



(12) 发明专利申请

(10) 申请公布号 CN 103312715 A

(43) 申请公布日 2013. 09. 18

(21) 申请号 201310237425. X

(22) 申请日 2013. 06. 14

(71) 申请人 西安交通大学

地址 710049 陕西省西安市咸宁西路 28 号

(72) 发明人 邹建华 朱正东 张祥雨 朱甫

刘景龙

(74) 专利代理机构 西安通大专利代理有限责任

公司 61200

代理人 汪人和

(51) Int. Cl.

H04L 29/06 (2006. 01)

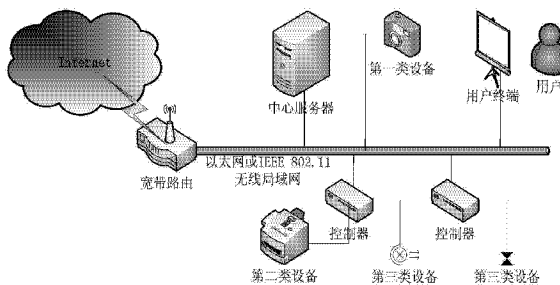
权利要求书2页 说明书13页 附图3页

(54) 发明名称

一种面向 Web 服务的家庭网络系统架构

(57) 摘要

本发明公开一种面向 Web 服务的家庭网络系统架构,该架构包括物理架构和逻辑架构两部分,在该架构中,使用 IEEE 802. 3 和 IEEE 802. 11 协议构建家庭局域网,使用家庭网关解决语义差异,在家庭网关设计时,本发明提出将家庭网关分为宽带路由和中心服务器两个部分,在中心服务器上部署家庭网关软件系统,在实现的中心服务器家庭网关软件系统中采用了虚拟设备技术,该技术同时为最终用户和第三方应用程序提供操作界面和 API 接口,并完成了上述功能的设计细节、实现算法和编码工作,本发明实现了家庭网络众多功能中最重要的设备互联功能,解决了设备的语义差异性,并且该系统架构具有充分的灵活性和可扩展性。



1. 一种面向 Web 服务的家庭网络系统架构,其特征在于:包括物理架构和逻辑架构,所述物理架构包括广域网以及局域网,局域网包括家庭网关、家庭设备以及控制器,所述家庭网关包括宽带路由和中心服务器,中心服务器与家庭网关的宽带路由相连,家庭设备直接或者通过控制器与家庭网关的宽带路由相连,所述逻辑架构分为层次模型、功能模块划分和协议设计。

2. 如权利要求 1 所述一种面向 Web 服务的家庭网络系统架构,其特征在于:所述局域网是家庭网络中通过使用 IEEE 802.3 协议的有线局域网和使用 IEEE 802.11 协议的无线局域网共同组成的局域网络部分,该局域网络部分支持统一的 HTTP/Web Service 协议;

所述家庭网关是为用户提供宽带接入服务,同时也是家庭网络本身的接入、管理和控制协调中心;

所述家庭设备包括第一类设备、第二类设备以及第三类设备,第一类设备使用的协议与局域网相同,第一类设备直接接入局域网中,第二类设备不具有标准的网络接口,或是虽有接口,但不符合以太网或 IEEE802.11 标准,同时,第二类设备控制逻辑复杂,第三类设备控制逻辑简单,仅支持开关操作,而且,第三类设备不具有网络接口,第二、三类设备通过控制器接入局域网中;

所述控制器将对应设备专有的通信协议与局域网使用的协议进行相互转换,实现双向互通;同时,控制器也负责解析对应设备的控制逻辑。

3. 如权利要求 1 所述一种面向 Web 服务的家庭网络系统架构,其特征在于:所述中心服务器负责运行家庭网关软件系统,完成家庭网络的管理。

4. 如权利要求 3 所述一种面向 Web 服务的家庭网络系统架构,其特征在于:所述中心服务器包括家庭 PC 和家庭网关软件系统;所述家庭 PC 是用于运行家庭网关软件系统的个人电脑;

所述家庭网关软件系统实现家庭设备发现、设别和离线,完成家庭设备互联。

5. 如权利要求 4 所述一种面向 Web 服务的家庭网络系统架构,其特征在于:所述家庭网关软件系统的主要组件包括家庭网络软件平台和设备互联协议的实现软件;

所述家庭网络软件平台是用户用来设置、控制和访问家庭设备的 Web 服务软件;

所述设备互联协议的实现软件完成设备接入、设备识别和设备离开的判定,是设备与设备之间,设备与中心服务器之间的约定。

6. 如权利要求 1 所述一种面向 Web 服务的家庭网络系统架构,其特征在于:所述控制器包括控制装置及其所使用的设备驱动程序;

所述设备驱动程序包括家庭设备的驱动程序和控制器的驱动程序。

7. 如权利要求 1 所述一种面向 Web 服务的家庭网络系统架构,其特征在于:所述层次模型是最高层的用户到最底层的硬件设备,反映了系统的纵向架构及逻辑结构的分层组织;

所述功能模块划分反映了系统的横向架构,即系统包含的功能模块,各模块分别起的作用,以及分别部署的物理设备;

所述协议设计主要确定各物理或逻辑单元之间的通信协议规范。

8. 如权利要求 7 所述一种面向 Web 服务的家庭网络系统架构,其特征在于,所述层次模型包括应用层、虚拟设备层、服务层、网络层和物理设备层;

所述应用层包括家庭网络用户操作界面和服务提供商开发的第三方应用程序；所述虚拟设备层将家庭设备抽象成虚拟设备，并利用服务层驱动程序库提供的信息解析从服务层获取的设备通信数据的含义；另外，虚拟设备层向应用层提供一组 API，用户通过 API 操纵这些虚拟设备，虚拟设备层完成语义转换，并对应到具体物理设备的操纵；

所述服务层提供了操作家庭网络的较为底层的功能，服务层中的监控模块时刻监控家庭网络中各设备的连接状况，实现设备发现功能；设备逻辑模块通过设备驱动数据库提供的信息完成设备类型识别的功能，同时确定与该设备进行通信所需的语义规范，这些功能均向虚拟设备层提供接口，完成与设备之间收发数据的功能；

所述网络层是指家庭网络系统的局部网络层协议，在本系统架构中，局域网采用 HTTP 以及 Web Service/SOAP 协议；

所述物理设备层包含第一类设备以及第二、三类设备对应的控制器，其中控制器用来进行协议转换，解决设备的语法差异性，物理设备层的设备与家庭网络的局域网直接相连，位于架构的最底层。

9. 如权利要求 7 所述一种面向 Web 服务的家庭网络系统架构，其特征在于，所述功能模块划分包括中心服务器的网络监控模块、设备支持模块和 Web 页面与 API 模块，以及控制器的通信模块和协议转换脚本解释模块；

