



(12)发明专利

(10)授权公告号 CN 105472042 B

(45)授权公告日 2018.09.21

(21)申请号 201610027711.7

(22)申请日 2016.01.15

(65)同一申请的已公布的文献号
申请公布号 CN 105472042 A

(43)申请公布日 2016.04.06

(73)专利权人 中煤电气有限公司
地址 101300 北京市顺义区林河大街9号

(72)发明人 张昊 邓珑 杨会龙 王亚智
潘志国

(74)专利代理机构 北京正理专利代理有限公司
11257

代理人 付生辉 谢小延

(51)Int.Cl.
H04L 29/08(2006.01)

(56)对比文件

CN 104954469 A,2015.09.30,
CN 103944924 A,2014.07.23,
CN 102255934 A,2011.11.23,
CN 104092767 A,2014.10.08,
US 2014149485 A1,2014.05.29,

审查员 肖鸿

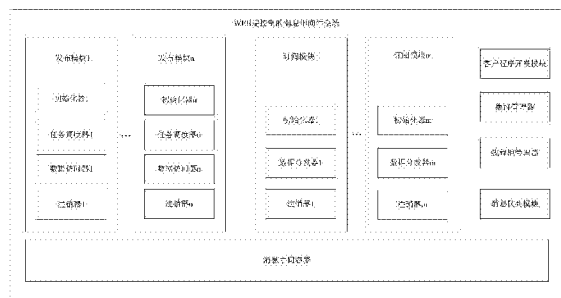
权利要求书2页 说明书8页 附图2页

(54)发明名称

WEB端控制的消息中间件系统及其数据传送方法

(57)摘要

本发明公开一种WEB端控制的消息中间件系统及其数据传送方法,数据传送方法包括如下步骤:S1、初始化WEB端控制的消息中间件系统;S2、根据发送客户端的设置,利用WEB端控制的消息中间件系统采用推送或拉取的方式从发送客户端接收数据并将该数据转换为JSON格式的消息;S3、根据接收客户端的设置,利用WEB端控制的消息中间件系统将对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据。本发明所述技术方案可实现企业内或企业间各个异构系统、不同的数据存储格式之间进行数据相互交换。



1. 一种WEB端控制的消息中间件系统,其特征在于,该系统包括:

至少一个与发送客户端一一对应的发布模块,根据对应的发送客户端的设置,采用推送或拉取的方式从对应的发送客户端接收数据并将该数据转换为JSON格式的消息;

至少一个与接收客户端一一对应的订阅模块,根据对应的接收客户端的设置,将对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据;

消息队列模块,生成数据传送两端的一组发送客户端与接收客户端的消息队列;

线程池管理器,对发布模块、订阅模块、消息队列模块产生的多个线程进行池化管理;

消息中间件容器,调度与管理各发布模块和订阅模块;

客户端程序开发模块,实现发布模块和订阅模块以推送的方式进行数据传送,并将订阅模块未发送成功的数据转换为消息保存到内存的消息队列中。

2. 根据权利要求1所述WEB端控制的消息中间件系统,其特征在于,所述发布模块包括:

初始化器,在发布模块初始构建时提供发布模块的全局配置信息作为其它功能单元的全局变量;

任务调度器,触发执行采用推送或拉取的方式从对应的发送客户端接收数据任务的调度指令;

数据访问器,按照调度指令采用推送或拉取的方式从发布模块对应的发送客户端接收数据并将该数据转换为JSON格式的消息;

注销器,在发布模块停止工作或关闭服务时释发相应外部资源。

3. 根据权利要求2所述WEB端控制的消息中间件系统,其特征在于,所述数据访问器为以下数据访问器的一种或任意组合:本地文件数据访问器、远程文件数据访问器、FTP数据访问器、webservice数据访问器、数据库最后更新时间数据访问器、快照表数据库访问器。

4. 根据权利要求1所述WEB端控制的消息中间件系统,其特征在于,所述订阅模块包括:

初始化器,在订阅模块初始构建时提供订阅模块的全局配置信息作为其它功能单元的全局变量;

数据分发器,将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据;

