一客户端和第二客户端,第二场景数据用于将第一游戏场景更新为第二游戏场景;第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景。也就是说,该实施例的服务器获取第一客户端同步过来的场景编辑信息,修改游戏场景的场景数据,然后将修改后的场景数据分发给编辑该游戏场景的第一客户端和第二客户端,从而保证每个客户端呈现的都是目前最新的游戏场景,实现了多人同时编辑同一个场景的目的,避免了串行对游戏场景进行编辑所浪费的等候时间,也避免了将游戏场景划分多个子区域而容易导致合并出错和不能及时沟通的问题,从而解决了对游戏场景进行编辑的效率低的技术问题,达到了提高对游戏场景进行编辑的效率的技术效果。

[0091] 下面对该实施例的上述方法进行进一步介绍。

[0092] 作为一种可选的实施方式,在步骤S302,第一客户端针对第一游戏场景进行编辑,包括:第一客户端对第一游戏场景中的对象数据进行运算,得到场景编辑信息。

[0093] 在该实施例的游戏场景编辑中,采用帧同步的技术,第一客户端对第一游戏场景中的对象数据进行逻辑和计算,将计算好的场景编辑信息上传至服务器,以使服务器获取该场景编辑信息,以对第一游戏场景的第一场景数据进行修改,得到第二场景数据。其中,对象数据可以是第一客户端在对第一游戏场景中的对象进行物理碰撞的计算时所用到的数据。

[0094] 作为一种可选的实施方式,在步骤S302,第一客户端向服务器发送场景编辑信息之前,该编辑方法还包括:第一客户端将对应的虚拟摄像机的位置信息发送至服务器,使得服务器将第一客户端对应的虚拟摄像机的位置信息同步至第二客户端,以及使得第二客户端将对应的虚拟摄像机的位置调整至与第一客户端对应的虚拟摄像机的位置一致,其中,每个客户端呈现的游戏场景为每个客户端对应的虚拟摄像机捕获的游戏场景画面。

[0095] 在该实施例中,第一客户端呈现的游戏场景为第一客户端对应的虚拟摄像机捕获的游戏场景画面,第二客户端呈现的游戏场景第二客户端对应的虚拟摄像机捕获的游戏场景画面。下面以第一客户端进行介绍。第一客户端对应的虚拟摄像机用于捕获对应视角下的游戏场景画面并呈现在第一客户端上,并且该视角下的游戏场景画面包括第一客户端需要编辑的内容。举例而言,第一客户端对应的虚拟摄像机用于呈现第一客户端对应的俯视

视角下的游戏场景画面,并且该俯视视角下的游戏场景画面包括了客户端需要编辑的内容;第一客户端对应的虚拟摄像机用于呈现第一客户端对应的左视角下的游戏场景画面,并且该左视角下的游戏场景画面包括第一客户端需要编辑的游戏场景画面;第一客户端对应的虚拟摄像机用于呈现第一客户端对应的右视角下的游戏场景画面,并且该右视角下的游戏场景画面包括客户端需要编辑的内容。

[0096] 该实施例的第一客户端会将对应的虚拟摄像机的位置信息发送至服务器,由服务器进一步同步至第二客户端,从而使得第二客户端将对应的虚拟摄像机的位置调整至与第一客户端对应的虚拟摄像机的位置一致,这样第二客户端可以通过该第一位置信息显示上述第一客户端在对应的视角下的游戏场景画面,进而可以实时确定第一客户端对游戏场景开发项目的编辑进度,来调整自己对游戏场景开发项目的工作内容,从而提高了对游戏场景的编辑效率。

[0097] 作为一种可选的实施方式,在步骤S304,第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景之后,该编辑方法还包括:在第一客户端由第二游戏场景更新为第三游戏场景的情况下,向服务器发送场景恢复请求,其中,场景恢复请求用于请求获取服务器已备份的第二场景数据;第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第三游戏场景,更新为第二游戏场景。