所述网络监控模块主要用于设备发现，监控局域网上有没有新设备接入，有没有设备刚刚离线，其位于层次模型的服务层；

所述设备支持模块用于识别设备类型，用正确的逻辑与设备进行通信，完成设备的虚拟化，其位于层次模型的服务层和虚拟设备层；

所述 Web 页面与 API 模块提供了家庭网络的用户界面，同时也向第三方应用程序提供了 API 接口，其位于层次模型的应用层和虚拟设备层；

所述通信模块提供一组调用接口，可以分别从连接家庭局域网的一个端口和连接家庭设备的一个端口上收发数据，其位于物理设备层；

所述协议转换脚本解释模块完成设备与局域网之间的语法差异问题，该模块先编写脚本解释器，存储协议转换脚本，利用解释器间接解释执行该脚本完成协议转换。

10. 如权利要求 7 所述一种面向 Web 服务的家庭网络系统架构，其特征在于，所述协议设计主要包括设备专有协议、HTTP 协议和 SOAP 协议；

所述设备专有协议是指控制器与第二、三类设备之间的通信所使用的协议，与具体设备的型号相关；

所述 HTTP 协议是家庭局域网统一使用的协议；

所述 SOAP 协议是指中心服务器与控制器进行通信时所使用的 Web 服务技术，即采用了基于 HTTP 的简单对象访问协议。

一种面向 Web 服务的家庭网络系统架构

技术领域

[0001] 本发明属于计算机应用领域,具体涉及一种 Internet 网络环境下面向 Web 服务的家庭网络系统架构。

背景技术

[0002] 数字家庭网络这一概念诞生于 1994 年,之后随着互联网的发展,家电产品数字化的革新,家庭网络逐渐成为一个热门的研究课题。英特尔、IBM、微软等大公司早已开始进行相关领域的研究,也进行了如“维纳斯”计划等实践尝试。DLNA、UPnP 论坛等企业联盟和组织也在为家庭网络的标准化而努力。目前,针对家庭网络架构的研究是智能家居的主要研究方向之一。

[0003] 然而,直到今天家庭网络技术仍然没有得到广泛应用。这其中的原因很多,除了成本和市场的因素以外,一个主要原因是当前的家庭网络技术存在很多弊端,大大限制了它的应用和普及,主要包括如下:

[0004] (1)标准混杂,不统一。家庭网络各个设备之间的通信需要遵守共同的协议才能正常进行。目前参与研究和制定标准的单位众多,然而遗憾的是,没有一个获得业界的公认。这样就造成了各个家庭网络厂商各自使用各自的私有标准,互相不兼容,而家电厂商大多处在观望中,不愿冒险大量生产符合某个标准的产品。

[0005] (2)架构死板,销售方式不灵活。由于没有统一的标准,家庭网络厂商想支持多种家庭设备是十分困难的。于是,大多数家庭网络厂商主要采用捆绑的方式进行销售,即让用户一次性购买家庭网络控制器和厂商指定的其他家庭设备,其架构本身大都也仅支持这些设备,一般不允许用户自行购买设备接入家庭网络中。

[0006] (3)安装部署麻烦。目前的家庭网络产品一般都不具备“即插即用”功能,安装时需要专业人员上门进行大量的配置工作。当用户今后要添加新设备时,即便该设备可以支持当前使用的家庭网络系统,用户也很难自己进行配置工作,必须求助于厂商的技术人员,无法实现“插上即可使用”的功能。

[0007] (4)远程控制功能对网络环境要求高。目前部分家庭网络产品提供远程控制功能,用户不在家时也可以通过互联网或 3G 网络控制家庭设备。然而此类功能往往要求用户的家庭必须拥有公网 IP 地址(可能是固定的,也可能是动态的),这对于某些采用共享方式接入(如小区宽带)的家庭来说是不具备的。

[0008] (5)缺少多媒体功能。目前家庭网络产品主要还是以对开关设备或模拟设备的控制为主,较少涉及视频、音频等多媒体设备的操作和控制。

[0009] (6)扩展性差。当前多数家庭网络产品本质上说都是一个写死的系统,一旦生产出来,几乎没有扩展功能的余地。第三方厂商很难利用已有系统作为平台开发具有新功能的产品。

发明内容

[0010] 本发明的目的在于提供一种 Internet 环境下面向 Web 服务的家庭网络系统架构, 本发明实现了家庭网络众多功能中最重要的设备互联功能, 解决了设备的语义差异性, 并且该系统架构具有充分的灵活性和可扩展性。

[0011] 为了达到上述目的, 本发明采用了以下技术方案:

[0012] 该架构包括物理架构和逻辑架构两个部分。通过该架构来规范家庭网络系统结构的设计。该架构使用 IEEE 802.3 和 IEEE 802.11 协议构建家庭局域网络, 使用家庭网关解决语义差异。在家庭网关设计时, 本发明提出在家庭网关的中心服务器上部署家庭网关软件系统, 采用了虚拟设备技术精心设计驱动文件支持不同的设备, 并为最终用户和第三方应用程序提供操作界面和 API 接口。本发明完成了上述功能的设计细节、实现算法以及编码工作。具体如下所述:

[0013] 该架构包括物理架构和逻辑架构, 所述物理架构是根据所要支持的设备特性设计完成的硬件之间的连接架构, 所述物理架构包括广域网以及局域网, 局域网包括家庭网关、家庭设备以及控制器和用户终端, 所述家庭网关包括宽带路由和中心服务器, 中心服务器与家庭网关的宽带路由相连, 家庭设备直接或者通过控制器与家庭网关的宽带路由相连, 所述逻辑架构分为层次模型、功能模块划分和协议设计, 反映系统各物理设备的横向架构、纵向架构及逻辑结构的分层组织, 以及各物理或逻辑单元之间的通信协议规范。

[0014] 所述广域网是家庭网络出口端所连接的公共的、大范围的网络。最常见的有 Internet, 以及 3G 手机网络等;

[0015] 所述局域网是家庭网络中通过使用 IEEE 802.3 协议的有线局域网和使用 IEEE 802.11 协议的无线局域网共同组成的局域网络部分, 该局域网络部分支持统一的 HTTP/Web Service 协议;

[0016] 所述家庭网关是为用户提供宽带接入服务, 同时也是家庭网络本身的接入、管理和控制协调中心, 以及家庭网络的逻辑和物理的中心; 一方面, 用户对外部网络进行访问, 其数据流都需要通过家庭网关; 另一方面, 家庭网络平台的软件部分以及服务提供商提供的应用程序也都部署在家庭网关上, 实现对各设备的管理。可以说, 设计家庭网关是构建家庭网络的最重要的一个环节。

[0017] 所述家庭设备是接入家庭网络的硬件设备及家用电器, 如摄像头、空调等。所述家庭设备包括第一类设备、第二类设备以及第三类设备, 第一类设备使用的协议与局域网相同, 第一类设备直接接入局域网中, 第二类设备不具有标准的网络接口, 或是虽有接口, 但不符合以太网或 IEEE 802.11 标准, 同时, 第二类设备控制逻辑复杂, 第三类设备控制逻辑简单, 仅支持开关操作, 而且, 不具有网络接口, 第二、三类设备通过控制器接入局域网中; 其与控制器的关系可以是一对一的, 也可以是多对一的。

[0018] 所述控制器是一种硬件设备, 用于将对应设备专有的通信协议与局域网使用的协议进行相互转换, 实现双向互通; 同时, 控制器也负责解析对应设备的控制逻辑; 控制器的设计也是构建家庭网络的关键环节之一。

