

设备控制方法、装置、服务器及存储介质

本申请要求于 2021 年 08 月 19 日提交的申请号为 202110956934.2、发明名称为“设备控制方法、装置、服务器及存储介质”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及物联网技术领域，特别涉及一种设备控制方法、装置、服务器及存储介质。

背景技术

随着物联网 (internet of things, IoT) 技术的飞速发展, IoT 设备可以通过接入 IoT 平台, 实现在任何时间、任何地点的人、机、物相通, 同时 IoT 应用也能够通过 IoT 平台对相应的 IoT 设备进行控制。相关技术中, IoT 应用的开发者通过在 IoT 平台上开发设备对应的数字孪生模型, IoT 设备接入平台后与对应的数字孪生模型绑定, 使得 IoT 应用和 IoT 设备能够通过该数字孪生模型, 进行控制命令下发和数据上报, 从而实现对 IoT 设备的控制。

上述技术中, 由于不同厂家生产的同类设备设置有不同的数据格式, 应用开发者需要通过不同的数字孪生模型, 用不同数据格式的命令控制不同厂家的设备, 应用和设备之间强耦合, 导致控制 IoT 设备的过程十分繁琐。

发明内容

为了解决网络设备的控制过程繁琐的问题, 本申请实施例提供了一种设备控制方法、装置、服务器及计算机可读存储介质。该技术方案如下:

第一方面, 提供了一种设备控制方法, 由服务器执行, 该方法包括:

接收控制设备的第一控制命令, 该第一控制命令携带第一网络设备的设备标识和第一控制体, 该第一控制命令用于指示该第一网络设备执行该第一控制体对应的操作, 该第一控制体符合标准数据格式, 该标准数据格式由该第一网络设备的上位设备类型对应的标准模型描述;

基于该第一网络设备的设备标识, 确定该第一网络设备对应的第一数字孪生模型, 该第一数字孪生模型描述该第一网络设备所支持的数据格式, 该第一数字孪生模型记录了该标准数据格式和该第一网络设备所支持的数据格式之间的第一对应关系;

基于该第一对应关系, 将该第一控制体转换为该第一网络设备所支持的数据格式, 将转换后的第一控制体发送至该第一网络设备。

本申请实施例所提供的设备控制方法, 对于任一类型的网络设备, 该类型的网络设备包括多种不同的型号, 通过该类型下任一型号的网络设备的数字孪生模型中记录的该类型网络设备的标准数据格式与该型号网络设备所支持的数据格式之间的对应关系, 对控制设备发送的控制命令的控制体进行格式转换。该方法使控制设备能够通过统一的标准数据格式的控制命令的控制体, 控制不同型号的网络设备, 且无需将不同型号的网络设备支持的数据格式与

标准数据格式进行适配，网络设备就能够识别控制设备发送的控制命令的控制体，实现了网络设备与物联网应用之间的解耦，从而简化了网络设备的控制过程。

在一些实施例中，该方法还包括：

接收控制设备的第二控制命令，该第二控制命令携带第二网络设备的设备标识和第二控制体，该第二控制命令用于指示该第二网络设备执行该第二控制体对应的操作，该第二控制体符合该标准数据格式，该第一网络设备的型号与该第二网络设备的型号不同，该第一网络设备的上位设备类型与该第二网络设备的上位设备类型相同；基于该第二网络设备的设备标识，确定该第二网络设备对应的第二数字孪生模型，该第二数字孪生模型描述该第二网络设备所支持的数据格式，该第二数字孪生模型记录了该标准数据格式和该第二网络设备所支持的数据格式之间的第二对应关系；基于该第二对应关系，将该第二控制体转换为该第二网络设备所支持的数据格式，将转换后的第二控制体发送至该第二网络设备。

本申请实施例提供的方法，能够通过上位设备类型的标准数据格式和第二网络设备所支持的数据格式之间的对应关系，对第二控制命令的控制体进行格式转换并发送至第二网络设备，从而实现通过统一的标准数据格式的控制命令的控制体，对同一类型的不同型号的网络设备进行控制。

在一些实施例中，该方法还包括：

接收该第一网络设备的状态数据和该第一网络设备的设备标识，该状态数据为该第一网络设备所支持的数据格式；基于该第一对应关系，将该状态数据转换为该标准数据格式，将转换后的状态数据发送至该控制设备。通过对网络设备的状态数据进行格式转换，使控制设备能够以统一的标准数据格式，识别不同型号的网络设备的状态数据。

在一些实施例中，该接收控制设备的第一控制命令之前，该方法还包括：

接收该第一网络设备的设备接入请求，该设备接入请求携带该第一网络设备的型号和该第一网络设备的设备标识；根据该第一网络设备的型号，确定该第一网络设备的型号对应的第一数字孪生模型；建立该第一网络设备的设备标识和该第一数字孪生模型之间的第三对应关系。通过建立设备标识与第一数字孪生模型之间的第三对应关系，使第一网络设备与对应的第一数字孪生模型绑定，从而能够实现对第一网络设备的控制体的格式转换。

在一些实施例中，该建立该第一网络设备的设备标识和该第一数字孪生模型之间的第三对应关系之后，该方法还包括：

基于该第一数字孪生模型，创建该第一网络设备的运行实例。通过创建运行实例，能够存储该第一网络设备的状态数据，从而能够查询该第一网络设备的历史状态。

在一些实施例中，该第一对应关系包含该第一数字孪生模型的数字孪生元模型与该标准模型的标准元模型之间的对应关系；

该基于该第一对应关系，将该第一控制体转换为该第一网络设备所支持的数据格式，包括：

基于该第一数字孪生模型的数字孪生元模型与该标准模型的标准元模型之间的对应关系，确定该上位设备类型的一个或多个标准元模型；基于该第一数字孪生模型和该上位设备类型的一个或多个标准元模型，将该第一控制体转换为该第一网络设备所支持的数据格式。通过确定第一网络设备对应的标准元模型，从而能够通过该标准元模型提供的标准数据格式和第一网络设备支持的格式，对第一控制体进行格式转换，从而实现对第一网络设备的控制。

在一些实施例中，该第一数字孪生模型包括一个或多个数字孪生元模型，该数字孪生元模型包括属性元模型、命令元模型、事件元模型、数据元模型、组件元模型和关系元模型中的一种或多种。通过将该第一数字孪生模型分为多个数字孪生元模型，能够将该数字孪生模型分为多个子模型，从而能够更加灵活地描述第一网络设备类型的多个特性，使得与对应的标准模型进行关联时能够更加便利。

