



(12) 发明专利申请

(10) 申请公布号 CN 114629971 A

(43) 申请公布日 2022.06.14

(21) 申请号 202210178386.X

(22) 申请日 2022.02.25

(71) 申请人 浙江爱充网络科技有限公司

地址 310012 浙江省杭州市西湖区天际大厦1104室

(72) 发明人 张东阳 韩旺坤 胡群峰

(74) 专利代理机构 杭州杭诚专利事务所有限公司 33109

专利代理师 叶绍华

(51) Int. Cl.

H04L 69/22 (2022.01)

H04L 43/10 (2022.01)

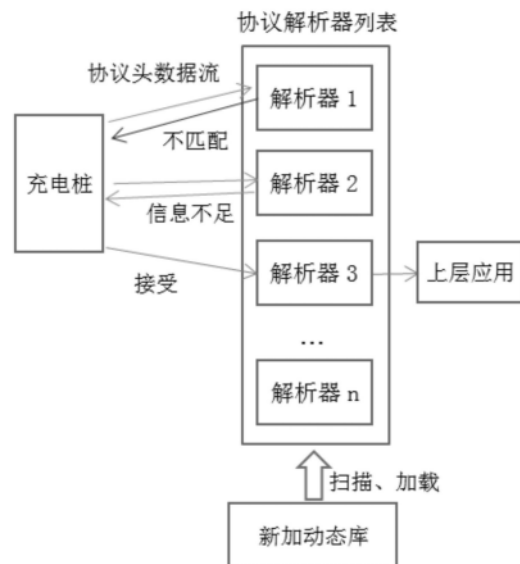
权利要求书1页 说明书5页 附图2页

(54) 发明名称

一种电动汽车充电服务平台的桩服务程序热升级系统及方法

(57) 摘要

本发明提供了一种电动汽车充电服务平台的桩服务程序热升级系统及方法,系统包括电动汽车充电服务平台,所述电动汽车充电服务平台通过服务程序与若干充电桩通信连接,所述服务程序从下到上包括应用层、表示层、会话层和网络层,所述表示层设有协议解析器列表,所述协议解析器列表对应若干协议解析器,相邻层之间设有通用的调用接口,并禁止跨层调用;所述网络层用于实现电动汽车充电服务平台与充电桩的网络通信;所述会话层用于接收对端数据流,并保存对端通信协议信息;所述表示层采用轮询的方式用于匹配对端通信协议信息对应的协议解析器。本发明保持到桩的连接不断,可以做到升级时平台服务无需重启、充电服务升级时充电业务不中断。



1. 一种电动汽车充电服务平台的桩服务程序热升级系统,其特征是,包括电动汽车充电服务平台,所述电动汽车充电服务平台通过服务程序与若干充电桩通信连接,所述服务程序从下到上包括应用层、表示层、会话层和网络层,所述表示层设有协议解析器列表,所述协议解析器列表对应若干协议解析器,相邻层之间设有通用的调用接口,并禁止跨层调用;所述网络层用于实现电动汽车充电服务平台与充电桩的网络通信;所述会话层用于接收对端数据流,并保存对端通信协议信息;所述表示层采用轮询的方式用于匹配对端通信协议信息对应的协议解析器,应用层用于接收并应用对端数据流,充电桩对应的会话层仅保存协议名称,会话层需要访问协议解析器时,通过协议名称从解析器管理者获取解析器对象,调用后立即释放对象引用。

2. 一种电动汽车充电服务平台的桩服务程序热升级方法,采用权利要求1所述的一种电动汽车充电服务平台的桩服务程序热升级系统,其特征是,包括以下步骤:

S1,会话层收到对端数据流后,判断是否保存了对端通信协议信息,若是,则根据协议名称从解析器管理者处获取协议解析器实例,再进入S3,若否进入S2;

S2,采用轮询的方法从协议解析器列表中搜索相应的协议解析器,判断是否成功匹配到协议解析器,若是,再进入S3,若否反馈匹配失败;

