

模拟演练系统中消息机制的研究与设计^①

王玉红^{1,2}, 廉东本², 孙咏², 苏谟²

¹(中国科学院 研究生院, 北京 100049)

²(中国科学院 沈阳计算技术研究所, 沈阳 110168)

摘要: 应急救援模拟演练系统中的模块众多, 相互之间的信息交互错综复杂, 研究了一套适合模拟演练系统的消息机制, 包括消息是如何由消息体和消息类型构成的, 消息工厂如何产生消息的, 消息管理器如何操纵消息与组件、角色以及系统其它部分交互的, 最后举例分析请求同步消息在系统中的处理, 阐释了消息与系统的关联和对系统起的作用。

关键词: 组件; 角色; 消息管理器; 消息类型; 消息注册

Design of Message Mechanism in the Simulation Training System

WANG Yu-Hong^{1,2}, LIAN Dong-Ben², SUN Yong², SU Mo²

¹(Graduate School, Chinese Academy of Sciences, Beijing 100049, China)

²(Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)

Abstract: There are many modules in the Emergency Rescue Simulated Training System, and information exchange between them is complex. This paper studies a set of message mechanism suitable for the simulation training system, which includes how the message is composed of message body and message type, how the message is created by message factory, message manager how to manipulate the message interacting with components, roles, and other parts. Finally, give an example for the analysis of the request synchronous message processing during the system and illustrate the relevance of messages with the system and the role it played in the system.

Key words: component; actor; message manager; message types; message registered

随着经济社会的快速发展, 新能源、新材料、新产品大量涌现, 高层建筑火灾、化学品爆炸等特殊环境下的灾害频繁发生。为减少人员伤亡, 财产损失, 提高在特殊火灾和灾害事故处置行动中的成功率, 必须加强培训基地和模拟训练设施建设, 推动训练手段的模拟化和训练场地的基地化。因此, 很有必要建设一套能逼真地模拟真实灾害现场的应急救援模拟演练系统^[1]。

任何可工作的应急救援模拟演练系统必须能够支持系统中的用户、信息实体之间的信息交互, 使系统中的每位用户(包括原有的用户和新加入的用户)都能收到当前状态的刷新消息, 这样用户便能跟踪虚拟世界状态的变化, 并在此基础上积极的参与到不断变

化的虚拟世界中去^[2]。为了有效的传递消息, 必须研究消息的传递策略和方法, 也就是有关消息传递机制的问题。

在模拟演练系统中消息管理器决定角色与整个世界的交互, 控制着消息的过滤与分发, 因此设计一个好的消息传递机制不仅决定着其它模块能否按照预期实现, 也是整个模拟演练系统成功的关键。

1 应急救援模拟演练系统概述

本系统是针对 2013 年沈阳市全运会的安保系统制定的, 通过对各类灾害的数值模拟仿真和人员行为的数值模拟仿真, 在虚拟空间中仿真灾害发生、发展的过程以及人们在灾害环境中可能做出的各种反应, 并

^① 基金项目: 国家水体污染控制与治理科技重大专项(2012ZX07505003)

收稿时间: 2012-07-05; 收到修改稿时间: 2012-08-23

在演练平台上,在最大限度仿真实际灾害的条件下,开展应急救援演练.在此基础上,制定各类企事业单位的数字化应急预案,对现有预案进行推演.

该系统可以用来训练各级决策与指挥员、事故处置人员,发现应急处置过程中存在的问题,检验和评估应急预案的可操作性和实用性,提高应急能力^[3].

应急救援模拟演练系统也可以经过适当的改良应用于石油、电力、钢铁、冶金、煤矿、化工、航空、地铁、公路、水域、自然灾害,反恐等领域.

2 消息的结构

2.1 消息工厂

模拟演练系统里的消息都是在消息工厂里创建的.通过消息工厂可以使整个系统的消息类型由消息工厂统一管理,使程序员之间的分工明确,程序架构清晰,有利于维护.