[0098] 在该实施例中,为了避免第二场景数据丢失,该第二场景数据可以定时在服务器中进行备份。在该实施例中,在第二游戏场景更新为第三游戏场景的情况下,比如,第一客户端(或者为其它客户端)向服务器发送场景编辑信息,使得服务器基于场景编辑信息对第二游戏场景的第二场景数据进行编辑,得到第三场景数据,第一客户端和第二客户端获取服务器下发的第三场景数据,将原来显示的第二游戏场景更新第三游戏场景,但这可能是由于第一客户端的不慎操作而导致的将第二游戏场景更新为第三游戏场景的结果。由于该实施例的第二场景数据已经在服务器中进行了备份,因而在客户端向服务器发送场景恢复请求的情况下,第一客户端和第二客户端可以获取服务器响应该场景恢复请求发送的备份的第二场景数据,这样第一客户端和第二客户端可以恢复显示第二游戏场景,从而避免了因为某个其它客户端的不慎操作,将已编辑好的场景全部删掉,或者将别人的编辑好的场景不小心覆盖的情况出现,从而可以提高游戏场景的编辑效率。

[0099] 作为一种可选的实施方式,步骤S302,第一客户端针对第一游戏场景进行编辑,包括:响应于针对编辑模式的选择操作,确定目标编辑模式;基于目标编辑模式,对第一游戏场景进行编辑,获得场景编辑信息。

[0100] 在该实施例中,第一客户端可以处于不同的编辑模式,可以响应针对编辑模式的选择操作,从不同的编辑模式中选择目标编辑模式,并将第一客户端当前所处的编辑模式切换至目标编辑模式,该目标编辑模式可以用于指示第一客户端对游戏场景进行编辑的权限,进而第一客户端在目标编辑模式下,对第一游戏场景进行编辑操作,将产生操作信息和编辑内容确定为场景编辑信息,进而向服务器发送场景编辑信息。

[0101] 作为一种可选的实施方式,目标编辑模式包括以下至少之一:近景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式;远景编辑模式,用于针对第一游戏场景内与对应的虚拟摄像机之间的距离小于

预设的第二距离的对象进行编辑的模式,预设的第二距离大于预设的第一距离;全局参数编辑模式,用于基于第一游戏场景的全局参数生成场景编辑信息;浏览模式,用于禁止对第一游戏场景中的对象进行编辑的模式。

[0102] 在该实施例中,当第一客户端处于上述近景编辑模式下时,第一客户端可以针对第一游戏场景内与对应的虚拟摄像机之间的距离小于预设的第一距离的对象进行编辑的模式,其中,与对应的虚拟摄像机之间的距离小于预设的第一距离,也即,第一客户端对应的虚拟摄像机的视锥体内近处的距离,比如,第一客户端可以对视锥体内近处一定距离内的物体进行编辑,从而产生场景编辑信息。

[0103] 当第一客户端处于上述远景编辑模式时,第一客户端可以针对第一游戏场景内与对应的虚拟摄像机之间的距离大于预设的第二距离的对象进行编辑,从而生成场景编辑信息,其中,对应的虚拟摄像机,也即,客户端对应的虚拟摄像机的视锥体内远处的距离,比如,客户端可以对能看到的远处场景的天空盒等远景物体进行编辑,从而生成场景编辑信息。需要说明的是,该实施例的远景编辑模式并不是只能对远处的对象进行编辑,而是指除了近景的对象可以编辑之外,远景的对象也可以编辑。

[0104] 当客户端处于全局参数编辑模式时,第一客户端可以针对第一游戏场景的全局参数进行编辑,第一客户端可以修改游戏场景的全局参数的变量,比如,修改游戏场景的灯光方向、灯光的强度、全局的烘焙参数等。

