说明书

应用编程接口的访问方法、装置、存储介质及计算机系统

技术领域

本说明书涉及计算机技术领域，尤其涉及一种应用编程接口的访问方法、装置、存储介质及计算机系统。

背景技术

当应用系统可以被客户端或其他系统调用时，通常需要暴露 API（Application Programming Interface，应用编程接口），这些 API 代表着应用系统的各种功能点。如果在两个系统、或客户端与系统之间，加上一个中介者来协调 API 的调用，使得进入系统的请求都经过它，这个中介者就是 API 网关（Application Programming Interface Gateway）。通过 API 网关，可以更好地管理和控制 API 的调用，提高系统的安全性和性能。但是，现有的通过 API 网关访问 API 的方式，仍然存在不能满足不同的用户对于 API 接口的差异性的访问要求的问题。

发明内容

本说明书实施例提供一种应用编程接口的访问方法、装置、存储介质及计算机系统，以部分解决上述现有技术存在的问题。

本说明书提供了一种应用编程接口的访问方法，通过应用编程接口网关执行，包括：接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系；根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

本说明书提供了一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，其特征在于，所述处理器执行所

述程序时实现如下步骤：接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系；根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

本说明书提供了一种应用编程接口的访问装置，所述装置包括：接收单元，用于接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系；发送单元，用于根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

本说明书提供了一种计算机可读非易失性存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述的应用编程接口的访问方法。

本说明书实施例采用的上述至少一个技术方案能够达到以下有益效果中的至少一项：使得不同的用户可以通过己方分别定义的用户字段，访问相同应用编程接口的效果，提高用户访问应用编程接口的用户体验。

附图说明

此处所说明的附图用来提供对本说明书的进一步理解，构成本说明书的一部分，本说明书的示意性实施例及其说明用于解释本说明书，并不构成对本说明书的不当限定。在附图中：

图 1 为一种应用接口访问方案的示意图；

图 2 为本说明书中一种应用编程接口的访问方法的示意图；

图 3 为本说明书中一种多模态数据处理方法的流程示意图；

图 4A 为本说明书中接受一个用户输入的用户字段的示意图；

图 4B 为本说明书中接受另一个用户输入的用户字段的示意图；

图 5A 为本说明书中接受一个用户选取的因子字段的示意图；

图 5B 为本说明书中接受另一个用户选取的因子字段的示意图；

图 6 为本说明书中一种获取因子字段的字段内容的流程示意图；

图 7 为本说明书中一种应用编程接口的访问装置的结构示意图；

图 8 为本说明书实施例提供的计算机系统的结构示意图。

具体实施方式

为使本说明书的目的、技术方案和优点更加清楚，下面将结合本说明书具体实施例及相应的附图对本说明书技术方案进行清楚、完整地描述。显然，所描述的实施例仅是本说明书一部分实施例，而不是全部的实施例。基于本说明书中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本说明书保护的范围。

如前所述，当应用系统可以被客户端或其他系统调用时，通常需要暴露 API（Application Programming Interface），这些 API 代表着应用系统的各种功能点。如果在两个系统或客户端与系统之间，加上一个中介者来协调 API 的调用，使得进入系统的请求可以都经过它，这个中介者就是 API 网关（Application Programming Interface Gateway）。图 1 为一种应用接口访问方案的示意图。如图 1 所示，例如各个希望访问服务端的不同 API 的客户端，都将其访问请求发送到 API 网关，由 API 网关将它们的访问请求转发给服务端的不同应用编程接口进行处理，并在接收到服务端发送的处理结果后，将处理结果返回给各个客户端。

通过 API 网关，可以更好地管理和控制 API 的调用，提高系统的安全性和性能。但是，现有的通过 API 网关访问 API 的方式，仍然存在难以满足不同的用户对于 API 接口的差异性的访问要求的问题。例如，当不同的用户在访问同一 API 的时候，希望获取其中不同字段的值，或者不同的用户希望 API 接口中的字段根据自己的定义设定名称。现有的通过 API 网关访问 API 的方案难以达到用户希望的上述技术效果。