第二方面，提供了一种设备控制装置，该装置包括多个功能模块，用于执行如第一方面所提供的设备控制方法中的对应步骤。

通过该设备控制装置，实现对于任一类型下任一型号的网络设备，基于该型号的网络设备的数字孪生模型中记录的该类型网络设备的数据格式与该型号网络设备所支持的数据格式之间的对应关系，对控制设备发送的控制命令的控制体进行格式转换，从而使控制设备能够通过统一的标准数据格式的控制命令的控制体，控制不同型号的网络设备，且无需将不同型号的网络设备支持的数据格式与标准数据格式进行适配，网络设备就能够识别控制设备发送的控制命令的控制体，实现了网络设备与物联网应用之间的解耦，从而简化了网络设备的控制过程。

第三方面，提供了一种服务器，该服务器包括通信总线、至少一个网络接口、存储器以及至少一个处理器，该通信总线用于在上述组件之间传送信息，该至少一个网络接口用于与其它设备或通信网络通信，该存储器保存有程序代码，该至少一个处理器通过读取并执行该存储器中保存的程序代码实现上述第一方面中设备控制方法的任一步骤。

通过该服务器，实现对于任一类型下任一型号的网络设备，基于该型号的网络设备的数字孪生模型中记录的该类型网络设备的数据格式与该型号网络设备所支持的数据格式之间的对应关系，对控制设备发送的控制命令的控制体进行格式转换，从而使控制设备能够通过统一的标准数据格式的控制命令的控制体，控制不同型号的网络设备，且无需将不同型号的网络设备支持的数据格式与标准数据格式进行适配，网络设备就能够识别控制设备发送的控制命令的控制体，实现了网络设备与物联网应用之间的解耦，从而简化了网络设备的控制过程。

第四方面，提供了一种计算机可读存储介质，当该计算机可读存储介质中的程序代码由服务器的处理器执行时，使得服务器能够执行上述第一方面中设备控制方法的任一步骤。

通过该计算机可读存储介质，实现对于任一类型下任一型号的网络设备，基于该型号的网络设备的数字孪生模型中记录的该类型网络设备的数据格式与该型号网络设备所支持的数据格式之间的对应关系，对控制设备发送的控制命令的控制体进行格式转换，从而使控制设备能够通过统一的标准数据格式的控制命令的控制体，控制不同型号的网络设备，且无需将不同型号的网络设备支持的数据格式与标准数据格式进行适配，网络设备就能够识别控制设备发送的控制命令的控制体，实现了网络设备与物联网应用之间的解耦，从而简化了网络设备的控制过程。

附图说明

图1是本申请实施例提供的一种设备控制方法的实施环境示意图；

图2是本申请实施例提供的一种服务器200的结构示意图；

图3是本申请实施例提供的一种物联网平台的结构示意图；

图 4 是本申请实施例提供的一种设备控制方法的步骤流程图；
图 5 是本申请实施例提供的一种获取标准模型的步骤流程图；
图 6 是本申请实施例提供的一种标准模型创建界面；
图 7 是本申请实施例提供的一种获取数字孪生模型的步骤流程图；
图 8 是本申请实施例提供的一种设备控制方法的示意图；
图 9 是本申请实施例提供的一种发送控制命令的步骤流程图；
图 10 是本申请实施例提供的一种设备控制装置的结构示意图。

具体实施方式

为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步地详细描述。

为了便于理解本申请实施例的技术过程，下面对本申请实施例所涉及的一些关键术语和名词进行解释：

(1) 物联网：物联网是物物相连的互联网，能够实现物品与物品、人与物品之间的互连，是实现智能化识别、定位、跟踪、监控和管理的一种网络。

(2) 物联网平台：物联网平台是一种基于互联网的、连接物体和实现物联网功能的平台，物联网平台具有多种功能，如网络设备和物联网应用的通信、设备接入、设备管理、数据存储、物联网应用接入等。

(3) 数字孪生模型：数字孪生模型是基于数字孪生模型语言实现的虚拟模型，用于描述网络设备的特性。

(4) 数字孪生模型语言（digital twin model language, DTML）：一种基于链接数据的 javascript 对象表示方法（javascript object notation for linked data, JSON-LD）的计算机程序语言，用于描述数字孪生模型。

下面对本申请实施例提供的设备控制方法的实施环境进行介绍。

图 1 是本申请实施例提供的一种设备控制方法的实施环境示意图，该实施环境包括：控制设备 101、服务器 102 和网络设备 103。控制设备 101 和服务器 102 能够通过有线网络或无线网络进行直接或间接地连接，服务器 102 和网络设备 103 能够通过有线网络或无线网络进行直接或间接地连接，本申请实施例对此不作限定。

控制设备 101 可以是智能手机、平板电脑、笔记本电脑、台式计算机等，但并不局限于此。在控制设备 101 安装有物联网应用，该物联网应用可以是工业应用、物流应用和生活应用等。控制设备 101 能够基于该物联网应用，通过服务器 102 向网络设备 103 发送控制命令，控制网络设备执行对应操作，并通过该服务器 102 接收网络设备 103 发送的状态数据。

服务器 102 为控制服务器，该服务器 102 可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还能够是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、内容分发网络（content delivery network, CDN）、以及大数据和人工智能平台等基础云计算服务的云服务器。该服务器 102 部署有物联网平台，服务器 102 能够通过该物联网平台，在控制设备 101 和网络设备 103 之间转发控制命令和状态数据。

网络设备 103 可以是不同厂家生产的不同类型的网络设备，如摄像机、智能冰箱、空调

等，本实施例对此不作限定。该网络设备 103 能够通过服务器 102 接收控制设备 101 的控制命令，基于该控制命令执行对应的操作，并通过服务器 102 向控制设备 101 发送状态数据。

下面对本申请实施例涉及的服务器的结构进行介绍。

图 2 是本申请实施例提供的一种服务器 200 的结构示意图。图 2 所示的服务器 200 用于执行该设备控制方法所涉及的操作，如图 2 所示，该服务器 200 包括至少一个处理器 201、通信总线 202、存储器 203 以及至少一个网络接口 204。

处理器 201 例如是通用中央处理器（central processing unit, CPU）、网络处理器（network processor, NP）、图形处理器（graphics processing unit, GPU）、神经网络处理器（neural-network processing units, NPU）、数据处理单元（data processing unit, DPU）、微处理器或者一个或多个用于实现本申请方案的集成电路。例如，处理器 201 包括专用集成电路（application-specific integrated circuit, ASIC）、可编程逻辑器件（programmable logic device, PLD）或其组合。PLD 例如是复杂可编程逻辑器件（complex programmable logic device, CPLD）、现场可编程逻辑门阵列（field-programmable gate array, FPGA）、通用阵列逻辑（generic array logic, GAL）或其任意组合。

通信总线 202 用于在上述组件之间传送信息。通信总线 202 可选地分为地址总线、数据总线、控制总线等。为便于表示，图 2 中仅用一条粗线表示，但并不表示仅有一根总线或一种类型的总线。