[0105] 当第一客户端处于浏览模式时,第一客户端不能对第一游戏场景中的对象进行编辑,但第一客户端可以获取由服务器同步过来的其它客户端对应的虚拟摄像机的位置信息,可以显示其它客户端在对应的视角下的第一游戏场景,从而可以看到其它客户端的编辑操作是如何进行的,这样可以方便某个工作人员想要参考和学习其他人通过客户端编辑的游戏场景,或者是编辑游戏场景所使用的手法和技巧,为后续对游戏场景进行编辑而做铺垫,从而提高了对游戏场景进行编辑的效率。

[0106] 需要说明的是,该实施例采用上述多种模式的原因是,每个客户端对游戏场景进行编辑的内容和关注的点,都是不一样,每个客户端在编辑的时候使用不同的模式,可以在不影响开发效率的前提下,避免冲突操作。

[0107] 可选地,基于目标编辑模式,对第一游戏场景进行编辑时,该编辑方法还包括:确定第一客户端当前编辑的第一游戏场景的目标对象,将目标对象设置为锁定状态,其中,目标对象在锁定状态下不能被除第一客户端以外的客户端编辑。

[0108] 在该实施例中,在第一客户端处于目标编辑模式下,比如,在处于近景编辑模式、远景编辑模式、全局参数编辑模式的情况下,可以将第一游戏场景中的目标对象设置为锁定状态,比如,对目标对象进行加锁操作,以将目标对象设置为锁定状态。在对目标对象设置为锁定状态之后,其它任何的客户端都不能对游戏场景中的该目标对象进行编辑操作,从而通过将目标对象设置为锁定状态可以防止其它客户端对其进行修改的目的。

[0109] 可选地,该编辑方法还包括:响应于编辑模式的切换操作或者响应于当前游戏场景的更新,解除目标对象的锁定状态。

[0110] 该实施例的第一客户端除了可以对第一游戏场景中的目标对象进行加锁操作之外,还可以对已进行处于锁定状态的目标对象进行解锁操作。可选地,只有在满足一定的解锁条件的情况下,才可以解除目标对象的锁定状态。可选地,第一客户端可以响应于编辑模

式的切换操作或者响应于当前游戏场景的更新,解除目标对象的锁定状态,比如,当改变了第一客户端的目标编辑模式,和/或,第一客户端对应的虚拟摄像机的视锥体不再呈现出当前的游戏场景画面时,就会解除目标对象的锁定状态。

[0111] 作为一种可选的实施方式,在第一客户端上呈现与目标编辑模式对应的标识信息。

[0112] 在该实施例中,当第一客户端处于不同的编辑模式时,可以呈现不同的标识信息,比如,该标识信息可以为颜色信息,不同的编辑模式对应不同的颜色信息,这样在客户端处于不同的编辑模式下,就可以显示出不同的颜色。

[0113] 可选地,在该实施例中,当第一客户端处于不同的编辑模式时,第一客户端对应的虚拟摄像机的颜色,以及对应的视锥体的半透显示(视锥体的显示可以配置是否开启)的颜色是不同的。

[0114] 该实施例的用于对游戏场景进行编辑的客户端都可以显示由服务器同步过来的其它客户端在编辑模式下的第一游戏场景,通过标识信息确定各个客户端是处在哪个编辑模式下,进而确定客户端具体在在编辑游戏场景的哪部分功能,比如,有些客户端在编辑当前游戏场景中的远景部分,有些客户端在编辑当前游戏场景中的近景部分,有些客户端在编辑当前游戏场景的全局参数,还有的客户端只是单纯浏览当前游戏场景的效果或者游戏场景开发项目的制作进度,从而使得对游戏场景的编辑透明化,提高了对游戏场景进行编辑的效率。

[0115] 在该实施例的游戏场景编辑中,可以利用帧同步技术,将游戏场景中的每个对象的位移旋转缩放信息等状态信息同步给用于对游戏场景进行编辑的每个客户端,然后将当前对第一游戏场景进行编辑的客户端的场景编辑信息同步到服务器上,来对当前的游戏场景的场景数据进行修改,然后服务器将修改后的场景数据再分发给每个客户端,从而可以提高开发同一个游戏场景的效率,使游戏场景的编辑不再是线性的过程,避免了需要通过增加人力来减短游戏场景开发项目的开发周期,从而解决了对游戏场景进行编辑的效率低的技术问题,达到了提高对游戏场景进行编辑的效率的技术效果。