S3,升级时,替换掉解析器管理者成员变量为新协议解析器对象实例,无需告知会话层和应用层。

3. 根据权利要求2所述的一种电动汽车充电服务平台的桩服务程序热升级方法,每个请求上下文环境中并不保存对象引用,仅保存桩编号或协议名称,需要调用各层对象时,根据桩编号或协议名称,从对象集中管理器中查找,每次消息发送完消息或处理完消息,立即释放掉对解析器实例的引用。

4. 根据权利要求2所述的一种电动汽车充电服务平台的桩服务程序热升级方法,所述S2具体包括以下步骤:

对端需要首先传协议帧,告知当前平台充电桩所采用的协议信息;

协议解析器加载到服务程序内存后,收到来自充电桩的协议头网络数据流时,采用轮询的方法从协议解析器列表中搜索相应的协议解析器;

各协议解析器收到协议帧后,对比协议头相应的协议标识以及CRC校验码,轮番解析此协议数据,根据匹配情况,返回三种结果:接受、匹配失败、信息不足。

5. 根据权利要求4所述的一种电动汽车充电服务平台的桩服务程序热升级方法,所述轮询的过程如下:

若有协议解析器返回接受,则表示匹配成功停止搜索,否则继续询问列表中的其它协议解析器。

若所有解析器都返回匹配失败,则表示协议不支持,返回错误给对端,静默一段时间后断开连接;

若返回信息不足,则表明协议帧数据并未全部到达,先不做任何处理,缓存住已经到达的部分数据,等待更多的网络流数据到达后,组成完整的数据帧,再次发起协议轮询。

6. 根据权利要求2所述的一种电动汽车充电服务平台的桩服务程序热升级方法,每种协议只有一个解析器对象,所有的协议解析器对象的生命周期,由全局唯一的解析器管理者统一管理。

## 一种电动汽车充电服务平台的桩服务程序热升级系统及方法

### 技术领域

[0001] 本发明涉及充电服务程序热升级技术领域,尤其是一种电动汽车充电服务平台的桩服务程序热升级系统及方法。

### 背景技术

[0002] 随着新能源汽车占比越来越高,作为新基建的一部分,越来越多的商家参与到电动汽车充电桩设计、制造和运营。

[0003] 参照中电联T/CEC 102标准,各充电桩厂商定制化开发自己的充电桩与充电平台的通信协议,不同厂商有不同的协议,同一厂家不同年代的充电桩通信协议也不一样。

[0004] 同一充电平台服务需要接入成千上万根充电桩,为了接入不同厂家的充电桩,充电平台需要解析各种不同的通信协议,当其中某家通信协议升级时,往往需要升级充电平台服务。

[0005] 目前,充电桩程序通过建立TCP直连方式或MQTT协议,连接各平台开发的后端桩服务程序。后端服务升级重启,将导致桩与服务之间的连接被断开,充电业务会出现短暂中断。

[0006] 充电平台运营商往往挑在凌晨升级,并事先告知客户和公司管理层。但是凌晨电价相对便宜,很多公交车、运营车辆仍在凌晨充电,服务的突然重启会影响到他们充电。而且运营车辆对充电服务平台的可用性要求很高,一旦充不上电,将影响公司运营业务正常开展。

[0007] 充电平台需要支持的充电桩通信协议越多,平台服务需要升级重启的次数越多,重启导致充电业务闪断的次数越多。

[0008] 为提高平台服务可用性,需要尽可能避免平台服务重启导致的服务中断。

### 发明内容

[0009] 本发明为了提高充电平台的可用性、提升充电用户体验,提供了一种电动汽车充电服务平台的桩服务程序热升级系统及方法,保持到桩的连接不断、充电桩和充电用户无感知,可以做到升级时平台服务无需重启、充电服务升级时充电业务不中断。

[0010] 为实现上述目的,提出以下技术方案:

