



(12) 发明专利

(10) 授权公告号 CN 113908554 B

(45) 授权公告日 2024.09.20

(21) 申请号 202111225554.8

A63F 13/45 (2014.01)

(22) 申请日 2021.10.21

(56) 对比文件

(65) 同一申请的已公布的文献号

CN 108093315 A, 2018.05.29

申请公布号 CN 113908554 A

CN 110858408 A, 2020.03.03

(43) 申请公布日 2022.01.11

审查员 黄秀萍

(73) 专利权人 福建天晴数码有限公司

地址 350000 福建省福州市君竹路83号科技发展中心大楼第四层Q476室(自贸试验区内)

(72) 发明人 刘德建 温荣泉 陈宏展

(74) 专利代理机构 福州旭辰知识产权代理事务所(普通合伙) 35233

专利代理师 程勇

(51) Int. Cl.

A63F 13/77 (2014.01)

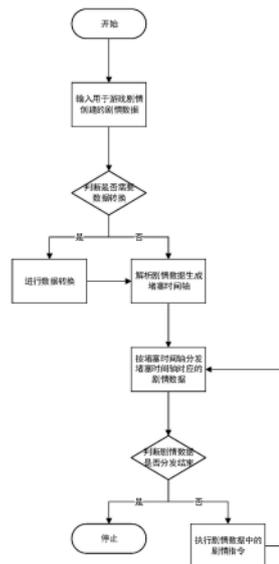
权利要求书4页 说明书9页 附图7页

(54) 发明名称

一种自动生成游戏剧情的实现方法及其系统

(57) 摘要

本发明提供了一种自动生成游戏剧情的实现方法,该方法为1、输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;进入步骤2;是,直接进入步骤2;2、解析转换后的数据生成堵塞时间轴;3、按堵塞时间轴分发堵塞时间轴对应的剧情数据;判断堵塞时间轴对应的剧情数据是否分发结束;是,停止结束流程;否,进行步骤4;4、执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据。本发明中的游戏提供了堵塞时间轴的交互方法,支持按时间线播放剧情、设置剧情至任一时刻进行播放,实现完备的游戏剧情播放与交互控制。



CN 113908554 B

1. 一种自动生成游戏剧情的实现方法,其特征在于:所述方法包括如下步骤:

步骤S1、输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;进入步骤S2;是,直接进入步骤S2;

步骤S2、解析转换后的数据生成堵塞时间轴;该堵塞时间轴为支持堵塞条件的时间轴,其中,堵塞时间轴剧情节点数据未设置堵塞条件节点时,时间轴将按照时间顺序正常运行;当设置了堵塞条件时,播放时间更新到堵塞条件节点时播放时间停止更新,等待堵塞条件满足后播放时间移动至下一个节点的开始位置继续播放;

步骤S3、按堵塞时间轴分发堵塞时间轴对应的剧情数据;判断堵塞时间轴对应的剧情数据是否分发结束;是,停止结束流程;否,进行步骤S4;

步骤S4、执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据;

所述解析转换后的数据生成堵塞时间轴进一步具体为:

步骤S21、输入转换后的用于游戏剧情创建的剧情数据;该剧情数据中包含多个子剧情数据,剧情数据中记录的数据包括剧情指令及其执行方式、剧情节点时间长度、是否堵塞节点、堵塞条件、是否支持跳过;

步骤S22、创建一个堵塞时间轴;

步骤S23、读取用于游戏剧情创建的剧情数据中的所有剧情节点数据;将所有剧情节点数据加入一个剧情节点数据队列中;

步骤S24、判断剧情节点数据队列是否为空,是,返回堵塞时间轴,并结束流程;否,进入步骤S25;

步骤S25、从剧情节点数据队列中读取一个剧情节点数据,将剧情节点数据加入堵塞时间轴的当前节点位置;堵塞时间轴的当前时间值向后移动到剧情节点数据中接入的时间值;

步骤S26、判断剧情节点数据中是否记录了堵塞值,该堵塞值为堵塞条件对应的数据值;否,进入步骤S27;是,堵塞时间轴中添加堵塞条件节点,设置堵塞值调整为剧情节点数据中记录的堵塞条件;堵塞时间轴当前时间轴向后移动一个堵塞节点单位时间值,进入步骤S27;

步骤S27、判断剧情数据节点队列中是否还有其它剧情数据节点,是,执行步骤S25;否,返回堵塞时间轴,并结束流程。

2. 根据权利要求1所述的一种自动生成游戏剧情的实现方法,其特征在于:所述步骤S1进一步具体包括:

步骤S11、输入用于游戏剧情的创建的剧情数据,判断输入剧情数据类型是否是Json数据,是,进入步骤S12,否,判断是否是Json文件路径,是,读取文件中的Json数据,并进入步骤S12,否,停止流程;

步骤S12、将Json数据识别为Json实例类型;

步骤S13、创建一个剧情数据结构;

步骤S14、将Json数据中的基础数据填充到剧情数据结构的对应字段值中;

步骤S15、无匹配字段的数据填充到剧情数据结构的拓展字段队列中;

步骤S16、判断剧情数据结构中的剧情数据是否是根节点数据,否,添加到父节点剧情

数据的子数据中;是,记录为根节点剧情数据;

步骤S17、判断Json数据是否包含子数据,是,进入步骤S13;否,返回创建的根节点剧情数据,并结束流程。

3. 根据权利要求1所述的一种自动生成游戏剧情的实现方法,其特征在于:所述堵塞时间轴包括:堵塞时间轴的进度值和时间节点集合,所述时间节点集合由多个时间节点组成,每个时间节点包括时间长度、进度值、以及堵塞条件。

4. 根据权利要求3所述的一种自动生成游戏剧情的实现方法,其特征在于:所述步骤S3进一步具体包括:

步骤S31、输入堵塞时间轴对应的剧情数据;

步骤S32、读取堵塞时间轴的进度值;

步骤S33、判断堵塞时间轴是否执行完成,是,堵塞时间轴对应的剧情数据分发结束并结束流程,否,读取当前进度值的剧情数据;

步骤S34、读取剧情数据中的剧情指令数据集合,将剧情指令数据集合中的数据加入一个剧情指令数据队列;

步骤S35、判断剧情指令数据队列是否为空,是,更新堵塞时间轴进度值到下一节点,并执行步骤S33,否,执行步骤S36;

步骤S36、从剧情指令数据队列中读取一个剧情指令数据,执行剧情指令数据;

步骤S37、判断剧情指令数据队列中是否还有其它剧情指令数据,是,执行步骤S36;否,更新堵塞时间轴进度值到下一节点,并执行步骤S33。

5. 根据权利要求3所述的一种自动生成游戏剧情的实现方法,其特征在于:步骤S4进一步具体包括:步骤S41、输入剧情指令数据,读取剧情指令数据中的剧情指令;

步骤S42、判断剧情指令是否有注册了对应的执行方式,否,剧情指令执行完成并结束流程,是,进入步骤S43,

步骤S43、执行对应的执行方式传入剧情指令数据;

步骤S44、判断是否同步执行对应的执行方式;否,等待异步执行方式执行完成,返回执行成功事件;剧情指令执行完成;是,返回执行成功事件,剧情指令执行完成。

6. 一种自动生成游戏剧情的实现系统,其特征在于:所述实现系统包括数据转换模块、数据解析模块、数据分发模块、以及数据执行模块;

所述数据转换模块,用于输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;执行数据解析模块;是,直接执行数据解析模块;

所述数据解析模块,用于解析转换后的数据生成堵塞时间轴;该堵塞时间轴为支持堵塞条件的时间轴,其中,堵塞时间轴剧情节点数据未设置堵塞条件节点时,时间轴将按照时间顺序正常运行;当设置了堵塞条件时,播放时间更新到堵塞条件节点时播放时间停止更新,等待堵塞条件满足后播放时间移动至下一个节点的开始位置继续播放;

所述数据分发模块,用于按堵塞时间轴分发堵塞时间轴对应的剧情数据;判断堵塞时间轴对应的剧情数据是否分发结束;是,停止结束流程;否,执行数据执行模块;

所述数据执行模块,用于执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据;