[0116] 下面结合优选的实施方式对本发明实施例的游戏场景的编辑方法进行举例介绍。

[0117] 在项目开发的过程中,为了提高项目开发的效率,会经常涉及多个美术人员和策划人员同时编辑一个场景。而在现有的游戏引擎中,如果多人同时对游戏场景进行编辑,就会出现资源冲突的问题。

[0118] 因此,通常对于一个游戏场景的开发,同一时间仅允许一个人进行编辑。在项目进度非常紧急的时候,这种开发方式是非常低效的。因此,需要一种支持多人同时编辑同一个游戏场景的合作方案。

[0119] 在相关技术中,多个客户端串行地编辑同一个游戏场景,这种方案在有多个游戏场景的时候,能够流水线式地开发,可以避免其中一个人等候另一个人所浪费的时间。但是在只有一个游戏场景的时候,这种等候的问题就变得比较明显和严重了。针对只有一个游戏场景的游戏,上述做法就必然会出现效率的问题。

[0120] 在相关技术中,将一个游戏场景划分成多个子区域,每个客户端编辑其中一个子区域,最后再将编辑好的所有子区域整合起来,得到最终的游戏场景。该编辑方法看似能够让多个客户端同时在一个游戏场景上进行工作,但是实际上是一个游戏场景划分开来,其

中一个客户端并不能显示另一客户端上的对游戏场景进行编辑的进度,从而使得一个工作人员并不能实时看到另一个人的工作进度,以根据其工作内容来调整自己的工作内容。也即,上述方法是将一个大游戏场景变成了几个小游戏场景,最后还需要将几个小游戏场景合并起来,而在合并的过程中可能就会出现问题,并且合并这个操作在不同的游戏引擎上的实现方式也不一样,也会影响效率。进一步地,在反复修改当前场景时也会遇到很多问题。

[0121] 而在本发明实施例中,可以实现多人同时编辑同一个场景的同一块区域,避免串行开发带来的效率的降低,也避免分多块子区域进行编辑,而在合并时容易出错和不能及时沟通的问题。

[0122] 在该实施例中,可以通过网络同步的功能,将游戏场景中的每个对象的位移旋转缩放信息同步给每个客户端,将每个客户端对应的虚拟摄像机的位置信息同步到其它客户端中,同时将每个客户端的场景编辑信息同步到服务器,服务器基于场景编辑信息修改当前的游戏场景的场景数据,然后服务器将场景数据再分发给其它编辑该游戏场景的客户端。该实施例的多个客户端可以同时操作同一份游戏场景的数据,这样就能保证每个客户端显示的都是目前最新的游戏场景。同时服务器会定时备份一下当前的游戏场景的场景数据,这样也能避免因为某个客户端的不慎操作,将该客户端的工作内容全部删掉,或者,将其它客户端的场景数据不小心覆盖的问题出现。

[0123] 该实施例的客户端一共有四种编辑模式可以设置,包括:浏览模式、近景编辑模式、远景编辑模式、全局参数编辑模式。该实施例的客户端可以随意并且自由地在上述几种不同地编辑模式下进行切换。

[0124] 当客户端处于浏览模式下时,客户端不能编辑游戏场景中的任何对象,只能看到其它客户端的编辑操作是怎样的,这样可以方便某个客户端想要参考和学习其它客户端编辑的游戏场景,或者是编辑游戏场景所使用的手法和技巧。