存储器 203 例如是只读存储器（read-only memory, ROM）或可存储静态信息和指令的其它类型的静态存储设备，又如是随机存取存储器（random access memory, RAM）或者可存储信息和指令的其它类型的动态存储设备，又如是电可擦可编程只读存储器（electrically erasable programmable read-only memory, EEPROM）、只读光盘（compact disc read-only memory, CD-ROM）或其它光盘存储、光碟存储（包括压缩光碟、激光碟、光碟、数字通用光碟、蓝光光碟等）、磁盘存储介质或其它磁存储设备，或者是能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其它介质，但不限于此。存储器 203 例如是独立存在，并通过通信总线 202 与处理器 201 相连接。或者可选地存储器 203 和处理器 201 集成在一起。

网络接口 204 使用任何收发器一类的装置，用于与其它设备或通信网络通信。网络接口 204 包括有线网络接口，可选地还包括无线网络接口。其中，有线网络接口例如为以太网接口。以太网接口例如是光接口、电接口或其组合。无线网络接口例如为无线局域网（wireless local area networks, WLAN）接口，蜂窝网络接口或其组合等。

在一些实施例中，处理器 201 包括一个或多个 CPU，如图 2 中所示的 CPU0 和 CPU1。

在一些实施例中，服务器 200 可选地包括多个处理器，如图 2 中所示的处理器 201 和处理器 205。这些处理器中的每一个例如是一个单核处理器，又如是一个多核处理器。这里的处理器可选地指一个或多个设备、电路、和/或用于处理数据（如计算机程序指令）的处理核。

在一些实施例中，服务器 200 还包括输出设备和输入设备。输出设备可选地以多种方式来显示信息。例如，输出设备可选地是液晶显示器（liquid crystal display, LCD）、发光二极管（light emitting diode, LED）显示设备、阴极射线管（cathode ray tube, CRT）显示设备或投影仪（projector）等。输入设备和处理器 201 通信，可选地以多种方式接收用户的输入。例如，输入设备可选地是鼠标、键盘、触摸屏设备或传感设备等。

可选地，处理器 201 通过读取存储器 203 中保存的程序代码 210 实现下述实施例中的方法，或者，处理器 201 通过内部存储的程序代码实现下述实施例中的方法。在处理器 201 通过读取存储器 203 中保存的程序代码 210 实现下述实施例中的方法的情况下，存储器 203 中保存实现本申请实施例提供的设备控制方法的程序代码。

处理器 201 实现上述功能的更多细节请参考下面各个方法实施例中的描述，在这里不再重复。

本申请实施例还提供了一种包括程序代码的计算机可读存储介质，例如包括程序代码的存储器 203，上述程序代码可由服务器 200 的处理器 201 执行以完成设备控制方法。可选地，计算机可读存储介质可以是只读内存（read-only memory，ROM）、随机存取存储器（random access memory，RAM）、只读光盘（compact-disc read-only memory，CD-ROM）、磁带、软盘和光数据存储设备等。

基于上述实施环境，下面对部署在服务器上的物联网平台进行介绍。

图 3 是本申请实施例提供的一种物联网平台的结构示意图，应理解，图 3 仅是示例性地展示了物联网平台的一种结构化示意图，本申请并不限定对物联网平台中各项功能的划分。如图 3 所示，该物联网平台提供 3 类服务，包括：设备接入管理服务、模型引擎服务和标准模型服务。下面分别对 3 类服务进行介绍。

设备接入管理服务用于将物联网应用、网络设备和对应的数字孪生模型进行关联，并提供设备接入、应用接入的功能，实现物联网应用和网络设备之间的互通。该设备接入管理服务还用于将接收到的控制命令和状态数据发送给模型引擎服务，接收模型引擎服务发送的经过格式转换的控制命令的控制体和状态数据并转发给对应的网络设备或物联网应用。

模型引擎服务是基于 DTML 实现的功能单元，用于同步标准模型服务中的标准模型，并基于标准模型生成标准应用程序接口（application programming interface，API），该模型引擎服务还用于存储并解析数字孪生模型，对设备接入管理服务发送的控制命令的控制体和状态数据进行数据格式的转换，并将转换后的控制命令的控制体和状态数据发送给设备接入管理服务，具体过程详见图 4 对应的实施例。

标准模型服务是基于 DTML 实现的功能单元，用于创建并存储标准模型。

在介绍了本申请实施例应用的系统架构以及物联网平台的结构的基础上，下面结合图 4，对本申请实施例提供的设备控制方法进行说明。图 4 是本申请实施例提供的一种设备控制方法的步骤流程图，该方法由服务器执行，如图 4 所示，该设备控制方法包括以下步骤 401 至 410。

401、服务器获取标准模型，该标准模型描述第一网络设备的上位设备类型的标准数据格式。

其中，第一网络设备的上位设备类型是指该第一网络设备所属的设备类型，第一网络设备可以是该设备类型下任一型号的网络设备，例如，该上位设备类型为摄像机，第一网络设备为第一型号的摄像机。

若需要为任一种上位设备类型提供标准的数据格式，从而简化该上位设备类型的网络设备的控制过程，则平台管理员在服务器上创建该上位设备类型的网络设备的标准模型。其中，该标准模型可以为标准元模型组，即该标准模型可以包括一个或多个标准元模型，每个标准元模型用于描述该上位设备类型的一个特性，该标准元模型包括标准属性元模型（property）、

标准命令元模型 (command)、标准事件元模型 (event)、标准数据元模型 (dataschema)、标准组件元模型 (component) 和标准关系元模型 (relationship) 中的一种或多种。通过将该标准模型分为多个标准元模型, 能够将该标准模型分为多个子模型, 从而能够更加灵活地描述上位设备类型的多个特性, 使得后续将数字孪生模型与该标准模型进行关联时能够更加便利。

在一些实施例中, 服务器获取标准模型的过程如图 5 所示, 物联网平台的标准模型服务提供有如图 6 所示的标准模型创建界面, 平台管理员能够通过该标准模型创建界面进行模型代码的编写, 从而创建该标准模型, 服务器响应于在该标准模型创建界面中对该标准模型的模型代码的提交操作, 标准模型服务将该模型代码进行存储, 也即是, 将该标准模型进行存储。可以理解的是, 标准模型可以包括一个或多个标准元模型, 不同上位设备类型所对应的标准模型可以包括不同的标准元模型, 平台管理员能够基于不同上位设备类型的特性, 创建不同的标准模型, 本申请实施例对此不作限定。

需要说明的是, 平台管理员在创建任一上位设备类型的网络设备的标准模型时, 可以首先通过查询多个该上位设备类型的网络设备的生产厂家提供的设备文档, 获知该上位设备类型的网络设备的功能和属性, 然后再进行标准模型的创建, 以使创建的标准模型能够准确并全面的描述该上位设备类型的网络设备的特性。