[0011] 一种电动汽车充电服务平台的桩服务程序热升级系统,包括电动汽车充电服务平台,所述电动汽车充电服务平台通过服务程序与若干充电桩通信连接,所述服务程序从下到上包括应用层、表示层、会话层和网络层,所述表示层设有协议解析器列表,所述协议解析器列表对应若干协议解析器,相邻层之间设有通用的调用接口,并禁止跨层调用;所述网络层用于实现电动汽车充电服务平台与充电桩的网络通信;所述会话层用于接收对端数据流,并保存对端通信协议信息;所述表示层采用轮询的方式用于匹配对端通信协议信息对应的协议解析器,应用层用于接收并应用对端数据流,充电桩对应的会话层仅保存协议名称,会话层需要访问协议解析器时,通过协议名称从解析器管理者获取解析器对象,调用后

立即释放对象引用。

[0012] 本发明充电接入服务负责与充电桩通信,将服务从下到上分成4层:应用层、表示层、会话层、网络层,协议解析器位于表示层。相邻层之间设计通用的调用接口,基于接口编程,不同层有不同的实现,不允许跨层调用。充电服务启动后,后台任务周期性扫描指定目录,尝试加载实现了相应层级调用接口的解析器动态库,仅保存各通信协议解析器的最新版本。扫描时,根据统一的命名规则,尝试分析协议解析器是否实现了通用的表示层接口,只有实现了分层接口的动态库才被加载进解析器列表。一个充电平台接入服务可以接入成千上万根充电桩,每根电桩一条网络连接,分别对应一个会话层实例、表示层实例、应用层实例。

[0013] 一种电动汽车充电服务平台的桩服务程序热升级方法,采用上述的一种电动汽车充电服务平台的桩服务程序热升级系统,包括以下步骤:

[0014] S1,会话层收到对端数据流后,判断是否保存了对端通信协议信息,若是,则根据协议名称从解析器管理者处获取协议解析器实例,再进入S3,若否进入S2;

[0015] S2,采用轮询的方法从协议解析器列表中搜索相应的协议解析器,判断是否成功匹配到协议解析器,若是,再进入S3,若否反馈匹配失败;

[0016] S3,升级时,替换掉解析器管理者成员变量为新协议解析器对象实例,无需告知会话层和应用层。

[0017] 作为优选,每个请求上下文环境中并不保存对象引用,仅保存桩编号或协议名称,需要调用各层对象时,根据桩编号或协议名称,从对象集中管理器中查找,每次消息发送完消息或处理完消息,立即释放掉对解析器实例的引用。

[0018] 作为优选,所述S2具体包括以下步骤:

[0019] 对端需要首先传协议帧,告知当前平台充电桩所采用的协议信息;

[0020] 协议解析器加载到服务程序内存后,收到来自充电桩的协议头网络数据流时,采用轮询的方法从协议解析器列表中搜索相应的协议解析器;

[0021] 各协议解析器收到协议帧后,对比协议头相应的协议标识以及CRC校验码,轮番解析此协议数据,根据匹配情况,返回三种结果:接受、匹配失败、信息不足。

[0022] 作为优选,所述轮询的过程如下:

[0023] 若有协议解析器返回接受,则表示匹配成功停止搜索,否则继续询问列表中的其它协议解析器。

[0024] 若所有解析器都返回匹配失败,则表示协议不支持,返回错误给对端,静默一段时间后断开连接;

[0025] 若返回信息不足,则表明协议帧数据并未全部到达,先不做任何处理,缓存住已经到达的部分数据,等待更多的网络流数据到达后,组成完整的数据帧,再次发起协议轮询。