[0125] 当客户端处于近景编辑模式下时,客户端可以对当前场景的视锥体内近处一定距离内的对象进行编辑。

[0126] 当客户端处于远景编辑模式下时,客户端可以对当前场景的远处的天空盒等远景对象进行编辑。

[0127] 当客户端在全局参数编辑模式下时,客户端可以修改游戏场景的全局参数的变量,比如,修改游戏场景的灯光方向、灯光的强度、全局的烘焙参数等。

[0128] 当该实施例在后面三种编辑模式下,客户端可以将能够编辑的对象设置为锁定状态,防止其它客户端对编辑的对象进行修改。当客户端改变了编辑模式,或摄像机的视锥体不再看当前的游戏场景,则处于锁定状态的对象会解锁,其中,当编辑模式改变的情况下,通常表示工作人员对对象的修改已经结束,所以需要已加锁的对象进行解锁,允许其它客户端进行编辑。

[0129] 可选地,在该实施例中,还可以引入多人语音通话功能,通过实时交流来进一步提高游戏场景的开发效率以及降低冲突的风险。

[0130] 图4是根据本发明实施例的一种每个客户端对应的虚拟摄像机存在的方式和呈现的内容的示意图。如图4所示,客户端1在对应的视角下显示的游戏场景为X,其中包括了客户端1需要编辑的内容,客户端1对应的虚拟摄像机为S1;客户端2在对应的视角下显示的游

戏场景为Y,其中包括了客户端2需要编辑的内容,客户端2对应的虚拟摄像机为S2;客户端3在对应的视角下显示的游戏场景为Z,其中包括了客户端3需要编辑的内容,客户端3对应的虚拟摄像机为S3。

[0131] 在该实施例中,在客户端处于不同的模式下,客户端的虚拟摄像机的颜色,以及视锥体的半透显示(视锥体的显示可以配置是否开启)的颜色都可以不同,可选地,图4所示的虚拟摄像机的颜色,以及视锥体的半透显示的颜色可以为蓝色。这样客户别的场编就知道哪个人在编辑哪部分的功能,有些人编辑远景,有些人编辑近景,有人的人编辑全局的参数,有的人编辑烘焙参数,还有的人只是单纯浏览当前的场景效果或者场景制作进度。

[0132] 该实施例的客户端采用多种模式的编辑的原因是,每个客户端编辑的内容和关注的部分都是不一样的,每个客户端在对游戏场景进行编辑的时候使用不同的模式,可以在不影响开发效率的前提下,避免冲突操作。

[0133] 需要说明的是,该实施例的场景编辑使用帧同步而不使用状态同步的原因是,在对游戏场景进行编辑的过程中,会有很多与物理碰撞相关的计算,这些计算的计算量是比较大的,如果将每个客户端的这些计算量直接放到服务器,则服务器的计算压力非常大,而且会有较大的延迟,所以将这些与物理碰撞相关的计算计算放到客户端来计算会更好,从而减轻了服务器的压力。

[0134] 该实施例将游戏场景中的每个对象的位移旋转缩放信息同步给其它每个客户端,然后将每个客户端的场景编辑信息同步到服务器上,来对象当前的游戏场景的场景数据进行修改,然后服务器将修改后的场景数据再分发给其它对游戏场景进行编辑的客户端,可以提高开发同一个游戏场景的效率,使游戏场景的编辑不再是线性的过程,可以提升监管美术场编的开发进度和开发方式,也可以用于教学,比如,分享美术场景编辑的开发技巧和开发思路,避免了需要通过增加人力来减短游戏场景开发项目的开发周期,从而解决了对游戏场景进行编辑的效率低的技术问题,达到了提高对游戏场景进行编辑的效率的技术效果。