消息工厂主要包含以下功能:

- ① 根据消息类型(必须已经在消息类型映射表注册)创建消息.
- ② 根据消息类型的名字、ID 来辨别消息类型映射表里面是否注册过此消息类型.
- ③ 根据已知消息,克隆此消息.
- ④ 注册未被注册的消息类型
- ⑤ 注销已经注册的消息类型.

2.2 消息注册

消息包括消息类型和消息体.消息的注册是由消息管理器负责的,通过消息工厂创建消息之后,就要把消息加入到消息队列里面去,然后每个消息都要分发给所有的组件,分发给注册过该消息类型的所有角色.

2.3 消息定义

1) 所谓消息体就是封装数据的一个容器,根据传输方式的不同可分为本地消息和网络消息.一类消息必须既可以被客户端接收又可以被服务器端接收,所以消息必须在整个系统中进行注册,以便服务器和所有的客户端都能够识别此类消息.由于服务器和客户端的交互都是通过消息,所以需要定义不同的消息体以满足系统的需要.消息体必须和一定的消息类型相关联才能使用.消息体结构如表 1 所示.

消息类型主要用来辨别此消息是否是想要接收的消息;消息源是发送者主机的信息;消息目的指目的

地主机的信息;参数列表类型包含了我们常用的基本数据类型和自定义类型.

表 1 消息体结构

| 消息 | 消息类型 | 消息源主机 | 消息目的主机 | 消息发送者ID | 消息接收者ID | 消息参数列表 |
|----|------|-------|--------|---------|---------|--------|
|----|------|-------|--------|---------|---------|--------|

消息拥有 `osg` 的内存管理机制,由系统统一管理消息的销毁操作.拥有可以序列化的特质(即可以和 `string`、`stream` 相互转化),进而可以转化为在网络上传输的数据.

2) 所谓消息类型实际上就是对消息体进行标识的 ID,接收者必须对消息的消息类型进行判断,确定是否是自己想要的消息,所以消息类型必须是独一无二的.为了判断消息的创建是否合法,消息类型也要在消息类型映射表中注册.消息类型的结构如表 2 所示:

表 2 消息类型描述

| 消息类型 | ID | 名字 | 种类 | 描述 | 消息指针 |
|------|----|----|----|----|------|
|------|----|----|----|----|------|

ID 是标志消息类型的,全系统唯一,需要从用户自定义的 ID 开始;名字指消息类型名字.种类指消息类型类别,表明这个消息类型是什么类型的消息,譬如 `info`、`command`、`request` 等.描述指这个消息类型起到什么作用.消息指针指向消息实体,即此消息类型关联到哪个消息实体,默认填系统自带消息,也可以填写自定义消息实体指针.

3) 消息类型分类

消息类型分类如表 3 所示:

表 3 消息类型分类

| | |
|---------|---------|
| Info | 通知型消息 |
| Tick | 每帧都发的消息 |
| Command | 命令性消息 |
| Request | 请求性消息 |
| Log | 日志类消息 |
| System | 系统类消息 |
| Task | 任务类型消息 |
| Alg | 算法类消息 |

通过为消息类型分类,可以更加规范的为消息类型命名.消息类型名字全为英文字母大写:消息类型的类别_目标消息类型_消息内容描述,如:

- ① `INFO_ACTOR_UPDATED`:通知 actor 更新.
- ② `INFO_ACTOR_SYN`: 通知 actor 同步.

- ③ INFO_FIRE_BEGIN:通知 fire 开始
- ④ REQUEST_ACTOR_RESCUE:请求 actor 救援.

3 消息管理器

消息管理器是模拟演练系统的核心,它有三大功能:第一,维护了一个所管辖范围内所有角色的列表,对所有的角色进行管理;第二,管理一个组件列表,管理已经注册的组件;第三,消息路由功能,给所有的组件和角色发送消息,使其每帧都能够相应的更新.

消息管理器分为客户端消息管理器和服务器端消息管理器.客户端消息管理器负责本地所有角色和组件的管理以及消息的传递.服务器端消息管理器记录了系统中所有的角色,除了处理本地的消息外,还向所有的客户端消息管理器传递消息,当客户端收到从服务器端传递过来的消息后,把消息转换为本地消息,然后传递给本地的角色和组件.