[0019] 所述用户终端是用来访问家庭网络管理系统, 对家庭网络进行配置和管理的 PC 机, 也可以是手机等移动设备。

[0020] 所述宽带路由是通常所述的交换机等设备, 负责完成宽带接入, 一般由电信部门和 ISP 提供完成设置, 常常不具备二次开发功能, 但具有能够自动或手动分配 IP 地址的功

能；

[0021] 所述中心服务器负责运行家庭网关软件系统，负责完成家庭网络的管理，是构建家庭网络的关键环节。

[0022] 所述中心服务器包括家庭 PC 和家庭网关软件系统；所述家庭 PC 是用于运行家庭网关软件系统的具有系统软硬件的个人电脑；

[0023] 所述家庭网关软件系统最主要的功能就是实现家庭设备发现、设别和离线，完成家庭设备互联，用于解析家庭网络软件平台，实现对家庭设备的访问和控制，是实现家庭网关的关键。

[0024] 所述家庭网关软件系统的主要组件包括家庭网络软件平台和设备互联协议的实现软件；

[0025] 所述家庭网络软件平台是用户用来设置、控制和访问家庭设备的 Web 服务软件；

[0026] 所述设备互联协议的实现软件完成设备接入、设备识别和设备离开的判定，是设备与设备之间，设备与中心服务器之间的约定。

[0027] 所述设备互联协议包括设备接入协议、设备离线协议和附加协议。

[0028] 所述设备接入协议是指家庭设备接入家庭网络后，与中心服务器、网关之间的互动约定。网关动态分配 IP 地址；设备的上线通知、在线通知以及给中心服务器发送设备信息，中心服务器发现设备并获得设备 MAC 地址、向设备发送上线确认、在线确认和设备识别确认信息；

[0029] 具体过程如下：

[0030] 1) 设备接入时发出 DHCP 请求，网关动态分配 IP 地址；设备也可以自行静态绑定 IP 地址。

[0031] 2) 设备的 IP 地址确定后，向网关发送上线通知进行设备发现，通知内容包括设备的 IP 地址和 MAC 地址。

[0032] 3) 中心服务器收到上线通知后，注册设备信息，同时检索中心服务器上的数据库，检查该设备的 MAC 地址曾经是否已完成设备识别，然后向设备发送上线确认，确认内容包括是否要求进行设备识别的信息。

[0033] 4) 如果设备在 10 秒内未收到上线确认信息，将返回步骤 2 重新进行设备发现。如果设备收到上线确认，则需根据上线确认的内容决定是否进行设备识别。

[0034] 5) 如果需要进行设备识别或设备信息的更新，设备向中心服务器发送设备信息，(其中，应包含设备类型、名称、状态等)，则转入步骤 6 执行。如果需要进行设备识别，但设备未发送设备信息，那么在中心服务器收到设备发送的无设备信息的数据信息后，该设备将被识别为未知设备。如果不需要进行设备识别，可以直接转入步骤 8。

[0035] 6) 中心服务器在收到设备信息后，注册该设备的相关信息，完成设备识别。然后，向设备发送信息确认。

[0036] 7) 如果设备在 10 秒内未收到信息确认，那么将返回步骤 5 执行，直到收到确认为止。如果连续 3 次发送设备信息均未收到确认，则回到步骤 2 执行。

[0037] 8) 设备每隔 10 秒向服务器发送在线通知。从步骤 5 之后，如果中心服务器在任意连续的 40 秒内未收到该设备的在线通知，则判定该设备离线，在中心服务器上注销该设备，并转入步骤 2 等待设备的上线通知。否则，向设备发送在线确认。

[0038] 9)如果设备在发送在线通知 10 秒后仍未收到在线确认,则继续重发在线通知。如果在 3 次发送后仍未收到,则判断为本设备已掉线,转入步骤 2 重新进行设备发现。

[0039] 对于本系统而言,最大的难题在于如何完成设备发现和设备识别。因此本节对其进行详细介绍。

[0040] 1) 设备发现

[0041] 设备发现有三大任务:

[0042] ①能够监控到设备是否断开连接。

[0043] ②能够监控到设备何时接入家庭网络。

[0044] ③记录已接入设备的 IP 地址和 MAC 地址。

[0045] 其中,第一个任务相对比较简单,只需要每隔一段时间 Ping 在线设备列表中的所有设备,如果有些设备没有响应,就意味着该设备离线,然后将此设备从在线设备列表中去除即可。

[0046] 第二个问题比较麻烦。因为在以太网中,各网络设备是相互对等的,没有哪台机器地位比其他设备高。因此,当一个设备连入时,它没有义务向其他设备发出通知。符合某些家庭网络标准的设备会主动进行设备注册,即通知家庭网关它刚刚连入,但是我们的系统支持的设备大多数不具备这种功能。

[0047] 为了解决这个问题,可能的方法有如下几种:

[0048] Ping 广播地址。在 IP 网中,每个网段的最后一个地址为广播地址,意为本网段的所有 IP。如果我们 Ping 这个 IP,那么理论上就相当于 Ping 本网段的所有机器,因此,通过分析返回包的 IP 地址,就可以得到当前在线设备的 IP 列表。然而,经过我们的测试,大多数网络设备(如摄像头、PC 等)都不对广播 Ping 响应(可能是因为安全问题)。因此本方法行不通。

[0049] 从宽带路由处获得设备是否在线的信息。该方法主要适用于无线网。按 IEEE802.11 的要求,无线设备接入无线基站(如无线路由)时,需要在基站处完成注册,表明其已经连入该无线网。该方法只能获得设备的 MAC 地址,不能获得 IP 地址。并且,需要考虑不同品牌、型号的宽带路由信息获取方法的差异性。

[0050] 使用 UDP 协议发送特别构造的数据包,目的地址是全局广播地址(255.255.255.255)。之前我们通过实验发现,很多网络摄像头 SDK 中都带有 IP 扫描工具,可以迅速发现局域网中有没有相应型号的摄像头。为了了解其工作原理,我们对此类工具进行抓包分析,该包为 UDP 包,源地址为本机 IP,目的地址为全局广播地址,包的内容不定,随摄像头型号不同而不同。实验发现当指定型号的摄像头收到特定的 UDP 包后,会往广播地址发一个反馈包,其中包含了它的 IP 地址。这些工具正是通过该反馈包获得指定设备的 IP 地址的。然而目前,我们只发现网络摄像头使用了这种方法,对于其他设备还没有见到。而且不同型号的摄像头,其包的格式有很大差别。总之,该方法应用面十分有限。

[0051] 监控网络中 ARP 包的收发情况。局域网中的两个设备要进行通信,首先需要知道对方的 MAC 地址。然而一般发送方只知道对方的 IP 地址,这时就要使用 ARP 协议。发送方首先构造一个 ARP 包,询问谁的 IP 地址是某个指定的地址。由于 ARP 包是广播形式发送的,于是当某个设备发现它的 IP 恰好就是 ARP 包询问的 IP 时,它就会发一个反馈包,告诉发送方自己的 MAC 地址。由此可见,ARP 包在局域网中是大量存在的,因为只要有通信,就