示例性地, 以该上位设备类型是摄像头为例进行说明。摄像头标准模型中包括 2 个标准元模型, 分别为标准命令元模型和标准属性元模型, 平台管理员能够通过标准模型创建界面编写模型代码, 响应于对摄像头标准模型的模型代码的提交操作, 标准模型服务将该摄像头标准模型进行存储。该摄像头标准模型的模型代码如下所示:

```
"rotateCommand":{ //标准命令元模型的元模型标识
  "displayName":"摄像头旋转",
  "input":{ //输入参数
    "@type": "object",
    "displayName":"参数: 水平旋转角度, 垂直旋转角度, 速度",
    "fields":{
      "horizontalAngle":{ //水平旋转角度参数的标识
        "@type":"integer"
      },
      "verticalAngle":{ //垂直旋转角度参数的标识
        "@type":"integer"
      },
      "speed":{ //速度参数的标识
        "@type":"integer"
      }
    }
  },
  "output":{ //输出参数
    "@type": "object",
    "fields": {
```

```

    "SUCCESS": {
        "@type": "integer"
    }
}
},

```

"status":{略} //标准属性元模型的元模型标识

其中，rotateCommand 所标识的标准元模型为标准命令元模型，用于提供摄像头的旋转命令的标准数据格式，该旋转命令包括 3 个输入参数，分别为水平旋转角度、垂直旋转角度和速度，该 3 个参数的标准数据格式分别为 horizontalAngle、verticalAngle 和 speed 对应的代码段中所描述的格式。status 所标识的标准元模型为标准属性元模型，用于提供摄像头状态数据的标准数据格式。

在一些实施例中，服务器能够基于标准模型生成标准应用程序接口，本步骤 401 还包括：标准模型服务将存储的标准模型发送给模型引擎服务，模型引擎服务接收并存储该标准模型，以完成标准模型的同步，并对该标准模型进行解析，为该标准模型的每个标准元模型生成标准应用程序接口。该标准应用程序接口能够用于物联网应用的开发，使得该物联网应用能够发送或接收标准数据格式的控制命令的控制体或状态数据。

示例性地，以为摄像头标准模型生成标准应用程序接口为例进行说明，模型引擎服务从标准模型服务同步该摄像头标准模型，对该摄像头标准模型进行解析，分别为 2 个标准元模型生成标准应用程序接口，包括旋转命令接口和状态属性接口。该 2 个标准元模型的标准应用程序接口如下所示：

Https://{host}/device/{deviceid}/rotateCommand //旋转命令接口

Https://{host}/device/{deviceid}/status //状态属性接口

402、服务器获取第一数字孪生模型，将该第一数字孪生模型和对应的标准模型关联，该第一数字孪生模型描述该第一网络设备所支持的数据格式。

对任意的上位设备类型，若需要通过物联网应用对该上位设备类型下任一型号的网络设备（例如，该上位设备类型下的第一型号的网络设备）进行控制，则应用开发者在物联网平台中创建该第一型号的网络设备对应的数字孪生模型，并开发物联网应用。

其中，该第一数字孪生模型为第一数字孪生元模型组，该第一数字孪生模型包括一个或多个数字孪生元模型，每个数字孪生元模型用于描述该第一网络设备的一个特性，数字孪生元模型包括属性元模型、命令元模型、事件元模型、数据元模型、组件元模型和关系元模型中的一种或多种。

在一些实施例中，应用开发者首先在物联网平台进行注册，物联网平台为该应用开发者生成用户标识，然后应用开发者创建第一网络设备对应的第一数字孪生模型。服务器获取第一数字孪生模型的过程如图 7 所示，该物联网平台提供有数字孪生模型创建界面，应用开发者能够通过该数字孪生模型创建界面，基于该第一网络设备所支持的数据格式，编写第一数字孪生模型中每个数字孪生元模型的模型代码，并通过物联网平台的标准模型服务，查询到

该第一网络设备的上位设备类型的标准模型，将所创建的第一数字孪生模型和对应的该上位设备类型的标准模型相关联，也即是，将所创建的第一数字孪生元模型组与对应的标准元模型组相关联，完成模型代码编写后，应用开发者能够通过该数字孪生创建界面提交该模型代码。

示例性地，对上述关联第一数字孪生模型和对应的标准模型的过程进行说明。应用开发者编写第一数字孪生模型中每个数字孪生元模型的模型代码时，在所创建的第一数字孪生模型的每个数字孪生元模型的代码中，添加“@type”元素，并在“@type”元素后添加每个数字孪生元模型对应的标准模型中标准元模型的元模型标识，从而能够通过“@type”元素，将所创建的第一数字孪生模型中每个数字孪生元模型与对应的标准模型中对应的标准元模型相关联，并在每个数字孪生元模型的每个参数对应的代码中，添加“@type”元素，并在“@type”元素后面添加每个数字孪生元模型中每个参数对应的标准元模型中对应参数的标识，从而通过“@type”元素，将每个数字孪生元模型中的每个参数与对应的标准元模型中对应的参数关联起来。通过关联第一数字孪生模型的每个数字孪生元模型和对应的标准模型的每个标准元模型，并关联每个数字孪生元模型的每个参数和对应的每个标准元模型的每个参数，实现了将第一数字孪生模型与对应的标准模型相关联。

由于第一数字孪生模型提供了第一网络设备所支持的数据格式，标准模型提供了标准数据格式，因此，第一数字孪生模型通过“@type”元素，记录了标准数据格式与第一网络设备所支持的数据格式之间的第一对应关系，且该第一对应关系包含了第一数字孪生模型的数字孪生元模型与标准模型的标准元模型之间的对应关系，从而使得模型引擎能够通过解析该第一数字孪生模型中每个数字孪生元模型的“@type”元素，实现第一网络设备所支持的数据格式与标准数据格式之间的转换。

在一些实施例中，设备接入服务响应于在该数字孪生模型创建界面的模型代码提交操作，向模型引擎服务发送模型创建请求，该模型创建请求携带该第一数字孪生模型的模型代码，该模型创建请求用于指示将创建的第一数字孪生模型进行存储，模型引擎服务存储该第一数字孪生模型，并且通过解析该第一数字孪生模型中的id字段，获取该第一数字孪生模型的模型标识，通过解析该第一数字孪生模型中每个数字孪生元模型的“@type”元素，识别与该数字孪生元模型关联的标准元模型。