服务器端消息管理器是消息处理与消息分发的主要承担者,对大量消息进行统一处理.消息的接收者主要是所有的组件和角色.由于数据大部分需要经过网络的传输,所以需要把消息转换成可在网络上传输的数据包,再把接收到的数据包转换成消息,用户可以定义自己的数据包.消息管理器的逻辑示意图如图 1 所示:

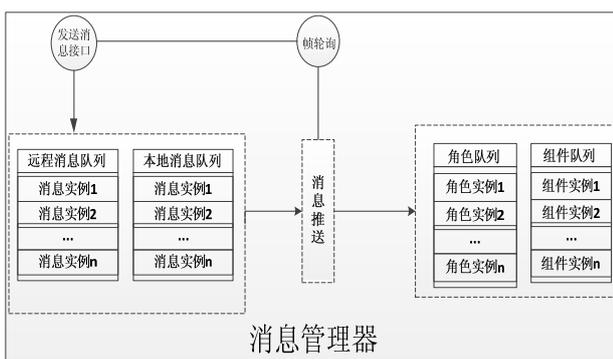


图 1 消息管理器的逻辑示意图

消息管理器的逻辑示意图说明:

角色和组件通过消息工厂创建消息,然后通过消息管理器的发送消息接口 `SendMessage()` 和 `SendNetworkMessage()` 将消息发送到本地消息队列和远程消息队列,在帧轮询的过程中,通过消息推送,把所

有的消息推送给组件实例,把角色注册过的相关消息发送到角色实例中.组件通过接口 `ProcessMessage(Message &msg)` 处理收到的本地消息,根据消息类型处理相关的消息,譬如: `if(msg.GetMessageType() == MessageType::INFO_ACTOR_UPDATE){进行相关处理}`,即如果接收到的消息的消息类型为 `INFO_ACTOR_UPDATE`,对此消息包含的数据进行处理;组件通过接口 `DispatchNetwork Message (Message & msg)` 处理收到的远程消息,处理方式和处理本地消息一样.角色通过以下三个接口注册消息,并对这些消息进行处理.三个接口的参数都有消息类型和 `Invoke` 名字.消息类型前面已做介绍,这里不再赘述, `Invoke` 就是一些专门处理消息的函数,其函数参数只能为消息类型,其定义如: `new Invokable (TICK_LOCAL_INVOKABLE,dtUtil::Make Functor(&Game Actor::OnTickLocal, GetGameActor()))`,前参为 `invoke` 名字,唯一,后参为相应的处理消息类型的函数.

① `RegisterForMessages(MessageType, invokable-name)`:注册消息类型和处理此类消息的方法.通过这个接口注册的消息类型,所有关注此类消息的角色都可接收到,并进行相应处理.

② `RegisterForMessagesAboutOtherActor(Message Type, UniqueId, invokablename)`:注册消息类型和消息目的 `ActorId` 的消息.所有消息的消息类型是此消息类型,目的 `ActorId` 是此 `UniqueID` 的消息 都可被接收到,并通过相应的 `invoke` 进行处理.

③ `RegisterForMessagesAboutSelf(MessageType, invokablename)`:注册所有目标 `actor` 是自己,消息类型是此消息类型的消息.所有的这种消息都可接收到,并被注册的 `invoke` 处理.角色接收到消息后,通过注册消息类型时对应的 `invoke` 进行处理.

4 消息的在系统中的应用

消息主要用于在系统之间、系统组件之间、角色之间传递数据.所以消息是整个模拟演练系统的核心,几乎每个模块中都要使用消息来传递数据.下面通过分析同步管理器中的请求同步消息的处理过程来分析消息在系统中的应用.