为了解决上述问题，在本申请中，可以预先保存用户命名的用户字段与系

统内部的原始字段（或称为因子字段）的对应关系。图 2 为本说明书中一种应用编程接口的访问方法的示意图。如图 2 所示，例如当 API 接收到用户例如通过客户端发送的访问 API 的请求时，来自客户端的访问请求可以包括希望获取其字段内容的用户字段。API 网关可以根据预先保存的用户字段与因子字段的对应关系，将用户字段变为因子字段后，发送到 API 的服务端。在服务端得到因子字段的字段数据后，将因子字段替换为对应的用户字段，并将用户字段和字段数据返回客户端。

相对于传统的通过 API 的方案中，所有用户或使用方都访问统一的接口，接口中所有可访问字段都是设定的，对于各个用户均是无法变更的。从而，无法解决希望不同的用户访问 API 的时候，可以获取不同字段的值；或者不同的用户可以通过不同的己方希望的字段名称获取 API 中相同的字段的值的技术问题。本申请中提供的应用编程接口的访问方法中，通过预先保存 API 的用户字段和因子字段的对应关系。在 API 网关收到客户端发送的访问 API 的请求时，可以将请求中用户字段转换为因子字段，并通过因子字段从 API 服务端获取因子字段的字段内容。并且，将因子字段的字段内容作为用户字段的字段内容，发送回客户端。从而实际达到不同的用户可以通过己方定义的不同用户字段，访问相同的 API 的效果，提高用户访问 API 的用户体验。

此外，不同的用户还可以从相同的 API 中选择不同的字段，用于己方访问该 API。使得同一 API 面对不同的用户，可以具有不同的字段组合，从而满足了不同用户的差异化字段组合的访问要求，进一步提高用户访问 API 的用户体验。

以下结合附图，详细说明本说明书各实施例提供的技术方案。

图 3 为本说明书中一种业务预测方法的流程示意图，通过第一方执行，具体可包括以下步骤：

S300：接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系。应用编程接口（API），通常是指用于调用方访问并调用应用系统的各种功能的软件接口。在不同的实

施例中，目标应用编程接口可以是用于调用不同的服务系统的不同具体业务功能或系统功能的编程接口。在不同的实施例中，目标用户可以是不同的具体用户，例如可以是个体用户或属于目标团体的用户。

在接收到目标用户发送的针对目标应用编程接口的请求（第一请求）后，该请求中包括目标用户希望通过目标应用编程接口访问的一个或多个数据字段。数据字段，是指使用方希望通过访问 API 返回的数据结构中的各个组成部分。由于 API 通常用于在不同的软件应用程序（例如客户端应用和服务器端应用，或者不同的系统服务端应用）之间传输数据。这些数据通常是以结构化格式（例如，Jason 和 XML）返回的，并且包含多个字段，每个字段都代表特定的数据或信息。在本说明书中，这些字段也被称为应用编程接口的字段。因子字段，是指在 API 中包括的原始字段，也就是由 API 提供方预先定义的字段，例如 API 提供方通常可以设定了原始字段的名称或标识。在不同的实施例中，目标应用编程接口包括的因子字段可以不同。

用户字段，则可以是由目标用户设定的与因子字段匹配的字段。可以预先保存目标用户针对因子字段设定的匹配的用户字段，以及用户字段与因子字段之间的对应关系，或称关联关系。在不同的实施例中，获取和保存上述关联关系的具体方式的可以不同。在一个实施例中，可以获取目标用户输入的、与所述目标应用编程接口内设的因子字段对应的用户字段；保存所述因子字段和所述用户字段之间的对应关系。具体的，在一个实施例中，还可以向用户展示因子字段，然后接收用户输入的与因子字段匹配的用户字段。图 4A 为本说明书中接受一个用户输入的用户字段的示意图。如图 4A 所示，目标应用接口例如可以包括多个因子字段，例如包括因子字段 1、因子字段 2…等。可以向用户（例如为用户 A）展示这些因子字段，然后接收用户 A 输入的与各个因子字段分别匹配的用户字段，例如与因子字段 1 匹配的用户字段 a1、与因子字段 2 匹配的用户字段 a2…等。不同的用户可以针对目标应用编程接口中相同的字段设定匹配的不同的用户字段。图 4B 为本说明书中接受另一个用户输入的用户字段的示意图。如图 4B 所示，还可以向例如为用户 B 展示因子字段，并接收用户 B 输入的与各个因子字段分别匹配的用户字段，例如与因子字段 1 匹配的用户字段 b1、与因子字段 2 匹配的用户字段 b2…等。对于相同的因子字段，通

