

ジオコーディング API 提供システム 構築マニュアル

第 1.0.0 版

2015/12/28

国立研究開発法人 防災科学技術研究所

変更履歴

Version	変更日付	変更内容
1.0.0	2015/12/28	初版作成

目次

1	はじめに	1
2	セットアップ方法	1
2.1	サーバ環境	1
2.2	ネットワーク設定	2
2.3	動作環境インストール	2
2.3.1	PostgreSQL/PostGIS のインストール (yum 利用)	2
2.3.2	Java のインストール	3
2.3.3	Servlet エンジンのインストール	4
2.4	システムのセットアップ	5
2.4.1	データベースの作成と初期データの投入	5
2.4.2	Web アプリケーションの配置	7
2.4.3	動作確認	9
2.5	オフライン環境での利用について	9
3	ジオコーダとしての利用方法	10
3.1	URL によるジオコーディング	10
3.2	e コミマップでの利用	10
3.2.1	e コミマップの設定変更	10
3.2.2	動作確認	10
3.3	ジオコーダへのアクセス制限の設定	11

1 はじめに

このガイドでは、「ジオコーディング API 提供システム」のセットアップ方法、および利用方法等について説明します。

本システムは、PostGIS で構築されるデータベースから「e コミマップ」の住所検索機能で利用可能な API を提供します。なお、PostGIS にデータを格納するための画面はありませんので、コンソールからコマンドを使って格納する必要があります。

2 セットアップ方法

2.1 サーバ環境

本システムをセットアップするためには、以下のソフトウェア環境とそれが動作する十分なハードウェアスペックが必要になります。

ハードウェアのスペックは運用規模に応じて余裕のある構成にしてください。

・ソフトウェア環境

名称	ソフトウェア詳細
OS	Red Hat Enterprise Linux ES 5.3 相当以上 CentOS 5.3 相当以上 Fedora 10 相当 (※64bitOS 推奨)
Web サーバ	Apache 2.2.3 以降
Java ランタイム	Java SE 5 以降 (Java version 1.5.0 以降)
Servlet エンジン	Jetty 6.1.8 以降 もしくは Tomcat5.5 以降
GIS 用データベース	PostgreSQL 9.1 以降

・ハードウェア環境 (利用環境によって異なります)

名称	ソフトウェア詳細
CPU	Intel Core2Duo E6600 以上
メモリ	2GB DDR2 以上 (4GB 以上推奨)
ハードディスク空き容量	10GB 以上

2.2 ネットワーク設定

- SELinux の設定

以下を実行して SELinux を無効にします。

```
/usr/sbin/setenforce 0
```

再起動後も無効になるように /etc/selinux/config を編集します。

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - SELinux is fully disabled.
SELINUX=disabled
# SELINUXTYPE= type of policy in use. Possible values are:
#     targeted - Only targeted network daemons are protected.
#     strict - Full SELinux protection.
SELINUXTYPE=targeted
```

- iptables の設定

ポート 80 と 8080 が開放されていない場合は、以下を実行してポートを開放します。

```
/sbin/iptables -I INPUT -p tcp -m tcp --dport 80 --syn -j ACCEPT
/sbin/iptables -I INPUT -p tcp -m tcp --dport 8080 --syn -j ACCEPT
/etc/init.d/iptables save
```

2.3 動作環境インストール

インターネットに接続可能なサーバのコンソールから動作環境のインストールを行います。

なお、e コミマップをインストール済みのサーバに、本システムをインストールする場合は、以下の PostgreSQL、Java、Servlet エンジンのインストールは不要です。

2.3.1 PostgreSQL/PostGIS のインストール (yum 利用)

- PostgreSQL/PostGIS のインストールと初期化

yum を利用して、PostgreSQL/PostGIS のパッケージをインストールし、インストールが完了したら初期化を行います。

```
yum -y update postgresql91*
yum -y install postgresql91-*
yum -y install geos*
yum -y install proj*
yum -y install postgis
```

```
yum -y install postgres-*  
/sbin/service postgresql initdb  
psql -U postgres -d template1 -f /usr/share/pgsql/contrib/postgis-64.sql  
psql -U postgres -d template1 -f /usr/share/pgsql/contrib/spatial_ref_sys.sql
```

- PostgreSQL モジュールのインストール

PostgreSQL で全文検索機能を利用するため「pg_bigm」モジュールをインストールします。Pg_bigm モジュールは、<http://pgbigm.osdn.jp/> から rpm またはソースをダウンロードします。

rpm の場合、以下のようにインストールします。