所述解析转换后的数据生成堵塞时间轴进一步具体为：

步骤S61、输入转换后的用于游戏剧情创建的剧情数据；该剧情数据中包含多个子剧情数据，剧情数据中记录的数据包括剧情指令及其执行方式、剧情节点时间长度、是否堵塞节点、堵塞条件、是否支持跳过；

步骤S62、创建一个堵塞时间轴；

步骤S63、读取用于游戏剧情创建的剧情数据中的所有剧情节点数据；将所有剧情节点数据加入一个剧情节点数据队列中；

步骤S64、判断剧情节点数据队列是否为空，是，返回堵塞时间轴，并结束流程；否，进入步骤S65；

步骤S65、从剧情节点数据队列中读取一个剧情节点数据，将剧情节点数据加入堵塞时间轴的当前节点位置；堵塞时间轴的当前时间值向后移动到剧情节点数据中接入的时间值；

步骤S66、判断剧情节点数据中是否记录了堵塞值，该堵塞值为堵塞条件对应的数据值；否，进入步骤S67；是，堵塞时间轴中添加堵塞条件节点，设置堵塞值调整为剧情节点数据中记录的堵塞条件；堵塞时间轴当前时间轴向后移动一个堵塞节点单位时间值，进入步骤S67；

步骤S67、判断剧情数据节点队列中是否还有其它剧情数据节点，是，执行步骤S65；否，返回堵塞时间轴，并结束流程。

7. 根据权利要求6所述的一种自动生成游戏剧情的实现系统，其特征在于：所述数据转换模块的实现方式进一步具体包括：

步骤S51、输入用于游戏剧情的创建的剧情数据，判断输入剧情数据类型是否是Json数据，是，进入步骤S52，否，判断是否是Json文件路径，是，读取文件中的Json数据，并进入步骤S52，否，停止流程；

步骤S52、将Json数据识别为Json实例类型；

步骤S53、创建一个剧情数据结构；

步骤S54、将Json数据中的基础数据填充到剧情数据结构的对应字段值中；

步骤S55、无匹配字段的数据填充到剧情数据结构的拓展字段队列中；

步骤S56、判断剧情数据结构中的剧情数据是否是根节点数据，否，添加到父节点剧情数据的子数据中；是，记录为根节点剧情数据；

步骤S57、判断Json数据是否包含子数据，是，进入步骤S53；否，返回创建的根节点剧情数据，并结束流程。

8. 根据权利要求6所述的一种自动生成游戏剧情的实现系统，其特征在于：所述堵塞时间轴包括：堵塞时间轴的进度值和时间节点集合，所述时间节点集合由多个时间节点组成，每个时间节点包括时间长度、进度值、以及堵塞条件。

9. 根据权利要求8所述的一种自动生成游戏剧情的实现系统，其特征在于：所述数据分发模块的实现方式进一步具体包括：

步骤S71、输入堵塞时间轴对应的剧情数据；

步骤S72、读取堵塞时间轴的进度值；

步骤S73、判断堵塞时间轴是否执行完成，是，堵塞时间轴对应的剧情数据分发结束并

结束流程,否,读取当前进度值的剧情数据;

步骤S74、读取剧情数据中的剧情指令数据集合,将剧情指令数据集合中的数据加入一个剧情指令数据队列;

步骤S75、判断剧情指令数据队列是否为空,是,更新堵塞时间轴进度值到下一节点,并执行步骤S73,否,执行步骤S76;

步骤S76、从剧情指令数据队列中读取一个剧情指令数据,执行剧情指令数据;

步骤S77、判断剧情指令数据队列中是否还有其它剧情指令数据,是,执行步骤S76;否,更新堵塞时间轴进度值到下一节点,并执行步骤S73。

10. 根据权利要求8所述的一种自动生成游戏剧情的实现系统,其特征在于:所述数据执行模块的实现方式进一步具体包括:步骤S81、输入剧情指令数据,读取剧情指令数据中的剧情指令;

步骤S82、判断剧情指令是否有注册了对应的执行方式,否,剧情指令执行完成并结束流程,是,进入步骤S83,

步骤S83、执行对应的执行方式传入剧情指令数据;

步骤S84、判断是否同步执行对应的执行方式;否,等待异步执行方式执行完成,返回执行成功事件;剧情指令执行完成;是,返回执行成功事件,剧情指令执行完成。

## 一种自动生成游戏剧情的实现方法及其系统

### 技术领域

[0001] 本发明涉及游戏制作技术领域,特别是一种自动生成游戏剧情的实现方法及其系统。

### 背景技术

[0002] 剧情被众多有信仰的玩家和开发者认为是游戏的灵魂。对于一个玩家,这没有任何不妥;而作为一名开发者,在进行剧情设计时,有一个无法忽视的问题——剧情的可交互性很低。游戏剧情可以用文字、图片、动画完美地表达,却难以用交互的方式表达得十分准确。现有生成游戏剧情的技术包括基于配置动态生成游戏剧情、基于Json生成游戏剧情等;现有技术对生成的局限性高,通常只针对特点的游戏剧情进行生成,且不支持以堵塞时间轴的方式进行交互控制,拓展困难,无法广泛使用。

### 发明内容

[0003] 为克服上述问题,本发明的目的是提供一种自动生成游戏剧情的实现方法,解决了游戏剧情单一方向执行、不支持以时间轴的方式控制、使用复杂不易维护、无法动态构建等问题。

[0004] 本发明采用以下方案实现:一种自动生成游戏剧情的实现方法,所述方法包括如下步骤:

[0005] 步骤S1、输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;进入步骤S2;是,直接进入步骤S2;

[0006] 步骤S2、解析转换后的数据生成堵塞时间轴;该堵塞时间轴为支持堵塞条件的时间轴,其中,堵塞时间轴剧情节点数据未设置堵塞条件节点时,时间轴将按照时间顺序正常运行;当设置了堵塞条件时,播放时间更新到堵塞条件节点时播放时间停止更新,等待堵塞条件满足后播放时间移动至下一个节点的开始位置继续播放;

[0007] 步骤S3、按堵塞时间轴分发堵塞时间轴对应的剧情数据;判断堵塞时间轴对应的剧情数据是否分发结束;是,停止结束流程;否,进行步骤S4;

[0008] 步骤S4、执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据。

[0009] 进一步的,所述步骤S1进一步具体包括:

[0010] 步骤S11、输入用于游戏剧情的创建的剧情数据,判断输入剧情数据类型是否是Json数据,是,进入步骤S12,否,判断是否是Json文件路径,是,读取文件中的Json数据,并进入步骤S12,否,停止流程;

[0011] 步骤S12、将Json数据识别为Json实例类型;

[0012] 步骤S13、创建一个剧情数据结构;

[0013] 步骤S14、将Json数据中的基础数据填充到剧情数据结构的对应字段值中;

[0014] 步骤S15、无匹配字段的数据填充到剧情数据结构的拓展字段队列中;

[0015] 步骤S16、判断剧情数据结构中的剧情数据是否是根节点数据,否,添加到父节点剧情数据的子数据中;是,记录为根节点剧情数据;

[0016] 步骤S17、判断Json数据是否包含子数据,是,进入步骤S13;否,返回创建的根节点剧情数据,并结束流程。

[0017] 进一步的,所述解析转换后的数据生成堵塞时间轴进一步具体为:

[0018] 步骤S21、输入转换后的用于游戏剧情创建的剧情数据;该剧情数据中包含多个子剧情数据,剧情数据中记录的数据包括剧情指令及其执行方式、剧情节点时间长度、是否堵塞节点、堵塞条件、是否支持跳过;

[0019] 步骤S22、创建一个堵塞时间轴;

[0020] 步骤S23、读取用于游戏剧情创建的剧情数据中的所有剧情节点数据;将所有剧情节点数据加入一个剧情节点数据队列中;

[0021] 步骤S24、判断剧情节点数据队列是否为空,是,返回堵塞时间轴,并结束流程;否,进入步骤S25;

[0022] 步骤S25、从剧情节点数据队列中读取一个剧情节点数据,将剧情节点数据加入堵塞时间轴的当前节点位置;堵塞时间轴的当前时间值向后移动到剧情节点数据中接入的时间值;