可能产生 ARP 包。于是,可以通过侦听互联网中的 ARP 包的收发情况,记下发送方和接受方的 IP 地址,经过一段时间以后,便可以获得局域网中所有接入设备的 IP 列表。然而,该方法也有不少问题。首先,它的耗时较长,需要很久才能得到完整的 IP 列表;并且我们实验发现,有些网络设备接入局域网后根本不进行主动通信,只是等着别人来访问它,这样便不会有和它相关的 ARP 包存在。因此,该方法无论是可靠性还是性能都是比较差的。

[0052] 循环 Ping 当前网段中的所有 IP。考虑到家庭网络中的设备一般不会超过 200 个,因此用 C 类 IP 就可以了,地址范围从 1 到 254 共 254 个不同地址(最后一个 255 是用来做广播地址的)。由于有效地址并不是很多,因此可以循环 Ping 每个 IP,通过回应情况,确定哪些 IP 被设备占用。

[0053] 在对上述 5 种方法进行了分析比较之后,确定了使用最后一种方法。之所以这样选择主要是由于这种方法的适应性较好,不依赖具体设备,而且较为方便快捷。不过在循环 Ping 时要注意使用多线程技术,因为如果 Ping 的那个 IP 没有设备,程序必须等待较长时间(至少 50ms 左右)才能确定该 IP 确实没有设备对应,期间线程被阻塞住,无法继续推进,这样的话,200 多个 IP Ping 下来需要很长时间。而通过多线程技术,则可以并行地 Ping 各个地址,速度会快很多。但是线程数太多的话,也会消耗系统资源,影响速度。我们经过试验,认为开 20 到 40 个线程会取得比较好的效果。

[0054] 另外,有些设备的 IP 地址是用户手工设置的,用户知道这些设备的 IP 地址。此时,我们可以添加 IP 列表功能,当用户知道所有设备的 IP 地址时,可以编辑此 IP 列表。启用该功能后, Ping 的范围被局限在该列表中,进一步加快了设备发现的速度。

[0055] 现在来讨论第 3 个问题,即如何获得设备的 IP 地址和 MAC 地址。IP 地址的列表在完成上述循环 Ping 后就已经得到了,因此关键在于如何从 IP 地址得到设备的 MAC 地址。其实,该问题的解决方法已经在第 2 个问题的解决方法 4 中提到了,即使用 ARP 协议。通过构造 ARP 包,询问对应 IP 的 MAC 地址,此时设备就会回传 ARP 应答包,其中包含设备的 MAC 地址。这样就解决了问题。

[0056] 2) 设备识别

[0057] 设备识别的任务是确定设备的类别,以及匹配与之通信所需要的驱动文件。

[0058] 每个设备都有一个独一无二的特征,即向设备发送一个指定的请求时,设备将返回一个独一无二的结果。在制作驱动文件时,会提取设备的此种特征,并把它记录在驱动文件中。该特征可以用来识别不同的设备。首先,程序遍历驱动文件,比如说遍历到 A 的驱动文件时,向待检测的设备发送 A 的特征请求,如果此设备的返回刚好和 A 驱动文件中的规定符合的话,就说明该设备的类型就是 A;否则,继续检验其他的驱动是否符合。

[0059] 由于在总体架构中,已经规定家庭局域网中全部使用 HTTP 或 Web Service 方式进行通信,因此,这些特征请求事实上就是 HTTP 请求字符串,很容易在驱动文件中加以表示。

[0060] 检测出的设备类型被记录在了设备详细信息表中,而不是在线设备列表中。之所以这样设计,是因为这样在某一设备断开后又重新接入家庭网络时,可以立刻知道其设备类型,而无需重新检测(由于 MAC 地址的唯一性,直接在设备详细信息表中查找 MAC 地址相同的项即可)。毕竟,设备识别过程还是较慢的。上述流程同样会在服务器启动后不断执行,直到服务器关机。

[0061] 所述设备离线协议是指设备主动从家庭网关注销的过程。因网络等原因导致在线

通知 / 在线确认无法及时发送的注销情况不在此列 ;

[0062] 具体过程如下 :

[0063] 1) 设备向中心服务器发送离线请求。

[0064] 2) 中心服务器收到离线请求后,在中心服务器上注销该设备,同时向设备发送离线确认。

[0065] 3) 如果设备在 10 秒后仍然没有收到离线确认,则转入 1 继续发送 ;如果连续 3 次发送均未收到确认,则强制离线。否则,直接执行离线过程。

[0066] 所述附加协议是指上述协议之外需要补充说明的协议内容。

[0067] 具体内容包括 :

[0068] 1) 所有请求和确认均通过 HTTP 协议发送,设备作为 Client 端,中心服务器作为 Server 端。上线通知 / 上线确认,设备信息 / 信息确认,在线通知 / 在线确认,离线请求 / 离线确认分别构成 HTTP 的 Request/Response 对。

[0069] 2) 在设备获得 IP 地址之后的任何时候,设备均可向中心服务器发送上线通知,表示设备放弃当前的连接状态,则需重新进行设备发现。

[0070] 3) 任何不按上述设备互连(时序)协议进行数据交互的行为均将导致连接被重置,或者导致 HTTP 状态码呈现为错误状态码。但这些行为都将不会影响中心服务器的工作状态,也就是说,如果设备在此时向中心服务器发送一个合法的请求,时序协议将继续。

[0071] 所述控制器包括控制装置及其所使用的设备驱动程序 ;

[0072] 所述控制装置是用来控制第二、三类设备的控制设备,根据不同的家庭设备提供不同的驱动程序 ;

[0073] 所述设备驱动程序包括家庭设备的驱动程序和控制器的驱动程序。

[0074] 所述层次模型是最高层的用户到最底层的硬件设备,反映了系统的纵向架构及逻辑结构的分层组织 ;

[0075] 所述功能模块划分反映了系统的横向架构,即系统包含的功能模块,各模块分别起的作用,以及分别部署的物理设备 ;

[0076] 所述协议设计主要确定各物理或逻辑单元之间的通信协议规范。

[0077] 所述层次模型包括应用层、虚拟设备层、服务层、网络层和物理设备层 ;

[0078] 所述应用层包括家庭网络用户操作界面和服务提供商开发的第三方应用程序 ;最终用户使用它们完成对家庭网络的访问和管理。

[0079] 所述虚拟设备层将家庭设备抽象成虚拟设备,并利用服务层驱动程序库提供的信息解析从服务层获取的设备通信数据的含义 ;另外,虚拟设备层向应用层提供一组 API,用户通过 API 操纵这些虚拟设备,就好像直接操纵真实设备一样 ;虚拟设备层完成语义转换,并对应到具体物理设备的操纵 ;

[0080] 所述服务层提供了操作家庭网络的较为底层的功能,服务层中的监控模块时刻监控家庭网络中各设备的连接状况,实现设备发现功能 ;设备逻辑模块通过设备驱动数据库提供的信息完成设备类型识别的功能,同时确定与该设备进行通信所需的语义规范,这些功能均向虚拟设备层提供接口,完成与设备之间收发数据的功能 ;