注销器,在订阅模块停止工作或关闭服务时释发相应外部资源。

5. 根据权利要求4所述WEB端控制的消息中间件系统,其特征在于,所述数据分发器为以下分发器的一种或任意组合:控制台分发器、数据库表分发器、web服务分发器、webservice分发器。

6. 根据权利要求1所述WEB端控制的消息中间件系统,其特征在于,该系统还包括:集群管理器,将各发布模块和订阅模块作为独立节点从而实现集群管理。

7. 一种如权利要求1所述WEB端控制的消息中间件系统的数据传送方法,其特征在于,该方法包括如下步骤:

S1、初始化WEB端控制的消息中间件系统;

S2、根据发送客户端的设置,利用WEB端控制的消息中间件系统采用推送或拉取的方式从发送客户端接收数据并将该数据转换为JSON格式的消息;

S3、根据接收客户端的设置,利用WEB端控制的消息中间件系统将对应的接收客户端所

需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据。

8. 根据权利要求7所述WEB端控制的消息中间件系统的数据传送方法,其特征在于,步骤S2进一步包括如下子步骤:

S2.1、利用发布模块中的任务调度器通过线程池管理器已经开启的相应线程,发布任务调度指令;

S2.2、利用数据访问器访问相应的发送客户端,并采用推送或拉取的方式从相应的发送客户端接收数据;

S2.3、利用数据访问器将该数据统一转换为消息队列中的JSON格式的消息,并将此消息发送至消息队列。

9. 根据权利要求7所述WEB端控制的消息中间件系统的数据传送方法,其特征在于,步骤S3进一步包括如下子步骤:

S3.1、利用订阅模块通过线程池管理器以阻塞的方式,从消息队列中获取其对应的接收客户端需要的消息;

S3.2、利用订阅模块调用其相应的数据分发器,由数据分发器将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据。

10. 根据权利要求9所述WEB端控制的消息中间件系统的数据传送方法,其特征在于,步骤S3还进一步包括如下子步骤:

S3.3、如果数据分发器向对应的接收客户端发送数据发送失败,客户端程序开发模块会再次将该数据转换为消息存入消息队列;

如果数据分发器发送的同个数据失败次数超过指定上限,则客户端程序开发模块会放弃该数据;

在客户端程序开发模块将发送失败的数据转换为消息存入消息队列的同时,将有另一个线程池调用该消息队列将曾经失败的消息再次发送,而如果消息队列中消息数量超过指定上限,客户端程序开发模块会将发送失败的数据转换为消息保存到WEB端控制的消息中间件系统的本地磁盘中。

WEB端控制的消息中间件系统及其数据传送方法

技术领域

[0001] 本发明涉及数据交换领域。更具体地,涉及一种WEB端控制的消息中间件系统及其数据传送方法。

背景技术

[0002] 随着企业信息化不断完善,解决系统与系统之间的数据共享与数据及时性、避免信息孤岛越来越为企业所重视。

[0003] 现有的中间件技术大部分以java语言作为编程平台,基于消息队列或消息传送的方式,主要功能是在应用程序之间提供可靠的消息传送,这些消息可以在不同的网络协议、不同的计算机系统和不同的应用软件之间传递。具体设计如下:

[0004] 1) 发送端开发:按照不同数据来源编写不同的客户端程序,通过把不同数据转换成统一的传输内容,通过消息队列或者消息发送给服务端。但是因为客户端往往涉及大量的逻辑处理,修改数据来源内容对应就要修改客户端内容。

[0005] 2) 服务端开发:服务端接收到数据以后,对应寻找订阅者也就是接收端,如果接收端状态是可以连接,将数据推送给订阅的接收端,否则将数据缓存在服务端。

[0006] 3) 接收端开发:接收端程序接收到服务端推送来的消息数据,按照指定规则进行解析,按照不同的业务逻辑进行处理。同样如果发送端数据内容或者协议修改了,接收端数据处理的环节对应需要修改。

[0007] 但现有的中间件技术存在如下问题:

[0008] 1) 需要客户端程序部署,运行维护复杂。

[0009] 2) 功能扩展能力不足,不能实现功能插拔。

[0010] 3) 大多没有断点续传功能,信息的完整性无法保障

[0011] 因此,需要提供一种WEB端控制的消息中间件系统及其数据传送方法。

发明内容

[0012] 本发明的目的在于提供一种WEB端控制的消息中间件系统及其数据传送方法,在通过消息中间件进行数据交换的过程中实现数据断点续传;不需要部署客户端程序,在WEB端进行发布订阅从而方便维护与管理;采用可插拔的功能单元,在最大程度上实现中间件的扩展性与可伸缩性;采用推拉结合的数据传送方式以确保数据的及时有效性;最终实现企业内或企业间各个异构系统、不同的数据存储格式之间进行数据相互交换,从而实现实时同步。

[0013] 为达到上述目的,本发明采用下述技术方案:

[0014] 一种WEB端控制的消息中间件系统,该系统通过发布/订阅架构方式,在WEB端实现发布/订阅控制,实现数据发布端和数据订阅端在空间、时间、数据处理过程三个维度实现解耦,该系统包括:

[0015] 至少一个与发送客户端一一对应的发布模块,根据对应的发送客户端的设置,采

用推送或拉取的方式从对应的发送客户端接收数据并将该数据转换为JSON格式的消息；

[0016] 至少一个与接收客户端一一对应的订阅模块，根据对应的接收客户端的设置，将对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据；

[0017] 消息队列模块，生成数据传送两端的一组发送客户端与接收客户端的消息队列，消息队列模块缺省集成ActiveMQ一个第三方消息队列工具，在整体设计上为保证系统的可伸缩性，也可以搭载其它消息队列产品，实现消息队列工具解耦；

[0018] 线程池管理器，对发布模块、订阅模块、消息队列模块产生的多个线程进行池化管理，对由这些线程实现调度和回收，提高系统性能；

[0019] 消息中间件容器，调度与管理各发布模块和订阅模块；

[0020] 客户端程序开发模块，实现发布模块和订阅模块以推送的方式进行数据传送，并将订阅模块未发送成功的数据转换为消息保存到内存的消息队列中。

[0021] 优选地，发布模块包括：

[0022] 初始化器，在发布模块初始构建时提供发布模块的全局配置信息作为其它功能单元的全局变量，实现系统启动时，配置参数一次性初始化，实现全局参数唯一性；

[0023] 任务调度器，触发执行采用推送或拉取的方式从对应的发送客户端接收数据任务的调度指令，实现触发执行采用推送或拉取的方式从对应的发送客户端接收数据任务的调度指令，用户亦可通过修改配置文件，实现任务调度器功能单元的切换；

[0024] 数据访问器，按照调度指令采用推送或拉取的方式从发布模块对应的发送客户端接收数据并将该数据转换为JSON格式的消息；

[0025] 注销器，在发布模块停止工作或关闭服务时释发相应外部资源，实现对应模块停止工作或停止服务时，一次性释放相应的外部资源。

[0026] 优选地，数据访问器为以下数据访问器的一种或任意组合：本地文件数据访问器、远程文件数据访问器、FTP数据访问器、webservice数据访问器、数据库最后更新时间数据访问器、快照表数据库访问器。

[0027] 优选地，订阅模块包括：

[0028] 初始化器，在订阅模块初始构建时提供订阅模块的全局配置信息作为其它功能单元的全局变量，实现系统启动时，配置参数一次性初始化，实现全局参数唯一性；

[0029] 数据分发器，将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据；

[0030] 注销器，在订阅模块停止工作或关闭服务时释发相应外部资源，实现对应模块停止工作或停止服务时，一次性释放相应的外部资源。

[0031] 优选地，数据分发器为以下分发器的一种或任意组合：控制台分发器、数据库表分发器、web服务分发器、webservice分发器。

[0032] 优选地，该系统还包括：集群管理器，将各发布模块和订阅模块作为独立节点从而实现集群管理，将订阅模块和发布模块实现单独部署，在集群中的每个节点都是一个单独的服务器，运行自己的服务，节点之间可以相互通信。

[0033] 一种上述WEB端控制的消息中间件系统的数据传送方法，该方法包括如下步骤：

[0034] S1、初始化WEB端控制的消息中间件系统；

[0035] S2、根据发送客户端的设置,利用WEB端控制的消息中间件系统采用推送或拉取的方式从发送客户端接收数据并将该数据转换为JSON格式的消息;