需要说明的是，该第一数字孪生模型的模型标识基于第一网络设备的型号得到，在一些实施例中，应用开发者创建该第一网络设备的第一数字孪生模型之前，从该第一网络设备的生产厂家得到该第一网络设备的型号，在有些情况下，该第一网络设备的型号可以为字符串。可选地，应用开发者在创建第一数字孪生模型时，用第一网络设备的型号中的目标字段编写模型代码中的id字段，从而创建该第一数字孪生模型的模型标识，或者，应用开发者在数字孪生模型创建界面提交模型代码时，同时提交该第一网络设备的型号，设备接入服务响应于在该数字孪生模型创建界面的模型代码和第一网络设备的型号的提交操作，将该第一网络设备的型号中的目标字段作为该第一数字孪生模型的模型标识，并基于该模型标识在该模型代码中生成id字段。

示例性地，以创建A01型号摄像头的数字孪生模型为例进行说明，应理解，A01仅是对该摄像头的型号的简称，该摄像头的型号为一个字符串。应用开发者获取该A01型号摄像头的型号的字符串，将该字符串中的目标字段作为模型标识，基于A01型号摄像头支持的数据

格式，在数字孪生模型创建界面上，编写 A01 型号摄像头的数字孪生模型的模型代码，该数字孪生模型包括 2 个数字孪生元模型，分别为命令元模型 commandA01 和属性元模型 PropertyA01，通过“@type”元素将 commandA01 与标准元模型 rotateCommand 相关联，将 PropertyA01 与标准元模型 status 相关联。该 A01 型号摄像头的模型代码如下所示：

```
{
  "id":"dtml:dahua:Camera;1", //模型标识
  "properties":{ //属性元模型
    "propertyA01":{
      "@type":"status" //与 status 所标识的标准元模型相关联
    }
  },
  "command":{ //命令元模型
    "commandA01":{
      "@type":"rotateCommand", //与 rotateCommand 所标识的标准元模型相关联
      "displayName":"摄像头旋转",
      "input":{
        "@type": "object",
        "displayName":"参数：水平旋转角度，垂直旋转角度，速度",
        "fields":{
          "ha": { //水平旋转角度参数
            "@type":"integer, horizontalAngle " //与 horizontalAngle 相关联
          },
          "va": { //垂直旋转角度参数
            "@type":"integer, verticalAngle " //与 verticalAngle 相关联
          },
          "speed": { //速度参数
            "@type":"integer, speed " //与 speed 相关联
          }
        }
      }
    }
  }
}
```

其中，commandA01 所标识的数字孪生元模型为命令元模型，用于提供的 A01 型号摄像头的旋转命令所支持的数据格式，该旋转命令包括 3 个输入参数，分别为水平旋转角度、垂直旋转角度和速度，对于该 3 个参数，A01 型号摄像头所支持的数据格式分别为 ha、va 和 speed 对应的代码段所描述的格式。propertyA01 所标识的数字孪生元模型为属性元模型，用于提供 A01 型号摄像头的状态数据所支持的数据格式。

需要说明的是，同一上位设备类型的不同型号的网络设备所支持的数据格式是不同的，因而需要基于不同型号的网络设备所在支持的数据格式，创建不同的数字孪生模型，并将不同型号的网络设备的数字孪生模型与该上位设备类型的标准模型关联。例如，以上位设备类型是摄像头为例进行说明，如图 8 所示，对于 A01 型号摄像头、B01 型号摄像头和 C01 型号摄像头，可以分别创建该 3 个型号的摄像头对应的数字孪生模型，并分别将该 3 个型号的摄像头对应的数字孪生模型与摄像头标准模型相关联。

通过将多个型号的网络设备的数字孪生模型与网络设备的上位设备类型的标准模型相关联，使应用开发者在为网络设备开发物联网应用时，无需针对不同型号的网络设备支持的数据格式进行逐一适配，就能够让网络设备识别控制设备发送的控制命令的控制体，从而达到了简化控制网络设备的过程的目的。

403、第一网络设备向服务器发送设备接入请求，该设备接入请求携带该第一网络设备的型号和第一网络设备的设备标识。

其中，该第一网络设备的型号的目标字段为该第一网络设备对应的第一数字孪生模型的模型标识。

在一些实施例中，应用开发者完成该第一网络设备的第一数字孪生模型的创建后，会将该服务器的地址信息告知该第一网络设备的生产厂家，生产厂家能够将该服务器的地址信息预置在第一网络设备的配置文件中，且该第一网络设备中还预置有该第一网络设备的型号和设备标识，在该第一网络设备首次启动并接入网络的时候，能够基于该服务器的地址信息，向服务器发送设备接入请求，该设备接入请求携带该第一网络设备的型号和设备标识。

404、服务器接收第一网络设备的设备接入请求，将第一网络设备与对应的第一数字孪生模型关联。

在一些实施例中，该步骤 404 包括以下 2 个过程：

过程 1、将第一网络设备与对应的第一数字孪生模型关联。

在一些实施例中，设备接入管理服务响应于该第一网络设备的设备接入请求，获取该设备接入请求携带的第一网络设备的型号中的目标字段，确定与该目标字段相同的模型标识，该模型标识对应的数字孪生模型即为该第一网络设备对应的第一数字孪生模型，将该第一网络设备的设备标识与该模型标识对应存储，从而建立第一网络设备的设备标识和第一数字孪生模型之间的第三对应关系，将该模型标识发送给模型引擎服务。

过程 2、创建第一网络设备的运行实例。

在一些实施例中，模型引擎服务基于该模型标识对应的第一数字孪生模型，创建该第一网络设备的运行实例，生成该运行实例对应的实例标识，该运行实例用于存储该第一网络设备的状态数据。模型引擎服务向设备接入管理服务发送该第一网络设备的实例标识，设备接入管理服务存储该实例标识、模型标识与该第一网络设备的设备标识的对应关系信息，从而将该第一网络设备与对应的第一数字孪生模型和对应的运行实例进行关联。

405、控制设备向服务器发送该第一网络设备的第一控制命令，该第一控制命令携带第一网络设备的设备标识和第一控制体，该第一控制命令用于指示该第一网络设备执行第一控制体对应的操作，该第一控制体为标准数据格式。

在一些实施例中，控制设备通过物联网应用发送标第一控制命令，该物联网应用是应用开发者基于标准应用程序接口开发的应用程序，该第一控制命令携带该第一网络设备的设备

标识和标准格式的第一控制体。可选地，该物联网应用提供有控制界面，该物联网应用的用户能够通过该控制界面，订阅该第一网络设备，并向服务器发送该第一网络设备的第一控制命令。

示例性地，如图9所示，以控制设备向服务器发送对A01型号摄像头的旋转命令为例进行说明。控制设备显示物联网应用的控制界面，并在该控制界面上显示多个已接入物联网平台的网络设备，响应于对该A01型号摄像头的点击操作，显示该A01型号摄像头的旋转命令的参数输入界面，响应于在该参数输入界面的提交操作，基于该参数输入界面中的命令参数向服务器生成旋转命令的控制体，并向服务器发送rotateCommand命令，该rotateCommand命令携带A01型号摄像头的设备标识和旋转命令的控制体。