[0023] 步骤S26、判断剧情节点数据中是否记录了堵塞值,该堵塞值为堵塞条件对应的数据值;否,进入步骤S27;是,堵塞时间轴中添加堵塞条件节点,设置堵塞值调整为剧情节点数据中记录的堵塞条件;堵塞时间轴当前时间轴向后移动一个堵塞节点单位时间值,进入步骤S27;

[0024] 步骤S27、判断剧情数据节点队列中是否还有其它剧情数据节点,是,执行步骤S25;否,返回堵塞时间轴,并结束流程。

[0025] 进一步的,所述堵塞时间轴包括:堵塞时间轴的进度值和时间节点集合,所述时间节点集合由多个时间节点组成,每个时间节点包括时间长度、进度值、以及堵塞条件。

[0026] 进一步的,所述步骤S3进一步具体包括:

[0027] 步骤S31、输入堵塞时间轴对应的剧情数据;

[0028] 步骤S32、读取堵塞时间轴的进度值;

[0029] 步骤S33、判断堵塞时间轴是否执行完成,是,堵塞时间轴对应的剧情数据分发结束并结束流程,否,读取当前进度值的剧情数据;

[0030] 步骤S34、读取剧情数据中的剧情指令数据集合,将剧情指令数据集合中的数据加入一个剧情指令数据队列;

[0031] 步骤S35、判断剧情指令数据队列是否为空,是,更新堵塞时间轴进度值到下一节点,并执行步骤S33,否,执行步骤S36;

[0032] 步骤S36、从剧情指令数据队列中读取一个剧情指令数据,执行剧情指令数据;

[0033] 步骤S37、判断剧情指令数据队列中是否还有其它剧情指令数据,是,执行步骤S36;否,更新堵塞时间轴进度值到下一节点,并执行步骤S33。

[0034] 进一步的,步骤S4进一步具体包括:步骤S41、输入剧情指令数据,读取剧情指令数据中的剧情指令;

[0035] 步骤S42、判断剧情指令是否有注册了对应的执行方式,否,剧情指令执行完成并

结束流程,是,进入步骤S43,

[0036] 步骤S43、执行对应的执行方式传入剧情指令数据;

[0037] 步骤S44、判断是否同步执行执行方式;否,等待异步执行方式执行完成,返回执行成功事件;剧情指令执行完成;是,返回执行成功事件,剧情指令执行完成。

[0038] 本发明还提供了一种自动生成游戏剧情的实现系统,所述实现系统包括数据转换模块、数据解析模块、数据分发模块、以及数据执行模块;

[0039] 所述数据转换模块,用于输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;执行数据解析模块;是,直接执行数据解析模块;

[0040] 所述数据解析模块,用于解析转换后的数据生成堵塞时间轴;该堵塞时间轴为支持堵塞条件的时间轴,其中,堵塞时间轴剧情节点数据未设置堵塞条件节点时,时间轴将按照时间顺序正常运行;当设置了堵塞条件时,播放时间更新到堵塞条件节点时播放时间停止更新,等待堵塞条件满足后播放时间移动至下一个节点的开始位置继续播放;

[0041] 所述数据分发模块,用于按堵塞时间轴分发堵塞时间轴对应的剧情数据;判断堵塞时间轴对应的剧情数据是否分发结束;是,停止结束流程;否,执行数据执行模块;

[0042] 所述数据执行模块,用于执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据。

[0043] 进一步的,所述数据转换模块的实现方式进一步具体包括:

[0044] 步骤S51、输入用于游戏剧情的创建的剧情数据,判断输入剧情数据类型是否是Json数据,是,进入步骤S52,否,判断是否是Json文件路径,是,读取文件中的Json数据,并进入步骤S52,否,停止流程;

[0045] 步骤S52、将Json数据识别为Json实例类型;

[0046] 步骤S53、创建一个剧情数据结构;

[0047] 步骤S54、将Json数据中的基础数据填充到剧情数据结构的对应字段值中;

[0048] 步骤S55、无匹配字段的数据填充到剧情数据结构的拓展字段队列中;

[0049] 步骤S56、判断剧情数据结构中的剧情数据是否是根节点数据,否,添加到父节点剧情数据的子数据中;是,记录为根节点剧情数据;

[0050] 步骤S57、判断Json数据是否包含子数据,是,进入步骤S53;否,返回创建的根节点剧情数据,并结束流程。

[0051] 进一步的,所述解析转换后的数据生成堵塞时间轴进一步具体为:

[0052] 步骤S61、输入转换后的用于游戏剧情创建的剧情数据;该剧情数据中包含多个子剧情数据,剧情数据中记录的数据包括剧情指令及其执行方式、剧情节点时间长度、是否堵塞节点、堵塞条件、是否支持跳过;

[0053] 步骤S62、创建一个堵塞时间轴;

[0054] 步骤S63、读取用于游戏剧情创建的剧情数据中的所有剧情节点数据;将所有剧情节点数据加入一个剧情节点数据队列中;

[0055] 步骤S64、判断剧情节点数据队列是否为空,是,返回堵塞时间轴,并结束流程;否,进入步骤S65;

[0056] 步骤S65、从剧情节点数据队列中读取一个剧情节点数据,将剧情节点数据加入堵

塞时间轴的当前节点位置;堵塞时间轴的当前时间值向后移动到剧情节点数据中接入的时间值;

[0057] 步骤S66、判断剧情节点数据中是否记录了堵塞值,该堵塞值为堵塞条件对应的数据值;否,进入步骤S67;是,堵塞时间轴中添加堵塞条件节点,设置堵塞值调整为剧情节点数据中记录的堵塞条件;堵塞时间轴当前时间轴向后移动一个堵塞节点单位时间值,进入步骤S67;

[0058] 步骤S67、判断剧情数据节点队列中是否还有其它剧情数据节点,是,执行步骤S65;否,返回堵塞时间轴,并结束流程。

[0059] 进一步的,所述堵塞时间轴包括:堵塞时间轴的进度值和时间节点集合,所述时间节点集合由多个时间节点组成,每个时间节点包括时间长度、进度值、以及堵塞条件。

[0060] 进一步的,所述数据分发模块的实现方式进一步具体包括:

[0061] 步骤S71、输入堵塞时间轴对应的剧情数据;

[0062] 步骤S72、读取堵塞时间轴的进度值;

[0063] 步骤S73、判断堵塞时间轴是否执行完成,是,堵塞时间轴对应的剧情数据分发结束并结束流程,否,读取当前进度值的剧情数据;

[0064] 步骤S74、读取剧情数据中的剧情指令数据集合,将剧情指令数据集合中的数据加入一个剧情指令数据队列;

[0065] 步骤S75、判断剧情指令数据队列是否为空,是,更新堵塞时间轴进度值到下一节点,并执行步骤S73,否,执行步骤S76;

[0066] 步骤S76、从剧情指令数据队列中读取一个剧情指令数据,执行剧情指令数据;

[0067] 步骤S77、判断剧情指令数据队列中是否还有其它剧情指令数据,是,执行步骤S76;否,更新堵塞时间轴进度值到下一节点,并执行步骤S73。

[0068] 进一步的,所述数据执行模块的实现方式进一步具体包括:步骤S81、输入剧情指令数据,读取剧情指令数据中的剧情指令;

[0069] 步骤S82、判断剧情指令是否有注册了对应的执行方式,否,剧情指令执行完成并结束流程,是,进入步骤S83,

[0070] 步骤S83、执行对应的执行方式传入剧情指令数据;

[0071] 步骤S84、判断是否同步执行执行方式;否,等待异步执行方式执行完成,返回执行成功事件;剧情指令执行完成;是,返回执行成功事件,剧情指令执行完成。

[0072] 本发明的有益效果在于:本发明通过将游戏剧情Json数据生成统一的通用剧情数据结构,数据解析模块生成堵塞时间轴,当收到用户开启剧情后,数据分发模块按堵塞时间轴分发剧情数据,数据执行模块执行剧情数据中的剧情指令,实现完备的游戏剧情播放与交互控制系统。

