QGIS Server 3.34 User Guide

QGIS Project

2024 年 04 月 24 日

目 次

第 1 章　はじめに **3**

第 2 章　最初のステップ **5**

2.1 Debian ベースのシステムへのインストール 5
2.1.1 Apache HTTP サーバー 7
2.1.2 NGINX HTTP Server 9
2.1.3 Xvfb 15
2.2 Windows へのインストール 16
2.3 プロジェクトを供給する 18
2.4 プロジェクトを設定する 19
2.4.1 WMS 機能 21
2.4.2 WMTS 機能 23
2.4.3 WFS/OAPIF 機能 24
2.4.4 WCS 機能 25
2.4.5 OWS を微調整する 26
2.5 サードパーティとの統合 26
2.5.1 QGIS Desktop との統合 26
2.5.2 MapProxy との統合 26
2.5.3 QWC2 との統合 27

第 3 章　サービス **29**

3.1 基本 29
3.1.1 SERVICE 30
3.1.2 REQUEST 30
3.1.3 MAP 30
3.1.4 FILE_NAME 31
3.1.5 短い名前 31
3.2 Web マップサービス (WMS) 32
3.2.1 GetCapabilities 33
3.2.2 GetMap 33
3.2.3 GetFeatureInfo 43
3.2.4 GetLegendGraphic 47
3.2.5 GetStyle(s) 64
3.2.6 DescribeLayer 65
3.2.7 GetPrint 66
3.2.8 GetProjectSettings 70
3.2.9 GetSchemaExtension 71
3.2.10 外部 WMS レイヤ 72
3.2.11 赤線引き 73
3.3 Web Feature Service (WFS) 75

3.3.1 GetCapabilities . . . 75
3.3.2 GetFeature . . . 76
3.3.3 DescribeFeatureType . . . 85
3.3.4 Transaction . . . 86
3.4 Web Coverage Service (WCS) . . . 89
3.4.1 GetCapabilities . . . 89
3.4.2 DescribeCoverage . . . 91
3.4.3 GetCoverage . . . 93
3.5 ウェブマップタイルサービス (WMTS) . . . 95
3.5.1 GetCapabilities . . . 95
3.5.2 GetTile . . . 96
3.5.3 GetFeatureInfo . . . 97
3.6 OGC API 機能 . . . 98
3.6.1 リソース表現 . . . 99
3.6.2 エンドポイント . . . 100
3.6.3 ページ付け . . . 104
3.6.4 地物フィルタリング . . . 104
3.6.5 地物の並び替え . . . 106
3.6.6 属性の選択 . . . 106
3.6.7 HTML ページをカスタマイズする . . . 106

第 4 章 カタログ **109**

第 5 章 プラグイン **113**
5.1 プラグインのリスト . . . 113
5.2 プラグインの場所 . . . 113
5.3 インストール . . . 113
5.3.1 ZIP を使って手動で行う . . . 113
5.3.2 コマンドラインツールを使う . . . 114
5.4 HTTP サーバー構成 . . . 115
5.4.1 Apache . . . 115
5.5 プラグインの使用方法 . . . 115

第 6 章 高度な構成 **117**
6.1 ログを記録する . . . 117
6.2 環境変数 . . . 117
6.3 設定の要約 . . . 124
6.4 サービスファイルへの接続 . . . 126
6.5 お使いの Linux サーバーにフォントを追加 . . . 126

第 7 章 開発用サーバ **127**

第 8 章 コンテナによるデプロイメント **129**
8.1 シンプルな docker イメージ . . . 129
8.1.1 最初の実行 . . . 131
8.1.2 使用可能なサンプル . . . 132
8.1.3 クリーンアップ . . . 133
8.2 Docker スタック . . . 133

8.2.1 Swarm/docker-compose 133
8.2.2 Kubernetes 134
8.3 クラウドデプロイメント 139
8.3.1 AWS ユースケース 140

第 9 章 よくある質問 **141**

第1章　はじめに

QGIS Server は、WMS、WFS、OGC API Features 1.0 (WFS3)、WCS、セマンティックマッピングのための高度な地図作成機能のオープンソース実装の１つです。QGIS Server は C++で記述された FastCGI / CGI (Common Gateway Interface) アプリケーションであり、Web サーバ（Apache、Nginx など）と連携して動作します。また、新機能を高速かつ効果的に開発、展開することを可能にするため Python プラグインに対応しています。

QGIS Server は、GIS ロジックや地図のレンダリングのためのバックエンドとして、QGIS を使用しています。また、グラフィックスおよびプラットフォームに依存しない C++ プログラミングのために Qt ライブラリが使用されています。他の WMS のソフトウェアとは対照的に、QGIS Server は、サーバ設定とユーザー定義の地図作成ルールの両方のために、設定言語として地図作成ルールを使用します。

QGIS Desktop と QGIS Server は同じ可視化ライブラリを使用しているため、ウェブで公開される地図はデスクトップ GIS と同様の外観になります。

次のセクションでは、Linux (Debian、Ubuntu および派生ディストロ) と Windows に QGIS Server をセットアップするためのサンプル構成を記載します。サーバプラグイン開発に関する詳細は server_plugins を参照してください。

Free Software Foundation により発行された、GNU Free Documentation License, Version 1.3 またはより新しいバージョンの条件の下で、この文書を複製、頒布、修正することが許可されます。ただし、変更不可部分、表紙テキスト、裏表紙テキストは含まれません。

ライセンスの複製は gnu_fdl セクションに含まれています。

第2章　最初のステップ

2.1 Debian ベースのシステムへのインストール

ここでは (Ubuntu と派生ディストロを含む) Debian ベースのシステムにおいて最低限の動作をするようなインストール手順を短く簡単に紹介します。とはいえ、他の多くのディストロや OS も QGIS Server のためのパッケージを提供しています。

注釈: Ubuntu では、管理者権限を必要とするコマンドの前に sudo を付けることで、そのコマンドを一般ユーザーとして使用することができます。Debian では、sudo を使用せず、管理者 (root) として作業することができます。

要件や Debian ベースのシステムに QGIS Server をインストールするために必要な QGIS 公式レポジトリを追加する手順は QGIS のインストーラー ページに記載されています。少なくとも最新の LTR (長期リリース) をインストールすることをおすすめします。

対象バージョンのレポジトリを設定し、QGIS Server をインストールできたなら、以下のファイルで正常にインストールされたことを確認できます。

```
/usr/lib/cgi-bin/qgis_mapserv.fcgi
```

下記のように出力されるなら、サーバは正常にインストールされたと言えます。

注釈: QGIS のバージョンによっては、`qgis_mapserv.fcgi` を実行した時の出力はいくらか異なることがあります。

```
QFSFileEngine::open: No file name specified
Warning 1: Unable to find driver ECW to unload from GDAL_SKIP environment variable.
Warning 1: Unable to find driver ECW to unload from GDAL_SKIP environment variable.
Warning 1: Unable to find driver JP2ECW to unload from GDAL_SKIP environment variable.
Warning 1: Unable to find driver ECW to unload from GDAL_SKIP environment variable.
Warning 1: Unable to find driver JP2ECW to unload from GDAL_SKIP environment variable.
Content-Length: 206
Content-Type: text/xml; charset=utf-8

<ServiceExceptionReport version="1.3.0" xmlns="https://www.opengis.net/ogc">
 <ServiceException code="Service configuration error">Service unknown or unsupported</
```

(次のページに続く)

(前のページからの続き)

```
↪ServiceException>
</ServiceExceptionReport>
```

注釈: 下記の場合は QGIS がステータス 400 というコードを返しており、有効な HTTP セッションが存在しないためにリクエストが失敗したことを正しく認識していることが分かります。これはバグではなく、サーバが適切に機能していることを示すものです。

```
Application path not initialized
Application path not initialized
Warning 1: Unable to find driver ECW to unload from GDAL_SKIP environment variable.
Warning 1: Unable to find driver ECW to unload from GDAL_SKIP environment variable.
Warning 1: Unable to find driver JP2ECW to unload from GDAL_SKIP environment variable.
"Loading native module /usr/lib/qgis/server/libdummy.so"
"Loading native module /usr/lib/qgis/server/liblandingpage.so"
"Loading native module /usr/lib/qgis/server/libwcs.so"
"Loading native module /usr/lib/qgis/server/libwfs.so"
"Loading native module /usr/lib/qgis/server/libwfs3.so"
"Loading native module /usr/lib/qgis/server/libwms.so"
"Loading native module /usr/lib/qgis/server/libwmts.so"
QFSFileEngine::open: No file name specified
Content-Length: 102
Content-Type: application/json
Server:  QGIS FCGI server - QGIS version 3.22.6-Białowieża
Status:  400
[{"code":"Bad request error","description":"Requested URI does not match any␣
↪registered API handler"}]
```

サンプルプロジェクトを追加してみましょう。自作することもできますが、下記のように トレーニング用のデモデータ を使用することもできます。

```
mkdir /home/qgis/projects/
cd /home/qgis/projects/
wget https://github.com/qgis/QGIS-Training-Data/archive/release_3.22.zip
unzip release_3.22.zip
mv QGIS-Training-Data-release_3.22/exercise_data/qgis-server-tutorial-data/world.qgs .
mv QGIS-Training-Data-release_3.22/exercise_data/qgis-server-tutorial-data/
↪naturalearth.sqlite .
```

もちろん、お好きな GIS ソフトを使ってこのファイルを開き、設定や利用可能なレイヤを見ることもできます。

QGIS サーバーを正しく配置するには HTTP サーバーが必要です。推奨される選択肢は Apache または NGINX です。

2.1.1 Apache HTTP サーバー

注釈: 以下では、qgis.demo をあなたのサーバーの名前または IP アドレスに置き換えてください。

1. Apache と mod_fcgid をインストールします：

```
apt install apache2 libapache2-mod-fcgid
```

2. QGIS Server はデフォルトのウェブサイト上で実行することができますが、これ専用に以下のように仮想ホスト を設定してみましょう。

   1. /etc/apache2/sites-available ディレクトリに、次の内容の qgis.demo.conf というファイルを作ります：

```
<VirtualHost *:80>
  ServerAdmin webmaster@localhost
  ServerName qgis.demo

  DocumentRoot /var/www/html

  # Apache logs (different than QGIS Server log)
  ErrorLog ${APACHE_LOG_DIR}/qgis.demo.error.log
  CustomLog ${APACHE_LOG_DIR}/qgis.demo.access.log combined

  # Longer timeout for WPS... default = 40
  FcgidIOTimeout 120

  FcgidInitialEnv LC_ALL "en_US.UTF-8"
  FcgidInitialEnv PYTHONIOENCODING UTF-8
  FcgidInitialEnv LANG "en_US.UTF-8"

  # QGIS log
  FcgidInitialEnv QGIS_SERVER_LOG_STDERR 1
  FcgidInitialEnv QGIS_SERVER_LOG_LEVEL 0

  # default QGIS project
  SetEnv QGIS_PROJECT_FILE /home/qgis/projects/world.qgs

  # QGIS_AUTH_DB_DIR_PATH must lead to a directory writeable by the Server's␣
↪FCGI process user
  FcgidInitialEnv QGIS_AUTH_DB_DIR_PATH "/home/qgis/qgisserverdb/"
  FcgidInitialEnv QGIS_AUTH_PASSWORD_FILE "/home/qgis/qgisserverdb/qgis-auth.
↪db"

  # Set pg access via pg_service file
```

(次のページに続く)

(前のページからの続き)

```
  SetEnv PGSERVICEFILE /home/qgis/.pg_service.conf
  FcgidInitialEnv PGPASSFILE "/home/qgis/.pgpass"

  # if qgis-server is installed from packages in debian based distros this␣
↪is usually /usr/lib/cgi-bin/
  # run "locate qgis_mapserv.fcgi" if you don't know where qgis_mapserv.fcgi␣
↪is
  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory "/usr/lib/cgi-bin/">
    AllowOverride None
    Options +ExecCGI -MultiViews -SymLinksIfOwnerMatch
    Require all granted
  </Directory>

  <IfModule mod_fcgid.c>
  FcgidMaxRequestLen 26214400
  FcgidConnectTimeout 60
  </IfModule>

</VirtualHost>
```

注釈: 上記の設定オプションのいくつかは、サーバの **環境変数** と pg_service ファイル のセクションで説明しています。

2. では QGIS Server のログと認証データベースを格納するディレクトリを作成しましょう：

```
mkdir -p /var/log/qgis/
chown www-data:www-data /var/log/qgis
mkdir -p /home/qgis/qgisserverdb
chown www-data:www-data /home/qgis/qgisserverdb
```

注釈: `www-data` は Debian ベースのシステムでは Apache のユーザーで、Apache がこれらの場所やファイルにアクセスできる必要があります。`chown www-data...` コマンドはそれぞれのディレクトリとファイルの所有者を `www-data` に変更します。

3. これで、まだ有効になっていなければ、仮想ホストと ``fcgid`` mod を有効にすることができます：

```
a2enmod fcgid
a2ensite qgis.demo
```

4. 新しい設定が反映されるように Apache を再起動してください：

```
systemctl restart apache2
```

5. これで Apache は、http://qgis.demo へのリクエストに応答する必要があることがわかります。*qgis.demo* が誰かわかるようにクライアントシステムをセットアップする必要もあります。そのためには、hosts ファイルに `127.0.0.1 qgis.demo` を追加します。

```
# Replace 127.0.0.1 with the IP of your server.
sh -c "echo '127.0.0.1 qgis.demo' >> /etc/hosts"
```