[0026] 作为优选,每种协议只有一个解析器对象,所有的协议解析器对象的生命周期,由全局唯一的解析器管理者统一管理,有利于统一升级。

[0027] 本发明的有益效果是:保持到桩的连接不断、充电桩和充电用户无感知,可以做到升级时平台服务无需重启、充电服务升级时充电业务不中断。

## 附图说明

- [0028] 图1是实施例的轮询示意图；  
[0029] 图2是实施例的系统构架示意图。

## 具体实施方式

[0030] 实施例：

[0031] 本实施例提出一种电动汽车充电服务平台的桩服务程序热升级系统，包括电动汽车充电服务平台，所述电动汽车充电服务平台通过服务程序与若干充电桩通信连接，所述服务程序从下到上包括应用层、表示层、会话层和网络层，所述表示层设有协议解析器列表，所述协议解析器列表对应若干协议解析器，相邻层之间设有通用的调用接口，并禁止跨层调用；所述网络层用于实现电动汽车充电服务平台与充电桩的网络通信；所述会话层用于接收对端数据流，并保存对端通信协议信息；所述表示层采用轮询的方式用于匹配对端通信协议信息对应的协议解析器，应用层用于接收并应用对端数据流，充电桩对应的会话层仅保存协议名称，会话层需要访问协议解析器时，通过协议名称从解析器管理者获取解析器对象，调用后立即释放对象引用。本实施例还提出一种电动汽车充电服务平台的桩服务程序热升级方法，采用上述的一种电动汽车充电服务平台的桩服务程序热升级系统，包括以下步骤：S1，会话层收到对端数据流后，判断是否保存了对端通信协议信息，若是，则根据协议名称从解析器管理者处获取协议解析器实例，再进入S3，若否进入S2；S2，采用轮询的方法从协议解析器列表中搜索相应的协议解析器，判断是否成功匹配到协议解析器，若是，再进入S3，若否反馈匹配失败；S3，升级时，替换掉解析器管理者成员变量为新协议解析器对象实例，无需告知会话层和应用层。

[0032] 本实施例的具体实施方式参考图1和图2，平台的电桩接入服务对外互联分为两部分：

[0033] 1) 物联网部分，通过Netty网络框架与充电桩TCP直连，同一服务程序最多可连入2万根充电桩。

[0034] 2) 互联网部分，通过Java Spring框架，接收充电启动停止、查询充电进度等各种请求；同时通过消息队列中间件比如RabbitMQ，发送电桩异步应答、充电实时数据上报等消息给其它微服务。

[0035] Netty Channel封装了网络层，每根充电桩一条长连接，对应一个Channel实例，通过Channel实例发送和接收消息，与充电桩网络通信。

[0036] 参考图2最右边：

[0037] EpNettySession、PresentationProxy、AppUpdateHandlerImpl分别对应会话层、表示层、应用层，相互持有以便跨层调用。

[0038] 为每根充电桩分别创建一个Channel、EpNettySession、PresentationProxy、AppUpdateHandlerImpl对象。为了热升级方便，表示层应用代理模式，将PresentationProxy设计为请求代理，并不持有协议解析器EpPresentation对象。PresentationProxy作为代理，同样实现了表示层的接口，但是它只负责获取相应的协议解析器并转发请求，并不真正实现表示层相关的业务逻辑。

[0039] 同一电桩协议有多根充电桩，同一EpPresentation实例针对不同桩，需要引用不

同的上下文对象,故设计PresentationContext封装电桩请求上下文。

[0040] 当EpPresentation要发送消息到某指定的桩时,通过PresentationContext得到桩相关的会话层、应用层对象或桩编号等请求上下文。PresentationContext的引入,让多个充电桩动态依赖而不是持有协议解析器成为可能。

[0041] 连接时断时续,与连接对应的Channel是动态和易逝的,但会话层比较稳定,只有在桩持续失联时才会释放会话。所以Channel实例的生命周期与EpNettySession不一致,不设计成相互持有的强依赖关系。Channel和会话层都记下桩ID,借助EpSessionAdmin建立动态依赖,需调用对方时通过桩ID查找到相应对象实例。

[0042] EpSessionAdmin作为Channel和Session的容器,以桩ID为键,以键值对方式保存Channel和Session对象。当连接连入且根据协议解析出对端桩ID后,保存对应Channel和Session。连接断开时移除Channel,当连接断开一段时间后会话超时,移除桩对应的Session。

[0043] 为做到热升级,容易变化的协议解析部分封装在子类中,子类实现ProtocolImpl抽象类。PresentationLoader周期性扫描指定Jar包目录,加载各种ProtocolImpl的实现类,子类继承于ProtocolImpl,实现了协议名称和版本的查询接口,实现协议匹配检查接口等,PresentationLoader仅保留各协议最新版解析器。