[0081] 所述网络层是指家庭网络系统的局部网络层协议,在本系统架构中,局域网采用 HTTP 以及 Web Service/SOAP 协议 ;

[0082] 所述物理设备层包含第一类设备以及第二、三类设备的控制器,其中控制器用来进行协议转换,解决设备的语法差异性,物理设备层的设备与家庭网络的局域网直接相连,位于架构的最底层。

[0083] 所述功能模块划分包括中心服务器的网络监控模块、设备支持模块和 Web 页面与 API 模块,以及控制器的通信模块和协议转换脚本解释模块。

[0084] 所述网络监控模块主要用于设备发现,监控局域网上有没有新设备接入,有没有设备刚刚离线,其位于层次模型的服务层;

[0085] 所述设备支持模块用于识别设备类型,用正确的逻辑与设备进行通信,完成设备的虚拟化,其位于层次模型的服务层和虚拟设备层;

[0086] 所述 Web 页面与 API 模块提供了家庭网络的用户界面,同时也向第三方应用程序提供了 API 接口,其位于层次模型的应用层和虚拟设备层;

[0087] 所述通信模块提供一组调用接口,可以分别从连接家庭局域网的一个端口和连接家庭物理设备的一个端口上收发数据,其位于物理设备层;

[0088] 所述协议转换脚本解释模块完成设备与局域网之间的语法差异问题,该模块先编写脚本解释器,存储协议转换脚本,利用解释器间接解释执行该脚本完成协议转换,这是因为设备所采用的是私有协议,几乎每个设备都不相同,采用这种间接方式,只需要更新其存储的脚本,无需改变控制器的程序。

[0089] 所述协议设计主要包括设备专有协议、HTTP 协议和 SOAP 协议;

[0090] 所述设备专有协议是指控制器与第二、第三类设备之间的通信所使用的协议,与具体设备的型号相关;

[0091] 所述 HTTP 协议是家庭局域网统一使用的协议,使用该协议主要因为它是网络应用层中使用最广泛的协议之一,方便实现 Web 形式的访问和控制;并且,HTTP 完全基于文本,实现简单,大多数第一类设备,如网络摄像头等,都内置有 Web 服务器;在控制器中,通信模块也内置了 Web 服务器组件,为使用 HTTP 协议提供硬件支持;

[0092] 所述 SOAP 协议是指中心服务器与控制器进行通信时所使用的 Web 服务技术,即采用了基于 HTTP 的简单对象访问协议(SOAP),使用 Web 服务和 SOAP 协议,是因为针对控制器的调用主要表现为远程过程调用(RPC)的形式,另外,几乎所有的 HTTP 服务器软件都支持 Web 服务;

[0093] 通过以上协议设计,在中心服务器上可以部署面向 Web 服务的服务器软件,家庭网络平台的 API 同样可以 Web 服务的形式提供。以至于第三方应用程序便可以用标准的 SOAP 协议以远程调用的形式使用这些 API,而最终用户则可以通过浏览器以 HTTP 协议的方式访问家庭网络。

[0094] 与现有技术相比,本发明具有以下有益效果:

[0095] (1) 该架构将各种不同的电子设备连接起来,解决了设备之间信息格式和语义高度不统一的问题。

[0096] 本发明通过合理的架构设计解决了设备互联问题,解决了设备和家庭网络之间的语法和语义差异性。在物理架构设计阶段主要解决了语法差异,而在逻辑架构设计阶段主要解决了语义差异。这是实现各设备之间消息共享和协作的前提,否则,各设备之间缺乏协作,各自使用各自专有的方式进行通信和工作,形成一个个“信息孤岛”,这样家庭网络的作

用将会大打折扣。可以说,实现设备互联是家庭网络最重要的一个环节。

[0097] 设计的控制器通过依靠“协议转换脚本解释模块”以间接方式完成协议的转换工作,很好地解决了设备与局域网之间的语法差异,设计的虚拟设备驱动程序很好地解决了设备与设备之间的语义差异。

[0098] (2) 设备发现和设备识别等功能与用户设备访问、管理等功能相对独立,提供了一个友好的用户界面,能够明显提高用户访问系统的响应速度对各设备进行有效的控制。

[0099] 家庭网关软件系统实现了设备发现、设备管理、设备访问与控制等多种功能,同时也是用户访问家庭网络的入口,是家庭网络系统中最关键的一环。由系统时钟驱动的一个监护进程不断与设备进行通信,更新设备的相关信息,存储在数据库中;而用户则可以随时从数据库中查看信息,并且只在必要时,直接与设备进行通信。这种方式可以明显提高用户访问系统时系统的响应速度,也符合该系统作为一个软硬件平台的定位。

[0100] (3) 家庭网络平台起到了类似于家庭“操作系统”的作用,连接了最终用户和服务提供商,为提供商提供了开发软件的接口。

[0101] 用户希望家庭网络提供各种除上述功能以外更多的功能,如包括电话服务功能、音频与视频功能和遥控与控制功能等。而为了实现这些具体的功能,一方面需要向用户提供添加/删除应用软件等功能;另一方面需要向服务提供商提供一组 API 接口,提供商可以调用这组接口实现对家庭网络各资源的调度管理,进而开发出相应的应用软件提供给最终用户。在此过程中,设备的差异性统一交由家庭网络平台进行处理,而对于提供商和最终用户而言,只需决定需要进行什么操作,而无需知道需要什么协议进行通信,也无需担心设备在控制逻辑上的不同。即第三方应用程序可以通过家庭网络提供的界面或接口对接入的设备进行相应的控制。

附图说明

[0102] 图 1 为本发明家庭网络物理架构图;

[0103] 图 2 为本发明家庭网络系统的层次模型图;

[0104] 图 3 为本发明家庭网络系统功能模块划分图;

[0105] 图 4 为本发明家庭网络各组成单元之间的通信协议图;

[0106] 图 5 为本发明中心服务器软件系统结构框图。

具体实施方式

[0107] 下面结合附图对本发明做进一步详细描述。

[0108] 通过研究发现,采用什么样的架构,与具体需求尤其是所要支持的设备特性密切相关。家庭中所要支持的设备可分为如下三类:

[0109] 1) 第一类设备:自身包含以太网接口或具有 IEEE 802.11 无线网卡,如大多数网络摄像头。这类设备往往包含 HTTP 服务器,用户或程序可以访问其自带的 Web 页面实现设备信息获取和控制。

[0110] 2) 第二类设备:不具有标准的网络接口,或是虽有接口,但不符合以太网或 IEEE 802.11 标准,如普通空调、电视等。这些设备的另一个特点是控制逻辑一般较为复杂,如对于空调来说,不仅有简单的开关功能,还有模式切换、升温降温等功能。这些设备往往使用