[0073] 本发明通过游戏剧情Json数据或动态构建的通用剧情数据生成堵塞时间轴类型的游戏剧情数据,减少了游戏剧情与交互逻辑开发消耗,缩短了游戏剧情开发周期,加快了开发进度。且使用本发明中游戏剧情具有统一的调用风格,更少的游戏剧情相关文件与代码,更易维护,后期维护成本更低。

[0074] 本发明中的游戏提供了堵塞时间轴的交互方法,支持按时间线播放剧情、设置剧情至任一时刻进行播放;任意类型的游戏项目均可通过本发明快速接入并使用游戏剧情,

开发维护简单高效,且支持自定义拓展剧情指令,适配游戏剧情的各种使用需求。

### 附图说明

[0075] 图1是本发明的方法流程示意图。

[0076] 图2是本发明的方法中进行数据转换操作的流程示意图。

[0077] 图3是本发明的方法中进行数据解析生成堵塞时间轴对应的剧情数据流程示意图。

[0078] 图4是本发明的堵塞时间轴的数据结构示意图。

[0079] 图5是本发明的方法中按堵塞时间轴分发剧情数据流程示意图。

[0080] 图6是本发明的方法中进行执行剧情数据中的剧情指令的流程示意图。

[0081] 图7是本发明的系统原理框图。

### 具体实施方式

[0082] 下面结合附图对本发明做进一步说明。

[0083] 本发明可应用于所有类型游戏剧情的生成与交互过程,和常规的游戏剧情的开发不同,本发明通过将游戏剧情的描述信息进行提取生成堵塞时间轴,支持基于堵塞时间轴的剧情交互逻辑。

[0084] 请参阅图1所示,本发明的一种自动生成游戏剧情的实现方法,所述方法包括如下步骤:

[0085] 步骤S1、输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;进入步骤S2;是,直接进入步骤S2;

[0086] 步骤S2、解析转换后的数据生成堵塞时间轴;该堵塞时间轴为支持堵塞条件的时间轴,其中,堵塞时间轴剧情节点数据未设置堵塞条件节点时,时间轴将按照时间顺序正常运行;当设置了堵塞条件时,播放时间更新到堵塞条件节点时播放时间停止更新,等待堵塞条件满足后播放时间移动至下一个节点的开始位置继续播放;

[0087] 步骤S3、按堵塞时间轴分发堵塞时间轴对应的剧情数据;判断堵塞时间轴对应的剧情数据是否分发结束;是,停止结束流程;否,进行步骤S4;

[0088] 步骤S4、执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据。

[0089] 下面结合一具体实施例对本发明作进一步说明:

[0090] 步骤S1、输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;进入步骤S2;是,直接进入步骤S2;剧情数据包含用于游戏剧情创建的剧情数据、堵塞时间轴对应的剧情数据、剧情指令数据等;

[0091] 请参阅图2所示,所述步骤S1进一步具体包括:

[0092] 步骤S11、输入用于游戏剧情的创建的剧情数据,判断输入剧情数据类型是否是Json数据,是,进入步骤S12,否,判断是否是Json文件路径,是,读取文件中的Json数据,并进入步骤S12,否,停止流程;

[0093] 步骤S12、将Json数据识别为Json实例类型;

[0094] 步骤S13、创建一个剧情数据结构;

- [0095] 步骤S14、将Json数据中的基础数据填充到剧情数据结构的对应字段值中；
- [0096] 步骤S15、无匹配字段的数据填充到剧情数据结构的拓展字段队列中；
- [0097] 步骤S16、判断剧情数据结构中的剧情数据是否是根节点数据，否，添加到父节点剧情数据的子数据中；是，记录为根节点剧情数据；
- [0098] 步骤S17、判断Json数据是否包含子数据，是，进入步骤S13；否，返回创建的根节点剧情数据，并结束流程。
- [0099] 步骤S2、解析转换后的数据生成堵塞时间轴；该堵塞时间轴为支持堵塞条件的时间轴，
- [0100] 请参阅图3所示，所述解析转换后的数据生成堵塞时间轴进一步具体为：
- [0101] 步骤S21、输入转换后的用于游戏剧情创建的剧情数据；该剧情数据中包含多个子剧情数据，剧情数据中记录的数据包括剧情指令及其执行方式、剧情节点时间长度、是否堵塞节点、堵塞条件、是否支持跳过；
- [0102] 步骤S22、创建一个堵塞时间轴；堵塞时间轴是时间轴，存在多个时间节点，堵塞条件节点是一种包含堵塞条件的时间节点，在堵塞条件未达成前时间暂停，达成后进入下一节点；
- [0103] 步骤S23、读取用于游戏剧情创建的剧情数据中的所有剧情节点数据；将所有剧情节点数据加入一个剧情节点数据队列中；
- [0104] 步骤S24、判断剧情节点数据队列是否为空，是，返回堵塞时间轴，并结束流程；否，进入步骤S25；
- [0105] 步骤S25、从剧情节点数据队列中读取一个剧情节点数据，将剧情节点数据加入堵塞时间轴的当前节点位置；堵塞时间轴的当前时间值向后移动到剧情节点数据中接入的时间值；
- [0106] 步骤S26、判断剧情节点数据中是否记录了堵塞值，该堵塞值为堵塞条件对应的数据值；否，进入步骤S27；是，堵塞时间轴中添加堵塞条件节点，设置堵塞值调整为剧情节点数据中记录的堵塞条件；堵塞时间轴当前时间轴向后移动一个堵塞节点单位时间值，进入步骤S27；堵塞值的定义是剧情数据中堵塞条件参数对应的数据值，在剧情数据中可以设置或不设置，设置时产生堵塞，不设置则按时间顺序执行；
- [0107] 步骤S27、判断剧情数据节点队列中是否还有其它剧情数据节点，是，执行步骤S25；否，返回堵塞时间轴，并结束流程。参见图4，其中，所述堵塞时间轴包括：堵塞时间轴的进度值和时间节点集合，所述时间节点集合由多个时间节点组成，每个时间节点包括时间长度、进度值、以及堵塞条件。本发明中堵塞条件的定义包括但不限于角色到达指定位置、用户点击界面、用户与某个角色交互等。
- [0108] 步骤S3、按堵塞时间轴分发堵塞时间轴对应的剧情数据；判断堵塞时间轴对应的剧情数据是否分发结束；是，停止结束流程；否，进行步骤S4；
- [0109] 解析转换后的数据生成堵塞时间轴的示例如下：
- [0110] 堵塞时间轴上，堵塞时间轴中添加堵塞条件节点，设置堵塞值调整为剧情节点数据中记录的堵塞条件：
- [0111] 时间0节点播放进场动画，时间长度为5；
- [0112] 时间5节点添加了堵塞条件等待用户点击，时间长度为1；