过针对不同的用户设定不同的用户字段，使得不同的用户，可以根据其希望的例如字段名称或标识，访问相同的 API 并获取 API 中字段的字段内容，提高其用户体验。在一个具体的实施例中，还可以在向用户展示因子字段时，具体的展示因子字段的名称、标识，以及字段描述等，便于用户确定其希望对应的用户字段。

在一个实施例中，预先保存的可以是目标应用编程接口的部分因子字段与匹配目标用户的用户字段之间的对应关系，所述部分因子字段由所述目标用户从目标应用编程接口的全部因子字段中确定出；所述部分因子字段中包括所述目标用户字段。图 5A 为本说明书中接受一个用户选取的因子字段的示意图。如图 5A 所示，目标应用接口例如可以包括多个因子字段，例如包括因子字段 1、因子字段 2...因子字段 10。可以向用户（例如为用户 A）展示全部因子字段，然后接收用户 A 选择的例如因子字段 1~因子字段 5，作为目标应用编程接口针对的用户 A 的因子字段。不同的用户可以针对目标应用编程接口选择不同的因子字段集合。图 5B 为本说明书中接受另一个用户选取的因子字段的示意图。在如图 5B 所示，可以向用户（例如为用户 B）展示与图 5A 中相同的全部因子字段，然后接收用户 B 选择的例如因子字段 1、因子字段 2、因子字段 6、因子字段 7，作为目标应用编程接口针对用户 B 的因子字段。在一个具体的实施例中，还可以在向用户展示因子字段集合时，具体的展示各个因子字段之间的关联，便于用户根据不同因子字段的相互关系，确定希望保留的因子字段。对于相同的 API，通过针对不同的用户设定不同的因子字段集合，使得不同的用户可以根据需要的字段集合，访问 API 并获取所需字段的字段内容，而无需获取 API 中其不需要的字段内容。不仅提高了用户的体验，也减少了数据处理消耗的计算资源、以及数据传输消耗的网络资源。

在一些场景中，根据目标应用编程接口对应的业务，目标应用编程接口的全部因子字段可以包括必选因子字段（即访问该目标应用编程接口必须获取的因子字段）和可选因子字段（即访问该目标应用编程接口可以获取、也可以不获取的因子字段）。在这种场景下，用户选取的因子字段可以包括全部必选因子字段、以及全部或部分可选因子字段。例如图 5A 和图 5B 中所示，例如因子字段 1 和因子字段 2 为必选字段，因此用户 A 和用户 B 均选择它们作为己方

使用的因子字段。因子字段3~因子字段10为可选字段，因此用户A可以仅选择了其中的因子字段3、因子字段4、因子字段5作为己方使用的因子字段，用户B可以仅选择了其中的因子字段6、因子字段7作为己方使用的因子字段。

应用编程接口网关，是客户端或其它系统访问API的中介者，对于API的访问请求可以都经过它。在一个实施例中，本申请的应用编程接口的访问方法可以通过应用编程接口网关执行。在一些场景中，应用编程接口网关可以用于协调例如针对多种业务的多个应用编程接口的访问。因此，在一个实施例中，应用编程接口网关可以用于接收针对多个应用编程接口的请求，多个应用编程接口中可以包括目标应用编程接口。