重要: `qgis.demo.conf` と `/etc/hosts` ファイルの両方が、セットアップが機能するように設定されている必要があることを忘れないでください。`/etc/hosts` ファイルにアクセスし、`qgis.demo` の名前をサーバーマシンがネットワーク上に持つ IP（ローカルマシンからのみアクセスできるローカル IP `127.0.0.1` は指定しないでください）に指定することで、ネットワーク上の他のクライアント（Windows や macos マシンなど）から QGIS Server へのアクセスをテストすることもできます。。`*nix` マシンでは `hosts` ファイルは `/etc` にあり、Windows では `C:\Windows\System32\drivers\etc` ディレクトリの下にあります。Windows では hosts ファイルを開く前に管理者権限でテキストエディタを起動する必要があります。

これで QGIS Server が http://qgis.demo で利用できます。確認するには、簡単には次をブラウザで入力します：

```
http://qgis.demo/cgi-bin/qgis_mapserv.fcgi?SERVICE=WMS&VERSION=1.3.0&
↪REQUEST=GetCapabilities
```

2.1.2 NGINX HTTP Server

注釈: 以下では、`qgis.demo` をあなたのサーバーの名前または IP アドレスに置き換えてください。

NGINX を使用して QGIS Server を使用することもできます。Apache とは異なり、NGINX は FastCGI プロセスを自動的に起動しません。FastCGI プロセスは他の何かによって起動されます。

NGINX をインストールする:

```
apt install nginx
```

- 最初の選択肢として、**spawn-fcgi** または **fcgiwrap** を使って QGIS Server プロセスを起動し、管理することができます。どちらも公式の Debian パッケージがあります。X サーバーが動作しておらず、例えば印刷が必要な場合には、 *xvfb* を使用することができます。
- もう一つの選択肢は、今日ほとんどの Linux ディストリビューションが使用している GNU/Linux の init システムである **Systemd** に依存することです。この方法の利点の 1 つは、他のコンポーネントやプロセスを必要としないことです。これは、シンプルでありながら堅牢で、本番環境でのデプロイメントに効率的であることを意図しています。

NGINX の設定

前の設定で使った **include fastcgi_params;** は、 `/etc/nginx/fastcgi_params` からパラメータを追加するために重要です:

```
fastcgi_param  QUERY_STRING       $query_string;
fastcgi_param  REQUEST_METHOD     $request_method;
fastcgi_param  CONTENT_TYPE       $content_type;
fastcgi_param  CONTENT_LENGTH     $content_length;

fastcgi_param  SCRIPT_NAME        $fastcgi_script_name;
fastcgi_param  REQUEST_URI        $request_uri;
fastcgi_param  DOCUMENT_URI       $document_uri;
fastcgi_param  DOCUMENT_ROOT      $document_root;
fastcgi_param  SERVER_PROTOCOL    $server_protocol;
fastcgi_param  REQUEST_SCHEME     $scheme;
fastcgi_param  HTTPS              $https if_not_empty;

fastcgi_param  GATEWAY_INTERFACE  CGI/1.1;
fastcgi_param  SERVER_SOFTWARE    nginx/$nginx_version;

fastcgi_param  REMOTE_ADDR        $remote_addr;
fastcgi_param  REMOTE_PORT        $remote_port;
fastcgi_param  SERVER_ADDR        $server_addr;
fastcgi_param  SERVER_PORT        $server_port;
fastcgi_param  SERVER_NAME        $server_name;

# PHP only, required if PHP was built with --enable-force-cgi-redirect
fastcgi_param  REDIRECT_STATUS    200;
```

さらに、いくつかの qgis-server-envar を使用して QGIS Server を設定することができます。NGINX の設定ファイル `/etc/nginx/nginx.conf` では、以下のように `fastcgi_param` 命令を使ってこれらの変数を定義する必要があります：

```
location /qgisserver {
    gzip            off;
    include         fastcgi_params;
    fastcgi_param   QGIS_SERVER_LOG_STDERR  1;
    fastcgi_param   QGIS_SERVER_LOG_LEVEL   0;
    fastcgi_pass    unix:/var/run/qgisserver.socket;
 }
```

FastCGI ラッパー

警告: **fcgiwrap** はすでに Systemd サービスにラップされているので、**spawn-fcgi** よりもセットアップが簡単です。しかし、spawn-fcgi を使用するよりもはるかに遅いソリューションになります。fcgiwrap では、リクエストごとに新しい QGIS Server プロセスが作成されるため、QGIS プロジェクトファイルの読み込みや解析を含む QGIS Server の初期化処理がリクエストごとに行われます。spawn-fcgi を使用すると、QGIS Server プロセスはリクエストの間中存続するため、パフォーマンスが大幅に向上します。そのため、spawn-fcgi は実運用での使用を推奨します。

spawn-fcgi

spawn-fcgi を使いたい場合は:

1. 最初のステップはパッケージをインストールすることです:

```
apt install spawn-fcgi
```

2. 次に以下のブロックを NGINX サーバー設定に導入します:

```
location /qgisserver {
    gzip            off;
    include         fastcgi_params;
    fastcgi_pass    unix:/var/run/qgisserver.socket;
}
```

3. そして NGINX を再起動して新しい設定を有効にします:

```
systemctl restart nginx
```

4. 最後に、spawn-fcgi 用のデフォルトのサービスファイルがないので、ターミナルで QGIS Server を手動で起動する必要があります：

```
spawn-fcgi -s /var/run/qgisserver.socket \
                -U www-data -G www-data -n \
                /usr/lib/cgi-bin/qgis_mapserv.fcgi
```

これで QGIS Server が http://qgis.demo/qgisserver から利用できます。

注釈: spawn-fcgi を使用する場合、サーバを実行する前に環境変数を直接定義することができます。例: `export QGIS_SERVER_LOG_STDERR=1`

もちろん、起動時や好きな時に QGIS Server を起動する init スクリプトを追加することもできます。例えば、**systemd** を使用して：

1. この内容でファイル `/etc/systemd/system/qgis-server.service` を編集します:

```
[Unit]
Description=QGIS server
After=network.target

[Service]
;; set env var as needed
;Environment="LANG=en_EN.UTF-8"
;Environment="QGIS_SERVER_PARALLEL_RENDERING=1"
;Environment="QGIS_SERVER_MAX_THREADS=12"
;Environment="QGIS_SERVER_LOG_LEVEL=0"
;Environment="QGIS_SERVER_LOG_STDERR=1"
;; or use a file:
;EnvironmentFile=/etc/qgis-server/env

ExecStart=spawn-fcgi -s /var/run/qgisserver.socket -U www-data -G www-data -n /
↪usr/lib/cgi-bin/qgis_mapserv.fcgi

[Install]
WantedBy=multi-user.target
```

2. 次にこのサービスを有効にして起動します:

```
systemctl enable --now qgis-server
```

警告: spawn-fcgi は、上のコマンドで QGIS Server プロセスをひとつだけ起動します。

fcgiwrap

fcgiwrap を使うのは **spawn-fcgi** を設定するよりとても簡単ですが、ずっと遅いです。

1. まず、対応するパッケージをインストールする必要があります:

```
apt install fcgiwrap
```

2. 次に以下のブロックを NGINX サーバー設定に導入します:

```
location /qgisserver {
    gzip            off;
    include         fastcgi_params;
    fastcgi_pass    unix:/var/run/fcgiwrap.socket;
    fastcgi_param   SCRIPT_FILENAME /usr/lib/cgi-bin/qgis_mapserv.fcgi;
}
```

3. 最後に NGINX と **fcgiwrap** を再起動し、新しい設定を反映させます:

```
systemctl restart nginx
systemctl restart fcgiwrap
```

これで QGIS Server が http://qgis.demo/qgisserver から利用できます。

Systemd

QGIS Server を完全に使うには、特に印刷のために、X Server が動作している必要があります。すでに X Server が動作している場合は、systemd サービスを使うことができます。

この方法は QGIS Server をデプロイするために 2 つの Systemd ユニットに依存しています: Socket ユニット と Service ユニット 。

1. **QGIS Server Socket ユニット** は、NGINX が QGIS Server を起動して通信するために使用するファイルシステムソケットを定義して作成します。ソケットユニットは `Accept=false` で設定する必要があります。これは `accept()` システムコールの呼び出しがサービスユニットによって作成されたプロセスに委譲されることを意味します。ソケットユニットは `/etc/systemd/system/qgis-server@.socket` にあり、これは実際にはテンプレートです：

```
[Unit]
Description=QGIS Server Listen Socket (instance %i)

[Socket]
Accept=false
ListenStream=/var/run/qgis-server-%i.sock
SocketUser=www-data
SocketGroup=www-data
SocketMode=0600

[Install]
WantedBy=sockets.target
```

2. ソケットを有効にして起動します：

```
for i in 1 2 3 4; do systemctl enable --now qgis-server@$i.socket; done
```

3. **QGIS Server Service ユニット** は QGIS Server プロセスを定義し、起動します。重要なのは、Service プロセスの標準入力が Socket ユニットによって定義されたソケットに接続されていることです。これは `/etc/systemd/system/qgis-server@.service` にあるサービスユニットの設定で `StandardInput=socket` を使用して設定する必要があります:

```
[Unit]
Description=QGIS Server Service (instance %i)

[Service]
User=www-data
```

(次のページに続く)

(前のページからの続き)

```
Group=www-data
StandardOutput=null
StandardError=journal
StandardInput=socket
ExecStart=/usr/lib/cgi-bin/qgis_mapserv.fcgi
EnvironmentFile=/etc/qgis-server/env

[Install]
WantedBy=multi-user.target
```

注釈: QGIS Server **環境変数** は別のファイル `/etc/qgis-server/env` で定義されます。以下のようになります：

```
QGIS_PROJECT_FILE=/etc/qgis/myproject.qgs
QGIS_SERVER_LOG_STDERR=1
QGIS_SERVER_LOG_LEVEL=3
```

---

4. ソケットサービスを開始します：

```
for i in 1 2 3 4; do systemctl enable --now qgis-server@$i.service; done
```

5. 最後に、NGINX HTTP サーバーの設定を紹介します：

```
upstream qgis-server_backend {
   server unix:/var/run/qgis-server-1.sock;
   server unix:/var/run/qgis-server-2.sock;
   server unix:/var/run/qgis-server-3.sock;
   server unix:/var/run/qgis-server-4.sock;
}

server {
   ...

   location /qgis-server {
       gzip off;
       include fastcgi_params;
       fastcgi_pass qgis-server_backend;
   }
}
```

6. 新しい設定が反映されるように NGINX を再起動します：

```
systemctl restart nginx
```

チュートリアル を共有してくれた Oslandia に感謝します。

2.1.3 Xvfb

QGIS Server を完全に使用するためには、特に印刷のために X Server が動作している必要があります。サーバー上では通常インストールしないことを推奨しますので、xvfb を使って仮想 X 環境を構築してください。

サーバをグラフィック/X11 環境で動作させている場合は、xvfb をインストールする必要はありません。詳細は https://www.itopen.it/qgis-server-setup-notes/ 。

1. パッケージをインストールする:

```
apt install xvfb
```

2. この内容でサービスファイル /etc/systemd/system/xvfb.service を作ります:

```
[Unit]
Description=X Virtual Frame Buffer Service
After=network.target

[Service]
ExecStart=/usr/bin/Xvfb :99 -screen 0 1024x768x24 -ac +extension GLX +render -
↪noreset

[Install]
WantedBy=multi-user.target
```

3. xvfb.service の状態を有効にし、開始し、チェックします：

```
systemctl enable --now xvfb.service
systemctl status xvfb.service
```

4. その後、HTTP サーバーに応じて、**DISPLAY** パラメーターを設定するか、直接 **xvfb-run** を使います。

   - Apache を使う:

     1. *Fcgid* 設定に追加します (*Apache HTTP* **サーバー** を参照)：

        ```
        FcgidInitialEnv DISPLAY       ":99"
        ```

     2. 新しい設定が反映されるように Apache を再起動します：

        ```
        systemctl restart apache2
        ```

   - NGINX を使う

     - xvfb-run を使った spawn-fcgi を実行する：

       ```
       xvfb-run /usr/bin/spawn-fcgi -f /usr/lib/cgi-bin/qgis_mapserv.fcgi \
                                    -s /tmp/qgisserver.socket \
                                    -G www-data -U www-data -n
       ```

