

【公報種別】特許法第 17 条の 2 の規定による補正の掲載

【部門区分】第 6 部門第 3 区分

【発行日】令和 3 年 10 月 7 日 (2021.10.7)

【公表番号】特表 2020-536320 (P2020-536320A)

【公表日】令和 2 年 12 月 10 日 (2020.12.10)

【年通号数】公開・登録公報 2020-050

【出願番号】特願 2020-518519 (P2020-518519)

【国際特許分類】

G 0 6 F 8/60 (2018.01)

G 0 6 F 9/455 (2006.01)

【F I】

G 0 6 F 8/60

G 0 6 F 9/455 1 5 0

【手続補正書】

【提出日】令和 3 年 8 月 25 日 (2021.8.25)

【手続補正 1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項 1】

コンテナ環境内の登録されたサービスエンドポイントに対するアプリケーションプログラミングインターフェイス (API) 関数を与える方法であって、前記方法は、

API レジストリにおいて API 定義を受けるステップを含み、前記 API 定義は、

コンテナ環境にデプロイされたコンテナにカプセル化された第 1 のサービスのエンドポイントと、

1 つ以上の API 関数とを含み、前記方法はさらに、

前記 API レジストリが、前記 1 つ以上の API 関数と前記サービスのエンドポイントとの間のバインディングを作成するステップと、

前記 API レジストリが、前記第 1 のサービスの使用を求める要求を第 2 のサービスから受けるステップと、

前記 API レジストリが、前記 1 つ以上の API 関数を前記第 2 のサービスに与えるステップとを含む、方法。

【請求項 2】

前記 1 つ以上の API 関数を含む前記第 2 のサービスのためのクライアントライブラリを生成するステップをさらに含む、請求項 1 に記載の方法。

【請求項 3】

前記クライアントライブラリは、前記 API 関数を実現するメンバ関数とともにオブジェクトクラスを含む、請求項 2 に記載の方法。

【請求項 4】

前記クライアントライブラリは、前記第 1 のサービスのエンドポイントに対するダイレクトコールを含む、請求項 2 または 3 に記載の方法。

【請求項 5】

前記クライアントライブラリは、前記 1 つ以上の API 関数のうちの 1 つが前記第 2 のサービスによってコールされたときに前記第 1 のサービスのエンドポイントを実行時にアップデートするコードを含む、請求項 2 から 4 のいずれか 1 項に記載の方法。

【請求項 6】

前記クライアントライブラリは、前記 1 つ以上の A P I 関数の 1 つ以上のパラメータとして受けたデータを、前記第 1 のサービスのエンドポイントに対する H T T P コールに再フォーマットする、請求項 2 から 5 のいずれか 1 項に記載の方法。

【請求項 7】

前記クライアントライブラリは、前記第 1 のサービスのエンドポイントの前記 H T T P コールに加えるための追加情報を第 3 のサービスに要求する、請求項 6 に記載の方法。

【請求項 8】

前記クライアントライブラリは、前記第 1 のサービスに対する失敗コールをリトライするコードを含む、請求項 2 から 7 のいずれか 1 項に記載の方法。

【請求項 9】

前記クライアントライブラリは、結果値を前記第 1 のサービスの結果セットから抽出し前記結果値を前記 1 つ以上の A P I 関数に対するリターン値としてパッケージングするコードを含む、請求項 2 から 8 のいずれか 1 項に記載の方法。

【請求項 10】

1 つ以上のプロセッサによって実行されると前記 1 つ以上のプロセッサに動作を実行させる命令を含むプログラムであって、前記動作は、

A P I レジストリにおいて A P I 定義を受けることを含み、前記 A P I 定義は、

コンテナ環境にデプロイされたコンテナにカプセル化された第 1 のサービスのエンドポイントと、

1 つ以上の A P I 関数とを含み、前記動作はさらに、

前記 A P I レジストリが、前記 1 つ以上の A P I 関数と前記サービスのエンドポイントとの間のバインディングを作成することと、

前記 A P I レジストリが、前記第 1 のサービスの使用を求める要求を第 2 のサービスから受けることと、

前記 A P I レジストリが、前記 1 つ以上の A P I 関数を前記第 2 のサービスに与えることとを含む、プログラム。

【請求項 11】

前記第 1 のサービスのエンドポイントは、

I P アドレスと、

ポート番号とを含む、請求項 10 に記載のプログラム。

【請求項 12】

前記コンテナ環境は、コンテナにカプセル化された複数のサービスを含む、請求項 10 または 11 に記載のプログラム。

【請求項 13】

前記 A P I レジストリは、前記コンテナ環境内のコンテナにカプセル化されたサービスとしてデプロイされる、請求項 10 から 12 のいずれか 1 項に記載のプログラム。

【請求項 14】

前記 A P I レジストリは、

統合開発環境 ( I D E ) において開発中のサービスと、

前記コンテナ環境において既にデプロイされているサービスとが、利用できる、請求項 10 から 13 のいずれか 1 項に記載のプログラム。

【請求項 15】

前記動作はさらに、前記 A P I レジストリが、前記コンテナ環境にデプロイされているコンテナ化されたサービスの任意の利用できるエンドポイントを識別することを含む、請求項 10 から 14 のいずれか 1 項に記載のプログラム。

【請求項 16】

1 つ以上のプロセッサと、

1 つ以上のメモリデバイスとを備えるシステムであって、前記 1 つ以上のメモリデバイスは、前記 1 つ以上のプロセッサによって実行されると前記 1 つ以上のプロセッサに動作を実行させる命令を含み、前記動作は、

A P I レジストリにおいて A P I 定義を受けることを含み、前記 A P I 定義は、  
コンテナ環境にデプロイされたコンテナにカプセル化された第 1 のサービスのエンド  
ポイントと、

1 つ以上の A P I 関数とを含み、前記動作はさらに、

前記 A P I レジストリが、前記 1 つ以上の A P I 関数と前記サービスのエンドポイント  
との間のバインディングを作成することと、

前記 A P I レジストリが、前記第 1 のサービスの使用を求める要求を第 2 のサービスか  
ら受けることと、

前記 A P I レジストリが、前記 1 つ以上の A P I 関数を前記第 2 のサービスに与えるこ  
とを含む、システム。

【請求項 17】

前記動作はさらに、前記 A P I レジストリに登録されたすべてのサービスに対する A P  
I 関数のリストをユーザインターフェイスに表示させることを含む、請求項 16 に記載の  
システム。

【請求項 18】

前記第 1 のサービスの使用を求める前記第 2 のサービスからの要求は、前記ユーザイン  
ターフェイスに表示された前記 1 つ以上の A P I 関数のうちの 1 つの選択を受けること  
によって受ける、請求項 17 に記載のシステム。

【請求項 19】

前記ユーザインターフェイスはコマンドラインインターフェイスを含む、請求項 17 ま  
たは 18 に記載のシステム。

【請求項 20】

前記ユーザインターフェイスはグラフィカルユーザインターフェイスを含む、請求項 1  
7 または 18 に記載のシステム。