同步管理器中的请求同步消息的处理流程如图 2 所示:

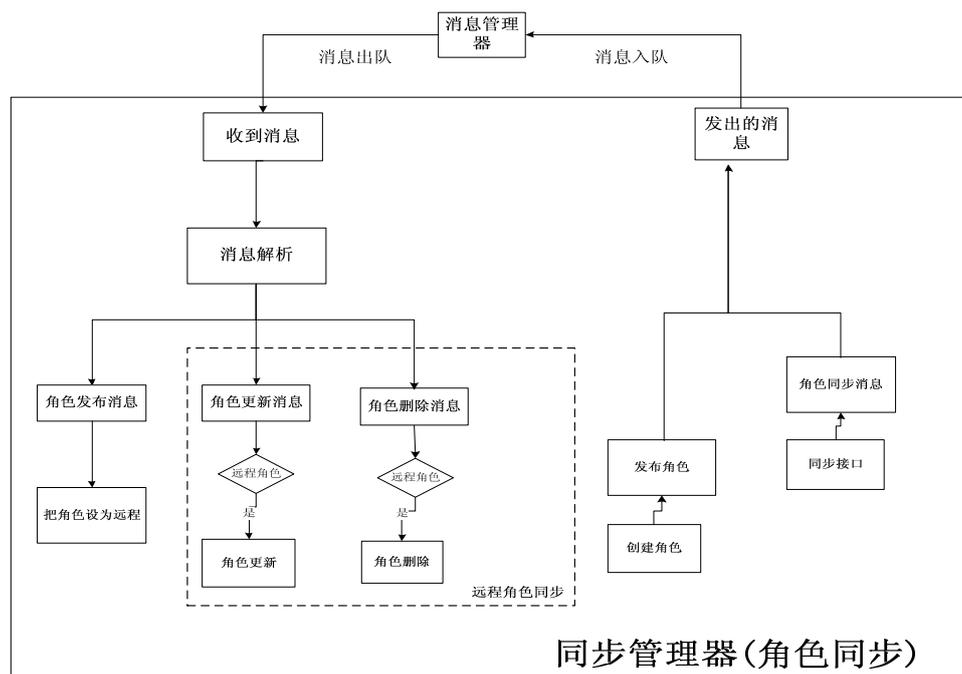


图 2 同步管理器逻辑图

同步管理器逻辑说明:

1) 当服务器端使用角色工厂创建一个角色的时候, 发出一个角色发布消息. 客户端同步管理器收到角色发布消息后, 对消息进行判断, 如果角色列表中没有该角色的时候就创建一个此类角色, 加入角色列表中.

2) 当角色属性数据发生变化后, 需发出角色更新消息, 同步管理器接收角色更新消息, 然后查找角色列表找到该角色, 把更新数据应用到该角色.

3) 当删除一个角色时候, 需要发出一个删除角色的消息, 同步管理器接收到角色删除消息后, 查找角色列表找到该角色, 然后从角色列表中删除该角色.

5 结语

消息要分发给所有的组件, 所以每个组件提供一个统一的接口处理消息, 消息因此具有很好的可扩展性. 另外, 只有经过注册的类型的消息才会被分发给角色, 通过查找消息映射表, 把各种类型的消息有针对性的分发出去, 极大的提高了消息处理的效率.

本文通过分析消息体, 消息类型以及消息工厂的结构, 分析消息管理器对消息的整体控制作用, 研究了模拟演练系统中的消息机制.

参考文献

- 1 郭华芳,李志文.消防模拟训练系统初探.消防科学与技术, 2007(1):86-88.
- 2 Thinpig.学习XML消息机制,编写一个简单的XML消息代理.中文信息,2002(2):82-83.
- 3 孙成江,刘林.应急救援模拟演练系统设计与实现初探.石油工业计算机应用,2010,67(3):3-6.
- 4 廖宁,马鸣辉.模拟演练系统综述.理论探索,2010,23(5):1-2.
- 5 郑白玫,施鹏飞.大型分布式虚拟环境中的消息传递机制的关键问题.计算机科学,2005,(5):1-2.
- 6 郭航,王兴伟,黄敏,蒋定德.基于多队列自适应的DTN 传染路由算法.小型微型计算机系统,2012(4)2-3.
- 7 卢本捷.分布式消息队列的理论、实现与应用.[博士学位论文].武汉:华中科技大学,2004.