[0044] 每个Channel里面,保存了协议名称和对端桩ID。对于新建的Channel,没有协议和桩ID信息,此时需要等对端发送的协议帧到达,然后Channel收到网络流数据后,轮询PresentationLoader中各ProtocolImpl实例,ProtocolImpl实例尝试根据各自协议定义,首先判断协议标识是否一致,然后根据长度字段切分出整个协议帧。如果帧有CRC校验,根据CRC计算帧数据是否正确。如果没有校验信息,需要等后续帧数据到达,再次比对协议标识。

[0045] 对同协议Jar包热升级时,需要用轮询搜索到类去替换掉之前加载到内存中的同名类。Java默认是从父ClassLoader中加载类,所以当jar包升级后,同包名和类名的类已经加载到内存中之后,默认没办法以旧换新。故实现时,需要自定义URLClassLoader的loadClass方法,对新升级本jar包中的类,让其不从父loader加载,而是从本次升级后的jar中加载Java类。

[0046] 构造了分层模型中与桩对应的网络层、会话层、表示层实例后,接着绑定应用层实例。应用层实例由上层创建和初始化,PresentationLoader提供一个注册方法,用于应用层注册回调函数,在协议解析器加载成功并解析出桩ID后,表示层通过此回调得到应用层实例,并彼此持有对方。

[0047] 从图2可以看到,协议解析器的生命周期由全局唯一的管理类实例PresentationLoader管理,其它类都是动态依赖,需要时根据协议名称去查找协议解析器。加载到新版本协议解析器时,PresentationLoader替换掉协议解析器后,无需通知其它层实例即可完成升级。

[0048] 消息从下到上逐层传递时,网络层每个Channel对象保存了根据协议帧解析出的桩ID,收到消息后,通过桩ID访问EpSessionAdmin即可得到对应的会话层对象。因为EPNettySession、PresentationProxy彼此持有对方,PresentationProxy、AppUpdateHandlerImpl彼此持有对方。网络层调用会话层,会话层处理后,再调用表示层,

表示层处理后,再调用应用层。

[0049] 以充电桩ID为主键,应用层保存了各桩对应应用层实例。当要通知桩启动或停止充电、查询桩实时充电状态时,根据桩ID从EpCommClientService获取AppUpdateHandlerImpl实例,让请求从上到下依次经过各层处理后到达充电桩。