- [0113] 时间6节点添加了主角播放动作A动画,时间长度为2;
- [0114] 时间6节点添加了NPC1播放动作B动画,时间长度为2;
- [0115] 时间7节点添加了NPC2播放动作C动画,时间长度为2;
- [0116] 则播放堵塞时间轴时的效果:
- [0117] 0时刻开始播放进场动画,播放了5单位时间;
- [0118] 等待用户点击,播放时间值跳到6时刻;
- [0119] 主角播放动作A动画、NPC1播放动作B动画;
- [0120] 时间播放到7时刻时,NPC2播放动作C动画。
- [0121] 请参阅图5,所示,本发明的所述步骤S3进一步具体包括:
- [0122] 步骤S31、输入堵塞时间轴对应的剧情数据;
- [0123] 步骤S32、读取堵塞时间轴的进度值;
- [0124] 步骤S33、判断堵塞时间轴是否执行完成,是,堵塞时间轴对应的剧情数据分发结束并结束流程,否,读取当前进度值的剧情数据;
- [0125] 步骤S34、读取剧情数据中的剧情指令数据集合,将剧情指令数据集合中的数据加入一个剧情指令数据队列;本发明中剧情指令的定义包括但不限于显示角色、设置角色的位置旋转或大小、角色的动作播放、角色寻路、角色换装、角色特效播放、摄像机镜头的移动缩放或抖动、剧情UI显示等。
- [0126] 步骤S35、判断剧情指令数据队列是否为空,是,更新堵塞时间轴进度值到下一节点,并执行步骤S33,否,执行步骤S36;
- [0127] 步骤S36、从剧情指令数据队列中读取一个剧情指令数据,执行剧情指令数据;
- [0128] 步骤S37、判断剧情指令数据队列中是否还有其它剧情指令数据,是,执行步骤S36;否,更新堵塞时间轴进度值到下一节点,并执行步骤S33。
- [0129] 步骤S4、执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据:
- [0130] 请参阅图6所示,本发明的步骤S4进一步具体包括:步骤S41、输入剧情指令数据,读取剧情指令数据中的剧情指令;
- [0131] 步骤S42、判断剧情指令是否有注册了对应的执行方式,否,剧情指令执行完成并结束流程,是,进入步骤S43,
- [0132] 步骤S43、执行对应的执行方式传入剧情指令数据;
- [0133] 步骤S44、判断是否同步执行执行方式;否,等待异步执行方式执行完成,返回执行成功事件;剧情指令执行完成;是,返回执行成功事件,剧情指令执行完成。
- [0134] 本发明基于游戏剧情Json数据生成统一的通用剧情数据结构,然后通过数据解析模块生成堵塞时间轴剧情数据。当收到用户开启剧情后,数据分发模块按堵塞时间轴分发剧情数据,数据执行模块执行剧情数据中的剧情指令,实现完备的游戏剧情播放与交互控制系统。
- [0135] 请参阅图7所示,本发明还提供了一种自动生成游戏剧情的实现系统,所述实现系统包括数据转换模块、数据解析模块、数据分发模块、以及数据执行模块;
- [0136] 所述数据转换模块,用于输入用于游戏剧情创建的剧情数据,判断输入剧情数据类型是否是Json数据,否,对输入剧情数据进行数据转换;执行数据解析模块;是,直接执行

数据解析模块；

[0137] 所述数据解析模块,用于解析转换后的数据生成堵塞时间轴;该堵塞时间轴为支持堵塞条件的时间轴,其中,堵塞时间轴剧情节点数据未设置堵塞条件节点时,时间轴将按照时间顺序正常运行;当设置了堵塞条件时,播放时间更新到堵塞条件节点时播放时间停止更新,等待堵塞条件满足后播放时间移动至下一个节点的开始位置继续播放;

[0138] 所述数据分发模块,用于按堵塞时间轴分发堵塞时间轴对应的剧情数据;判断堵塞时间轴对应的剧情数据是否分发结束;是,停止结束流程;否,执行数据执行模块;

[0139] 所述数据执行模块,用于执行剧情数据中的剧情指令,继续按堵塞时间轴分发堵塞时间轴对应的剧情数据。

[0140] 数据转换模块:该模块为数据解析模块提供数据转换的能力,实现将Json数据或Json文件路径转换为统一的剧情数据。

[0141] 数据解析模块:该模块提供解析数据生成堵塞时间轴剧情数据的能力,实现将统一的剧情数据解析为堵塞时间轴剧情数据,为数据分发模块提供数据源。

[0142] 数据分发模块:该模块提供按堵塞时间轴分发剧情数据的能力,实现从剧情数据中提起剧情指令数据集合并进行分发,推动剧情指令的执行。

[0143] 数据执行模块:该模块提供剧情指令执行的能力,实现执行剧情指令,将剧情指令中描述的数据应用到场景中的过程。

[0144] 所述数据转换模块的实现方式进一步具体包括:

[0145] 步骤S51、输入用于游戏剧情的创建的剧情数据,判断输入剧情数据类型是否是Json数据,是,进入步骤S52,否,判断是否是Json文件路径,是,读取文件中的Json数据,并进入步骤S52,否,停止流程;

[0146] 步骤S52、将Json数据识别为Json实例类型;

[0147] 步骤S53、创建一个剧情数据结构;

[0148] 步骤S54、将Json数据中的基础数据填充到剧情数据结构的对应字段值中;

[0149] 步骤S55、无匹配字段的数据填充到剧情数据结构的拓展字段队列中;

[0150] 步骤S56、判断剧情数据结构中的剧情数据是否是根节点数据,否,添加到父节点剧情数据的子数据中;是,记录为根节点剧情数据;

[0151] 步骤S57、判断Json数据是否包含子数据,是,进入步骤S53;否,返回创建的根节点剧情数据,并结束流程。

[0152] 所述解析转换后的数据生成堵塞时间轴进一步具体为:

[0153] 步骤S61、输入转换后的用于游戏剧情创建的剧情数据;该剧情数据中包含多个子剧情数据,剧情数据中记录的数据包括剧情指令及其执行方式、剧情节点时间长度、是否堵塞节点、堵塞条件、是否支持跳过;

[0154] 步骤S62、创建一个堵塞时间轴;

[0155] 步骤S63、读取用于游戏剧情创建的剧情数据中的所有剧情节点数据;将所有剧情节点数据加入一个剧情节点数据队列中;

[0156] 步骤S64、判断剧情节点数据队列是否为空,是,返回堵塞时间轴,并结束流程;否,进入步骤S65;

[0157] 步骤S65、从剧情节点数据队列中读取一个剧情节点数据,将剧情节点数据加入堵

塞时间轴的当前节点位置;堵塞时间轴的当前时间值向后移动到剧情节点数据中接入的时间值;

[0158] 步骤S66、判断剧情节点数据中是否记录了堵塞值,该堵塞值为堵塞条件对应的数据值;否,进入步骤S67;是,堵塞时间轴中添加堵塞条件节点,设置堵塞值调整为剧情节点数据中记录的堵塞条件;堵塞时间轴当前时间轴向后移动一个堵塞节点单位时间值,进入步骤S67;

[0159] 步骤S67、判断剧情数据节点队列中是否还有其它剧情数据节点,是,执行步骤S65;否,返回堵塞时间轴,并结束流程。所述堵塞时间轴包括:堵塞时间轴的进度值和时间节点集合,所述时间节点集合由多个时间节点组成,每个时间节点包括时间长度、进度值、以及堵塞条件。

[0160] 所述数据分发模块的实现方式进一步具体包括:

[0161] 步骤S71、输入堵塞时间轴对应的剧情数据;

[0162] 步骤S72、读取堵塞时间轴的进度值;

[0163] 步骤S73、判断堵塞时间轴是否执行完成,是,堵塞时间轴对应的剧情数据分发结束并结束流程,否,读取当前进度值的剧情数据;

[0164] 步骤S74、读取剧情数据中的剧情指令数据集合,将剧情指令数据集合中的数据加入一个剧情指令数据队列;

[0165] 步骤S75、判断剧情指令数据队列是否为空,是,更新堵塞时间轴进度值到下一节点,并执行步骤S73,否,执行步骤S76;本发明中剧情指令的定义包括但不限于显示角色、设置角色的位置旋转或大小、角色的动作播放、角色寻路、角色换装、角色特效播放、摄像机镜头的移动缩放或抖动、剧情UI显示等。

[0166] 步骤S76、从剧情指令数据队列中读取一个剧情指令数据,执行剧情指令数据;

[0167] 步骤S77、判断剧情指令数据队列中是否还有其它剧情指令数据,是,执行步骤S76;否,更新堵塞时间轴进度值到下一节点,并执行步骤S73。

[0168] 所述数据执行模块的实现方式进一步具体包括:步骤S81、输入剧情指令数据,读取剧情指令数据中的剧情指令;

[0169] 步骤S82、判断剧情指令是否有注册了对应的执行方式,否,剧情指令执行完成并结束流程,是,进入步骤S83,

[0170] 步骤S83、执行对应的执行方式传入剧情指令数据;

[0171] 步骤S84、判断是否同步执行执行方式;否,等待异步执行方式执行完成,返回执行成功事件;剧情指令执行完成;是,返回执行成功事件,剧情指令执行完成。

[0172] 本发明适用于所有类型游戏剧情的生成与交互过程,包括但不限于2D游戏中的游戏剧情的生成与交互流程、3D游戏中的游戏剧情的生成与交互流程、AR、VR游戏或应用中的游戏剧情的生成与交互流程、其它类型游戏中的游戏剧情的生成与交互流程。

