

サーバ・クライアント型による緯度経度解決モジュールの設計と構築 Implementation of Geocoding Module using a Client-Server Model

遠峰 隆史[†]
Takashi Tomine

齋藤 功[†]
Isao Saito

嶋津 恵子[†]
Keiko Shimazu

1. 背景

様々な地図情報サイトより地点名からその場所の地図を表示するサービスが提供されている。また、インターネット上で検索可能な情報と、その情報に関連する地理的な位置を地図上に表示するサービスが一般的になっている。

Google[1] では Google Maps API[2] を提供している。この API では、JavaScript を使って、自分の Web ページに Google Maps から取得した地図情報を表示したり、Google Maps と同じポップアップに自由にコンテンツや情報を加えたり、マーカーを使って地図上に印を付けたりすることができる。これにより、誰でも個人の Web サイトにおいて、地図を用いて様々な情報を提供する事が可能になった。

2. 問題点

前章で述べた Google Maps API を用いるためには、マーカーやポップアップなどを出力したい位置、表示したい地図の位置の緯度と経度の値を JavaScript に記述する。Google Maps API は、JavaScript に記述された緯度と経度の値を元に地図と加えられた情報を地図上に配置して出力する。API を使うユーザは、緯度と経度の値をあらかじめ調べ、その値を JavaScript を含むソースコードに記述しておかなければならない。しかし、ユーザにとって緯度と経度の値をあらかじめ調べておくのは困難である。

緯度と経度の値を調べるためのサービスである、Geocoding.jp[3] というサイトでは、緯度と経度の値を検索するための API[4] が提供されている。しかし、この API では 5 秒間に 2 度以上のアクセスをする事を控えるようアナウンスされている。そのため、動的に情報が生成されるコンテンツを作成する際、地図を表示するたびに Geocoding.jp の API にアクセスする事はできない。また、Geocoding.jp の API は Web からアクセスして利用するよう設計されているため、プログラムの内部においてこの API を利用するためには、http のアクセスをするための実装をしなければならず、利用するためにかかるコストは大きい。

このように、Google Maps API を利用する際に、地図上に示したい位置の緯度と経度の値をあらかじめ調べておかなければならない。そのため、コンテンツが登録され、コンテンツを動的に生成する前に、あらかじめ緯度と経度を検索し、Google Maps API を利用するサイトがあるホスト内に情報を保存しておく必要がある。

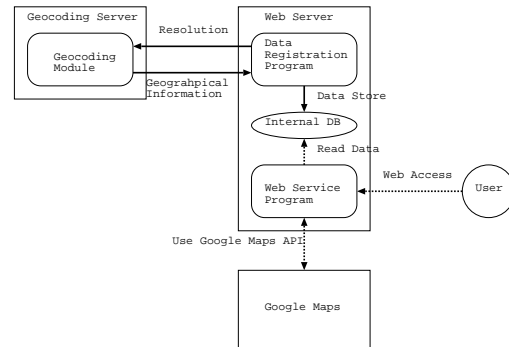


図 1: 設計概要

3. 目的

本研究では、前章で述べた問題を解決するため、コンテンツが登録され、コンテンツの情報がストアされると同時に、緯度と経度の値を同時に保存できるような仕組みを構築する事を目的とする。また、緯度と経度の値を解決する仕組みをサーバとして外部に設置する事により、解決された緯度と経度の情報を、様々なホストから利用可能とする。

4. 設計

前章で述べた目的を達成するための設計を行う。

4.1 設計概要

設計の概要を図 1 に示す。

本研究では、緯度経度解決を行うための仕組みを、Geocoding Module としてモジュール化する。このモジュールをインターネット上でアクセス可能なホストにサーバとして実装する。これにより、インターネット上の様々なホストから解決された緯度経度情報を利用可能にする。同時に、緯度経度情報を利用するホストにおいても、緯度経度情報を解決するにあたって発生する実装コストを、インターネットを用いたソケット通信を行う事によって軽減させる事が可能となる。