     - HTTP サーバー設定の **DISPLAY** 環境変数を使用します。

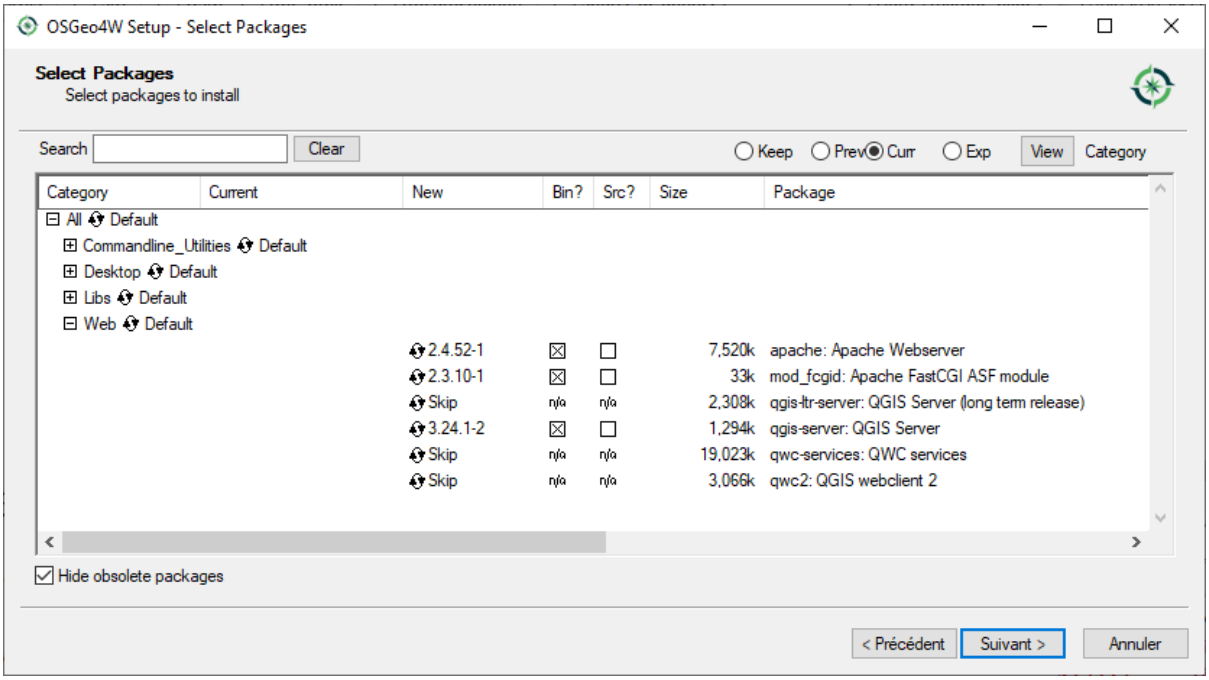
```
fastcgi_param  DISPLAY       ":99";
```

2.2 Windows へのインストール

QGIS Server は、OSGeo4W ネットワークインストーラー (https://qgis.org/en/site/forusers/download.html) の 64 ビット版を使用して Windows システムにインストールすることもできます。

簡単な手順は以下の通りです：

1. OSGeo4W インストーラーをダウンロードして実行する
2. "Advanced Install"に従い、**QGIS Desktop**, **QGIS Server apache**, **mod_fcgid** パッケージをインストールします。



3. Apache は Windows ではサービスとして直接インストールされません。次が必要です：
    1. Right-click the `OSGeo4W.bat` file at the root of the `C:\OSGeo4W\` folder (if the default installation paths have been used) and select *Run as administrator*
    2. コンソールで、`apache-install.bat` を実行し、次が出力されます

```
> apache-install.bat
Installing the 'Apache OSGeo4W Web Server' service
The 'Apache OSGeo4W Web Server' service is successfully installed.
Testing httpd.conf....
Errors reported here must be corrected before the service can be started.
...
```

レポートにあるようにサービスは開始されています。しかし、カスタム設定がないため、サーバーの実行に失敗する可能性があります。

4. Edit the `C:\OSGeo4w\apps\apache\conf\httpd.conf` file with the following changes (various other combinations are possible):

目的	既存の config	置換
(オプション) IP やポートを使ってリッスンするアドレスをカスタマイズします。エントリーはいくつでも追加できます。	`Listen ${SRVPORT}`	`Listen localhost:8080`
スクリプトファイルのある場所を示します	`ScriptAlias /cgi-bin/ "${SRVROOT}/cgi-bin/"`	`ScriptAlias /cgi-bin/ "C:/OSGeo4W/apps/qgis/bin/"`

スクリプトフォルダーに許可を与えます	`<Directory "${SRVROOT}/cgi-bin">`
	`  AllowOverride None`
	`  Options None`
	`  Require all granted`
	`</Directory>`
		`<Directory "C:/OSGeo4W/apps/qgis/bin">`
		`  SetHandler cgi-script`
		`  AllowOverride None`
		`  Options ExecCGI`
		`  Require all granted`
		`</Directory>`

スクリプトファイルに使うために拡張を有効にします。アンコメントして完成させます	`#AddHandler cgi-script .cgi`	`AddHandler cgi-script .cgi .exe`

さらに OSGeo4W カスタム設定変数を加えます	`# parse OSGeo4W apache conf files`
	`IncludeOptional "C:/OSGeo4W/httpd.d/httpd_*.conf"`
		`# parse OSGeo4W apache conf files`
		`IncludeOptional "C:/OSGeo4W/httpd.d/httpd_*.conf"`
		`SetEnv GDAL_DATA "C:/OSGeo4W/share/gdal"`
		`SetEnv QGIS_AUTH_DB_DIR_PATH "C:/OSGeo4W/apps/qgis/resources"`

5. Apache ウェブサーバーを再起動します

```
> apache-restart.bat
```

6. ブラウザウィンドウを開き、QGIS Server への GetCapabilities リクエストをテストします。

localhost:8080 をリッスンするように設定した IP とポートに置き換えてください。

```
http://localhost:8080/cgi-bin/qgis_mapserv.fcgi.exe?SERVICE=WMS&VERSION=1.3.0&
↪REQUEST=GetCapabilities
```

機能が :file:`XML`ファイルで返されるはずです。これでサーバを使用する準備が整いました。

2.3 プロジェクトを供給する

これで QGIS Server がインストールされ、稼働しているので、あとは使うだけです。

言うまでもありませんが、作業には QGIS プロジェクトが必要です。もちろん、連絡先情報を定義したり、CRS に正確な制限を加えたり、あるいはいくつかのレイヤを除外したりして、プロジェクトを完全にカスタマイズすることもできます。これについては **プロジェクトを設定する** で後述します。

ただしここでは、上で説明したように、既に設定され、ダウンロードされている /home/qgis/projects/world.qgs にある簡単なプロジェクトを使うことにします。

プロジェクトを開き、レイヤをざっと見てみると、4 つのレイヤが利用できることが分かります：

- airports
- places
- countries
- countries_shapeburst

今のところ、完全なリクエストを理解する必要はありませんが、ウェブブラウザで次のように *countries* レイヤを取得することで、QGIS Server のおかげで前のレイヤのいくつかを含むマップを取得することができます：

- 上記の手順に従って Apache HTTP Server をインストールした場合：

```
http://qgis.demo/cgi-bin/qgis_mapserv.fcgi?
  MAP=/home/qgis/projects/world.qgs&
  LAYERS=countries&
  SERVICE=WMS&
  VERSION=1.3.0&
  REQUEST=GetMap&
  CRS=EPSG:4326&
  WIDTH=400&
  HEIGHT=200&
  BBOX=-90,-180,90,180
```

- 上記の手順に従って NGINX HTTP サーバーをインストールした場合：

```
http://qgis.demo/qgisserver?
  MAP=/home/qgis/projects/world.qgs&
  LAYERS=countries&
```

(次のページに続く)

(前のページからの続き)

```
SERVICE=WMS&
VERSION=1.3.0&
REQUEST=GetMap&
CRS=EPSG:4326&
WIDTH=400&
HEIGHT=200&
BBOX=-90,-180,90,180
```

次の画像が表示されれば、QGIS Server は正しく動作しています：

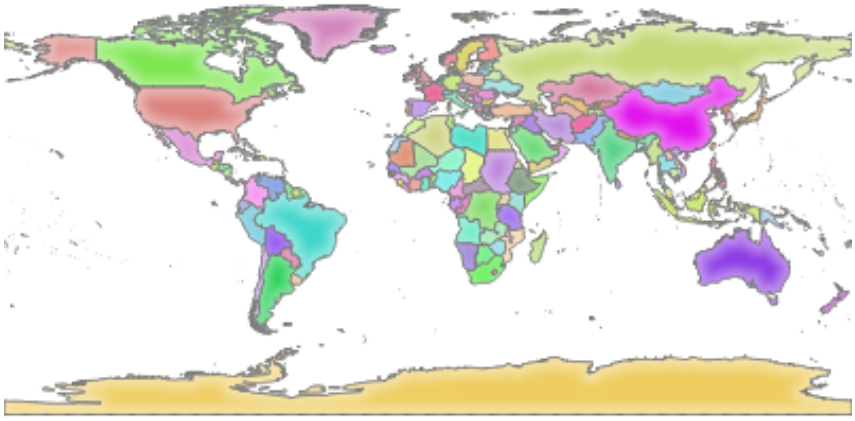


図 2.1: 基本 GetMap リクエストに対するサーバーの応答

なお、**MAP** パラメータを与える代わりに、**QGIS_PROJECT_FILE** 環境変数を定義して、デフォルトでプロジェクトを使用することもできます (qgis-server-envar を参照)。

spawn-fcgi を使う例：

```
export QGIS_PROJECT_FILE=/home/qgis/projects/world.qgs
spawn-fcgi -f /usr/lib/bin/cgi-bin/qgis_mapserv.fcgi \
           -s /var/run/qgisserver.socket \
           -U www-data -G www-data -n
```

2.4 プロジェクトを設定する

新しい QGIS Server WMS、WFS、OAPIF、または WCS を提供するには、いくつかのデータを含む QGIS プロジェクトファイルを作成するか、現在のプロジェクトの 1 つを使用する必要があります。QGIS のレイヤの色とスタイルを定義し、プロジェクトの CRS を定義します。次に、**プロジェクト　プロパティ...** ダイアログの *QGIS* **サーバー** メニューを開き、*Service Capabilities* タブに OWS に関する情報を入力します。

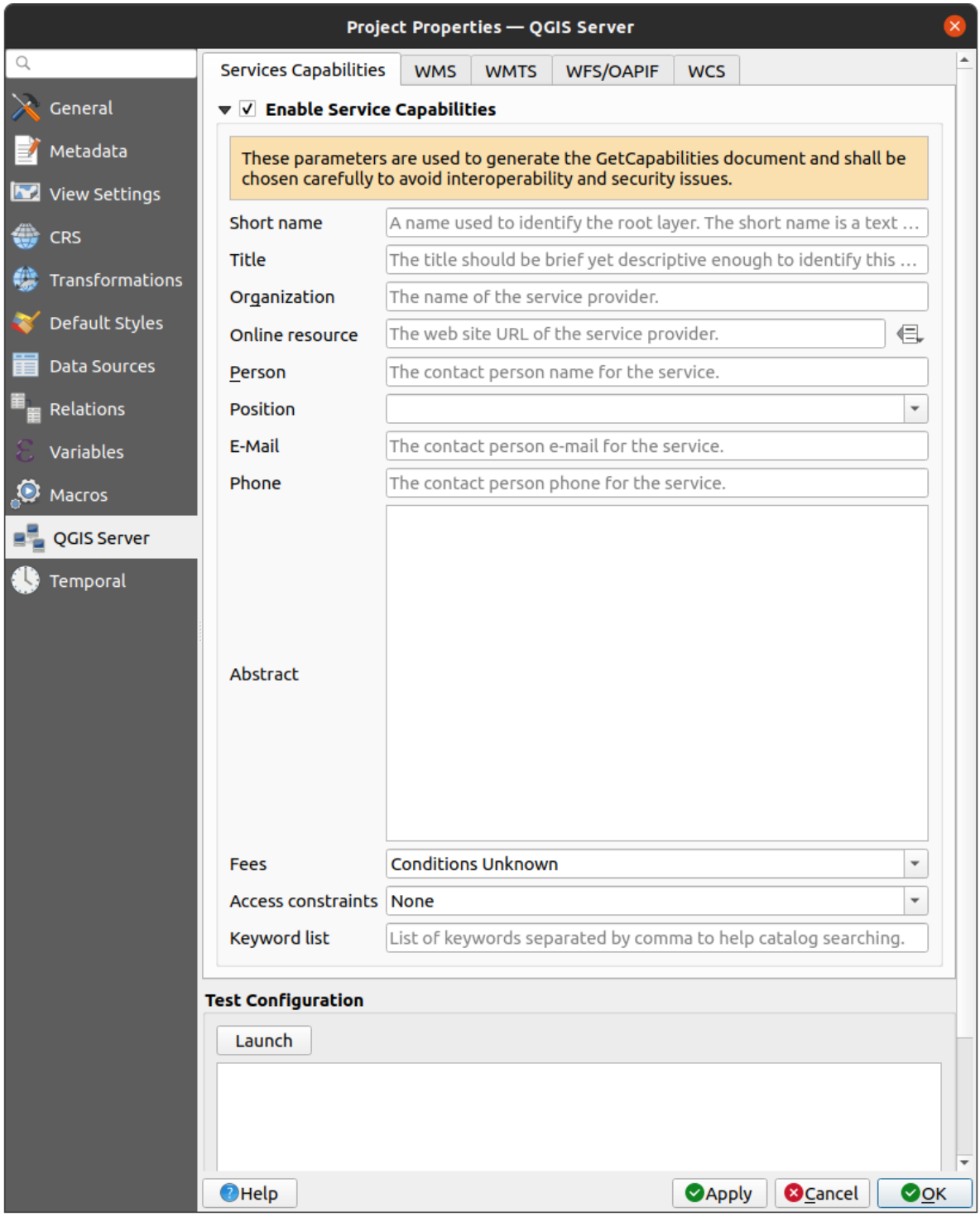


図 2.2: QGIS サーバーの WMS / WFS / WCS プロジェクトの定義

サービス機能が無効になっている場合は、最初に *Service Capabilities* **を有効化** する必要があります。これは WMS、WFS、WCS の GetCapabilities レスポンスに表示されます。 *Service Capabilities* **を有効化** にチェックを入れない場合、QGIS Server は `cgi-bin` フォルダにある `wms_metadata.xml` ファイルにある情報を使用します。