[0036] S3、根据接收客户端的设置,利用WEB端控制的消息中间件系统将对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据。

[0037] 优选地,步骤S2进一步包括如下子步骤:

[0038] S2.1、利用发布模块中的任务调度器通过线程池管理器已经开启相应线程,发布任务调度指令;

[0039] S2.2、利用数据访问器访问相应的发送客户端,并采用推送或拉取的方式从相应的发送客户端接收数据;

[0040] S2.3、利用数据访问器将该数据统一转换为消息队列中的JSON格式的消息,并将此消息发送至消息队列。

[0041] 优选地,步骤S3进一步包括如下子步骤:

[0042] S3.1、利用订阅模块通过线程池管理器以阻塞的方式,从消息队列中获取其对应的接收客户端需要的消息;

[0043] S3.2、利用订阅模块调用其相应的数据分发器,由数据分发器将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据。

[0044] 优选地,步骤S3还进一步包括如下子步骤:

[0045] S3.3、如果数据分发器向对应的接收客户端发送数据发送失败,客户端程序开发模块会再次将该数据转换为消息存入消息队列;

[0046] 如果数据分发器发送的同个数据失败次数超过指定上限,则客户端程序开发模块会放弃该数据;

[0047] 在客户端程序开发模块将发送失败的数据转换为消息存入消息队列的同时,将有另一个线程池调用该消息队列将曾经失败的消息再次发送,而如果消息队列中消息数量超过指定上限,客户端程序开发模块会将发送失败的数据转换为消息保存到WEB端控制的消息中间件系统的本地磁盘中。

[0048] 本发明的有益效果如下:

[0049] 本发明所述技术方案在通过消息中间件进行数据交换的过程中实现数据断点续传,从而保证数据完整性;不需要部署客户端程序,在WEB端进行发布订阅从而方便维护与管理;采用可插拔的功能单元,在最大程度上实现中间件的扩展性与可伸缩性;采用推拉结合的方式进行数据传送以确保数据的及时有效性;最终实现企业内或企业间各个异构系统、不同的数据存储格式之间进行数据相互交换,从而实现实时同步。

附图说明

[0050] 下面结合附图对本发明的具体实施方式作进一步详细的说明。

[0051] 图1示出WEB端控制的消息中间件系统的示意图。

[0052] 图2示出WEB端控制的消息中间件系统中推拉采用推拉结合的方式进行数据传送的示意图。

[0053] 图3示出WEB端控制的消息中间件系统的数据传送方法的流程图。

具体实施方式

[0054] 为了更清楚地说明本发明,下面结合优选实施例和附图对本发明做进一步的说明。附图中相似的部件以相同的附图标记进行表示。本领域技术人员应当理解,下面所具体描述的内容是说明性的而非限制性的,不应以此限制本发明的保护范围。

[0055] 如图1和图2所示,本实施例提供的WEB端控制的消息中间件系统,包括:

[0056] 至少一个与发送客户端一一对应的发布模块,根据对应的发送客户端的设置,采用推送或拉取的方式从对应的发送客户端接收数据并将该数据转换为JSON格式的消息;

[0057] 至少一个与接收客户端一一对应的订阅模块,根据对应的接收客户端的设置,将对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据;

[0058] 消息队列模块,生成数据传送两端的一组发送客户端与接收客户端的消息队列,其中,消息队列模块包括消息队列适配器和消息队列工具,通过消息队列适配器将各组消息队列与消息队列工具集成,消息队列用于存放待发送的消息,消息队列可以序列化并且性能高效,以满足断电、网络故障及高并发的要求;本系统缺省集成ActiveMQ一个第三方消息队列工具,在整体设计上为保证系统的可伸缩性,也可以搭载其它消息队列产品;

[0059] 线程池管理器,对发布模块、订阅模块、消息队列模块产生的多个线程进行池化管理,其中,在本系统中无论是发布模块、订阅模块、消息队列模块,每个实例都会对应一组线程组,而上述每个功能单元均对应各自独立的线程进行运算与处理;这就需要本系统支持多线程并且对为了提升性能,对多个线程进行池化管理;线程池管理器的目的是管理各功能的线程,对这些线程进行调度与回收等;