红外遥控器等进行控制,无网络接口。

[0111] 3) 第三类设备:控制逻辑简单,仅支持开关操作,如电灯等。显然,大多数这种设备没有网络接口。另外,这种设备的数量在家庭网络中往往是比较多的。

[0112] 为了支持上述三类设备,需要分析它们的差异性,针对不同的情况设计不同的策略来处理。设备的差异性主要有两种:语法差异和语义差异。前者是指设备控制所使用的协议或物理连接方式与家庭网络系统中所使用的不同。比如,一般家用电视采用遥控器进行控制,使用的红外信号;而我们的家庭网络系统中使用的是以太网。后者是指协议所承载的内容含义不同。比如,网络摄像头和无线报警灯可能都采用了 HTTP 协议,但是传输数据的实际含义肯定是不同的。对于三类设备,其差异性特征如表 1 所示:

[0113] 表 1 三类设备的差异性特征

[0114]

差异性	第一类设备	第二类设备	第三类设备
是否有语法差异	否	是	是
是否有语义差异	是	是	否

[0115] 物理架构设计阶段主要解决语法差异,而语义差异则在逻辑架构设计阶段解决。对于第一类设备,由于其使用的协议与家庭网络采用的相同,因此可以直接接在家庭网络的局域网上,家庭网络平台只需要处理其语义差异即可。而对于第二类设备,我们需要开发专用的控制器,比如红外控制器等,用于实现系统所用的网络协议与专有协议(如红外信号等)的相互转换,然后用控制器的专有协议端控制设备,网络协议端与家庭网络相连,进而间接实现家庭网络与设备之间的相互通信。当然,在物理上互联之后,平台还必须解决此类设备在语义上的差异性。而对于第三类设备,与第二类设备类似也必须引入控制器,只是控制器的私有协议端的信号类型非常简单,为开关信号,设备的控制语义也很简单,只有开和关两种状态。鉴于此,第三类设备的控制器表现为智能开关,使用网络信号来控制设备的通断;同时考虑到第三类设备数量较多,因此可以采用一对多的连接方式,使用一个多路控制开关来控制多个设备,以降低控制器成本。

[0116] 综合以上分析,我们可以设计出该家庭网络系统的物理架构和逻辑架构,其中,逻辑架构从从层次模型、模块划分和协议设计三个方面进行介绍。

[0117] 图 1 给出了本发明家庭网络系统的物理架构,广域网被具体化为 Internet,而局域网则被具体化为以太网或 IEEE802.11 无线局域网。相对于家庭网络的一般架构,本架构的最大不同是将家庭网关分拆为宽带路由和中心服务器两部分。这是因为家庭网关兼具宽带接入和家庭网络管理两大功能,而前者一般由电信部门和 ISP 提供的宽带路由完成,本身不具备二次开发能力。因此要构建家庭网络的软件系统,实现家庭网络管理功能,就只能部署在另外的机器上,我们称之为中心服务器。中心服务器采用 PC 机作为硬件平台,这是因为部署家庭网络的家庭一般都有 PC 机,而且相对于嵌入式平台,PC 机在计算速度、存储容量上都有不小的优势,便于实现更复杂的功能,也省去了重新开发嵌入式平台的开发成本和硬件成本。后文中,在提及对家庭网关进行设计时,默认就是指对中心服务器进行设计;反之亦然。

[0118] 采用什么样的物理架构,与具体需求尤其是所要支持的设备特性密切相关。星形网络架构要求掌握大量的设备资料,而且需要对每一种设备单独编程,工作量很大,这不符合实际情况。而且,更希望使用常见的、标准化的协议来构建家庭网络,而不是使用私有的协议,因此,总线式的以太网以及 IEEE 802.11 无线网更适合。然而,所要支持的设备并不一定采用这种协议。星形网络架构的系统,其支持的电器设备符合 Echonet 标准,可以通过一个 Echonet UPnP 网关来进行协议转换;但在需求中,所要支持的设备甚至不包含网络接口(如一般家用的空调、电灯等),即使有接口,也未必使用标准的以太网协议或 IEEE 802.11 无线协议,其协议类型可能是多种多样的。归结而言,所要支持的设备可分为如下三类:

[0119] 第一类设备:自身包含以太网接口或具有 IEEE 802.11 无线网卡,如大多数网络摄像头。这类设备往往包含 HTTP 服务器,用户或程序可以访问其自带的 Web 页面实现设备信息获取和控制。

[0120] 第二类设备:不具有标准的网络接口,或是虽有接口,但不符合以太网或 IEEE 802.11 标准,如普通空调、电视等。这些设备的另一个特点是控制逻辑一般较为复杂,如对于空调来说,不仅有简单的开关功能,还有模式切换、升温降温等功能。这些设备往往使用红外遥控器等控制,无网络接口。

[0121] 第三类设备:控制逻辑简单,仅支持开关操作,如电灯等。显然,大多数这种设备没有网络接口。另外,这种设备的数量在家庭网络中往往是比较多的。

[0122] 为了支持上述三类设备,需要分析它们的差异性,针对不同的情况设计不同的策略来处理。设备的差异性主要有两种:语法差异和语义差异。前者是指设备控制所使用的协议或物理连接方式与家庭网络系统中所使用的不同。比如,一般家用电视采用遥控器进行控制,使用的红外信号;而家庭网络系统中使用的是以太网。后者是指协议所承载的内容含义不同。比如,网络摄像头和无线报警灯可能都采用了 HTTP 协议,但是传输数据的实际含义肯定是不同的。

[0123] 物理架构设计阶段主要解决语法差异,而语义差异则在逻辑架构设计阶段解决。对于第一类设备,由于其使用的协议与家庭网络采用的相同,因此可以直接接在家庭网络的局域网,家庭网络平台只需要处理其语义差异即可。而对于第二类设备,需要开发专用的控制器,比如红外控制器等,用于实现系统所用的网络协议与专有协议(如红外信号等)的相互转换,然后用控制器的专有协议端控制设备,网络协议端与家庭网络相连,进而间接实现家庭网络与设备之间的相互通信。当然,在物理上互联之后,平台还必须解决此类设备在语义上的差异性。而对于第三类设备,与第二类设备类似也必须引入控制器,只是控制器的私有协议端的信号类型非常简单,为开关信号,设备的控制语义也很简单,只有开和关两种状态。鉴于此,第三类设备的控制器表现为智能开关,使用网络信号来控制设备的通断;同时考虑到第三类设备数量较多,因此可以采用一对多的连接方式,使用一个多路控制开关来控制多个设备,以降低控制器成本。

[0124] 如图 2, 3, 4 给出了家庭网络系统的逻辑架构,以功能逻辑的视角,从层次模型、模块划分和协议设计三个方面介绍家庭网络系统。

[0125] 1) 层次模型

[0126] 层次模型反映了系统的纵向架构,即从最高层的用户到最底层的硬件设备,系统

中的各个逻辑结构是如何分层组织的。如图 2。

[0127] 其中,各层的功能与意义如下所述:

[0128] (1)应用层。本层对最终用户直接可见,包含了家庭网络的用户操作界面和服务提供商开发的第三方应用程序。最终用户使用它们完成对家庭网络的访问和管理。

[0129] (2)虚拟设备层。该层是解决设备语义差异性的关键层。该层从服务层获取设备通信的数据,利用服务层中驱动程序库提供的信息,解析这些数据的含义,然后将设备抽象成一个个“虚拟设备”;同时,该层向应用层提供一组 API,用户界面或第三方应用程序通过 API 操纵这些虚拟设备,就好像是直接操纵真实设备一样,而由虚拟设备层完成语义转换,对应到具体物理设备的操纵。比如,该层可以提供一个虚拟摄像头,应用层可以调用统一的接口操纵这个摄像头,完成拍照、录像等摄像头公有的功能,而无需关心家庭网络中实际连接的是 A 品牌还是 B 品牌的摄像头,也无需关心它们在通信方式、数据语义上的差别;这些语义上的差别由虚拟设备层统一处理。

[0130] (3)服务层。该层提供了操作家庭网络的较为底层的功能。首先,它包含网络监控模块,时刻监控家庭网络中各设备的连接状况,实现设备发现功能;其次,该层包含设备逻辑模块,通过设备驱动数据库提供的信息,完成设备类型识别的功能,同时确定与该设备进行通信所需的语义规范。上述功能均向虚拟设备层提供了接口,完成和设备之间收发数据的功能。

[0131] (4)LAN。即家庭网络系统的局域网。在本系统中,该局域网的应用层协议采用 HTTP 以及 Web Service/SOAP 协议。

[0132] (5)物理设备/控制器层。该层包含第一类设备以及二、三类设备的控制器,其中控制器用来进行协议转换,解决设备的语法差异性。该层的设备与家庭网络的局域网直接相连,位于架构的最底层。

[0133] 2) 模块划分

[0134] 模块划分反映了系统的横向架构,即系统包含哪些功能模块,各模块分别起到什么作用,分别部署在哪个物理设备上等。本家庭网络系统的模块划分情况如图 3 所示。

[0135] 由图 3 我们可以看到,家庭网络系统中需要设计的对象有两个:中心服务器和控制器。对于前者,主要包含三个功能模块:

[0136] (1)网络监控模块。该模块主要用于设备发现,监控局域网上有没有新设备接入,有没有设备刚刚离线。其位于层次模型的服务层。

[0137] (2)设备支持模块。该模块用于识别设备类型,用正确的逻辑与设备进行通信,完成设备的虚拟化。其位于层次模型的服务层和虚拟化层。

[0138] (3)Web 页面与 API 模块。该模块提供了家庭网络的用户界面,同时也向第三方应用程序提供了 API 接口。其位于层次模型的应用层和虚拟化层。

[0139] 而对于控制器,其主要包含两个模块,均位于层次模型的物理设备/控制器层。

[0140] (1)通信模块。控制器一般包含两个端口,一个用来连接家庭网络的局域网,一个用来连接设备。该模块提供一组调用接口,可以分别从两个端口上收发数据。其中与局域网进行通信时,采用 HTTP/Web Service 的通信方式。

[0141] (2)协议转换脚本解释模块。控制器之所以能够解决设备与局域网之间的语法差异问题,就是靠该模块完成协议的转换工作。实现该功能主要有两种方式:直接编写协议转

换程序(直接方式),或者先编写脚本解释器,然后存储协议转换脚本,利用解释器解释执行该脚本完成协议转换(间接方式)。本项目中采用了后一种方式。这是因为设备所采用的是私有协议,几乎每个设备都不相同。而控制器属于嵌入式设备,如果采用直接方式把转换程序硬编码入其中,那么一旦用户更换设备,就必须刷新其固件,甚至必须更换控制器,非常麻烦。而采用间接方式的话,只需要更新其存储的脚本即可,无需改变控制器的程序。

[0142] 图 4 给出了本节主要介绍家庭网络中各物理或逻辑单元之间的通信协议规范。

[0143] 图中,控制器与第二、第三类设备之间的通信均使用设备专有的协议,与具体设备的型号相关。而在局域网部分,则全部统一为 HTTP 协议。之所以使用该协议,是因为它是网络中(应用层)使用最广泛的协议之一,方便实现 Web 形式的访问和控制。并且,HTTP 完全基于文本,实现简单,大多数第一类设备(如网络摄像头)都内置有 Web 服务器,我们在控制器中的通信模块中也内置了 Web 服务器组件,这就为 HTTP 协议的使用提供了硬件支持。

[0144] 特别地,中心服务器与控制器进行通信时均使用了 Web Service 技术,即采用了基于 HTTP 的简单对象访问协议(SOAP)。它使用 XML 对 HTTP 协议所承载的内容做了进一步规范。之所以使用 Web Service 和 SOAP 协议,是因为针对控制器的调用主要表现为远程过程调用(RPC)的形式。比如说,中心服务器要求关闭空调,那么它将调用对应的空调控制器的“Shutdown”函数,这表现为一种远程函数调用。而 SOAP 协议恰恰是实现 RPC 的标准规范协议之一。Web Service/SOAP 协议的另一个好处是构建容易,几乎所有的 HTTP 服务器软件都支持 Web Service。

[0145] 另外,我们在中心服务器上部署了 HTTP 和 Web Service 服务器软件,家庭网络平台的 API 同样以 Web Service 的形式提供。这样,第三方应用程序便可以用标准的 SOAP 协议以远程调用的形式使用这些 API,而最终用户则可以通过浏览器以 HTTP 协议的方式访问家庭网络。

[0146] 合理的架构设计是解决设备互联问题的关键,而要解决此问题,关键在于解决设备和家庭网络之间语法和语义的差异性。前文所述的物理和逻辑架构方案已经完整地解决了这些问题。

[0147] 图 5 给出了整个系统的两大模块组成:

[0148] 1) 系统监控模块,完成系统时钟驱动的设备信息更新功能;由系统时钟驱动的网络扫描(设备发现)和设备识别等功能与用户驱动的设备访问、管理等功能相对独立,它们之间主要通过一系列数据存储建立关系。这也体现了我们构建此系统的主要思路:即由一个监护进程不断与设备进行通信,更新设备的相关信息,存储在数据库中;而用户则可以随时从数据库中查看信息,并且只在必要时,直接与设备进行通信。这种方式可以明显提高用户访问系统时系统的响应速度,也符合该系统作为一个软硬件平台的定位。

[0149] 2) Web 页面与 Web Service API 功能,主要完成用户驱动的各种功能,以及向外界以 Web Service/SOAP 协议的方式提供 API。这两个模块共享位于系统中层的各种服务,主要包括与网络通信相关的服务,以及与设备、虚拟化相关的服务。位于系统底层,操作系统上层的是数据库系统,其中包括存储驱动文件的驱动库和存储系统所需其他信息的主数据库。数据库之上是数据访问层和驱动加载器,它们的作用是将操作数据库的方法进行封装,方便数据库的访问。

[0150] 实例:

[0151] 下面给出了 SOAP 协议的一个样例。该示例展示了函数 `String login(String username, String password)` 的 SOAP 协议格式,其中上半部分是调用格式,下半部分是返回值格式。

[0152]

```
POST /Services/UserService.asmx HTTP/1.1
Host: localhost
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <Login xmlns="http://tempuri.org/">
      <username>string</username>
      <password>string</password>
    </Login>
  </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LoginResponse xmlns="http://tempuri.org/">
      <LoginResult>string</LoginResult>
    </LoginResponse>
  </soap12:Body>
</soap12:Envelope>
```