[0135] 本发明实施例还提供了一种游戏场景的编辑装置。需要说明的是,该实施例的游戏场景的编辑装置可以用于执行本发明实施例图2所示的游戏场景的编辑方法。

[0136] 图5是根据本发明实施例的一种游戏场景的编辑装置的示意图。如图5所示,该游戏场景的编辑装置50包括:获取单元51、编辑单元52和第一发送单元53。

[0137] 获取单元51,用于获取第一客户端发送的场景编辑信息。

[0138] 编辑单元52,用于基于场景编辑信息对第一游戏场景的第一场景数据进行修改,得到第二场景数据,其中,第二场景数据用于将第一游戏场景更新为第二游戏场景。

[0139] 第一发送单元53,用于将第二场景数据发送至第一客户端和第二客户端,以使第一客户端和第二客户端显示第二游戏场景。

[0140] 本发明实施例还提供了另一种游戏场景的编辑装置。需要说明的是,该实施例的游戏场景的编辑装置可以用于执行本发明实施例图3所示的游戏场景的编辑方法。

[0141] 图6是根据本发明实施例的另一种游戏场景的编辑装置的示意图。如图6所示,该游戏场景的编辑装置60可以包括:第二发送单元61和接收单元62。

[0142] 第二发送单元61,用于使第一客户端针对第一游戏场景进行编辑,并将场景编辑信息发送至服务器,其中,场景编辑信息用于使服务器对第一游戏场景的第一场景数据进

行修改得到第二场景数据,并将第二场景数据发送至客户端,客户端包括第一客户端和第二客户端,第二场景数据用于将第一游戏场景更新为第二游戏场景。

[0143] 接收单元62,用于使第一客户端接收服务器发送的第二场景数据,并基于第二场景数据将显示的第一游戏场景,更新为第二游戏场景。

[0144] 在该实施例中,服务器获取客户端同步过来的场景编辑信息,修改游戏场景的场景数据,然后将修改后的场景数据分发给编辑该游戏场景的多个客户端,保证每个客户端呈现的都是目前最新的游戏场景,实现了多人同时编辑同一个场景的目的,避免了串行对游戏场景进行编辑所浪费的等候时间,也避免了将游戏场景划分多个子区域而容易导致合并出错和不能及时沟通的问题,从而解决了对游戏场景进行编辑的效率低的技术问题,达到了提高对游戏场景进行编辑的效率的技术效果。

[0145] 本发明的实施例还提供了一种存储介质,该存储介质中存储有计算机程序,其中,在计算机程序被处理器运行时控制存储介质所在设备执行本发明上述任一项方法实施例中的步骤。

[0146] 可选地,在本实施例中,上述存储介质可以包括但不限于:U盘、只读存储器(Read-Only Memory,简称为ROM)、随机存取存储器(Random Access Memory,简称为RAM)、移动硬盘、磁碟或者光盘等各种可以存储计算机程序的介质。

[0147] 本发明的实施例还提供了一种电子装置,包括存储器和处理器,该存储器中存储有计算机程序,该处理器被设置为运行计算机程序以执行上述任一项方法实施例中的步骤。

[0148] 可选地,上述电子装置还可以包括传输设备以及输入输出设备,其中,该传输设备和上述处理器连接,该输入输出设备和上述处理器连接。

[0149] 显然,本领域的技术人员应该明白,上述的本发明的各模块或各步骤可以用通用的计算装置来实现,它们可以集中在单个的计算装置上,或者分布在多个计算装置所组成的网络上,可选地,它们可以用计算装置可执行的程序代码来实现,从而,可以将它们存储在存储装置中由计算装置来执行,并且在某些情况下,可以以不同于此处的顺序执行所示出或描述的步骤,或者将它们分别制作成各个集成电路模块,或者将它们中的多个模块或步骤制作成单个集成电路模块来实现。这样,本发明不限制于任何特定的硬件和软件结合。