S302：根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

在不同的实施例中，目标应用编程的服务端可以是应用编程接口的服务端组件，或者是运行上述服务端组件的服务器。在不同的实施例中，运行服务端组件的服务器的具体类型也可以不同。在一个的实施例中，例如可以是运行物理服务器、逻辑服务器或云服务器中的一种或多种。在另一个实施例中，目标应用编程的服务端还可以是运行服务端组件的上述服务器的集群。

进一步的，上述图3所述的S302步骤，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，可具体如图4所示：S400：向目标应用编程接口对应的服务端发送第二请求，所述第二请求包括目标因子字段。由于在目标应用编程接口的服务端，数据通常通过因子字段进行组织或保存。因此，可以根据因子字段向服务端发出针对字段内容的请求。S402：接收所述服务端发送的目标因子字段的字段内容。在不同的具体实施例中，根据目标因子字段的不同，目标因子字段的具体字段内容可以不同，本说明书对此不做限制。

以上为本说明书实施例提供的应用编程接口的访问方法，基于同样的思路，本说明书还提供了相应的应用编程接口的访问装置。如图7所示，该应用编程接口的访问装置包括：获取模块702，用于接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用

于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系；构建模块 704，用于根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

为了描述的方便，描述以上装置时以功能分为各种模块分别描述。当然，在实施本说明书时可以把各模块的功能在同一个或多个软件和/或硬件中实现。

The specification also provides a computer-readable non-transitory storage medium, wherein the storage medium stores a computer program, which, when executed by a processor, can be used to perform one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Herein, the computer-readable non-transitory storage medium or media may include one or more semiconductor-based or other integrated circuits (ICs) (such, as for example, field-programmable gate arrays (FPGAs) or application-specific ICs (ASICs)), hard disk drives (HDDs), hybrid hard drives (HHDs), optical discs, optical disc drives (ODDs), magneto-optical discs, magneto-optical drives, floppy diskettes, floppy disk drives (FDDs), magnetic tapes, solid-state drives (SSDs), RAM-drives, SECURE DIGITAL cards or drives, any other suitable computer-readable non-transitory storage media, or any suitable combination of two or more of these, where appropriate. A computer-readable non-transitory storage medium may be volatile, non-volatile, or a combination of volatile and non-volatile, where appropriate.

The specification also provides a computer system. FIG. 8 illustrates an example computer system **800**. In particular embodiments, one or more computer systems **800** perform one or more steps of one or more methods described or illustrated herein. In particular embodiments, one or more computer systems **800** provide functionality described or illustrated herein. In particular embodiments, software running on one or more computer systems **800** performs one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Particular embodiments include one or more portions of one or more computer

systems **800**. Herein, reference to a computer system may encompass a computing device, and vice versa, where appropriate. Moreover, reference to a computer system may encompass one or more computer systems, where appropriate.

This disclosure contemplates any suitable number of computer systems **800**. This disclosure contemplates computer system **800** taking any suitable physical form. As example and not by way of limitation, computer system **800** may be an embedded computer system, a system-on-chip (SOC), a single-board computer system (SBC) (such as, for example, a computer-on-module (COM) or system-on-module (SOM)), a desktop computer system, a laptop or notebook computer system, an interactive kiosk, a mainframe, a mesh of computer systems, a mobile telephone, a personal digital assistant (PDA), a server, a tablet computer system, or a combination of two or more of these. Where appropriate, computer system **800** may include one or more computer systems **800**; be unitary or distributed; span multiple locations; span multiple machines; span multiple data centers; or reside in a cloud, which may include one or more cloud components in one or more networks. Where appropriate, one or more computer systems **800** may perform without substantial spatial or temporal limitation one or more steps of one or more methods described or illustrated herein. As an example and not by way of limitation, one or more computer systems **800** may perform in real time or in batch mode one or more steps of one or more methods described or illustrated herein. One or more computer systems **800** may perform at different times or at different locations one or more steps of one or more methods described or illustrated herein, where appropriate.