```
rpm -ivh pg_bigm-1.1.20131122-1.pg91.el6.x86_64.rpm
```

- PostgreSQL 設定の変更

/var/lib/pgsql/data/pg_hba.conf の、以下の箇所を修正してください。

```
# TYPE DATABASE USER CIDR-ADDRESS METHOD  
# "local" is for Unix domain socket connections only  
local all all trust  
# IPv4 local connections:  
host all all 127.0.0.1/32 trust  
# IPv6 local connections:  
host all all ::1/128 trust
```

必要に応じて /var/lib/pgsql/data/postgresql.conf を修正してください。

```
listen_addresses = 'localhost'  
port = 5432  
max_connections = 256
```

- PostgreSQL サービスへの登録と起動

```
/sbin/service postgresql start  
(問題なく起動できたら)  
/sbin/chkconfig postgresql on
```

2.3.2 Java のインストール

- Java 実行環境のダウンロード (※以下は 2012 年 3 月 15 日現在の情報です)

① <http://www.java.com/ja/download/manual.jsp> の「Linux x64 RPM *」を選択し、ファイルをダウンロード。

(※ 32bitOS の場合は「Linux RPM (自己解凍ファイル)」を選択)

Linux			
	Linux RPM (自己解凍ファイル) ファイルサイズ: 20.1 MB	手順	今すぐチェック
	Linux (自己解凍ファイル) ファイルサイズ: 20.6 MB	手順	Java のインストールが終了したら、お使いの Web ブラウザを再起動して、Java が
	Linux x64 * ファイルサイズ: 20.2 MB	手順	正しくインストールされていることを確認してください。
	Linux x64 RPM * ファイルサイズ: 19.6 MB	手順	

* Java アプレットと Java Web Start を使用する場合は、32 ビットバージョンを使用してください。

・Java のインストール

コンソールで以下を実行

※Linux 64bitOS の場合

```
sh jre-6u30-linux-x64-rpm. bin
```

※Linux 32bitOS の場合

```
sh jre-6u30-linux-i586-rpm. bin
```

バージョンの確認

```
java -version
```

※バージョンが異なる場合は /usr/bin/java を /usr/java/default/bin/java のリンクに入れ替えてください。

```
mv /usr/bin/java /usr/bin/java.bak  
ln -s /usr/java/default/bin/java /usr/bin/java
```

2.3.3 Servlet エンジンのインストール

ご利用のサーバ環境に合わせて、Servlet エンジンをインストールしてください。

- ・Jetty 6.1.8 以降 : <http://jetty.codehaus.org/jetty/>
- ・Tomcat 5.5 以降 : <http://tomcat.apache.org/>

2.4 システムのセットアップ

セットアップに際して、インストーラ (`geocoder-installer-1.0.0.tar.gz`) を入手してください。パッケージ一式は、WinSCP などでインストール先のサーバ上の任意のディレクトリ上に UP し、解凍してください。解凍すると、`webapps`、`db`、`httpd` の 3 つのディレクトリが格納されています。

ここでは e コミマップをインストール済のサーバに、本システムを追加でセットアップする場合の手順を説明します。

手順の中の以下のディレクトリ名は、以下の略称で表記します。

- e コミマップインストール先ディレクトリ : *map_dir*
- 本システムのパッケージ解凍先ディレクトリ : *geocoder_pkg*

2.4.1 データベースの作成と初期データの投入

解凍したパッケージの中の `db` ディレクトリ以下に、`geocoder_1.0.0.sql` ファイルがあることを確認してください。ファイルを確認できたら、以下の操作を行ってください。

- ジオコーダ用データベースの作成

以下は、データベース名を `geocoder` とした場合の例です。