2.4.1 WMS 機能

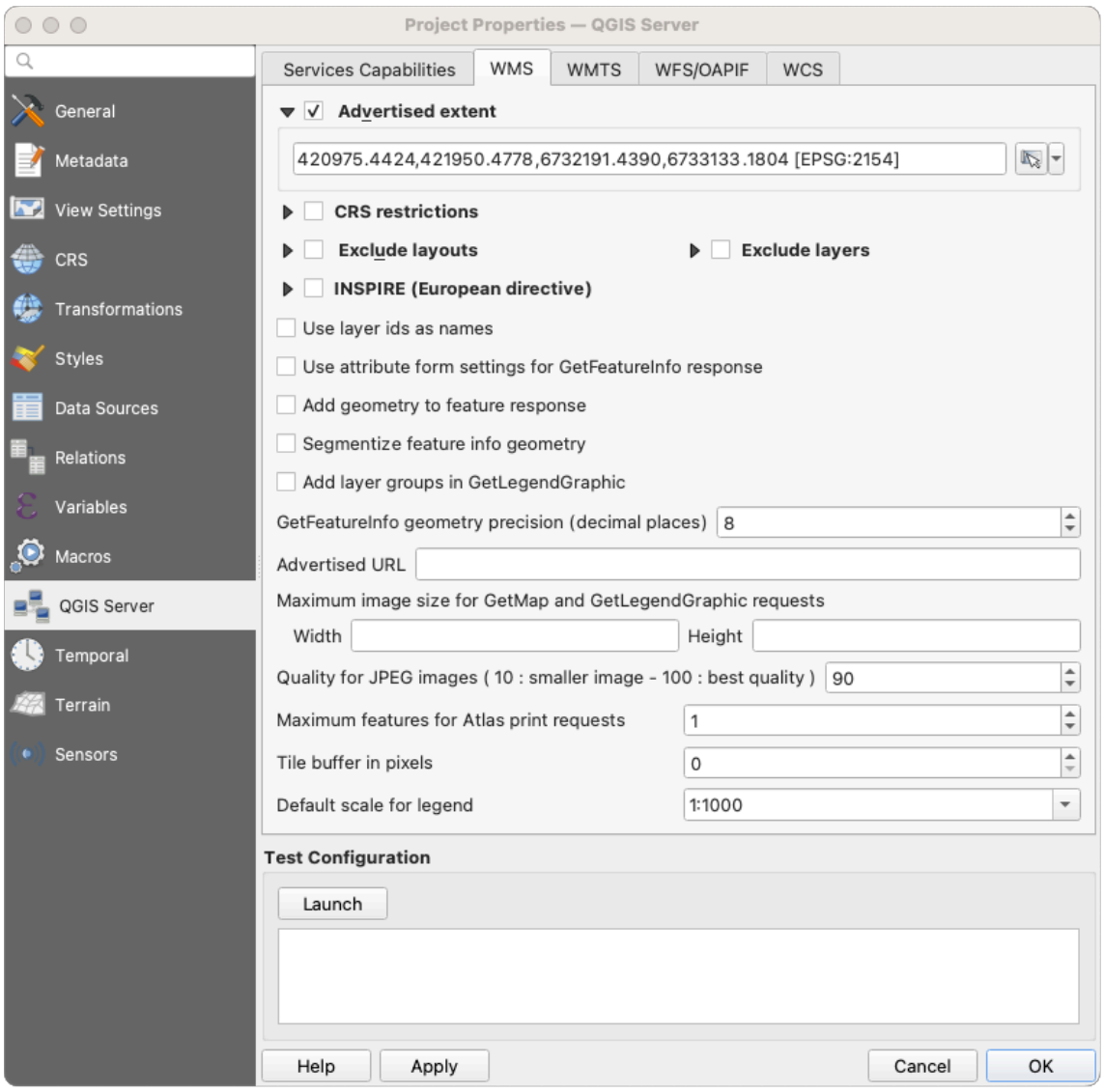


図 2.3: WMS タブにある定義

In the *WMS capabilities* tab, check *Advertised extent* to define the extent advertised in the WMS GetCapabilities response. The spatial extent selector widget helps you enter the extent as a `xmin, xmax, ymin, ymax` text or pick it from the map canvas, layers, bookmarks...

By checking *CRS restrictions*, you can restrict in which coordinate reference systems (CRS) QGIS Server will offer to render maps. It is recommended that you restrict the offered CRS as this reduces the size of the WMS GetCapabilities response. Use the button below to select those CRSs from the Coordinate Reference System Selector, or click *Used* to add the CRSs used in the QGIS project to the list.

プロジェクトで印刷レイアウトが定義されている場合、それらは `GetProjectSettings` レスポンスにリストされ、GetPrint リクエストで印刷レイアウトのいずれかをテンプレートとして使用して印刷を作成することができます。これは WMS 1.3.0 仕様に対する QGIS 固有の拡張です。WMS によって公開される印刷レイアウトを除外したい場合は、チェックボックス| **レイアウトの除外** をチェックし、下の ボタンを

クリックします。次に、除外レイアウトリストに追加するために、**印刷レイアウトの選択** ダイアログから印刷レイアウトを選択します。

WMS からレイヤまたはレイヤーグループを除外したい場合は、**レイヤの除外** をチェックし、下の ボタンをクリックします。これにより **制限されたレイヤとグループを選択する** ダイアログが開き、公開したくないレイヤとグループを選択することができます。複数のエントリを選択したい場合は `Shift` キーまたは `Ctrl` キーを使用してください。WMS の GetCapabilities レスポンスのサイズが小さくなり、クライアント側のロード時間が短縮されるため、不要なレイヤは公開しないことをお勧めします。

レイヤ *ID* **を名前に使う** をチェックすると、`GetCapabilities` レスポンスや `GetMap LAYERS` パラメータでレイヤを参照する際にレイヤ ID が使用されます。そうでない場合は、レイヤの名前、または定義されている場合は、短い名前 (ref:*vectorservermenu* を参照) が使用されます。

要求された GetFeatureInfo は、平文テキスト、XML、および GML で受け取ることができます。デフォルトは XML です。

必要であれば、**地物情報の応答に** *WKT* **ジオメトリを加える** をチェックします。これにより、各地物のバウンディングボックスが GetFeatureInfo レスポンスに含まれます。ref:`WITH_GEOMETRY` パラメータも参照してください。

多くの Web クライアントはジオメトリの円弧を表示できないため、GetFeatureInfo レスポンスでクライアントに送信する前にジオメトリをセグメント化するオプションがあります。これにより、そのようなクライアントでも地物のジオメトリを表示することができます（例 地物のハイライト表示）。このオプションを有効にするには、 **地物情報ジオメトリをセグメント化** をチェックする必要があります。

When a layer group is passed to `GetLegendGraphic` request, all of its leaf layers are added to the legend picture (however without the groups' labels). Check the *Add layer groups in GetLegendGraphic* option if you want to also insert the layer groups (and subgroups) names into the layer tree, just like in QGIS Desktop legend.

また、*GetFeatureInfo* **ジオメトリ精度** オプションを使用して、GetFeatureInfo ジオメトリの精度を設定することもできます。これにより、完全な精度を必要としない場合に帯域幅を節約することができます。

WMS GetCapabilities レスポンスで QGIS Server に特定のリクエスト URL を広告させたい場合は、**公開** *URL* フィールドに対応する URL を入力します。

さらに、:guilabel:`Maximums for GetMap request`の下にあるそれぞれのフィールドに最大幅と最大高さを入力することで、GetMap リクエストによって返されるマップの最大サイズを制限することができます。

JPEG **イメージの品質** の係数を変更することができます。品質係数は 0 から 100 の範囲でなければなりません。0 を指定すると最大圧縮、100 を指定すると非圧縮になります。

地図帳印刷する地物数の上限 フィールドを設定することで、1 回のリクエストで印刷する地図帳地物の上限を変更することができます。

QGIS Server をタイルモードで使用する場合（*TILED* **パラメータ** を参照）、**タイルのバッファ（ピクセル）** を設定することができます。推奨値は、QGIS プロジェクトで最大のシンボルまたは線幅の大きさです。

レイヤーの一つが 地図ヒントディスプレイ を使用している場合（つまり式を使ってテキストを表示するために）これは GetFeatureInfo 出力内で表示されます。レイヤーは、その属性のいずれかの値マップを使用している場合は、この情報には、GetFeatureInfo 出力に表示されます。

2.4.2 WMTS 機能

WMTS **機能** タブでは、WMTS として公開したいレイヤを選択し、PNG または JPEG として公開するかどうかを指定できます。

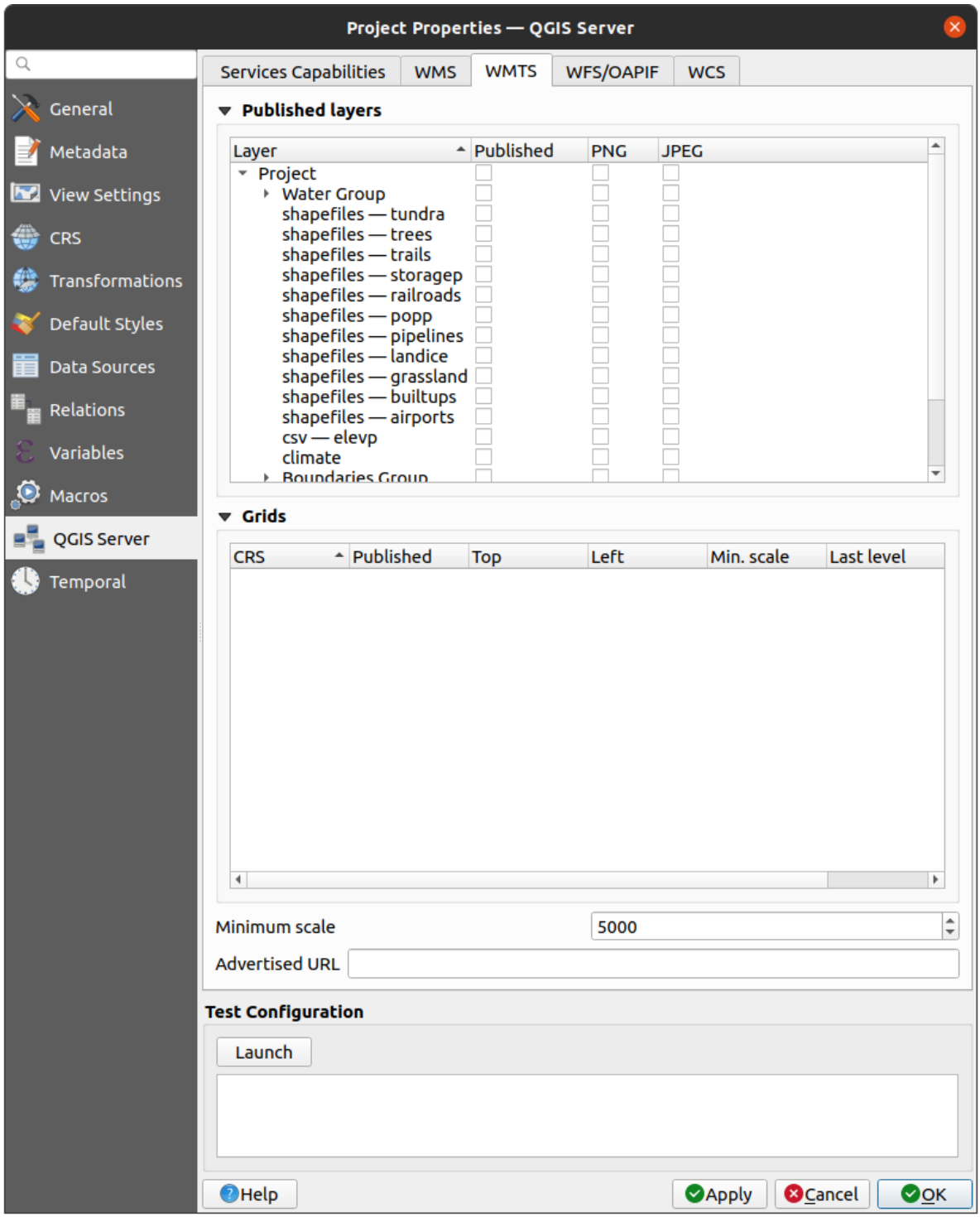


図 2.4: WMTS タブにある定義

WMTS **機能** セクションの **公開** *URL* フィールドに URL を入力すると、QGIS Server は WMTS GetCapabilities レスポンスでこの特定の URL を広告します。

2.4.3 WFS/OAPIF 機能

WFS/OAPIF **機能** タブでは、WFS または OAPIF として公開したいレイヤを選択し、更新、挿入、削除操作を許可するかどうかを指定することができます。

