Cloud Network Service Manager API

Technical White Paper V1.1

Table of Contents

Chapter 1 Background

Application Programing Interfaces (APIs) are pre-defined functions to enable applications and developers to access a set of routines via specific software or hardware without having to access the source codes or understand details of the internal work mechanism.

Chapter 2 Customer Benefits

Information of products and status can be constantly tracked and obtained via API invocation, digitalizing physical entities. Assets and business capabilities of an enterprise are unleashed via APIs because they can be searched, located, managed, and controlled in a digital way, thereby creating a liquid market featuring value exchange. As a result, a new API-enabled economic model is created in the manufacturing industry.

External software developers can understand and use APIs, which can be invoked by Internet software, mobile terminals, and browsers. Assets, data, services, and capabilities of an enterprise can be open APIs.

API economy features openness, overlapping, and value remolding. Enterprises digitalize their physical assets and open their data, services, and business capabilities through APIs to players in the ecosystem, achieving interconnection and communication of business capabilities and building a new coexistent value network.

API economy brings about disruptive influence because products and capabilities of an enterprise are exposed to more consumers when the enterprise implements Internet digitalization via APIs and builds an integrated API-based ecosystem.

Chapter 3 Technology Implementation
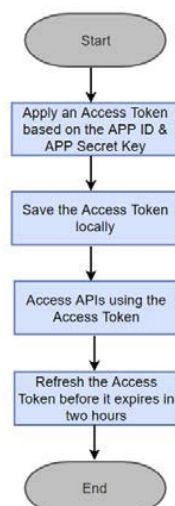
3.1 Overview

Our APIs compose of two parts, which are application layer APIs & raw data APIs. The following section describes the features of the APIs in terms of function description, input parameters and return values.

3.2 Workflow Infrastructure

The amount of data is relatively small while the business logic can be complex. This is due to the nature of statistical data.

- RESTful APIs. The APIs, based on the HTTP/HTTPS protocol and GET/POST method, are provided by the cloud network service manager for invoking. Each RESTful API matches a URL. The content is in JSON format. The APPs take initiative to invoke the API with the AccessToken mechanism for authority control.

- Push APIs. The cloud network service manager invokes specific APIs to push information.

Procedures of invoking application layer API:

```
                    ┌─────────┐
                    │  Start  │
                    └─────────┘
                         │
          ┌──────────────────────────────┐
          │ Apply an Access Token         │
          │ based on the APP ID &         │
          │ APP Secret Key                │
          └──────────────────────────────┘
                         │
          ┌──────────────────────────────┐
          │ Save the Access Token         │
          │ locally                       │
          └──────────────────────────────┘
                         │
          ┌──────────────────────────────┐
          │ Access APIs using the         │
          │ Access Token                  │
          └──────────────────────────────┘
                         │
          ┌──────────────────────────────┐
          │ Refresh the Access            │
          │ Token before it expires in    │
          │ two hours                     │
          └──────────────────────────────┘
                         │
                    ┌─────────┐
                    │   End   │
                    └─────────┘
```

 NOTE

The "lang" parameter can be included in each request to support multiple languages.

3.3 Advantages of the APIs

RESTful API

- The traffic is less owing to separated front-end and back-end.
- Security focuses on APIs. Security concerns are prevented as JSON is supported.
- All front-end languages are supported, including Android, iOS, and HTML5. The back-end only takes charge of data processing.
- Therefore, the front-end and back-end developers focus on their own development task and interact with each other with the help of the API documentation.
- Optimized server performance: Static pages used by the front-end can be obtained through Nginx. Therefore, the major server pressure is on APIs.

JSON Format Data

- JSON data is easy to read and write and consumes less bandwidth owing to the compressed format.
- JSON is easy to parse. Javascript client can read and search for JSON data using eval() without efforts.
- Multiple languages, involving ActionScript, C, C#, ColdFusion, Java, JavaScript, Perl, PHP, Python, and Ruby, are supported, facilitating parse at the server end.
- PHP-JSON and JSON-PHP can be used for direct data invocation of PHP-serialized programs. Objects and arrays of a PHP server can be directly converted to JSON data, facilitating access and retrieval of clients. Meantime, PEAR also supports JSON.
- JSON reduces the code development quantity and is easy to maintain.

Oauth Authentication and Authorization

- Multiple processes are supported so that each client can choose their specific process of obtaining access token. No third-party support, for example, a mobile device, is required and scalability is improved.
- Complex encryption algorithms are deleted. HTTPS ensures authentication security.
- The workload of developers is reduced as the authentication process requires two steps only.

Moreover, an access token upgrade scheme has been raised. A client obtains refresh_token that has a longer validity period while obtaining access_token. In this way, refresh_token can be used to update access_token.

5

Chapter 4 Application Scenario

After interconnecting with the cloud network service manager, the education platform enables integrated Wi-Fi management and control and data acquisition in the education system. The education area is classified into three-layer areas managed on the platform, which helps take statistics of flow change, personnel distribution, and Internet usage in each area and properly optimize network resources.