```
createdb -T template0 -U postgres -E UTF-8 geocoder
```

- テーブルの作成

```
cd geocoder_pkg/db  
psql -U postgres -f geocoder_1.0.0.sql geocoder
```

- 初期データの登録

上記操作によりデータベースには「`address_new_normalize`」テーブルが作成されます。このテーブルヘデータを登録することでジオコーディング結果が取得できるようになります。

「`address_new_normalize`」テーブル

カラム名	データ型	必須	説明
id	integer	○	連番 ID
level	smallint	○	住所レベル 0: 県名、1: 市名、2: 町名、3: 丁目、 4: 街区、5: 番地、6: 住居番号、 7: ランドマーク

カラム名	データ型	必須	説明
kencode	smallint		県コード
kenname	varchar(8)		県名
sicyocode	integer		市町村コード
sicyoname	varchar(64)		市町村名
sicyoyomi	varchar(64)		市町村名 (よみがな)
cyoazacode	smallint		町字コード
cyoazaname	varchar(64)		町字名
cyoazayomi	varchar(64)		町字名 (よみがな)
cyomecode	smallint		丁目コード
cyomename	varchar(64)		丁目名
gaiku	varchar(32)		街区
banchi	varchar(32)		番地
jyukyo	varchar(32)		住居番号
address	varchar(256)	○	表示文字列
normalize_address	varchar(256)	○	正規化された住所文字列 <ul style="list-style-type: none"> ・ 各種ハイフン→”-“ ・ 改行コード&スペース除去 ・ “高校”→”高等学校” ・ 末尾の”番地”, ”丁目”, ”号”を除去 ・ 数字に挟まれた”番”, ”番地”, ”番地の”, ”丁目”, ”丁目の”, ”の”→”-“
postcode	character(7)		郵便番号
the_geom	geometry	○	位置情報

SQLによりランドマークを登録する場合、以下のようにします。

```
# psql -U postgres geocoder
geocoder=# INSERT INTO address_new_normalize
( level, address, normalize_address, the_geom ) VALUES ( 7, '東京駅', '東京駅',
GeomFromText('POINT(139.766084 35.681382)',4326));
```

既存の GIS データ、住所と緯度経度の一覧表等によりデータベースに格納してください。

2.4.2 Web アプリケーションの配置

解凍したパッケージの中の `webapps` ディレクトリを、インストールした `Servlet` エンジンの Web アプリケーション配下に移動し、ブラウザで閲覧できるよう、設定を行ってください。

- `webapps` の配置

```
mkdir /map_dir/webapps/geocoder
cp -pr /geocoder_pkg/webapps/geocoder/* /map_dir/webapps/geocoder/
```

- Jetty の設定追加

`/map_dir/geoserver/etc/jetty.xml` に以下の内容を追加します。

```
<!-- for webapps/geocoder -->
<Call class="org.mortbay.jetty.webapp.WebAppContext" name="addWebApplications">
  <Arg><Ref id="contexts"/></Arg>
  <Arg><SystemProperty name="geocoder" default="."/>/webapps</Arg>
  <Arg><SystemProperty name="geocoder" default="."/>/etc/webdefault.xml</Arg>
  <Arg><Ref id="plusConfig"/></Arg>
  <Arg type="boolean">True</Arg>
  <Arg type="boolean">True</Arg>
</Call>
```

- apache の設定追加

```
cp -p /geocoder_pkg/httpd/geocoder.conf /etc/httpd/conf.d/
```

上記の `geocoder.conf` ファイルをコピー後、サーバの FQDN、e コミマップのインストール先ディレクトリ、e コミマップが使用している `GeoServer` の起動ポートに合わせて、以下の部分を変更します。以下は、サーバの FQDN が `server.domain.com`、`GeoServer` の起動ポートが `8080` の場合の例です。

```
ProxyPass /geocoder http://server.domain.com:8080/geocoder
ProxyPassReverse /geocoder http://server.domain.com:8080/geocoder
```

- `web.xml` の設定変更

`/map_dir/webapps/geocoder/WEB-INF/web.xml` の、以下の部分を変更します。以下は、`PostgreSQL` の起動ポートが `5432`、ジオコード用データベースの名前がジオコードデータベースへの接続ユーザ名が `postgres`、接続ユーザ名のパスワードが `pass`、の場合の例です。

```
<param-value>
```

```
jdbc:postgresql://localhost:5432/geocoder?user=postgres&password='pass'  
</param-value>
```

テンポラリディレクトリ (jp.ecom_plat.geocoder.TemporaryDirectory)、ログ出力先 (jp.ecom_plat.logCategoryRoot) の設定パラメータについても、必要に応じて適宜変更してください。

2.4.3 動作確認

データベースの作成と初期データの投入、Web アプリケーションの配置、が完了したら、`apache` と `e コミマップ` を再起動します。