在一些实施例中，设备接入管理服务接收控制设备的第一控制命令，该第一控制命令的用于指示第一网络设备执行第一控制体对应的操作。例如，设备接入管理服务接收控制设备的旋转命令，该旋转命令携带设备标识和控制体，该旋转命令中的控制体用于指示该设备标识对应的摄像头以目标速度旋转目标角度。

406、服务器接收控制设备的第一控制命令，基于该第一网络设备的设备标识，确定该第一网络设备对应的第一数字孪生模型，该第一数字孪生模型记录标准数据格式和第一网络设备所支持的数据格式之间的第一对应关系。

其中，该第一数字孪生模型通过“@type”元素，记录标准数据格式和第一网络设备所支持的数据格式之间的第一对应关系。

在一些实施例中，设备接入管理服务响应于接收到控制设备的第一控制命令，基于上述步骤404中存储的设备标识与模型标识之间的对应关系，将与该第一网络设备的设备标识对应的模型标识所指示的数字孪生模型，确定为该第一网络设备对应的第一数字孪生模型，将该第一控制命令和该第一数字孪生模型的模型标识转发至模型引擎服务。

407、服务器基于该第一对应关系，将该第一控制体转换为该第一网络设备所支持的数据格式并发送至第一网络设备。

其中，该第一对应关系包含第一数字孪生模型的数字孪生元模型与标准模型的标准元模型之间的对应关系，该标准模型可以为标准元模型组，该标准模型可以包括一个或多个标准元模型。相应地，服务器基于第一数字孪生模型的数字孪生元模型与标准模型的标准元模型之间的对应关系，确定该第一网络设备的上位设备类型的一个或多个标准元模型，基于第一数字孪生模型和上位设备类型的一个或多个标准元模型，将第一控制体转换为第一网络设备所支持的数据格式并发送至该第一网络设备。

在一些实施例中，模型引擎服务对该第一数字孪生模型进行解析，确定该第一数字孪生模型中命令元模型的“@type”元素对应的元模型标识对应的标准命令元模型，将该标准命令元模型确定为该第一网络设备的上位设备类型对应的标准元模型，模型引擎服务基于该第一数字孪生模型中命令元模型描述的数据格式、该标准命令元模型描述的标准数据格式，以及该命令元模型中每个参数与标准命令元模型中每个参数之间的对应关系，将该第一控制体从标准数据格式转换为该第一网络设备所支持的数据格式，并将转换后的第一控制指令发送至设备接入管理服务，该转换后的第一控制指令携带转换后的第一控制体和第一网络设备的设备标识，设备接入管理服务基于该设备标识，将该转换后的第一控制体发送至该第一网络设备，或者，将转后的第一控制指令发送至该第一网络设备。

示例性地,如图9所示,以向A01型号摄像头发送旋转命令为例进行说明。模型引擎服务接收rotateCommand命令,基于该A01型号摄像头对应的数字孪生模型的命令元模型中每个参数与对应的标准命令元模型中每个参数之间的对应关系,将rotateCommand命令的控制体中的3个参数进行格式转换,将horizontalAngle的格式转换为ha的格式,将verticalAngle的格式转换为va的格式,将speed的格式转换为speed的格式,得到commandA01命令的控制体,也即是,得到A01型号摄像头所支持的数据格式的控制体,将该commandA01命令的控制体和A01型号摄像头的设备标识发送至设备接入管理服务,设备接入管理服务将该commandA01命令的控制体发送至A01型号摄像头。

408、第一网络设备通过服务器向控制设备发送状态数据和第一网络设备的设备标识,该状态数据为该第一网络设备所支持的数据格式。

其中,该状态数据用于表示第一网络设备的状态,例如,若该第一网络设备为摄像头,该状态数据用于表示摄像头的旋转角度、剩余电量、温度等。

在一些实施例中,第一网络设备通过服务器向控制设备发送该第一网络设备的状态数据,该状态数据为该第一网络设备所支持的数据格式,例如,以该第一网络设备是任一型号的摄像头为例,如图8所示,不同型号的摄像头发送的状态数据的数据格式不同,A01型号摄像头发送的状态数据为propertyA01,B01型号摄像头发送的状态数据为propertyB01,C01型号摄像头发送的状态数据为propertyC01。

409、服务器接收第一网络设备的设备标识和状态数据,基于该第一网络设备的设备标识,确定该第一网络设备对应的第一数字孪生模型。

在一些实施例中,设备接入管理服务响应于接收到第一网络设备的设备标识和状态数据,基于与步骤406同理的方法,确定该第一网络设备对应的第一数字孪生模型,在此不再赘述。

410、服务器基于该第一对应关系,将该状态数据转换为标准数据格式,将转换后的状态数据发送至控制设备。

在一些实施例中,模型引擎服务基于与步骤407同理的方法,将该状态数据从该第一网络设备所支持的数据格式转换为标准数据格式,将转换后的状态数据进行存储并发送至设备接入管理服务,设备接入管理服务将该转换后的状态数据发送至控制设备。

需要说明的是,服务器能够用于对多个网络设备的控制命令进行格式转换,从而对该多个网络设备进行控制,对于相同上位设备类型下多个不同型号的网络设备,该多个不同型号的网络设备对应于不同的数字孪生模型。

在一些实施例中,服务器能够基于第二网络设备的第二数字孪生模型中记录的对应关系,对第二控制命令进行格式转换,从而实现对第二网络设备的控制,该第二网络设备的型号与第一网络设备的型号不同,该第二网络设备的上位设备类型与第一网络设备的上位设备类型相同。相应地,本申请实施例还包括:服务器获取第二数字孪生模型,将该第二数字孪生模型和对应的标准模型相关联,该第二网络设备对应的标准模型与第一网络设备对应的标准模型相同,该第二数字孪生模型描述该第二网络设备所支持的数据格式;接收控制设备的第二控制命令,第二控制命令携带第二网络设备的设备标识和第二控制体,该第二控制命令用于指示第二网络设备执行第二控制体对应的操作,该第二控制体符合标准数据格式;基于该第二网络设备的设备标识,确定第二网络设备对应的第二数字孪生模型,第二数字孪生模型记录了标准数据格式和第二网络设备所支持的数据格式之间的第二对应关系;基于该第二对应

关系，将第二控制体转换为第二网络设备所支持的数据格式并发送至第二网络设备。上述过程的具体实现方式与步骤 402 至步骤 407 同理，在此不再赘述。

需要说明的是，上述实施例是以控制设备和网络设备之间进行控制命令和状态数据的交互为例进行说明。在一些实施例中，本申请所提供的方法可以仅包括控制设备基于服务器向网络设备发送控制命令，或，仅包括网络设备基于服务器向控制设备发送状态数据。