[0060] 消息中间件容器,调度与管理各发布模块和订阅模块,为web端渲染动态页面提供数据支撑;

[0061] 客户端程序开发模块,实现发布模块和订阅模块以推送的方式进行数据传送,其中,本系统不但提供主动拉取数据,也通过客户端程序开发模块实现推送数据的功能;所谓客户端程序开发模块是通过java语言对webservice的封装,以方便调用方程序开发。通过客户端程序开发模块可以方便的发送消息,并且客户端程序开发模块还提供了断点续传的功能,将订阅模块未发送成功的数据转换为消息保存到内存的消息队列中,并且可以通过配置将消息序列化到本地磁盘。

[0062] 其中,

[0063] 发布模块包括:

[0064] 初始化器,在发布模块初始构建时提供发布模块的全局配置信息作为其它功能单元的全局变量,其中,发布模块中的初始化器仅在发布模块初始构建时调用一次,且经由初始化器初始后的变量不可修改;

[0065] 任务调度器,触发执行采用推送或拉取的方式从对应的发送客户端接收数据任务的调度指令,其中,本系统缺省提供的拉取的方式为以Corn格式按指定时间周期轮询或以秒为单位周期轮询(轮询发布模块对应的发送客户端有无发送请求),推送的方式为通过外部指令(发送客户端指令)触发,共以上三种调度模式;用户也可以针对具体场景,修改系统

环境配置单元,通过可插拔的方式插入更多的任务调度器功能单元;

[0066] 数据访问器,当任务调度器初触发后,会调取相应的数据访问器,通过数据访问器根据调度指令采用推送或拉取的方式从发布模块对应的发送客户端接收数据并将该数据转换为JSON格式的消息,其中,数据访问器带有API接口,API接口用于发布模块对应的发送客户端通信,与推送的方式为数据访问器直接从发布模块对应的发送客户端接收数据并将该数据转换为JSON格式的消息;拉取的方式为数据访问器轮询发布模块对应的发送客户端有无发送请求,若有则从发布模块对应的发送客户端接收数据并将该数据转换为JSON格式的消息;不同的客户端数据类型(通讯协议)与访问形式决定不同的数据访问器,其中,本系统缺省提供的数据访问器为以下数据访问器的一种或任意组合:本地文件数据访问器、远程文件数据访问器、FTP数据访问器、webservice数据访问器、数据库最后更新时间数据访问器、快照表数据库访问器等;

[0067] 注销器,在发布模块停止工作或关闭服务时释发相应外部资源,其中,发布模块中的注销器仅会在发布模块停止工作时调用一次,外部资源包括数据库连接池、文件流等。

[0068] 订阅模块包括:

[0069] 初始化器,在订阅模块初始构建时提供订阅模块的全局配置信息作为其它功能单元的全局变量,其中,订阅模块中的初始化器仅在订阅模块初始构建时调用一次,且经由初始化器初始后的变量不可修改;

[0070] 数据分发器,将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据,其中,数据分发器带有API接口,API接口用于与订阅模块对应的接收客户端通信,推送的方式为数据分发器直接将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并向对应的接收客户端发送数据;拉取的方式为数据分发器轮询订阅模块对应的接收客户端有无接收请求,若有则将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并向对应的接收客户端发送数据,本系统缺省提供的数据分发器为以下分发器的一种或任意组合:控制台分发器、数据库表分发器、web服务分发器、webservice分发器等;

[0071] 注销器,在订阅模块停止工作或关闭服务时释发相应外部资源,其中,订阅模块中的注销器仅会在发布模块停止工作时调用一次,外部资源包括数据库连接池、文件流等。

[0072] 本实施例提供的WEB端控制的消息中间件系统还优选包括集群管理器,将各发布模块和订阅模块作为独立节点从而实现集群管理,其中,考虑到消息中间件高并发的特性,本系统支持集群功能;集群中的每个节点都是一个单独的服务器,运行自己的服务,节点之间可以相互通信,而在用户看来,它只是在与一个单独的系统交互。