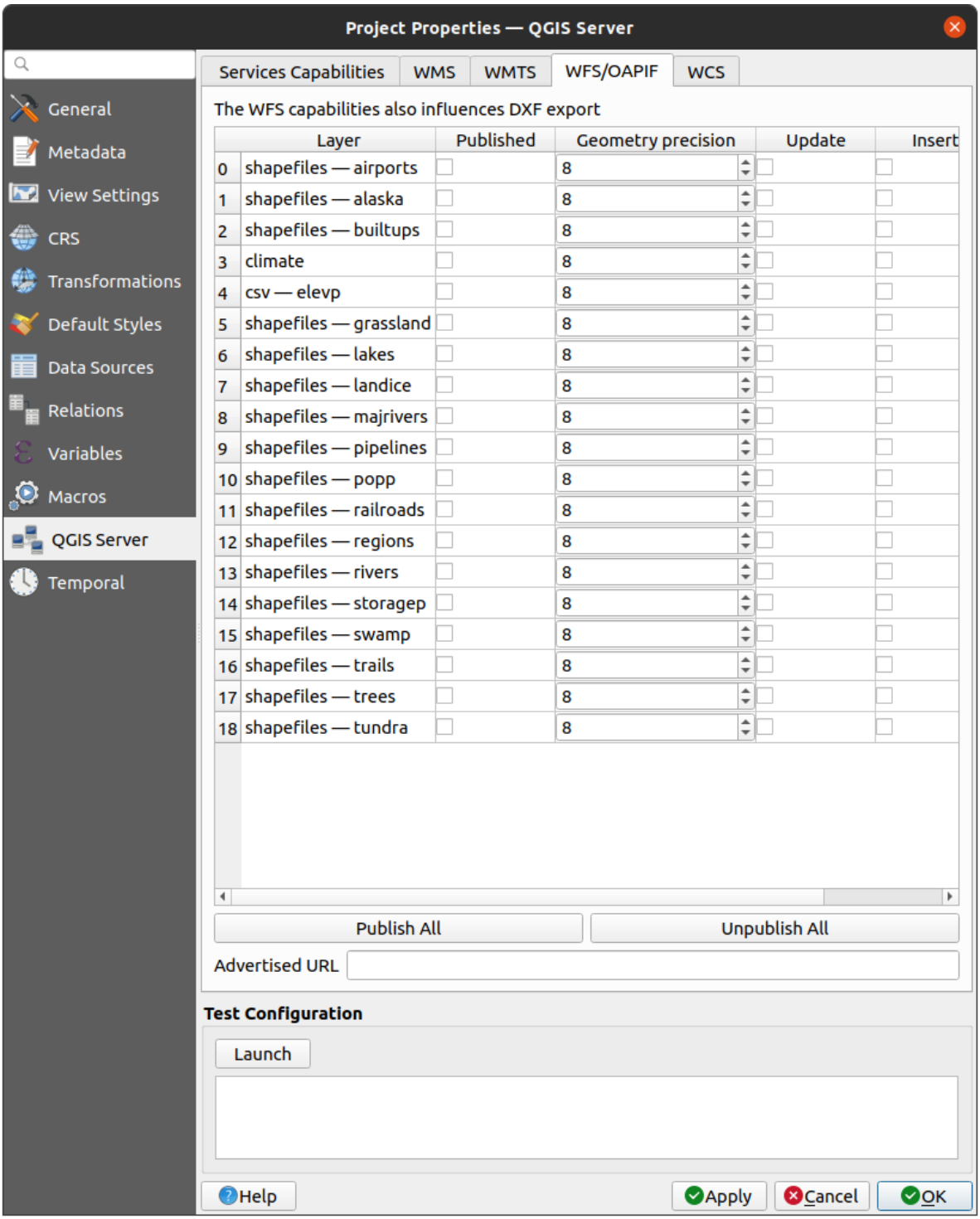


図 2.5: WFS/OAPIF タブにある定義

WFS **機能** セクションの **公開** *URL* フィールドに URL を入力すると、QGIS Server は WFS GetCapabilities レスポンスでこの特定の URL を広告します。

2.4.4 WCS 機能

WCS **機能** タブでは、WCS として公開したいレイヤを選択することができます。

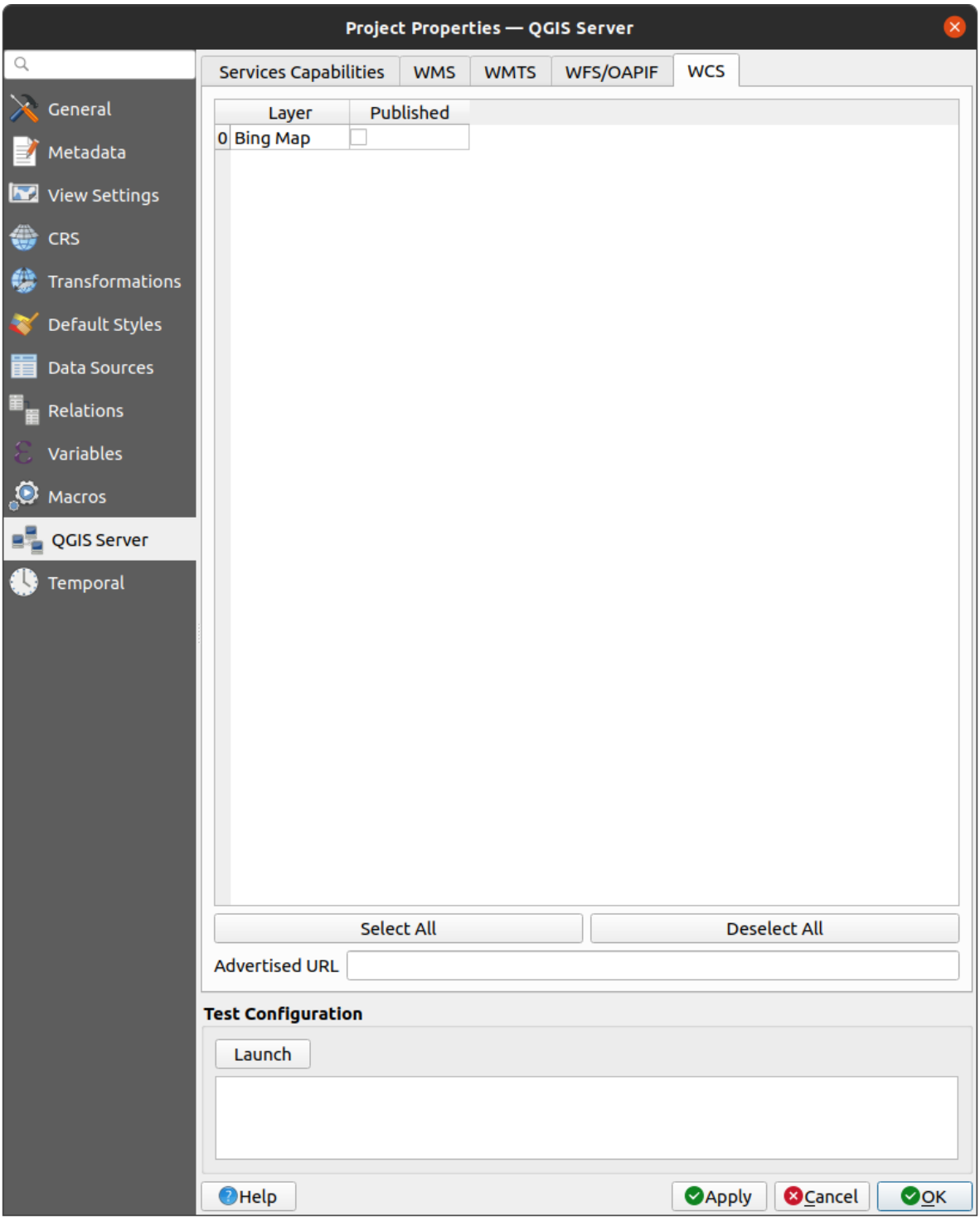


図 2.6: WCS タブにある定義

WCS **機能** セクションの **公開** *URL* フィールドに URL を入力すると、QGIS Server は WCS GetCapabilities レスポンスでこの特定の URL を広告します。

2.4.5 OWS を微調整する

ベクタレイヤの場合、 **レイヤ　　レイヤのプロパティ** ダイアログの **属性** メニューで、各属性を公開するかどうかを定義することができます。デフォルトでは、すべての属性が WMS と WFS によって公開されます。特定の属性を公開したくない場合は、 *WMS* または *WFS* 列の対応するチェックボックスのチェックを外してください。

プロジェクトファイルにテキスト注記または SVG 注記を追加することで、WMS が作成したマップに透かしを重ねることができます。注記の作成方法については sec_annotations セクションを参照してください。注記を WMS の出力に透かしとして表示するには、 **注記テキスト** ダイアログの **地図の位置固定** チェックボックスのチェックを外す必要があります。このチェックボックスには、注記ツールがアクティブな状態で注記をダブルクリックすることでアクセスできます。SVG 注記の場合、(**プロジェクト　　プロパティ**... ダイアログの **一般情報** メニューで）プロジェクトを絶対パスで保存するように設定するか、SVG 画像へのパスを有効な相対パスになるように手動で修正する必要があります。

2.5 サードパーティとの統合

QGIS Server は、WMS、WFS など の標準的な OGC ウェブサービスを提供します。 したがって、さまざまなエンドユーザーツールで使用することができます。

2.5.1 QGIS Desktop との統合

QGIS Desktop はマップデザイナーであり、QGIS Server はマップサーバーです。マップまたは QGIS プロジェクトは、OGC 標準を提供するために QGIS Server によって提供されます。これらの QGIS プロジェクトは、ファイルでもデータベース内のエントリでもかまいません（QGIS Desktop の **プロジェクト　　保存** *PostgreSQL* を使用します)。

さらに、QGIS Server で使用されているプロジェクトを更新するには、専用の更新ワークフローを確立する必要があります（プロジェクトファイルをサーバーの場所にコピーし、QGIS Server を再起動するなど)。今のところ、自動化された処理（メッセージキューサービスを介したサーバーのリロード）は実装されていません。

2.5.2 MapProxy との統合

MapProxy は、タイルキャッシュサーバーで、あらゆる WMS/WMTS マップサーバーを読み込んで提供することができるため、QGIS サーバーのウェブサービスに直接接続することができ、エンドユーザーエクスペリエンスを向上させることができます。

2.5.3 QWC2 との統合

QWC2 は QGIS Server 専用のレスポンシブ Web アプリケーションです。レイヤ選択、地物情報など、高度にカスタマイズされたマップビューアを構築するのに役立ちます。また、認証や印刷サービスのような多くのプラグインが利用可能で、完全なリストはこの リポジトリ にあります。

第3章　サービス

QGIS サーバーでは、 **Open Geospatial Consortium（OGC）** に記載されているように、標準プロトコルに従ってデータを提供できます。

- WMS 1.1.1 及び 1.3.0
- WFS 1.0.0 及び 1.1.0
- OGC API - Features (WFS3)
- WCS 1.0.0 及び 1.1.1
- WMTS 1.0.0

QGIS レンダリングエンジンのおかげでその振る舞いをカスタマイズする可能性を大いに高めるオリジナルの規格に加えて、追加のベンダーパラメータと要求がサポートされています。

3.1 基本

このセクションでは、サービスによって相互に共有されるコンセプトとパラメータについて説明します。これらの中には、OGC 仕様で定義されている標準的なものもあれば、QGIS Server に非常に特化したものもあります。

標準的なコンセプト:

コンセプト	説明
SERVICE	サービスの名称
REQUEST	リクエストの名称

ベンダーコンセプト:

コンセプト	説明
MAP	QGIS プロジェクトファイル
FILE_NAME	ダウンロードするファイルのファイル名
Short name	短い名前の定義

3.1.1 SERVICE

この標準的なパラメータは、特定の **リクエスト** に対して使用するサービス名を指定するもので、SERVICE=NAME のような形式で指定する必要があります。

WMS サービスの URL 例:

```
http://localhost/qgisserver?
SERVICE=WMS
&...
```

注釈: *WFS3 (OGC API Features)* のような REST 型のサービスでは使用できません。

3.1.2 REQUEST

この標準パラメータは、特定の **サービス** に対して実行するリクエストの名前を指定するもので、REQUEST=RequestName のように記述する必要があります。

GetCapabilities リクエストの URL 例:

```
http://localhost/qgisserver?
REQUEST=GetCapabilities
&...
```

注釈: *WFS3 (OGC API Features)* のような REST 型のサービスでは使用できません。

3.1.3 MAP

このベンダパラメータでは、使用する QGIS プロジェクトファイルを定義することができます。絶対パスか、サーバーの実行ファイル `qgis_mapserv.fcgi` の場所からの相対パスで指定します。リクエストは実際に動作するために QGIS プロジェクトを必要とするため、MAP はデフォルトで必須です。しかし、**QGIS_PROJECT_FILE** 環境変数を使用して、デフォルトの QGIS プロジェクトを定義することができます。この場合、MAP は必須パラメータではなくなります。さらに詳しい情報は server_env_ariables の章を参照してください。

URL の例:

```
http://localhost/qgisserver?
MAP=/tmp/QGIS-Training-Data/exercise_data/qgis-server-tutorial-data/world.qgs
&...
```

3.1.4 FILE_NAME

このベンダパラメータが設定されている場合、サーバーレスポンスは指定されたファイル名の添付ファイルとしてクライアントに送信されます。

XML **GetCapabilities** 文書を保存する URL 例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetCapabilities
&FILE_NAME=wms_capabilities.xml
&...
```

注釈: *WFS3 (OGC API Features)* のような REST 型のサービスでは使用できません。

3.1.5 短い名前

多くの要素が、**短い名前** と **タイトル** の両方を持っています。短い名前は機械同士の通信に使われるテキスト文字列で、タイトルは人間のために使われるものです。例えば、あるデータセットには *"Maximum Atmospheric Temperature"* という説明的なタイトルがあり、*"ATMAX"* と略された短い名前でリクエストされるかもしれません。タイトル、短い名前、要約を設定することができます:

- **レイヤ**: レイヤを右クリックし、 **プロパティ...** *QGIS* **サーバー** **説明** を選択します。
- **グループ**: グループ上で右クリックして **グループの** *WMS* **データを設定** を選択します
- **プロジェクト**: **プロジェクト** **プロパティ...** *QGIS* **サーバー** *Service Capabilities* に移動します。

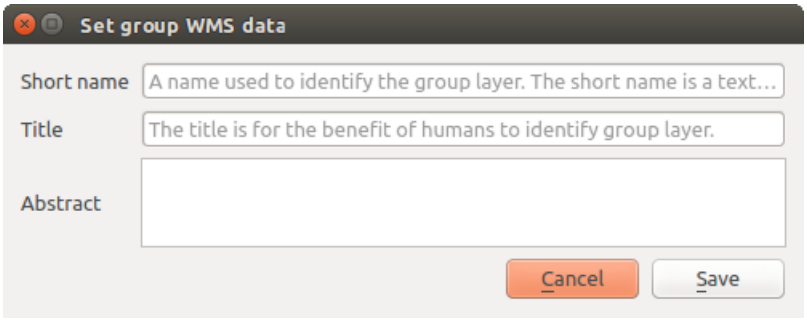


図 3.1: グループの WMS データを設定 ダイアログ

したがって、QGIS Server とやり取りする際に、これらの項目を識別するために短い名前を使用することができます。例えば、標準の LAYERS パラメータの場合:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
```