[0150] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

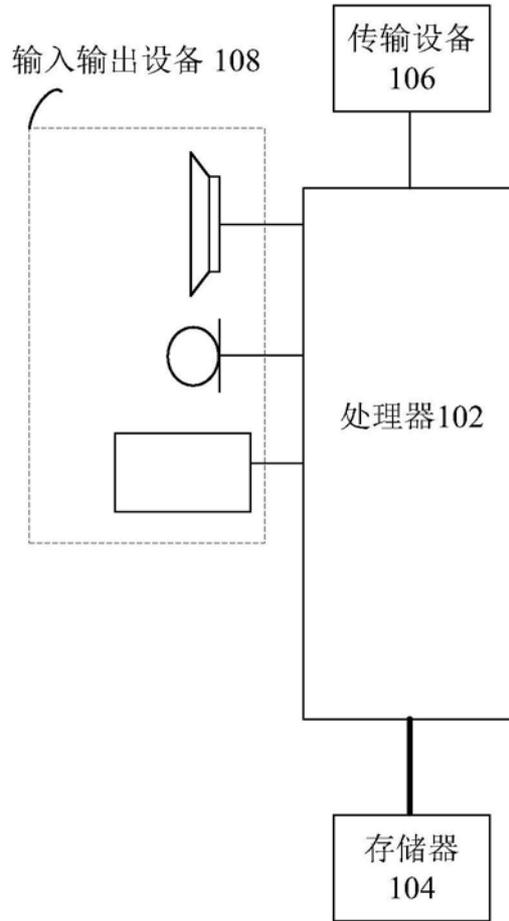


图1

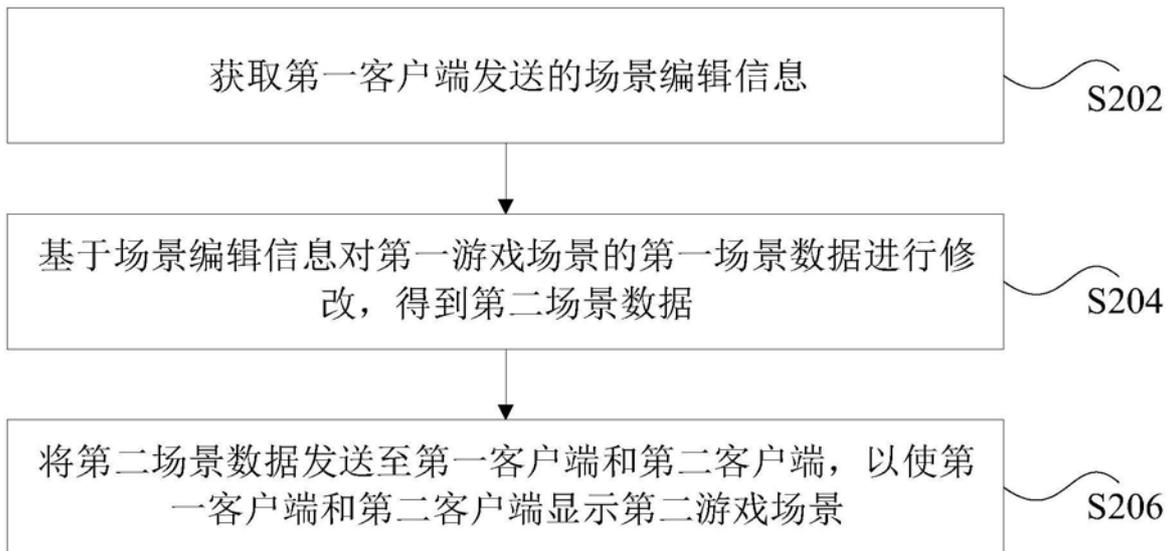


图2