In particular embodiments, computer system **800** includes a processor **802**, memory **804**, storage **806**, an input/output (I/O) interface **808**, a communication interface **810**, and a bus **812**. Although this disclosure describes and illustrates a particular computer system having a particular number of particular components in a particular arrangement, this disclosure contemplates any suitable computer system having any suitable number of any suitable components in any suitable arrangement.

In particular embodiments, processor **802** includes hardware for executing

instructions, such as those making up a computer program. As an example and not by way of limitation, to execute instructions, processor **802** may retrieve (or fetch) the instructions from an internal register, an internal cache, memory **804**, or storage **806**; decode and execute them; and then write one or more results to an internal register, an internal cache, memory **804**, or storage **806**. In particular embodiments, processor **802** may include one or more internal caches for data, instructions, or addresses. This disclosure contemplates processor **802** including any suitable number of any suitable internal caches, where appropriate. As an example and not by way of limitation, processor **802** may include one or more instruction caches, one or more data caches, and one or more translation lookaside buffers (TLB s). Instructions in the instruction caches may be copies of instructions in memory **804** or storage **806**, and the instruction caches may speed up retrieval of those instructions by processor **802**. Data in the data caches may be copies of data of **80** in memory **804** or storage **806** for instructions executing at processor **802** to operate on; the results of previous instructions executed at processor **802** for access by subsequent instructions executing at processor **802** or for writing to memory **804** or storage **806**; or other suitable data. The data caches may speed up read or write operations by processor **802**. The TLBs may speed up virtual-address translation for processor **802**. In particular embodiments, processor **802** may include one or more internal registers for data, instructions, or addresses. This disclosure contemplates processor **802** including any suitable number of any suitable internal registers, where appropriate. Where appropriate, processor **802** may include one or more arithmetic logic units (ALUs); be a multi-core processor; or include one or more processors **802**. Although this disclosure describes and illustrates a particular processor, this disclosure contemplates any suitable processor.

In particular embodiments, memory **804** includes main memory for storing instructions for processor **802** to execute or data for processor **802** to operate on. As an example and not by way of limitation, computer system **800** may load instructions from storage **806** or another source (such as, for example, another computer system

800) to memory **804**. Processor **802** may then load the instructions from memory **804** to an internal register or internal cache. To execute the instructions, processor **802** may retrieve the instructions from the internal register or internal cache and decode them. During or after execution of the instructions, processor **802** may write one or more results (which may be intermediate or final results) to the internal register or internal cache. Processor **802** may then write one or more of those results to memory **804**. In particular embodiments, processor **802** executes only instructions in one or more internal registers or internal caches or in memory **804** (as opposed to storage **806** or elsewhere) and operates only on data in one or more internal registers or internal caches or in memory **804** (as opposed to storage **806** or elsewhere). One or more memory buses (which may each include an address bus and a data bus) may couple processor **802** to memory **804**. Bus **812** may include one or more memory buses, as described below. In particular embodiments, one or more memory management units (MMUs) reside between processor **802** and memory **804** and facilitate accesses to memory **804** requested by processor **802**. In particular embodiments, memory **804** includes random access memory (RAM). This RAM may be volatile memory, where appropriate. Where appropriate, this RAM may be dynamic RAM (DRAM) or static RAM (SRAM). Moreover, where appropriate, this RAM may be single-ported or multi-ported RAM. This disclosure contemplates any suitable RAM. Memory **804** may include one or more memories **804**, where appropriate. Although this disclosure describes and illustrates particular memory, this disclosure contemplates any suitable memory.