本申请实施例所提供的设备控制方法，对于任一类型的网络设备，该类型的网络设备包括多种不同的型号，通过该类型下任一型号的网络设备的数字孪生模型中记录的该类型网络设备的数据格式与该型号网络设备所支持的数据格式之间的对应关系，对控制设备发送的控制命令的控制体进行格式转换。该方法使控制设备能够通过统一的标准数据格式的控制命令的控制体，控制不同型号的网络设备，且无需将不同型号的网络设备支持的数据格式与标准数据格式进行适配，网络设备就能够识别控制设备发送的控制命令的控制体，实现了网络设备与物联网应用之间的解耦，从而简化了网络设备的控制过程。

在上述实施例中，在创建数字孪生模型时，是通过“@type”来指示数字孪生模型中的数字孪生元模型与标准模型中的标准元模型之间的对应关系的；可以理解的是，还可以通过其他预先设定的标识符来指示上述对应关系，本申请对此不作限定。

图 10 是本申请实施例提供的一种设备控制装置的结构示意图，该设备控制装置用于执行上述设备控制方法执行时的步骤，参见图 10，该设备控制装置包括：接收模块 1001、确定模块 1002 和转换模块 1003。

接收模块 1001，用于接收控制设备的第一控制命令，该第一控制命令携带第一网络设备的设备标识和第一控制体，该第一控制命令用于指示该第一网络设备执行该第一控制体对应的操作，该第一控制体符合标准数据格式，该标准数据格式由该第一网络设备的上位设备类型对应的标准模型描述；

确定模块 1002，用于基于该第一网络设备的设备标识，确定该第一网络设备对应的第一数字孪生模型，该第一数字孪生模型描述该第一网络设备所支持的数据格式，该第一数字孪生模型记录了该标准数据格式和该第一网络设备所支持的数据格式之间的第一对应关系；

转换模块 1003，用于基于该第一对应关系，将该第一控制体转换为该第一网络设备所支持的数据格式并发送至该第一网络设备。

在一些实施例中，该接收模块 1001，用于接收控制设备的第二控制命令，该第二控制命令携带第二网络设备的设备标识和第二控制体，该第二控制命令用于指示该第二网络设备执行该第二控制体对应的操作，该第二控制体符合该标准数据格式，该第一网络设备的型号与该第二网络设备的型号不同，该第一网络设备的上位设备类型与该第二网络设备的上位设备类型相同；

该确定模块 1002，用于基于该第二网络设备的设备标识，确定该第二网络设备对应的第二数字孪生模型，该第二数字孪生模型描述该第二网络设备所支持的数据格式，该第二数字孪生模型记录了该标准数据格式和该第二网络设备所支持的数据格式之间的第二对应关系；

该转换模块 1003，用于基于该第二对应关系，将该第二控制体转换为该第二网络设备所支持的数据格式并发送至该第二网络设备。

在一些实施例中，该接收模块 1001，用于接收该第一网络设备的状态数据和该第一网络设备的设备标识，该状态数据为该第一网络设备所支持的数据格式；

该转换模块 1003，用于基于该第一对应关系，将该状态数据转换为该标准数据格式，将转换后的状态数据发送至该控制设备。

在一些实施例中，该接收模块 1001，用于接收该第一网络设备的设备接入请求，该设备接入请求携带该第一网络设备的型号和该第一网络设备的设备标识；

该确定模块 1002，用于根据该第一网络设备的型号，确定该第一网络设备的型号对应的第一数字孪生模型；

该装置还包括：建立模块，用于建立该第一网络设备的设备标识和该第一数字孪生模型之间的第三对应关系。

在一些实施例中，该第一对应关系包含该第一数字孪生模型的数字孪生元模型与该标准模型的标准元模型之间的对应关系；

该转换模块 1003，用于基于该第一数字孪生模型的数字孪生元模型与该标准模型的标准元模型之间的对应关系，确定该上位设备类型的一个或多个标准元模型；基于该第一数字孪生模型和该上位设备类型的一个或多个标准元模型，将该第一控制体转换为该第一网络设备所支持的数据格式。

在一些实施例中，该第一数字孪生模型包括一个或多个数字孪生元模型，该数字孪生元模型包括属性元模型、命令元模型、事件元模型、数据元模型、组件元模型和关系元模型中的一种或多种。

需要说明的是：上述图 10 对应的实施例提供的设备控制装置在进行设备控制时，仅以上述各功能模块的划分进行举例说明，实际应用中，可以根据需要而将上述功能分配由不同的功能模块完成，即将装置的内部结构划分成不同的功能模块，以完成以上描述的全部或者部分功能。另外，上述实施例提供的设备控制装置与设备控制方法实施例属于同一构思，其具体实现过程详见方法实施例，这里不再赘述。

本申请中术语“至少一个”的含义是指一个或多个，本申请中术语“多个”的含义是指两个或两个以上，例如，多个标准元模型是指两个或两个以上的标准元模型。

还应理解，术语“如果”可被解释为意指“当...时”(“when”或“upon”)或“响应于确定”或“响应于检测到”。类似地，根据上下文，短语“如果确定...”或“如果检测到[所陈述的条件或事件]”可被解释为意指“在确定...时”或“响应于确定...”或“在检测到[所陈述的条件或事件]时”或“响应于检测到[所陈述的条件或事件]”。

以上描述，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到各种等效的修改或替换，这些修改或替换都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以权利要求的保护范围为准。

本领域普通技术人员可以理解实现上述实施例的全部或部分步骤可以通过硬件来完成，也可以通过程序来指令相关的硬件完成，所述的程序可以存储于一种计算机可读存储介质中，上述提到的存储介质可以是只读存储器，磁盘或光盘等。

以上所述仅为本申请的可选实施例，并不用以限制本申请，凡在本申请的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本申请的保护范围之内。

权利要求书

1.一种设备控制方法，由服务器执行，其特征在于，所述方法包括：

接收控制设备的第一控制命令，所述第一控制命令携带第一网络设备的设备标识和第一控制体，所述第一控制命令用于指示所述第一网络设备执行所述第一控制体对应的操作，所述第一控制体符合标准数据格式，所述标准数据格式由所述第一网络设备的上位设备类型对应的标准模型描述；

基于所述第一网络设备的设备标识，确定所述第一网络设备对应的第一数字孪生模型，所述第一数字孪生模型描述所述第一网络设备所支持的数据格式，所述第一数字孪生模型记录了所述标准数据格式和所述第一网络设备所支持的数据格式之间的第一对应关系；