(次のページに続く)

(前のページからの続き)

```
&LAYERS=shortname1,shortname2
&...
```

3.2 Web マップサービス (WMS)

QGIS サーバに実装された **1.1.1** と **1.3.0** の WMS 標準は、QGIS プロジェクトから作成された地図又は凡例画像を要求するための HTTP インターフェースを提供します。典型的な WMS リクエストは、使用する QGIS プロジェクト、レンダリングするレイヤ、生成する画像の形式を決定します。**Styled Layer Descriptor (SLD)** ではベーシックサポートも利用できます。

仕様:

- WMS 1.1.1
- WMS 1.3.0
- SLD 1.1.0 WMS profile

QGIS Server が提供する標準リクエスト

リクエスト	説明
GetCapabilities	サーバーに関する情報を含む XML メタデータを返します
GetMap	マップを返します
GetFeatureInfo	ピクセル位置のデータ（ジオメトリと値）を取得します
GetLegend-Graphic	凡例シンボルを返します
GetStyle(s)	SLD によるスタイル記述の XML 文書を返します
DescribeLayer	ベクタレイヤとラスタレイヤについて、それぞれ WFS と WCS の有無に関する情報を返します

QGIS Server が提供するベンダーのリクエスト：

リクエスト	説明
GetPrint	QGIS のレイアウトを返します
GetProjectSettings	QGIS Server に関する特定の情報を返します
GetSchemaExtension	オプションの拡張機能に関する XML メタデータを返します

3.2.1 GetCapabilities

OGC WMS 1.1.1 及び 1.3.0 仕様に従った **GetCapabilities** リクエストの標準パラメータ:

パラメータ	必須	説明
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**GetCapabilities**)
VERSION	No	サービスのバージョン

GetCapabilities リクエストは以下のベンダーパラメータもサポートしています:

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetCapabilities
```

3.2.2 GetMap

OGC WMS 1.1.1 及び 1.3.0 仕様に従った **GetMap** リクエストの標準パラメータ:

パラメータ	必須	説明
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**GetMap**)
VERSION	Yes	サービスのバージョン
LAYERS	No	表示するレイヤ
STYLES	No	レイヤのスタイル
SRS / CRS	Yes	座標参照系
BBOX	Yes	マップの範囲
WIDTH	Yes	ピクセル単位の画像の幅
HEIGHT	Yes	ピクセル単位の画像の高さ
FORMAT	No	画像の形式
TRANSPARENT	No	透明な背景
SLD	No	スタイリングに使用する SLD の URL
SLD_BODY	No	スタイリングに使用するインライン SLD(XML)

QGIS Server では、標準的なものに加えて、 *redlining*、 *external WMS layers*、および以下の追加パラメータがサポートされています:

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル
BGCOLOR	No	背景色を指定します
DPI	No	出力解像度を指定します
IMAGE_QUALITY	No	JPEG 圧縮
OPACITIES	No	レイヤまたはグループの不透明度
FILTER	No	地物のサブセット
SELECTION	No	地物を強調します
FILE_NAME	No	ダウンロードするファイルのファイル名 `FORMAT=application/dxf` の場合のみ
FORMAT_OPTIONS	No	指定されたファイル形式のオプション `FORMAT=application/dxf` の場合のみ
TILED	No	*tiled* **モード** での作業

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetMap
&MAP=/home/qgis/projects/world.qgs
&LAYERS=mylayer1,mylayer2,mylayer3
&STYLES=style1,default,style3
&OPACITIES=125,200,125
&CRS=EPSG:4326
&WIDTH=400
&HEIGHT=400
&FORMAT=image/png
&TRANSPARENT=TRUE
&DPI=300
&TILED=TRUE
```

VERSION

このパラメータでは、使用するサービスのバージョンを指定することができます。`VERSION` パラメータに指定できる値は以下の通りです:

- `1.1.1`
- `1.3.0`

バージョン番号によって、次のパラメータで説明するように、若干の違いがあることが予想されます:

- `CRS` / `SRS`
- `BBOX`

LAYERS

マップ上に表示するレイヤーを指定するパラメータです。名前はカンマで区切る必要があります。

さらに、QGIS Server では、レイヤを選択するためのオプションがいくつか導入されています:

- the layer id: the project option allowing to select layers by their id is in *QGIS Server* *WMS* tab of the *Project* *Properties...* dialog. Check the *Use layer ids as names* checkbox to activate this option.
- **短い名前**

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&LAYERS=mylayerid1,mylayerid2
&...
```

STYLES

このパラメータは、レンダリングステップのレイヤのスタイルを指定するために使用されます。スタイルはカンマで区切る必要があります。デフォルトのスタイル名は `default` です。

SRS / CRS

このパラメータは、WMS **1.1.1** におけるマップ出力の空間参照系を示すもので、`EPSG:XXXX` のような形式で指定する必要があります。なお、現在のバージョンが **1.1.1** の場合は `CRS` もサポートされます。

WMS **1.3.0** では、`CRS` パラメータが望ましいですが、`SRS` もサポートされています。

もし `CRS` と `SRS` の両方のパラメータがリクエストに含まれていた場合、 `VERSION` パラメータで示される現在のバージョンが決定的に重要であることに注意してください。

次のケースでは、`CRS` が指示されていないため、`SRS` パラメータは `VERSION` パラメータが何であろうと保持されます:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&SRS=EPSG:2854
&...
```

次のケースでは、`VERSION` パラメータのため、`CRS` の代わりに ``SRS``パラメータが保持されます:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
```

(次のページに続く)

(前のページからの続き)

```
&VERSION=1.1.1
&CRS=EPSG:4326
&SRS=EPSG:2854
&...
```

次のケースでは、VERSION パラメータのため、SRS の代わりに CRS パラメータが保持されます：

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&CRS=EPSG:4326
&SRS=EPSG:2854
&...
```

BBOX

このパラメータを使用すると、現在の CRS に従った単位でマップの範囲を指定することができます。座標はカンマで区切る必要があります。

BBOX パラメータは min_a,min_b,max_a,max_b のように形成されますが、a と b の軸の定義は、現在の VERSION パラメータによって異なります:

- WMS **1.1.1** において、軸の順序は常に東／北です
- WMS **1.3.0** において、軸の順序は CRS の権限に依存します

例えば EPSG:4326 と WMS **1.1.1** の場合、a は経度（東）、b は緯度（北）で、次のようなリクエストとなります:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.1.1
&SRS=epsg:4326
&BBOX=-180,-90,180,90
&...
```

しかし、WMS **1.3.0** の場合、EPSG データベースで定義されている軸の順序は北/東なので、a は緯度、b は経度です：

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&CRS=epsg:4326
```

(次のページに続く)

(前のページからの続き)

```
&BBOX=-90,-180,90,180
&...
```

WIDTH

出力画像の幅をピクセル単位で指定することができるパラメータです。

HEIGHT

出力画像の高さをピクセル単位で指定することができるパラメータです。

FORMAT

このパラメータは、マップ画像の形式を指定するために使うことができます。使用可能な値は次の通りです:

- jpg
- jpeg
- image/jpeg
- image/png
- image/png; mode=1bit
- image/png; mode=8bit
- image/png; mode=16bit
- image/webp
- application/dxf: WFS サービスで読み取り権限を持つレイヤーのみが DXF 形式でエクスポートされます

  URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS&VERSION=1.3.0
&REQUEST=GetMap
&FORMAT=application/dxf
&LAYERS=Haltungen,Normschacht,Spezialbauwerke
&CRS=EPSG%3A21781
&BBOX=696136.28844801,245797.12108743,696318.91114315,245939.25832905
&WIDTH=1042
&HEIGHT=811
&FORMAT_OPTIONS=MODE:SYMBOLLAYERSYMBOLOGY;SCALE:250
&FILE_NAME=plan.dxf
```

TRANSPARENT

この ブール型パラメータを使って、背景の不透明度を指定することができます。使用可能な値は以下の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

ただし、FORMAT で示される画像の形式が PNG と異なる場合、このパラメータは無視されます。

BGCOLOR

このパラメータはマップ画像の背景色を指定することができます。ただし、PNG 画像の場合は TRANSPARENT パラメータと組み合わせることができません（透明度が優先されます）。色はリテラルでも 16 進数でもかまいません。

リテラル表記による URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&BGCOLOR=green
&...
```

16 進数表記による URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&BGCOLOR=0x00FF00
&...
```

DPI

このパラメータを使って、要求される出力分解能を指定することができます。

IMAGE_QUALITY

このパラメータは、JPEG 画像に対してのみ使用されます。デフォルトでは、JPEG の圧縮率は -1 です。

QGIS プロジェクトごとのデフォルトは、 **プロジェクト** **プロパティ...** ダイアログの *OWS Server* *WMS capabilities* メニューで変更することができます。GetMap リクエストでオーバーライドしたい場合は、IMAGE_QUALITY パラメータを使用することができます。

OPACITIES

不透明度の値をカンマで区切ったリストです。不透明度は、レイヤまたはグループレベルで設定することができます。設定可能な値は、0 （完全に透明）から 255 （完全に不透明）までの範囲です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&WIDTH=400
&HEIGHT=200
&CRS=EPSG:4326
&LAYERS=countries,places
&BBOX=42,-6,52,15
&OPACITIES=255,0
```

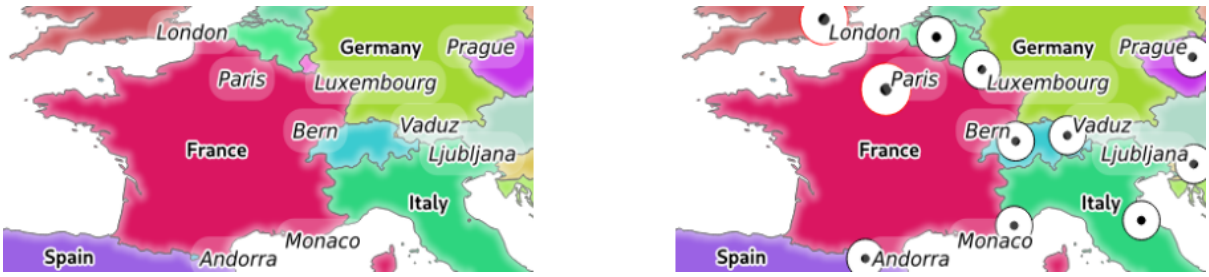


図 3.2: 左は OPACITIES=255,0、右は ``OPACITIES=255,255``

FILTER

FILTER パラメータを使用すると、レイヤーのサブセットを選択することができます。構文は基本的に QGIS のサブセット文字列と同じです。ただし、QGIS Server を介したデータベースへの SQL インジェクションを避けるために、いくつかの制限があります。パラメータに危険な文字列が見つかった場合、QGIS Server は次のエラーを返します：

```
<ServiceExceptionReport>
  <ServiceException code="Security">The filter string XXXXXXXXX has been rejected␣
↪because of security reasons.
```

(次のページに続く)

(前のページからの続き)

```
  Note: Text strings have to be enclosed in single or double quotes. A space between␣
↪each word / special character is mandatory.
  Allowed Keywords and special characters are IS,NOT,NULL,AND,OR,IN,=,<,=<,>,>=,!=,',
↪',(,),DMETAPHONE,SOUNDEX.
  Not allowed are semicolons in the filter expression.</ServiceException>
</ServiceExceptionReport>
```

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&WIDTH=400
&HEIGHT=300
&CRS=EPSG:4326
&BBOX=41,-6,52,10
&LAYERS=countries_shapeburst,countries,places
&FILTER=countries_shapeburst,countries:"name" = 'France';places: "name" = 'Paris'
```

図 3.3: FILTER パラメータを指定した GetMap リクエストに対するサーバーの応答

この例では、同じフィルター `"name" = 'France'` がレイヤ `countries` と `countries_shapeburst` に適用され、フィルター `"name" = 'Paris'` が `places` にのみ適用されています。

注釈: FILTER がある場合は、GetFeatureInfo で属性検索を行い、X/Y パラメータを省略することが可能です。QGIS Server は、マッチした地物の情報を返し、XML 出力に結合バウンディングボックスを生成し

ます。

SELECTION

SELECTION パラメータは、1 つまたは複数のレイヤから地物をハイライトすることができます。ベクタ地物は、地物 ID をカンマで区切ったリストを渡すことで選択することができます。

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&LAYERS=mylayer1,mylayer2
&SELECTION=mylayer1:3,6,9;mylayer2:1,5,6
&...
```

次の画像は、SELECTION オプションを使った GetMap リクエストの応答を示しています　例 `http://myserver.com/...&SELECTION=countries:171,65`。

これらの地物 ID は、ソースデータセットでは **France** と **Romania** に対応しているので、黄色でハイライトされています。

図 3.4: SELECTION パラメーターを指定した GetMap リクエストに対するサーバーの応答

FORMAT_OPTIONS

This parameter can be used to specify options for the selected format. Only for `FORMAT=application/dxf` in GetMap request. Takes a list of key:value pairs separated by semicolon:

- SCALE: シンボロジルール、フィルター、スタイルに使用されます（データの実際のスケーリングではありません - データは元の縮尺のままです）。
- MODE: QGIS Desktop の DXF エクスポートダイアログで提供されるエクスポートオプションに対応します。可能な値は `NOSYMBOLOGY`、`FEATURESYMBOLOGY`、`SYMBOLLAYERSYMBOLOGY` です。
- LAYERATTRIBUTES: specify a field or in case of many layers a comma separated list of fields that contains values for DXF layer names - if not specified, the original QGIS layer names are used.
- USE_TITLE_AS_LAYERNAME: 有効にすると、レイヤのタイトルがレイヤ名として使用されるようになります。
- CODEC: エンコードに使用するコーデックを指定します。デフォルトは ISO-8859-1 です。有効な値については、QGIS デスクトップの DXF エクスポートダイアログを確認してください。
- NO_MTEXT: ラベルに MTEXT の代わりに TEXT を使用します。

- FORCE_2D：2D 出力を強制します。ポリライン幅に必要です。

TILED

パフォーマンス上の理由から、QGIS Server はタイルモードで使用することができます。このモードでは、クライアントが複数の小さな固定サイズのタイルを要求し、それらを組み立ててマップ全体を形成します。この場合、2 つのタイルの境界付近のシンボルは、どちらかのタイルにしか存在しないため、カットされて表示されることがあります。

TILED パラメータを TRUE に設定すると、QGIS Server に **タイル** モードで動作し、QGIS プロジェクトで設定された **タイルバッファ** を適用するように指示します（ **プロジェクトを設定する** を参照）。

TILED が TRUE で、かつ QGIS プロジェクトで非ゼロのタイルバッファが設定されている場合、タイル境界でのカットシンボルを避けるために、タイル範囲外の地物が描画されます。

TILED のデフォルトは FALSE です。

3.2.3 GetFeatureInfo

OGC WMS 1.1.1 および 1.3.0 仕様に従った **GetFeatureInfo** リクエストの標準パラメータ:

パラメータ	必須	説明
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**GetFeatureInfo**)
VERSION	No	サービスのバージョン
QUERY_L	Yes	クエリするレイヤ
LAYERS	Yes	表示するレイヤ（*QUERY_LAYERS* と同じ）
STYLES	No	レイヤのスタイル
SRS / CRS	Yes	座標参照系
BBOX	No	マップの範囲
WIDTH	Yes	ピクセル単位の画像の幅
HEIGHT	Yes	ピクセル単位の画像の高さ
TRANSPARENT	No	透明な背景
INFO_FO	No	出力形式
FEATURE_CC	No	返す地物の最大数
I	No	クエリする点のピクセル列
X	No	*I* パラメータと同じだが、WMS 1.1.1 の場合
J	No	クエリする点のピクセル行
Y	No	*J* パラメータと同じだが、WMS 1.1.1 の場合
WMS_PR	No	ジオメトリを返す際に使用する精度（桁数）(*how to add geometry to feature response* を参照してください)。デフォルト値は -1 で、プロジェクトで定義された精度が使用されることを意味します。

QGIS Server は、標準的なものに加えて、以下の追加パラメータをサポートしています:

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル
FILTER	No	地物のサブセット
FI_POINT_TOLERANCE	No	ポイントレイヤに対するピクセル単位の許容範囲
FI_LINE_TOLERANCE	No	ラインレイヤに対するピクセル単位の許容範囲
FI_POLYGON_TOLERANCE	No	ポリゴンレイヤに対するピクセル単位の許容範囲
FILTER_GEOM	No	ジオメトリのフィルタリング
WITH_DISPLAY_NAME	No	Add the feature display name to the output
WITH_MAPTIP	No	出力にマップチップを追加します
WITH_GEOMETRY	No	出力にジオメトリを追加します

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetMap
&MAP=/home/qgis/projects/world.qgs
&LAYERS=mylayer1,mylayer2,mylayer3
&CRS=EPSG:4326
&WIDTH=400
&HEIGHT=400
&INFO_FORMAT=text/xml
&TRANSPARENT=TRUE
&QUERY_LAYERS=mylayer1
&FEATURE_COUNT=3
&I=250
&J=250
```

INFO_FORMAT

このパラメータは結果のフォーマットを指定するために使用されます。利用可能な値は以下のとおりです：

- text/xml
- text/html
- text/plain
- application/vnd.ogc.gml
- application/json

QUERY_LAYERS

このパラメータは、マップ上に表示するレイヤを指定します。名前はカンマで区切られます。

さらに、QGIS Server では、レイヤーを選択する以下のオプションを導入しています:

- 短い名前
- レイヤ id

詳しくは *GetMap* で定義されている LAYERS パラメータを参照してください。

FEATURE_COUNT

このパラメータは、レイヤーごとに返す地物の最大数を指定します。例えば、 QUERY_LAYERS が layer1, layer2 で、 FEATURE_COUNT が 3 の場合、レイヤ 1 から最大 3 つの地物が返されることになります。同様に、レイヤ 2 からは最大 3 つの地物が返されます。

デフォルトでは 1 つの地物だけが返されます。

I

WMS 1.3.0 で定義されたパラメータで、クエリポイントのピクセル列を指定することができます。

X

I と同じパラメータですが、WMS 1.1.1 で定義されています。

J

WMS 1.3.0 で定義されたパラメータで、クエリポイントのピクセル行を指定することができます。

Y

J と同じパラメータですが、WMS 1.1.1 で定義されています。

FI_POINT_TOLERANCE

このパラメータではラインレイヤでの許容誤差（ピクセル）を指定します。

FI_LINE_TOLERANCE

ラインレイヤの許容範囲をピクセル単位で指定するパラメータです。

FI_POLYGON_TOLERANCE

ポリゴンレイヤの許容範囲をピクセル単位で指定するパラメータです。

FILTER_GEOM

このパラメータは、地物が交差しなければならない WKT ジオメトリを指定します。

WITH_DISPLAY_NAME

This parameter specifies whether to add feature display name to the output.

利用可能な値は次の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

WITH_MAPTIP

このパラメータは、出力にマップチップを加えるかどうかを指定します。

利用可能な値は次の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

WITH_GEOMETRY

このパラメータは、出力にジオメトリを追加するかどうかを指定します。この機能を使用するには、まず QGIS プロジェクトで *Add geometry to feature response* オプションを有効にする必要があります。*Configure your project* を参照してください。

利用可能な値は次の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

3.2.4 GetLegendGraphic

OGC WMS 1.1.1 および 1.3.0 仕様に従った **GetLegendGraphic** リクエストの標準パラメータ

パラメータ	必須	説明
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**GetLegendGraphic**)
VERSION	No	サービスのバージョン
LAYERS	Yes	表示するレイヤ
STYLES	No	レイヤのスタイル
SRS / CRS	No	座標参照系
BBOX	No	マップの範囲
WIDTH	No	ピクセル単位の画像の幅
HEIGHT	No	ピクセル単位の画像の高さ
FORMAT	No	凡例の形式
TRANSPARENT	No	透明な背景

QGIS Server では、標準的なパラメータに加えて、凡例要素のサイズやレイヤタイトルやアイテムラベルのフォントプロパティを変更するための追加パラメータをサポートしています：

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル
SRCWIDTH	No	マップの幅
SRCHEIGHT	No	マップの高さ
SHOWFEATURECOUNT	No	地物の地物数を追加します
RULE	No	ルールシンボルをレンダリング
RULELABEL	No	アイテムラベルをレンダリング
BOXSPACE	No	凡例フレームと内容の間隔 (mm)
LAYERSPACE	No	レイヤの垂直間隔 (mm)
LAYERTITLESPACE	No	レイヤタイトルとアイテムの垂直間隔 (mm)
SYMBOLSPACE	No	シンボルとアイテムの垂直間隔 (mm)
ICONLABELSPACE	No	シンボルとラベルの水平間隔 (mm)
SYMBOLWIDTH	No	シンボルプレビューの幅 (mm)
SYMBOLHEIGHT	No	シンボルプレビューの高さ (mm)
LAYERTITLE	No	レイヤタイトルをレンダリング
LAYERFONTFAMILY	No	レイヤのフォントファミリー
LAYERFONTBOLD	No	レイヤタイトルを太字でレンダリング
LAYERFONTSIZE	No	レイヤタイトルのフォントサイズ (pt)
LAYERFONTITALIC	No	レイヤタイトルを斜体でレンダリング
LAYERFONTCOLOR	No	レイヤタイトルの色
ITEMFONTFAMILY	No	アイテムのフォントファミリー
ITEMFONTBOLD	No	アイテムラベルを太字でレンダリング
ITEMFONTSIZE	No	アイテムラベルのフォントサイズ (pt)
ITEMFONTITALIC	No	アイテムラベルを斜体でレンダリング
ITEMFONTCOLOR	No	アイテムラベルの色
ADDLAYERGROUPS	No	Adds the layer groups to JSON output

BBOX

このパラメータは凡例が作成される地理的な区域を指定するために使用することができます（その形式の説明は **ここ** にあります）が、 RULE パラメータと組み合わせることはできません。SRS/CRS パラメータは BBOX パラメータを使用する際に必須となります。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=countries,airports
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
```

注釈: BBOX パラメータが定義されている場合、凡例は **内容型の凡例** として参照されます。

WIDTH

このパラメータはデフォルトでは使用されませんが、 RULE パラメータが設定された場合には必須となります。この場合、出力画像の幅をピクセル単位で指定することができます。

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=getlegendgraphic
&LAYER=testlayer%20%C3%A8%C3%A9
&RULE=rule1
&WIDTH=30
&HEIGHT=30
```

HEIGHT

このパラメータはデフォルトでは使用されませんが、 RULE パラメータが設定されている場合は必須となります。この場合、出力画像の高さをピクセル単位で指定することができます。

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYER=testlayer%20%C3%A8%C3%A9
&RULE=rule1
&WIDTH=30
&HEIGHT=30
```

FORMAT

このパラメータを使用して、凡例画像の形式を指定することができます。使用可能な値は以下の通りです:

- image/jpeg
- image/png
- application/json

JSON の場合、シンボルは Base64 でエンコードされ、レイアウトやフォントに関するその他のオプションは、凡例がクライアント側で構築される必要があるため、考慮されません。RULE パラメータはこの形式と組み合わせることができません。

URL の例とそれに対応する JSON の出力:

```
http://localhost/qgisserver?
SERVICE=WMS&
REQUEST=GetLegendGraphic&
LAYERS=airports&
FORMAT=application/json
```

そして、それに対応する JSON の出力:

```
{
  "nodes":[
    {
      "icon":"<base64 icon>",
      "title":"airports",
      "type":"layer"
    }
  ],
  "title":""
}
```

SRCWIDTH

このパラメータは RULE パラメータが設定されているときに定義することができます。この場合、SRCWIDTH の値は WIDTH パラメータとして基となる GetMap リクエストに転送され、WIDTH パラメータは GetLegendGraphics の画像凡例の大きさに使われます。

SRCHEIGHT

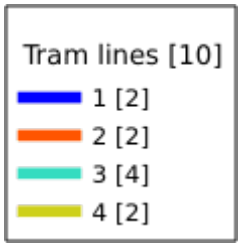
このパラメータは RULE パラメータが設定されているときに定義することができます。この場合、SRCHEIGHT の値は HEIGHT パラメータとして基となる GetMap リクエストに転送され、HEIGHT パラメータは GetLegendGraphics の画像凡例の大きさに使われます。

SHOWFEATURECOUNT

このパラメータは、凡例で地物カウントを有効にするために使用することができます。使用可能な値は以下の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

例:



RULE

This parameter is available on layers with *Rule-based* rendering and allows to build a legend with only the named rule symbol. It cannot be combined with BBOX parameter nor the JSON format. HEIGHT and WIDTH must be specified.