次節からは、Geocoding Module の設計の詳細について述べる。

4.2 Geocoding Module

Geocoding Module は、大きく分けて Interface 部、緯度経度情報ストア部、緯度経度解決部、初期設定部に分けられる。

4.2.1 Interface 部

ここでは、緯度経度情報を利用する他のホストからの緯度経度解決要求を受信する。要求はインターネットを用

[†]慶應義塾大学 デジタルメディア・コンテンツ統合研究機構, Research Institute for Digital Media and Content, Keio University

いたソケット通信で行う。そのため、Geocoding Module 自体を daemon として常駐させ、他のホストからの接続を待つ。この際に、接続を受け付ける port 番号は、後述の設定ファイルにて指定する。

緯度経度情報を利用するホストにおける実装コストを軽減させるため、緯度経度解決要求として受け付けるのは、緯度経度解決を行いたい住所、あるいは地名の 1 文字列とする。Geocoding Module からの応答は、要求に対応する緯度と経度の値のペアをコロンを用いて区切り、送信する。Geocoding Module において解決できなかった場合は、NULL 文字を送信する。応答の送信が完了すると、通信を終了する。このように、通信の内容を単純化する事により、1 処理あたりの処理時間を短縮する事が可能になる。

4.2.2 緯度経度情報ストア部

Geocoding Module は、Interface 部で受信した緯度経度解決要求に含まれる文字列が、Geocoding Module において管理されるデータベースの中に格納されているか、検索を行う。データベースの中に対応する情報がなかった場合には、緯度経度解決部において、緯度経度情報を取得し、データベースに格納する。これは、緯度経度解決部において利用される、外部の緯度経度情報提供サービスの負荷を下げるためである。

緯度経度解決部において緯度経度情報の解決が行えなかった場合には、検索に用いた文字列を緯度と経度の値が NULL のデータとしてデータベースに情報を格納する。

また、情報の正確さを保つために、データベースに情報を格納する際に登録した日時の情報も加え、初回登録時から一定期間たった情報については、再度、緯度経度解決部によって情報の再取得を行う。

4.2.3 緯度経度解決部

緯度経度解決部では、緯度経度情報ストア部からの情報に応じて、Interface 部で受信した緯度経度解決要求に含まれる文字列を、外部の緯度経度情報提供サービスへ問い合わせを行う。この際、問い合わせを行う緯度経度情報サービスは、後述する設定ファイルにて指定可能とする。

外部の緯度経度情報提供サービスは、複数の候補が当てはまる様な抽象的な文字列で検索を行うと、候補となる文字列を応答する場合がある。このように、一度で緯度経度情報を解決できない場合には、候補として挙げられた文字列を用いて、再帰的に検索を行い、緯度経度情報を取得する。

また、検索に用いた文字列に対応する緯度経度情報を取得できなかった場合には、エラーコードを緯度経度情報ストア部に返す。

4.2.4 初期設定部

Geocoding Module では、Module が daemon として起動する際に必要な、他のホストからの接続を待ち受ける port 番号の指定、および、緯度経度解決に用いる外部の緯度経度情報提供サービスを行っているサイトの情報を、設定ファイルに記述する。

Geocoding Module 起動の際に、初期設定部において設定ファイルを読み込み、他のホストからの接続の待ち受けを行うソケットの作成、緯度経度解決部の設定を行う。

5. 実装

本研究における、Geocoding Module の実装環境を表 1 に示す。

OS	Debian GNU/Linux (Sarge)
Linux Kernel	2.6.8
Language	Java
Database	PostgreSQL
Network Protocol	TCP/IP

表 1: 実装環境

6. まとめ

本研究では、Google Maps API を利用するために必要な、緯度と経度の値を解決するためのモジュールをネットワーク上に実装した。

本研究により、外部の緯度経度情報提供サービスの制約に囚われる事なく、動的な位置情報コンテンツの作成を、低い実装コストによって実現する事が可能になった。また、緯度と経度の値を解決するためのモジュールをネットワーク上に配置した事により、ネットワーク上の複数の Web サーバにおける位置情報コンテンツから Geocoding Module を利用することができ、動的な位置情報コンテンツを作成する際の実装コストを低減する事が可能になった。

参考文献

- [1] Google, "http://www.google.com/"
- [2] Google Maps API, "http://www.google.com/apis/maps/"
- [3] Geocoding.jp, "http://www.geocoding.jp/"
- [4] Geocoding.jp API, "http://www.geocoding.jp/api/"