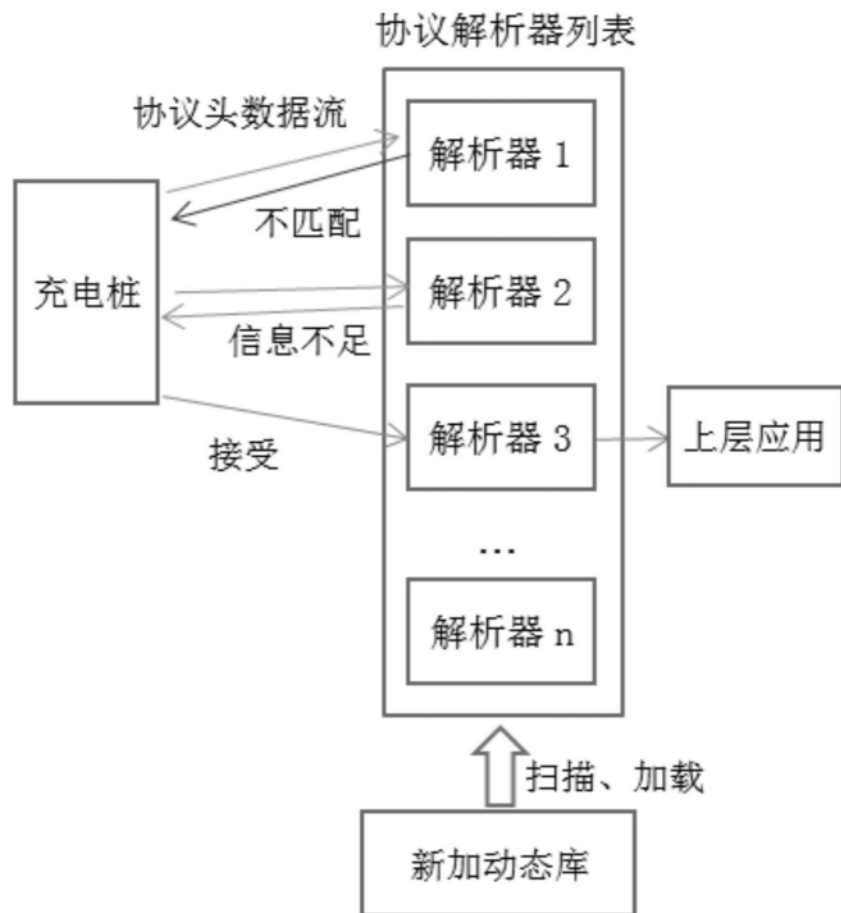


图1



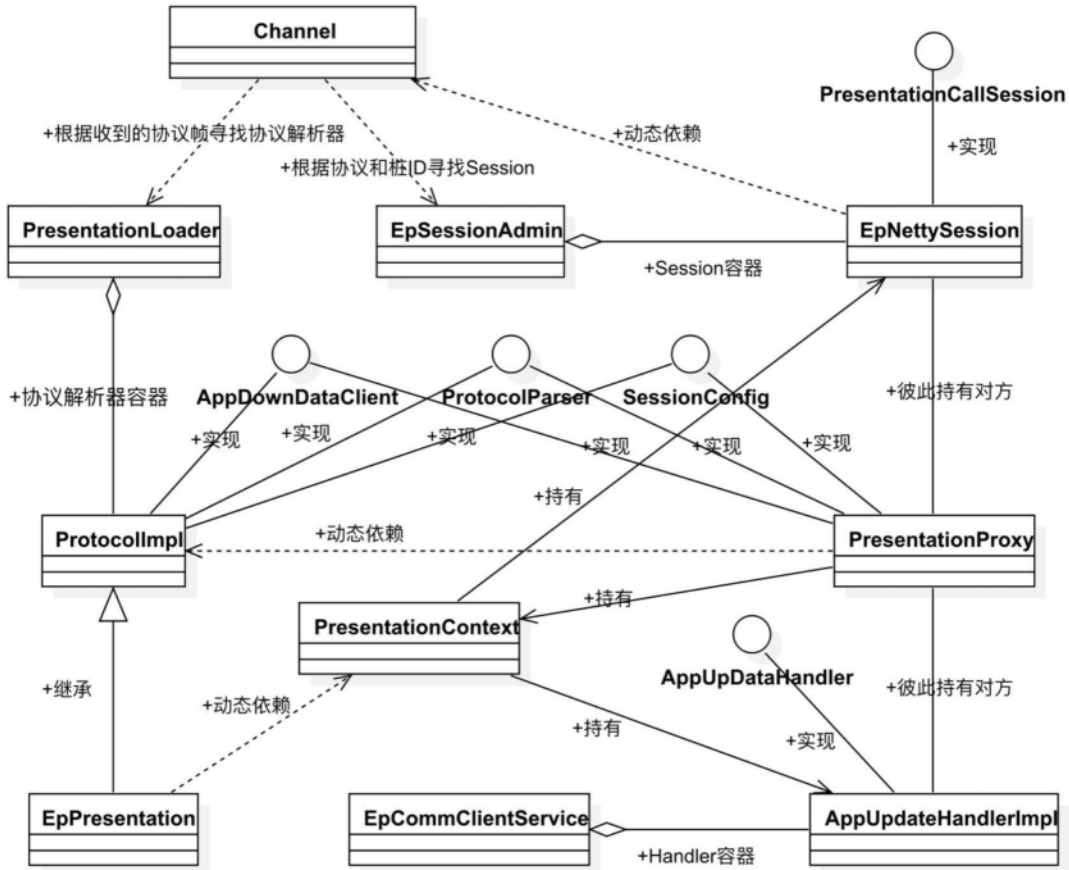


图2