[0173] 以上所述仅为本发明的较佳实施例,凡依本发明申请专利范围所做的均等变化与修饰,皆应属本发明的涵盖范围。

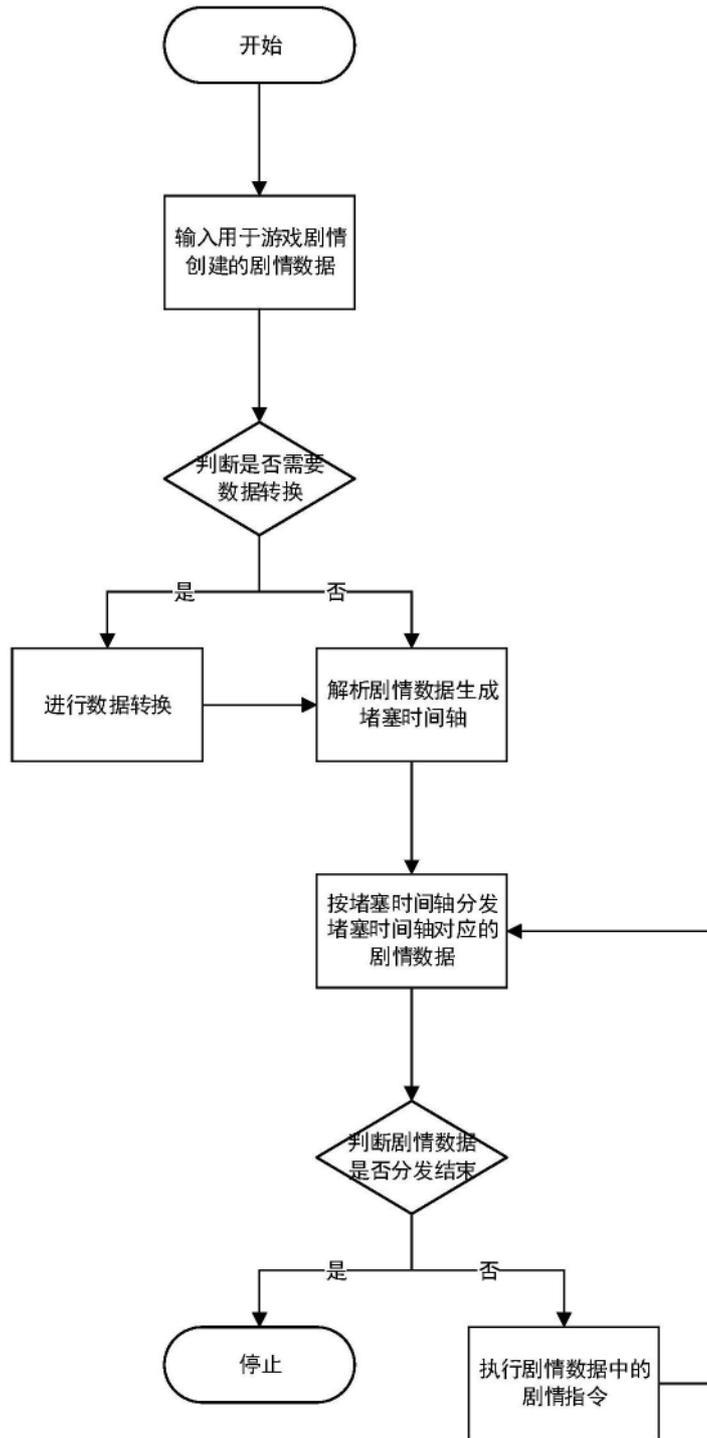


图1

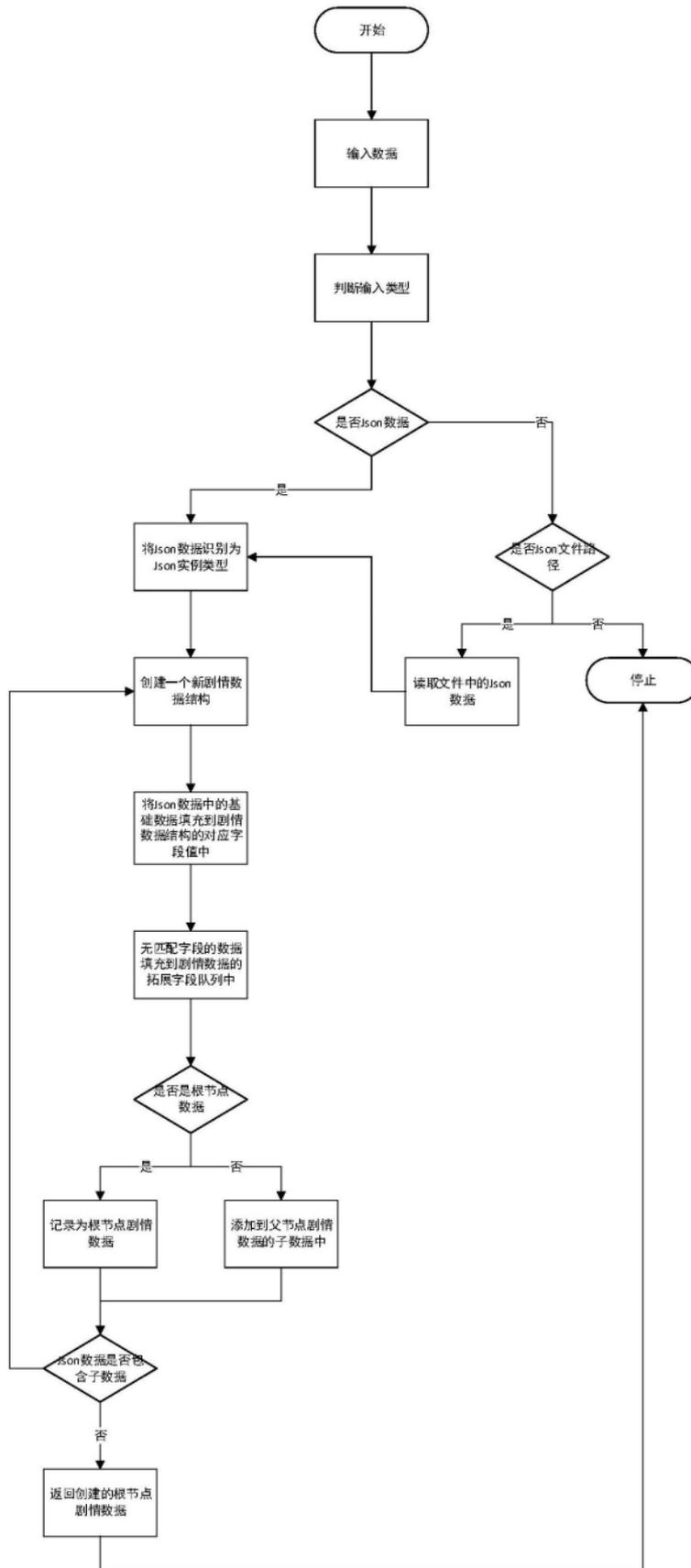