```
service httpd restart
service ecommapp restart
```

※`e コミマップ` をインストール済みのサーバに本システムを追加でセットアップするのではなく、単独でセットアップする場合は、`apache` と利用している `Servlet` エンジンの再起動が必要になります。

再起動が完了したら、以下の URL にアクセスし、それぞれジオコーダデータのアップロードページが表示されれば、セットアップは完了です。サーバの FQDN に応じて、以下の URL でアクセスします。

- ・アップロードページ : <http://FQDN/geocoder/upload.jsp>

ジオコーダデータアップロード

ファイルタイプ、文字コードを指定して、ジオコーダデータをアップロードしてください。

ファイルタイプ:	大字・町丁目レベル位置参照情報 (ZIP)
文字コード:	Shift_JIS
ファイル:	<input type="text"/> 参照...

アップロード

2.5 オフライン環境での利用について

ジオコーダデータ提供サーバは、セットアップはインターネット環境で行うことを想定していますが、セットアップ完了後は、オフライン環境で運用することも可能です。

3 ジオコーダとしての利用方法

データの登録された本システムは、ジオコーダとして利用することができます。以下に、利用例とその方法を示します。

3.1 URLによるジオコーディング

ジオコーダデータ提供サーバへ、以下の URL でアクセスすると、ジオコーディングの結果を JSON 形式で受け取ることができます（ジオコーダの Web アプリケーションを <http://FQDN/geocoder/> に配置した場合の例）。

<http://FQDN/geocoder/json?address=> URL エンコードされた検索したい住所文字列

出力される JSON のフォーマットは、Google Geocoder API (V3) の JSON Output Formats と同様のフォーマットとなっています。

<https://developers.google.com/maps/documentation/geocoding/#JSON>

3.2 e コミマップでの利用

本システムの情報を e コミマップ上に設定することにより、e コミマップから本システムが提供するジオコーディング情報を利用することができます。

3.2.1 e コミマップの設定変更

e コミマップへ管理者としてログインし、管理画面の「設定」タブにある「Geocoder」で設定を行います。

本システムを e コミマップと同じサーバ内に構築した場合は、「サーバ内のジオコーダー」をチェックします。e コミマップと異なるサーバに構築した場合は、「外部のジオコーダー」をチェックし、その URL (<http://FQDN/geocoder/json?>) を入力します。

最後に同画面にある「○○の設定を更新」ボタンをクリックして設定完了です。

Geocoder	
<input type="radio"/>	サーバの設定にしたがう()
<input type="radio"/>	Googleジオコーダー
<input type="radio"/>	サーバ内のジオコーダー (/geocoder/json?)
<input type="radio"/>	外部のジオコーダー (http://example.com/geocoder/json?)
URL:	
標高値表示	<input checked="" type="checkbox"/> 表示する
MGRSコード表示	<input type="checkbox"/> 表示する

図. e コミマップ管理画面

3.2.2 動作確認

設定変更が完了したら、e コミマップを再起動した後、e コミマップ上で任意のマップを開き、住所移動を行って、地図の中心に表示されるかを確認してください。

3.3 ジオコーダへのアクセス制限の設定

ジオコーダデータ提供サーバをインターネット上に公開している場合、想定していないユーザからのリクエストが行われたり、データのアップロードページ、ダウンロードページにアクセスされ、勝手にデータを登録・削除されたりする可能性があります。

ジオコーダデータ提供サーバの利用者を限定したい場合は、アクセス元の IP アドレスにより、制限をかけることが可能です。

ジオコーダデータ提供サーバの/etc/httpd/conf.d/geocoder.conf ファイルに、以下の内容を追加することで、アクセス元の IP アドレスによるアクセス制限を設定することができます（ジオコーダデータ提供サーバの Web アプリケーションが <http://FQDN/geocoder/> に配置されていて、「server.domain.com」、「192.168.1.1」からのアクセスのみを許可する場合の例です）。

```
<Location /geocoder>
    Order deny,allow
    Deny from all
    Allow from server.domain.com
    Allow from 192.168.1.1
</Location>
```

アクセス制限の設定を行うと、データのアップロードページ、ダウンロードページへも、許可した IP アドレスからのしかアクセスできなくなりますので、ご注意ください。

アクセス制限の設定の反映には、apache の再起動が必要になります。

```
service httpd restart
```