基于所述第一对应关系，将所述第一控制体转换为所述第一网络设备所支持的数据格式并发送至所述第一网络设备。

2.根据权利要求1所述的方法，其特征在于，所述方法还包括：

接收所述控制设备的第二控制命令，所述第二控制命令携带第二网络设备的设备标识和第二控制体，所述第二控制命令用于指示所述第二网络设备执行所述第二控制体对应的操作，所述第二控制体符合所述标准数据格式，所述第一网络设备的型号与所述第二网络设备的型号不同，所述第一网络设备的上位设备类型与所述第二网络设备的上位设备类型相同；

基于所述第二网络设备的设备标识，确定所述第二网络设备对应的第二数字孪生模型，所述第二数字孪生模型描述所述第二网络设备所支持的数据格式，所述第二数字孪生模型记录了所述标准数据格式和所述第二网络设备所支持的数据格式之间的第二对应关系；

基于所述第二对应关系，将所述第二控制体转换为所述第二网络设备所支持的数据格式并发送至所述第二网络设备。

3.根据权利要求1所述的方法，其特征在于，所述方法还包括：

接收所述第一网络设备的状态数据和所述第一网络设备的设备标识，所述状态数据为所述第一网络设备所支持的数据格式；

基于所述第一对应关系，将所述状态数据转换为所述标准数据格式，将转换后的状态数据发送至所述控制设备。

4.根据权利要求1所述的方法，其特征在于，所述接收控制设备的第一控制命令之前，所述方法还包括：

接收所述第一网络设备的设备接入请求，所述设备接入请求携带所述第一网络设备的型号和所述第一网络设备的设备标识；

根据所述第一网络设备的型号，确定所述第一网络设备的型号对应的第一数字孪生模型；建立所述第一网络设备的设备标识和所述第一数字孪生模型之间的第三对应关系。

5.根据权利要求4所述的方法，其特征在于，所述第一对应关系包含所述第一数字孪生

模型的数字孪生元模型与所述标准模型的标准元模型之间的对应关系；

所述基于所述第一对应关系，将所述第一控制体转换为所述第一网络设备所支持的数据格式，包括：

基于所述第一数字孪生模型的数字孪生元模型与所述标准模型的标准元模型之间的对应关系，确定所述上位设备类型的一个或多个标准元模型；

基于所述第一数字孪生模型和所述上位设备类型的一个或多个标准元模型，将所述第一控制体转换为所述第一网络设备所支持的数据格式。

6.根据权利要求1至5任一项所述的方法，其特征在于，所述第一数字孪生模型包括一个或多个数字孪生元模型，所述数字孪生元模型包括属性元模型、命令元模型、事件元模型、数据元模型、组件元模型和关系元模型中的一种或多种。

7.一种设备控制装置，其特征在于，所述装置包括：

接收模块，用于接收控制设备的第一控制命令，所述第一控制命令携带第一网络设备的设备标识和第一控制体，所述第一控制命令用于指示所述第一网络设备执行所述第一控制体对应的操作，所述第一控制体符合标准数据格式，所述标准数据格式由所述第一网络设备的上位设备类型对应的标准模型描述；

确定模块，用于基于所述第一网络设备的设备标识，确定所述第一网络设备对应的第一数字孪生模型，所述第一数字孪生模型描述所述第一网络设备所支持的数据格式，所述第一数字孪生模型记录了所述标准数据格式和所述第一网络设备所支持的数据格式之间的第一对应关系；

转换模块，用于基于所述第一对应关系，将所述第一控制体转换为所述第一网络设备所支持的数据格式并发送至所述第一网络设备。

8.根据权利要求7所述的装置，其特征在于，所述接收模块，用于接收所述控制设备的第二控制命令，所述第二控制命令携带第二网络设备的设备标识和第二控制体，所述第二控制命令用于指示所述第二网络设备执行所述第二控制体对应的操作，所述第二控制体符合所述标准数据格式，所述第一网络设备的型号与所述第二网络设备的型号不同，所述第一网络设备的上位设备类型与所述第二网络设备的上位设备类型相同；

所述确定模块，用于基于所述第二网络设备的设备标识，确定所述第二网络设备对应的第二数字孪生模型，所述第二数字孪生模型描述所述第二网络设备所支持的数据格式，所述第二数字孪生模型记录了所述标准数据格式和所述第二网络设备所支持的数据格式之间的第二对应关系；

所述转换模块，用于基于所述第二对应关系，将所述第二控制体转换为所述第二网络设备所支持的数据格式并发送至所述第二网络设备。

9.根据权利要求7所述的装置，其特征在于，所述接收模块，用于接收所述第一网络设备的状态数据和所述第一网络设备的设备标识，所述状态数据为所述第一网络设备所支持的数据格式；

所述转换模块,用于基于所述第一对应关系,将所述状态数据转换为所述标准数据格式,将转换后的状态数据发送至所述控制设备。

10.根据权利要求 7 所述的装置,其特征在于,所述接收模块,用于接收所述第一网络设备的设备接入请求,所述设备接入请求携带所述第一网络设备的型号和所述第一网络设备的设备标识;

所述确定模块,用于根据所述第一网络设备的型号,确定所述第一网络设备的型号对应的第一数字孪生模型;

所述装置还包括:建立模块,用于建立所述第一网络设备的设备标识和所述第一数字孪生模型之间的第三对应关系。

11.根据权利要求 10 所述的装置,其特征在于,所述第一对应关系包含所述第一数字孪生模型的数字孪生元模型与所述标准模型的标准元模型之间的对应关系;

所述转换模块,用于基于所述第一数字孪生模型的数字孪生元模型与所述标准模型的标准元模型之间的对应关系,确定所述上位设备类型的一个或多个标准元模型;基于所述第一数字孪生模型和所述上位设备类型的一个或多个标准元模型,将所述第一控制体转换为所述第一网络设备所支持的数据格式。

12.根据权利要求 7 至 11 任一项所述的装置,其特征在于,所述第一数字孪生模型包括一个或多个数字孪生元模型,所述数字孪生元模型包括属性元模型、命令元模型、事件元模型、数据元模型、组件元模型和关系元模型中的一种或多种。

13.一种服务器,其特征在于,所述服务器包括通信总线、至少一个网络接口、存储器以及至少一个处理器,所述通信总线用于在上述组件之间传送信息,所述至少一个网络接口用于与其它设备或通信网络通信,所述存储器保存有程序代码,所述至少一个处理器通过读取并执行所述存储器中保存的程序代码实现如权利要求 1-6 任一项所述的设备控制方法。

14.一种计算机可读存储介质,其特征在于,当所述计算机可读存储介质中的程序代码由服务器的处理器执行时,使得所述服务器能够执行如权利要求 1-6 任一项所述的设备控制方法。