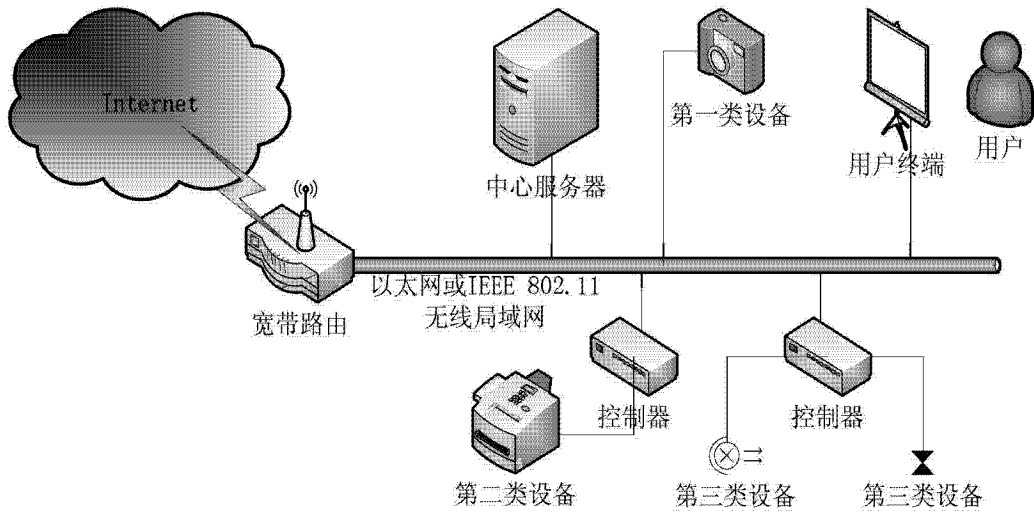



图 1

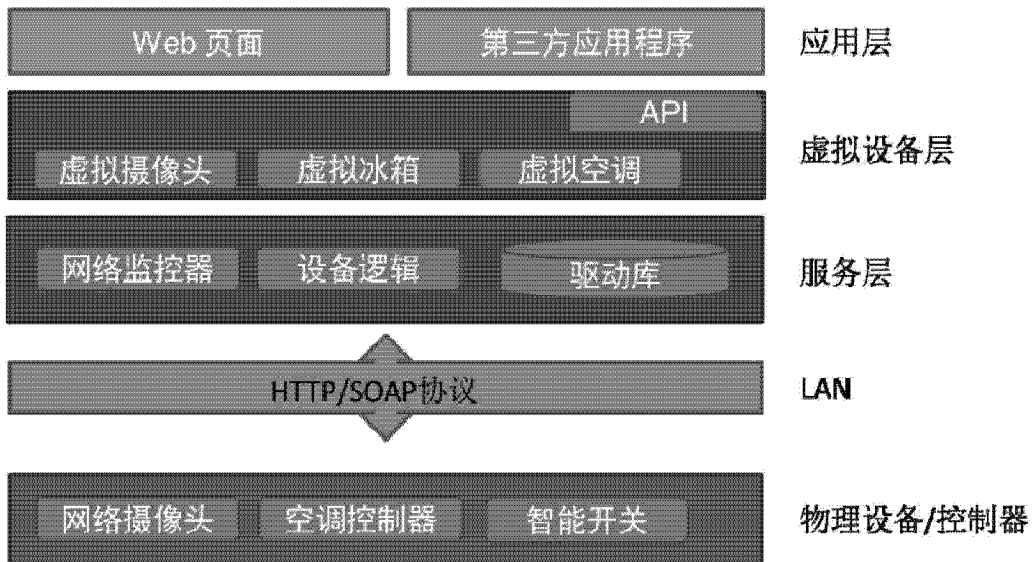


图 2

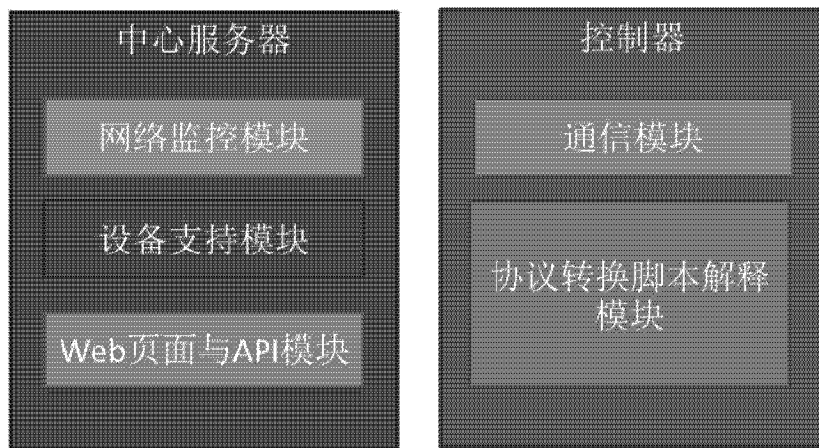


图 3

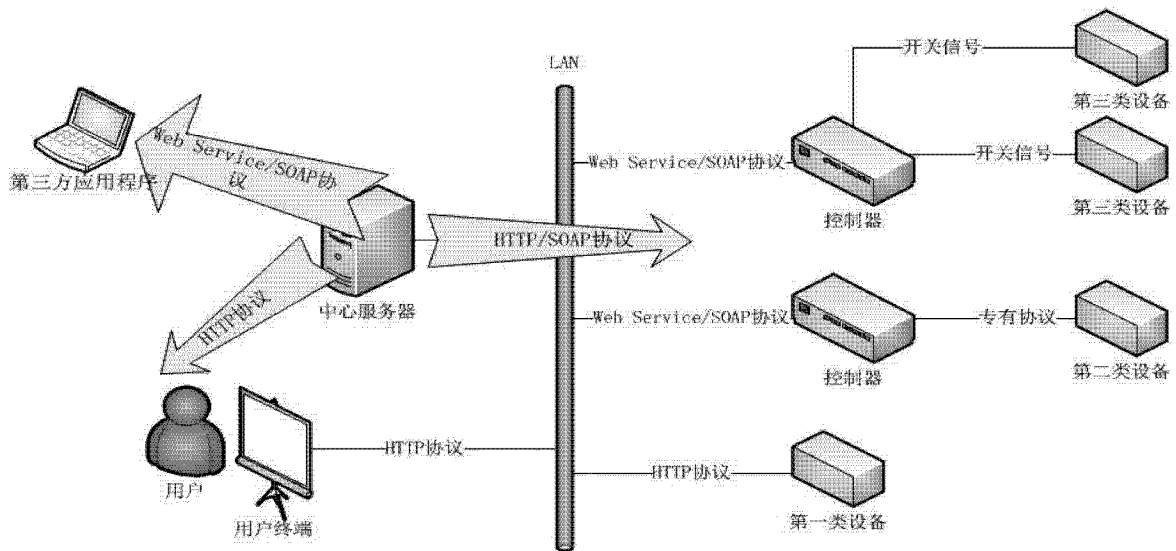


图 4



图 5