[0073] 本实施例提供的WEB端控制的消息中间件系统的配置文件的具体内容如下:


```

    {
      "producer":{
        "initProcessors":[
          "org.so.configuration.impl.initprocessor.ParameterInitProcessor"
        ],
[0074] "dataAccessors":[
          "org.so.configuration.impl.accessor.ProcedureDataAccessor",
          "org.so.configuration.impl.accessor.file.LocalSingleFileAccessor",
          "org.so.configuration.impl.accessor.file.SMBSingleFileAccessor",
          "org.so.configuration.impl.accessor.db.SelectJDBCDataAccessor",
          "org.so.configuration.impl.accessor.db.SelectLastModfiyTimeDataAccessor",
          "org.so.configuration.impl.accessor.db.SnapTableDataAccessor",
          "org.so.configuration.impl.accessor.bean.TaskBeanAccessor",
          "org.so.configuration.impl.accessor.soap.SoapDataAccessor"
        ],
        "schedulers":[
          "org.so.configuration.impl.scheduler.CycleSecondScheduler",
          "org.so.configuration.impl.scheduler.CommandScheduler",
          "org.so.configuration.impl.scheduler.CronScheduler"
        ],
        "destroyProcessors":[
        ]
      ],
[0075] "subscriber":{
        "initExecutors":[
        ],
        "dataDispatchers":[
          "org.so.configuration.impl.dispatcher.ConsolDataDispatcher",
          "org.so.configuration.impl.dispatcher.db.InsertOrUpdateJDBCDispatcher",
          "org.so.configuration.impl.dispatcher.http.HttpDataDispatcher",
          "org.so.configuration.impl.dispatcher.bean.RoleDataDispatcher"
        ],
        "destroyExecutors":[
        ]
      }
    }
  }

```

[0076] 如图3所示,本实施例提供的WEB端控制的消息中间件系统的数据传送方法,包括如下步骤:

- [0077] S1、初始化WEB端控制的消息中间件系统；
- [0078] S2、根据发送客户端的设置，利用WEB端控制的消息中间件系统采用推送或拉取的方式从发送客户端接收数据并将该数据转换为JSON格式的消息；
- [0079] S3、根据接收客户端的设置，利用WEB端控制的消息中间件系统将对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据。
- [0080] 其中，
- [0081] 步骤S1进一步包括如下子步骤：
- [0082] S1.1、WEB端控制的消息中间件系统启动时获取该系统的环境配置单元，得到所有功能单元的类型定义；
- [0083] S1.2、获取发布模块、订阅模块及数据库源的配置信息，从而构建发布模块和订阅模块，在构建过程中调用相应发布模块和订阅模块中的初始化器，初始化发布模块和订阅模块的全局参数；
- [0084] S1.3、逐一匹配各发布模块和订阅模块，从而形成各组的消息队列，通过消息队列适配器与消息队列工具集成；
- [0085] S1.4、利用线程池管理器初始化订阅模块的线程池，至此，系统启动初始化完成。
- [0086] 步骤S2进一步包括如下子步骤：
- [0087] S2.1、利用发布模块中的任务调度器通过线程池管理器已经开启相应线程，发布任务调度指令；
- [0088] S2.2、当发布模块在任务调度指令触发后，就会激活数据访问器，由数据访问器访问相应的发送客户端，并采用推送或拉取的方式从相应的发送客户端接收数据；
- [0089] S2.3、数据访问器得到数据后将该数据统一转换为消息队列中的JSON格式的消息，并通过消息队列适配器将此消息发送至消息队列。
- [0090] 步骤S3进一步包括如下子步骤：
- [0091] S3.1、利用订阅模块通过线程池管理器以阻塞的方式，从消息队列中获取其对应的接收客户端需要的消息；
- [0092] S3.2、利用订阅模块调用其相应的数据分发器，由数据分发器将订阅模块对应的接收客户端所需消息转换为对应的接收客户端指定的格式并采用推送或拉取的方式向对应的接收客户端发送数据。
- [0093] 步骤S3还进一步包括如下子步骤：
- [0094] S3.3、如果数据分发器向对应的接收客户端发送数据因为网络或对应的接收客户端等故障发送失败，客户端程序开发模块会再次将该数据转换为消息存入消息队列；
- [0095] 如果数据分发器发送的同个数据失败次数超过指定上限（本实施例中为10次），则客户端程序开发模块会放弃该数据；
- [0096] 在客户端程序开发模块将发送失败的数据转换为消息存入消息队列的同时，将有另一个线程池调用该消息队列将曾经失败的消息再次发送（也就是说失败过的消息发送与其他的消息发送在线程端是隔离的，从而避免消息积压而导致消息发送的不及时），而如果消息队列中消息数量超过指定上限（本实施例中为5000个），客户端程序开发模块会将发送失败的数据转换为消息保存到WEB端控制的消息中间件系统的本地磁盘中，从而保证数据