In particular embodiments, storage **806** includes mass storage for data or instructions. As an example and not by way of limitation, storage **806** may include a hard disk drive (HDD), a floppy disk drive, flash memory, an optical disc, a magneto-optical disc, magnetic tape, or a Universal Serial Bus (USB) drive or a combination of two or more of these. Storage **806** may include removable or non-removable (or fixed) media, where appropriate. Storage **806** may be internal or external to computer system **800**, where appropriate. In particular embodiments, storage **806** is non-

volatile, solid-state memory. In particular embodiments, storage **806** includes read-only memory (ROM). Where appropriate, this ROM may be mask-programmed ROM, programmable ROM (PROM), erasable PROM (EPROM), electrically erasable PROM (EEPROM), electrically alterable ROM (EAROM), or flash memory or a combination of two or more of these. This disclosure contemplates mass storage **806** taking any suitable physical form. Storage **806** may include one or more storage control units facilitating communication between processor **802** and storage **806**, where appropriate. Where appropriate, storage **806** may include one or more storages **806**. Although this disclosure describes and illustrates particular storage, this disclosure contemplates any suitable storage.

In particular embodiments, I/O interface **808** includes hardware, software, or both, providing one or more interfaces for communication between computer system **800** and one or more I/O devices. Computer system **800** may include one or more of these I/O devices, where appropriate. One or more of these I/O devices may enable communication between a person and computer system **800**. As an example and not by way of limitation, an I/O device may include a keyboard, keypad, microphone, monitor, mouse, printer, scanner, speaker, still camera, stylus, tablet, touch screen, trackball, video camera, another suitable I/O device or a combination of two or more of these. An I/O device may include one or more sensors. This disclosure contemplates any suitable I/O devices and any suitable I/O interfaces **808** for them. Where appropriate, I/O interface **808** may include one or more device or software drivers enabling processor **802** to drive one or more of these I/O devices. I/O interface **808** may include one or more I/O interfaces **808**, where appropriate. Although this disclosure describes and illustrates a particular I/O interface, this disclosure contemplates any suitable I/O interface.

In particular embodiments, communication interface **810** includes hardware, software, or both providing one or more interfaces for communication (such as, for example, packet-based communication) between computer system **800** and one or more other computer systems **800** or one or more networks. As an example and not

by way of limitation, communication interface **810** may include a network interface controller (NIC) or network adapter for communicating with an Ethernet or other wire-based network or a wireless NIC (WNIC) or wireless adapter for communicating with a wireless network, such as a WI-FI network. This disclosure contemplates any suitable network and any suitable communication interface **810** for it. As an example and not by way of limitation, computer system **800** may communicate with an ad hoc network, a personal area network (PAN), a local area network (LAN), a wide area network (WAN), a metropolitan area network (MAN), or one or more portions of the Internet or a combination of two or more of these. One or more portions of one or more of these networks may be wired or wireless. As an example, computer system **800** may communicate with a wireless PAN (WPAN) (such as, for example, a BLUETOOTH WPAN), a WI-FI network, a WI-MAX network, a cellular telephone network (such as, for example, a Global System for Mobile Communications (GSM) network), or other suitable wireless network or a combination of two or more of these. Computer system **800** may include any suitable communication interface **810** for any of these networks, where appropriate. Communication interface **810** may include one or more communication interfaces **810**, where appropriate. Although this disclosure describes and illustrates a particular communication interface, this disclosure contemplates any suitable communication interface.

In particular embodiments, bus **812** includes hardware, software, or both coupling components of computer system **800** to each other. As an example and not by way of limitation, bus **812** may include an Accelerated Graphics Port (AGP) or other graphics bus, an Enhanced Industry Standard Architecture (EISA) bus, a front-side bus (FSB), a HYPERTRANSPORT (HT) interconnect, an Industry Standard Architecture (ISA) bus, an INFINIBAND interconnect, a low-pin-count (LPC) bus, a memory bus, a Micro Channel Architecture (MCA) bus, a Peripheral Component Interconnect (PCI) bus, a PCI-Express (PCIe) bus, a serial advanced technology attachment (SATA) bus, a Video Electronics Standards Association local (VLB) bus, or another suitable bus or a combination of two or more of these. Bus **812** may include

one or more buses **812**, where appropriate. Although this disclosure describes and illustrates a particular bus, this disclosure contemplates any suitable bus or interconnect.