图2

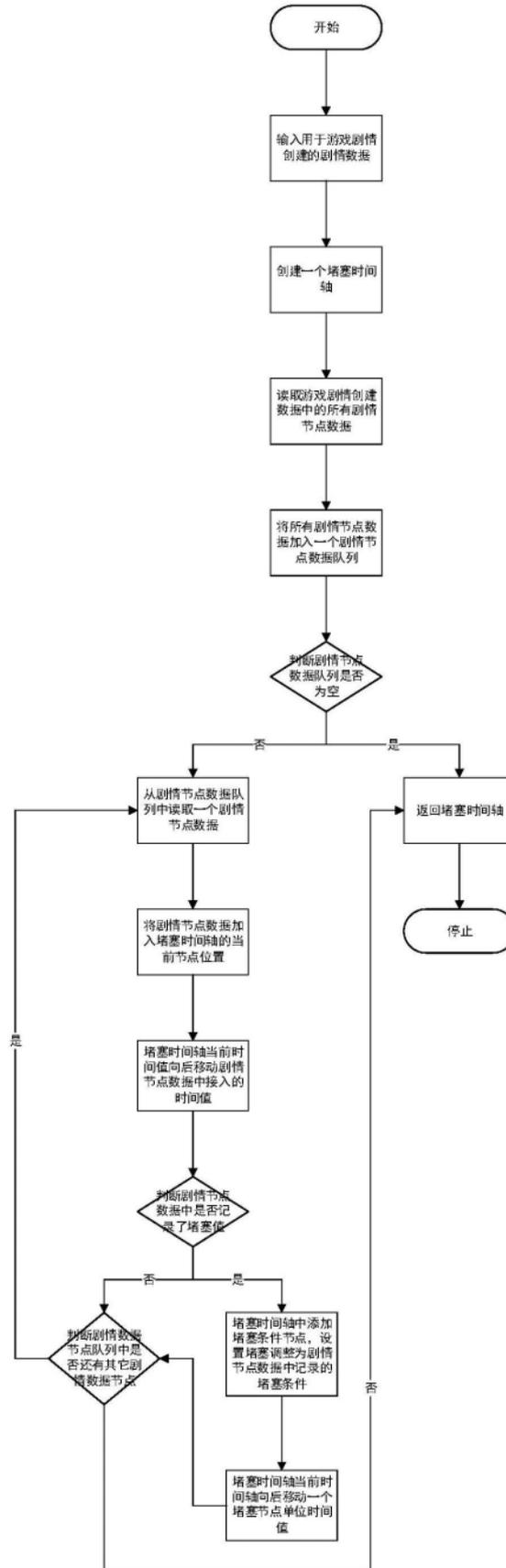


图3

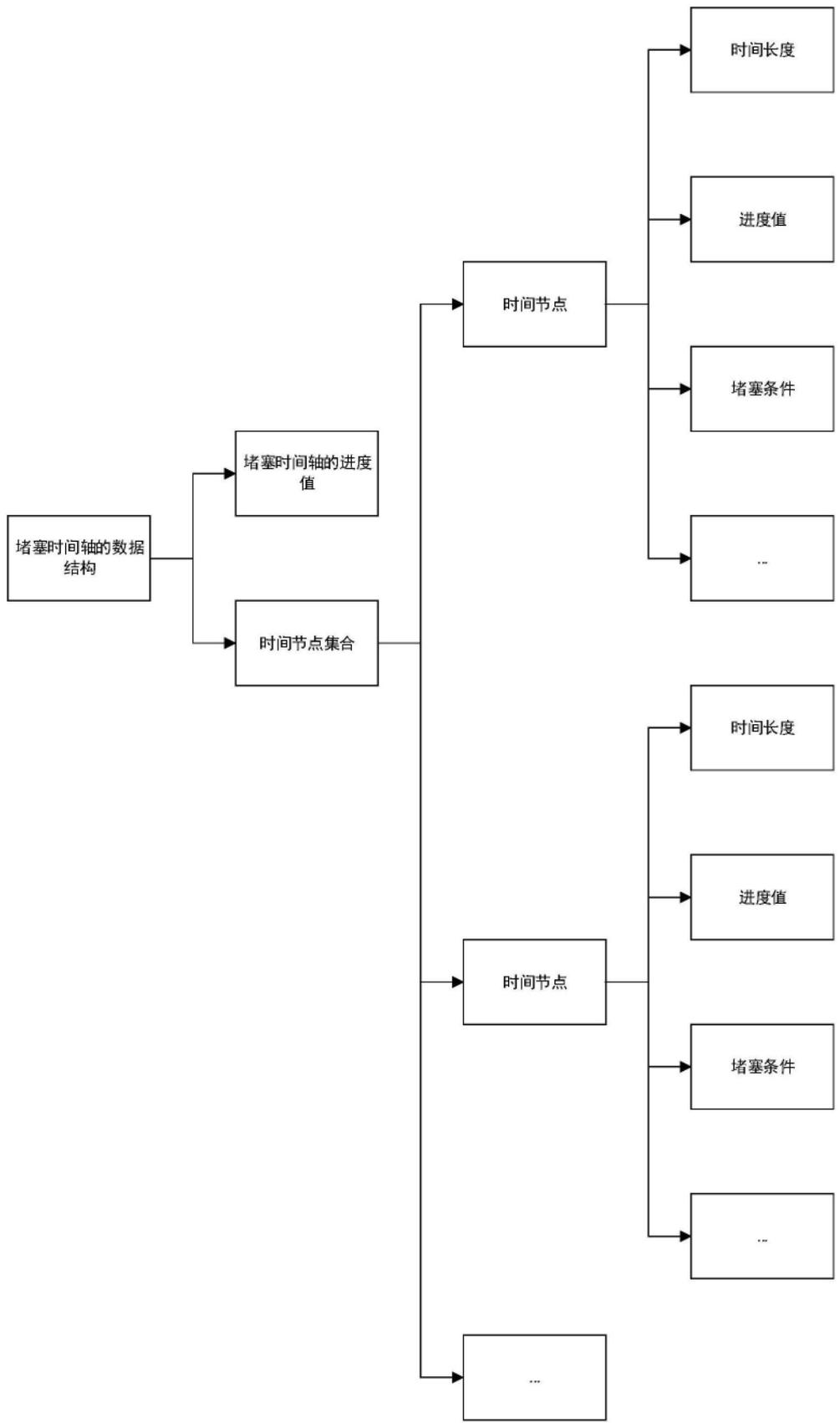


图4