的完整性。

[0097] 显然,本发明的上述实施例仅仅是为清楚地说明本发明所作的举例,而并非是对本发明的实施方式的限定,对于所属领域的普通技术人员来说,在上述说明的基础上还可以做出其它不同形式的变化或变动,这里无法对所有的实施方式予以穷举,凡是属于本发明的技术方案所引伸出的显而易见的变化或变动仍处于本发明的保护范围之列。

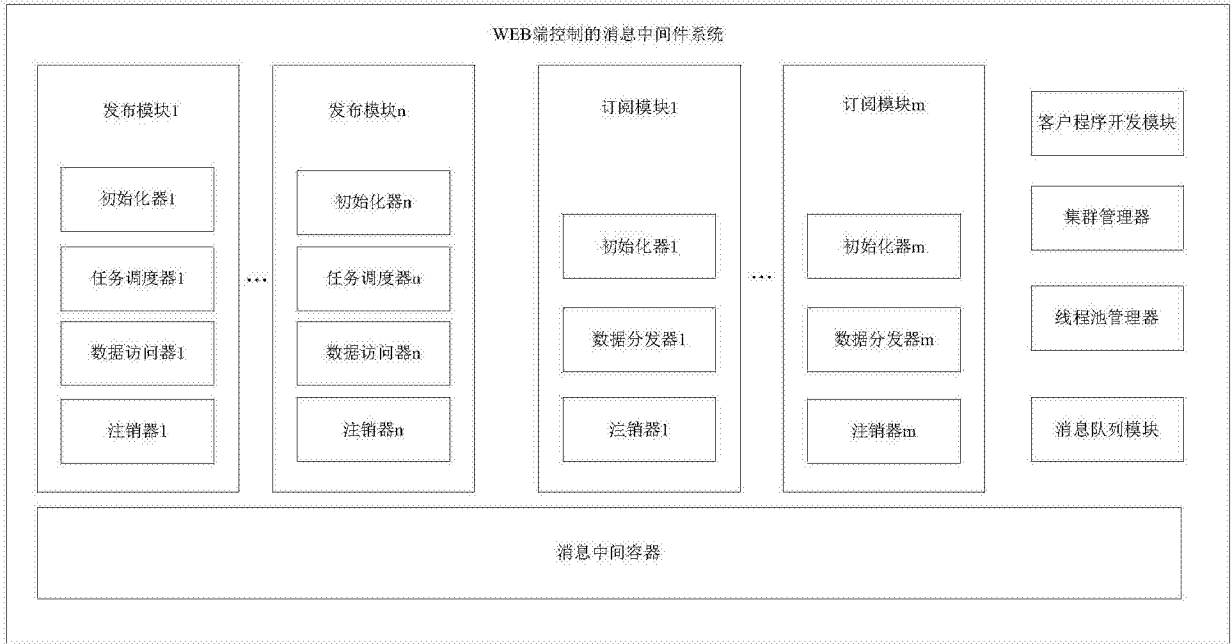


图1

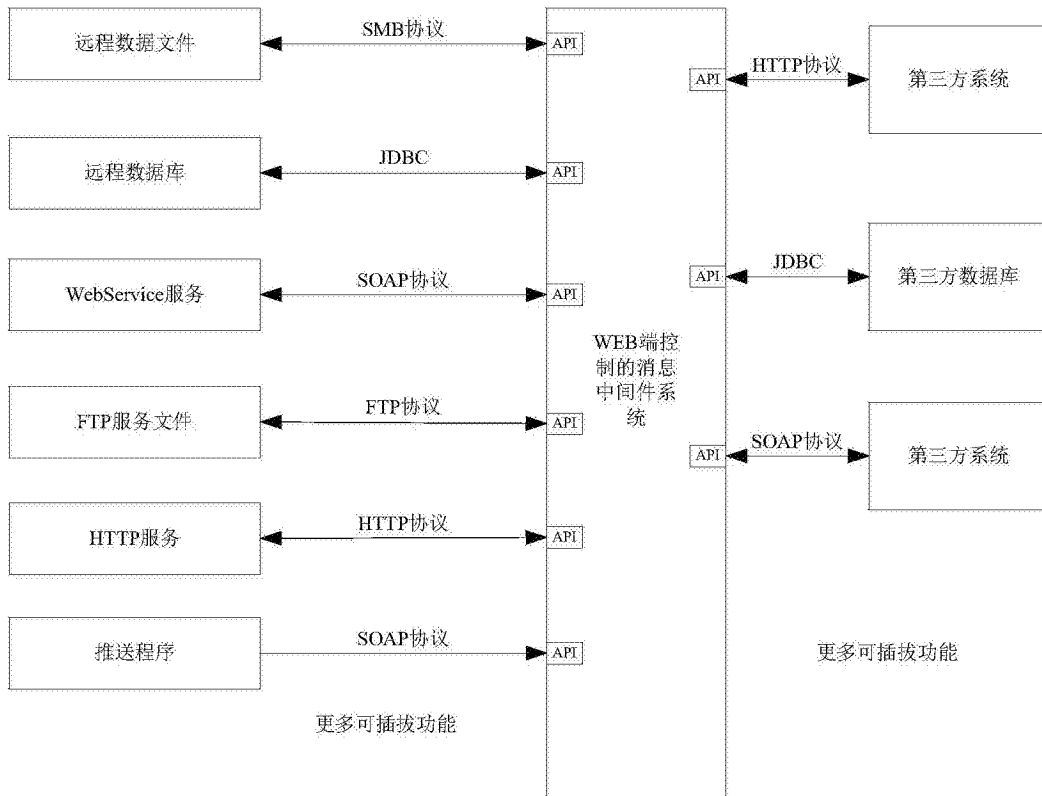


图2

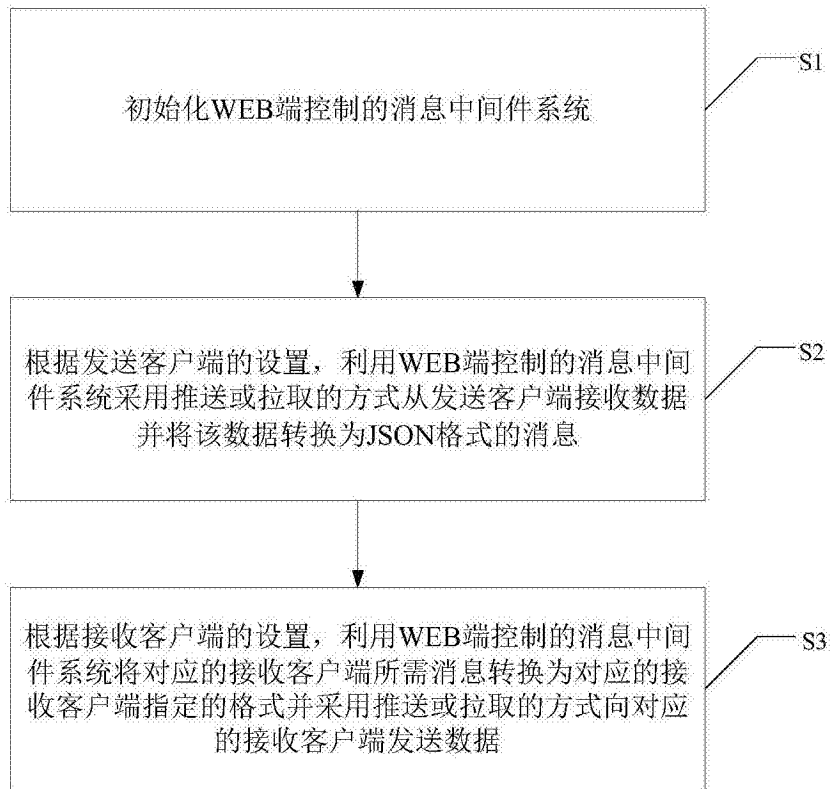


图3