还需要说明的是，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

本说明书中的各个实施例均采用递进的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。尤其，对于系统实施例而言，由于其基本相似于方法实施例，所以描述的比较简单，相关之处参见方法实施例的部分说明即可。

以上所述仅为本说明书的实施例而已，并不用于限制本说明书。对于本领域技术人员来说，本说明书可以有各种更改和变化。凡在本说明书的精神和原理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权利要求范围之内。

权利要求书

1、一种应用编程接口的访问方法，所述方法包括：

接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系；

根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

2、如权利要求 1 所述的方法，其中，根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，包括：

向目标应用编程接口对应的服务端发送第二请求，所述第二请求包括目标因子字段；

接收所述服务端发送的目标因子字段的字段内容。

3、如权利要求 1 所述的方法，其中，预先保存的目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系，包括：目标应用编程接口的部分因子字段与匹配目标用户的用户字段之间的对应关系，所述部分因子字段由所述目标用户从目标应用编程接口的全部因子字段中确定出；

所述部分因子字段中包括所述目标用户字段。

4、如权利要求 3 所述的方法，其中，目标应用编程接口的全部因子字段包括必选因子字段和可选因子字段；所述部分因子字段包括所述必选因子字段、以及全部或部分可选因子字段。

5、如权利要求 1 所述的方法，其中，目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系，通过以下方式预先保存：

获取目标用户输入的、与所述目标应用编程接口内设的因子字段对应的用户字段；

保存所述因子字段和所述用户字段之间的对应关系。

6、如权利要求 1 所述的方法，其中，所述目标用户包括个体用户或属于目标团体的用户。

7、如权利要求 1 所述的方法，通过应用编程接口网关执行。

8、如权利要求 7 所述的方法，其中，所述应用编程接口网关用于接收针对多个应用编程接口的请求，所述多个应用编程接口中包括所述目标应用编程接口。

9、一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，其特征在于，所述处理器执行所述程序时实现如下步骤：

接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系；

根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

10、如权利要求 9 所述的计算机系统，其中，根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，包括：

向目标应用编程接口对应的服务端发送第二请求，所述第二请求包括目标因子字段；接收所述服务端发送的目标因子字段的字段内容。

11、如权利要求 9 所述的计算机系统，其中，预先保存的目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系，包括：目标应用编程接口的部分因子字段与匹配目标用户的用户字段之间的对应关系，所述部分因子字段由所述目标用户从目标应用编程接口的全部因子字段中确定出；

所述部分因子字段中包括所述目标用户字段。

12、如权利要求 11 所述的计算机系统，其中，目标应用编程接口的全部因子字段包括必选因子字段和可选因子字段；所述部分因子字段包括所述必选因子字段、以及全部或部分可选因子字段。

13、如权利要求 9 所述的计算机系统，其中，目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系，通过以下方式预先保存：

获取目标用户输入的、与所述目标应用编程接口内设的因子字段对应的用

户字段；

保存所述因子字段和所述用户字段之间的对应关系。

14、如权利要求 9 所述的计算机系统，其中，所述目标用户包括属于目标团体的用户。

15、如权利要求 9 所述的计算机系统，用作为应用编程接口网关。

16、如权利要求 15 所述的计算机系统，其中，所述应用编程接口网关用于接收针对多个应用编程接口的请求，所述多个应用编程接口包括所述目标应用编程接口。

17、一种应用编程接口的访问装置，所述装置包括：

接收单元，用于接收目标用户发送的针对目标应用编程接口的第一请求，所述第一请求包含目标用户字段，根据预先保存的字段对应关系，确定所述目标用户字段对应的目标因子字段，所述字段对应关系用于指示目标应用编程接口内设的因子字段与匹配目标用户的用户字段之间的对应关系；

发送单元，用于根据目标因子字段，从目标应用编程接口对应的服务端获取目标因子字段的字段内容，将所述目标因子字段的字段内容作为目标用户字段的字段内容，发送给目标用户。

18、一种计算机可读非易失性存储介质，其特征在于，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述权利要求 1-8 任一项所述的方法。