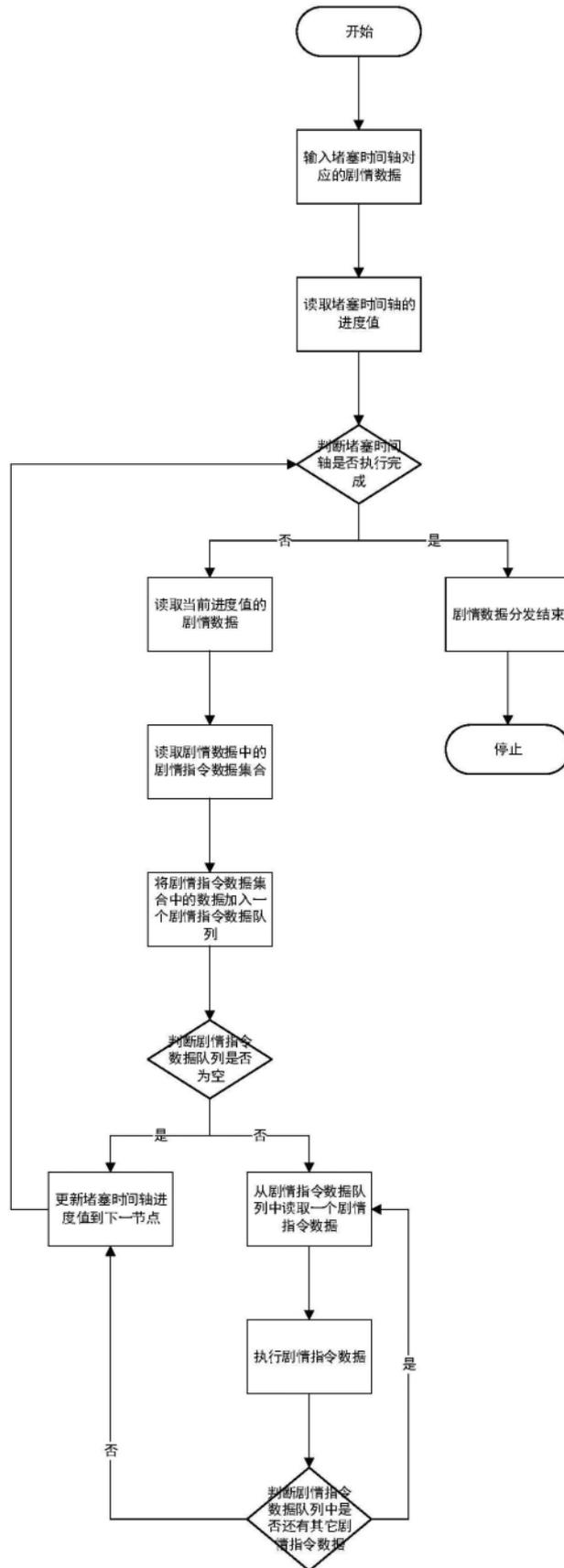


图5

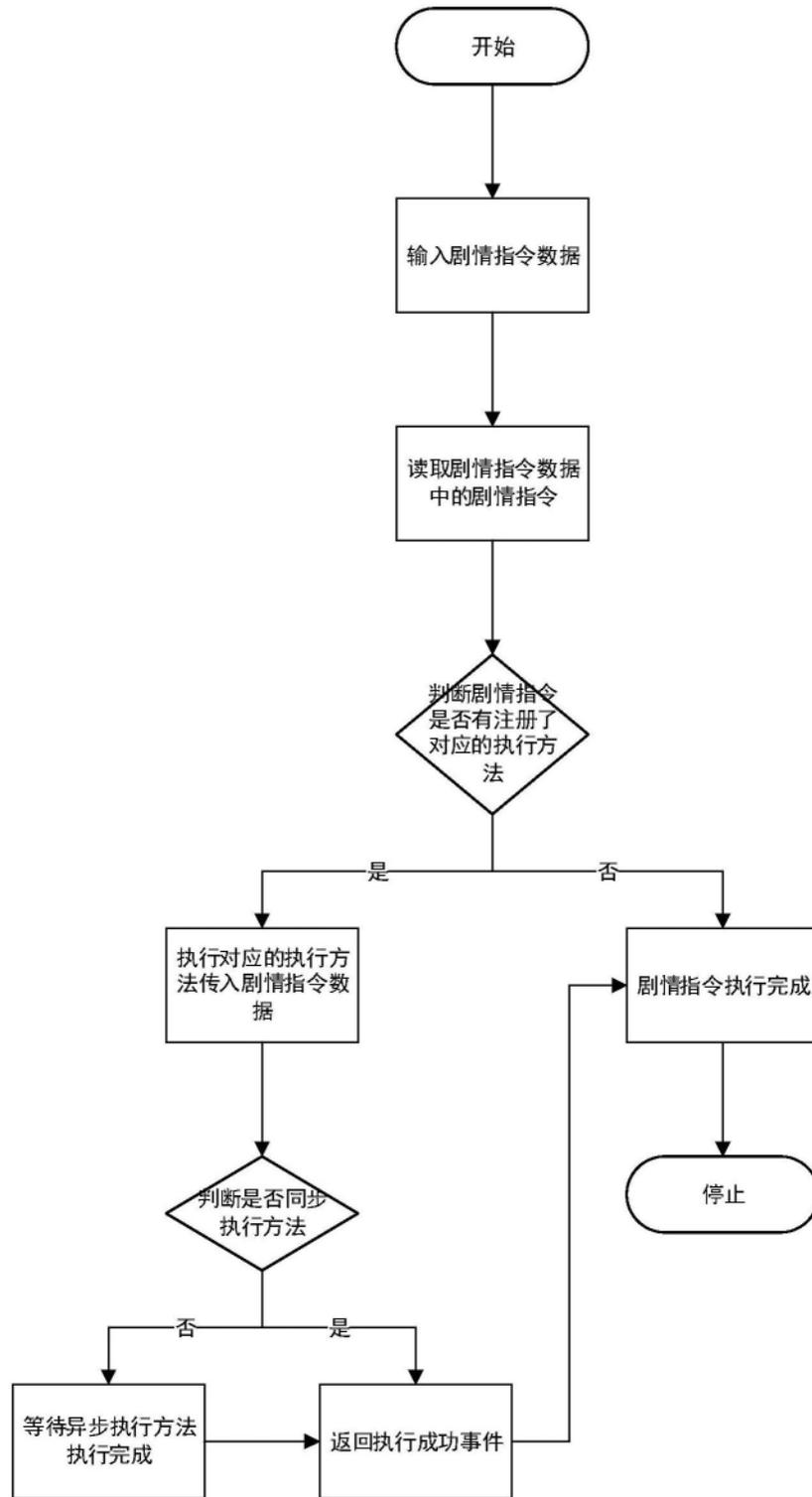


图6

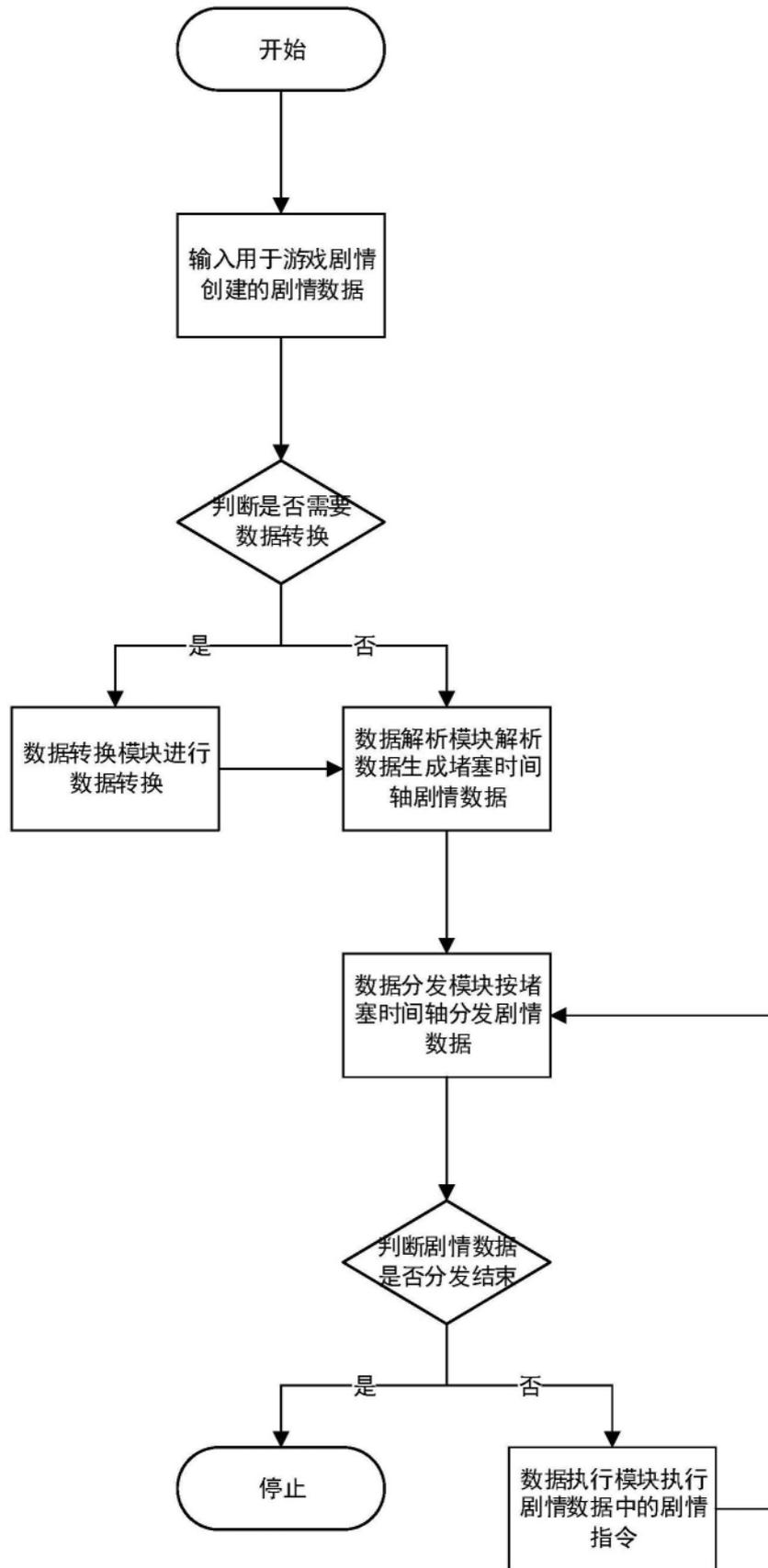


图7