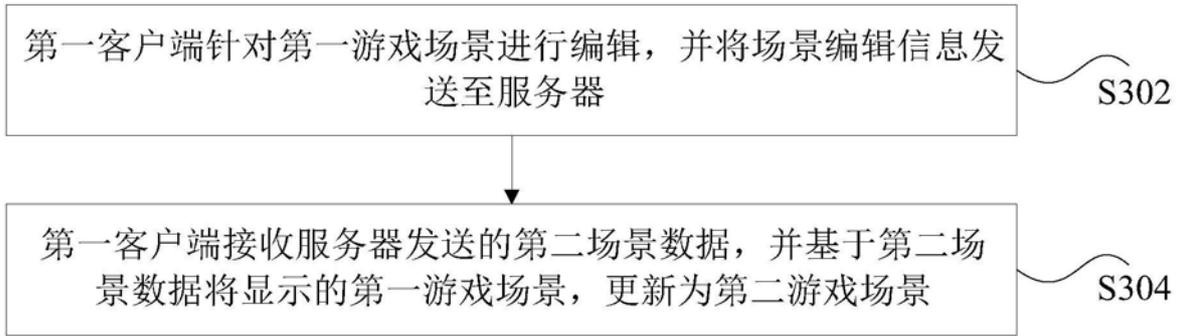


图3

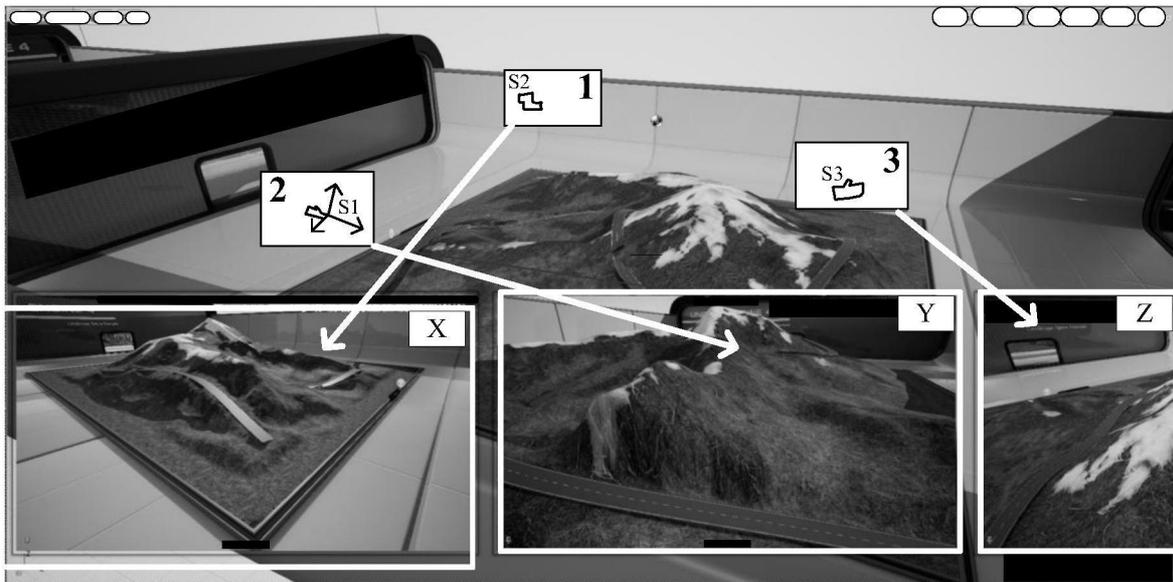


图4

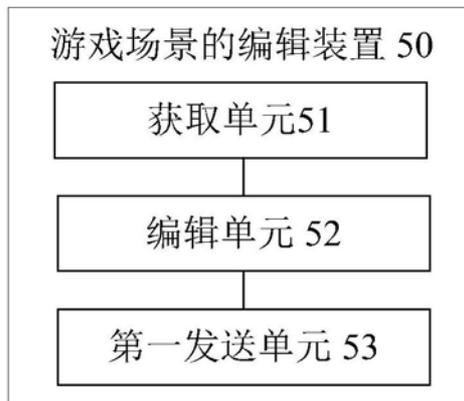


图5

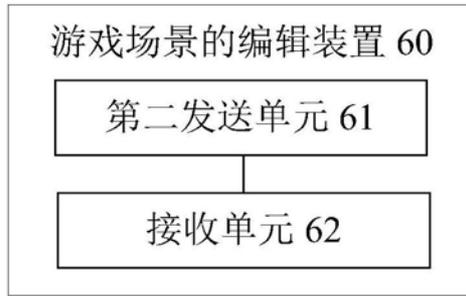


图6