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=mylayer,
&RULE=myrulename,
&WIDTH=20,
&HEIGHT=20
```

RULELABEL

このパラメータは、アイテムラベルの描画を制御することができます。使用可能な値は次の通りです（大文字と小文字は区別されません）：

- TRUE: アイテムラベルを表示します
- FALSE: アイテムラベルを隠します
- AUTO: **単一のシンボル** レンダリングのレイヤのアイテムラベルを隠します

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=countries,airports
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&RULELABEL=AUTO
```

図 3.5: 単一シンボルのレイヤにはラベルを使わずに凡例をレンダリング

BOXSPACE

このパラメータでは、凡例フレームと内容の間隔をミリメートル単位で指定することができます。デフォルトの間隔は 2 mm です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
```

(次のページに続く)

(前のページからの続き)

```
&REQUEST=GetLegendGraphic
&LAYERS=airports
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&BOXSPACE=0
```

図 3.6: 左は BOXSPACE=0、右は BOXSPACE=15

LAYERSPACE

このパラメータでは、レイヤ間の垂直間隔をミリメートル単位で指定することができます。デフォルトの間隔値は、3 mm です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&LAYERSPACE=0
```

図 3.7: 左は LAYERSPACE=0、右は LAYERSPACE=10

LAYERTITLESPACE

このパラメータでは、レイヤのタイトルとそれに続くアイテムの間の垂直間隔をミリメートル単位で指定することができます。デフォルトの間隔は 3 mm です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&LAYERTITLESPACE=0
```

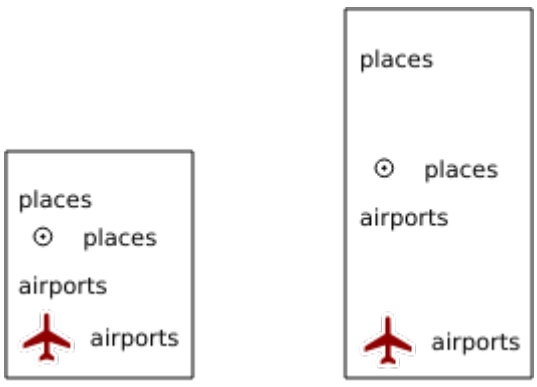


図 3.8: 左は LAYERTITLESPACE=0、右は LAYERTITLESPACE=10

SYMBOLSPACE

このパラメータでは、シンボルとそれに続くアイテムの間の垂直間隔をミリメートル単位で指定します。デフォルトの間隔は 2 mm です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=countries
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&SYMBOLSPACE=0
```

図 3.9: 左は SYMBOLSPACE=0、右は SYMBOLSPACE=5

ICONLABELSPACE

このパラメータでは、シンボルとラベルテキストの間の水平間隔をミリメートル単位で指定することができます。デフォルトの間隔は 2 mm です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=getlegendgraphic
&LAYERS=countries,
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
```

(次のページに続く)

(前のページからの続き)

```
&TRANSPARENT=TRUE
&ICONLABELSPACE=0
```

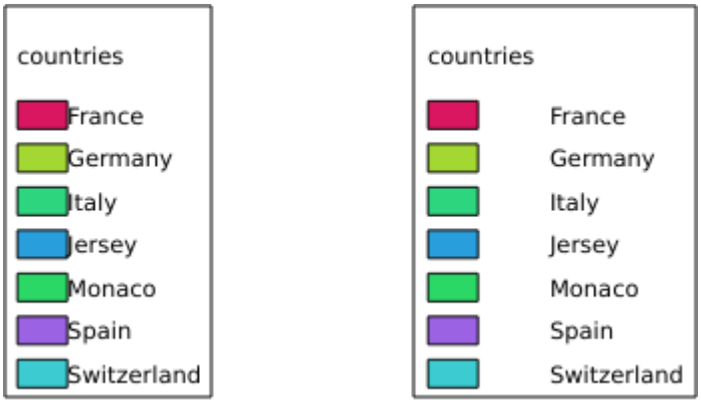


図 3.10: 左は ICONLABELSPACE=0、右は ICONLABELSPACE=10

SYMBOLWIDTH

このパラメータでは、シンボルプレビューの幅をミリメートル単位で指定することができます。デフォルトの幅は 7 mm です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=countries,
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&SYMBOLWIDTH=2
```

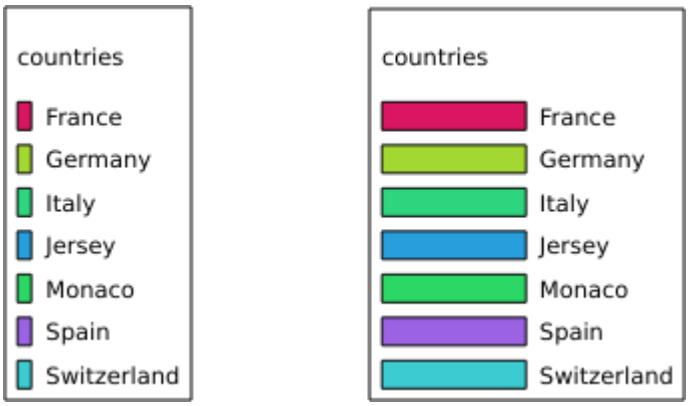


図 3.11: 左は SYMBOLWIDTH=2、右は SYMBOLWIDTH=20

SYMBOLHEIGHT

このパラメータでは、シンボルプレビューの高さをミリメートル単位で指定することができます。デフォルトでは、高さは 4 mm です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=countries,
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&SYMBOLHEIGHT=2
```

図 3.12: 左は SYMBOLHEIGHT=2、右は SYMBOLHEIGHT=6

LAYERTITLE

このパラメータは、レイヤのタイトルを描画するかどうかを指定します。

利用可能な値は次の通りです（大文字と小文字は区別されません）:

- TRUE （デフォルト値）
- FALSE

LAYERFONTFAMILY

このパラメータは、レイヤタイトルの描画に使用するフォントファミリーを指定します。

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=countries
&LAYERFONTFAMILY=monospace
```

LAYERFONTBOLD

このパラメータは、レイヤのタイトルを太字で描画するかどうかを指定します。使用可能な値は以下の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&LAYERFONTBOLD=TRUE
```

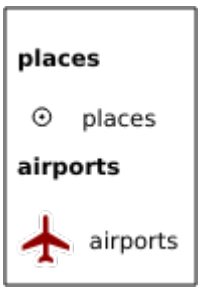


図 3.13: LAYERFONTBOLD=TRUE の凡例

LAYERFONTSIZE

このパラメータは、レイヤのタイトルを描画する際のフォントサイズをポイント単位で指定します。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&LAYERFONTSIZE=20
```

図 3.14: LAYERFONTSIZE=20 の凡例

LAYERFONTITALIC

このパラメータは、レイヤーのタイトルを斜体で描画するかどうかを指定します。使用可能な値は以下の通りです（大文字と小文字は区別されません）:

- TRUE
- FALSE

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&LAYERFONTITALIC=TRUE
```

図 3.15: LAYERFONTITALIC=TRUE の凡例

LAYERFONTCOLOR

このパラメータはレイヤタイトルの色を指定します。色はリテラル（red, green, ...）または 16 進数表記（0xFF0000, 0x00FF00, ...）で指定できます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&LAYERFONTCOLOR=0x5f9930
```

図 3.16: LAYERFONTCOLOR=0x5f9930 の凡例

ITEMFONTFAMILY

このパラメータは、アイテムラベルを描画するのに使うフォントファミリーを指定します。

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=countries
&ITEMFONTFAMILY=monospace
```

ITEMFONTBOLD

このパラメータは、アイテムラベルを太字で描画するかどうかを指定します。使用可能な値は次の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&ITEMFONTBOLD=TRUE
```

図 3.17: ITEMFONTBOLD=TRUE の凡例

ITEMFONTSIZE

このパラメータは、レイヤのタイトルを描画する際のフォントサイズをポイント単位で指定します。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&ITEMFONTSIZE=20
```

図 3.18: ITEMFONTSIZE=30 の凡例

ITEMFONTITALIC

このパラメータは、アイテムラベルを斜体で描画するかどうかを指定します。利用可能な値は次の通りです（大文字と小文字は区別されません）：

- TRUE
- FALSE

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&ITEMFONTITALIC=TRUE
```

図 3.19: ITEMFONTITALIC=TRUE の凡例

ITEMFONTCOLOR

このパラメータは、アイテムラベルの色を指定します。色はリテラル（red, green, ...）または 16 進数表記（0xFF0000, 0x00FF00, ...）で指定することができます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphic
&LAYERS=airports,places
&BBOX=43.20,-2.93,49.35,8.32
&CRS=EPSG:4326
&TRANSPARENT=TRUE
&ITEMFONTCOLOR=0x5f9930
```

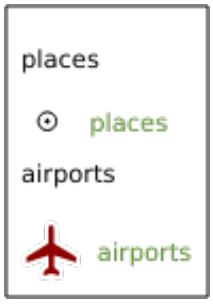


図 3.20: ITEMFONTCOLOR=0x5f9930 の凡例

ADDLAYERGROUPS

This parameter specifies if the JSON output will also display the names of the layers groups (and subgroups) in the legend entry. Possible values are:

- TRUE: display the groups labels
- FALSE (default): hide the groups labels

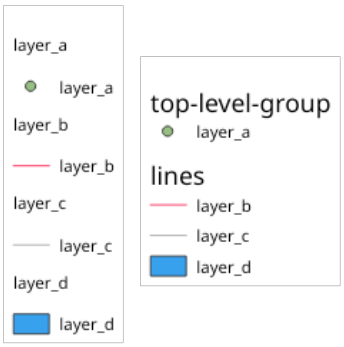


図 3.21: Legend without (left) and with (right) layer groups labels display

The corresponding JSON output showing groups name would look like:

```
{ "nodes":
  [{ "nodes":
     [{ "icon":
↪"iVBORw0KGgoAAAANSUhEUgAAABQAAAAUCAYAAACNiR0NAAAACXBIWXMAABYlAAAWJQFJUiTwAAAAUklEQVQ4jWNgGAXDHzASU
↪37l28evBKKisr/
↪0+a7IMiNi93C15DcUpgM4wYQ5nwuZAcMIQNvHv3LuO83C0kG0hysoFZRrJNo2AYAQC87BpkGQj1fwAAAABJRU5ErkJggg==
↪",
        "title":"layer_a", "type":"layer"
     },
     { "nodes":
       [{ "icon":
↪"iVBORw0KGgoAAAANSUhEUgAAABMAAAAUCAYAAABvVQZ0AAAACXBIWXMAABYlAAAWJQFJUiTwAAAAHUlEQVQ4jWNgGAWjYCQDx
↪",
          "title":"layer_b", "type":"layer"
        },
        { "icon":
↪"iVBORw0KGgoAAAANSUhEUgAAABMAAAAUCAYAAABvVQZ0AAAACXBIWXMAABYlAAAWJQFJUiTwAAAAHUlEQVQ4jWNgGAWjYCQDx
↪8aaEeMglEwYAAAaIoCzTtn5XoAAAAASUVORK5CYII=",
          "title":"layer_c", "type":"layer"
       }],
       "title":"lines", "type":"group"
     },
     { "icon":
↪"iVBORw0KGgoAAAANSUhEUgAAABMAAAATCAYAAAByUDbMAAAACXBIWXMAABYlAAAWJQFJUiTwAAAAKklEQVQ4jWNUVlX/
↪z0AlwMLAwMDAnXmQKoYxUcWUUcNGDRs1bNSwYWYYACXDAsvQaTuVAAAAAElFTkSuQmCC",
       "title":"layer_d", "type":"layer"
     }],
     "title":"top-level-group", "type":"group"
  }],
  "title":""
}
```


3.2.5 GetStyle(s)

OGC WMS 1.1.1 仕様に準拠した **GetStyle** (または **GetStyles**) リクエストの標準パラメータ：

パラメータ	必須	説明
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称（**GetStyle** 又は **GetStyles**）
LAYERS	Yes	クエリするレイヤ

GetStyle リクエストは、以下のベンダーパラメータもサポートしています:

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetStyles
&LAYERS=mylayer1,mylayer2
```

3.2.6 DescribeLayer

OGC WMS 1.1.1 及び 1.3.0 仕様に従った **DescribeLayer** リクエストの標準パラメータ:

パラメータ	必須	説明
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**DescribeLayer**)
LAYERS	Yes	説明するレイヤ
SLD_VERSION	Yes	SLD のバージョン

DescribeLayer リクエストは次のベンダーパラメータもサポートしています:

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=DescribeLayer
&SLD_VERSION=1.1.0
&LAYERS=mylayer1
```

XML 文書は次のようになります:

```
<DescribeLayerResponse xmlns="http://www.opengis.net/sld" xmlns:xsi="http://www.w3.
↪org/2001/XMLSchema-instance" xmlns:ows="http://www.opengis.net/ows" xmlns:xlink=
↪"http://www.w3.org/1999/xlink" xmlns:se="http://www.opengis.net/se"␣
↪xsi:schemaLocation="http://www.opengis.net/sld http://schemas.opengis.net/sld/1.1.0/
↪DescribeLayer.xsd">
    <Version>1.1.0</Version>
```

(次のページに続く)

(前のページからの続き)

```
    <LayerDescription>
        <owsType>wfs</owsType>
        <se:OnlineResource xlink:href="http://localhost/qgisserver" xlink:type="simple
↪"/>
        <TypeName>
            <se:FeatureTypeName>my_vector_layer</se:FeatureTypeName>
        </TypeName>
    </LayerDescription>
    <LayerDescription>
        <owsType>wcs</owsType>
        <se:OnlineResource xlink:href="http://localhost/qgisserver" xlink:type="simple
↪"/>
        <TypeName>
            <se:FeatureTypeName>my_raster_layer</se:FeatureTypeName>
        </TypeName>
    </LayerDescription>
</DescribeLayerResponse>
```

SLD_VERSION

このパラメータでは、SLD のバージョンを指定することができます。**値** *`1.1.0`* のみが利用できます。

3.2.7 GetPrint

QGIS サーバには、PDF またはピクセル形式のプリントレイアウト出力を作成する機能があります。公表されたプロジェクトでのプリントレイアウトがテンプレートとして使用されます。GETPRINT 要求では、クライアントは含まれているレイアウト地図とラベルのパラメータを指定する可能性があります。

GetPrint リクエストは、 *redlining*、 *external WMS layers* のほか、次のパラメータをサポートします:

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**GetPrint**)
VERSION	No	サービスのバージョン
LAYERS	No	表示するレイヤ
TEMPLATE	Yes	使用するレイアウトのテンプレート
SRS / CRS	Yes	座標参照系
FORMAT	No	出力形式
FORMAT_OPTIONS	No	指定されたファイル形式のオプション Only for `FORMAT=application/pdf`
ATLAS_PK	No	地図帳機能
STYLES	No	レイヤのスタイル
TRANSPARENT	No	透明な背景
OPACITIES	No	レイヤまたはグループの不透明度
SELECTION	No	地物を強調します
mapX:EXTENT	No	map *X* の範囲
mapX:LAYERS	No	map *X* のレイヤ
mapX:STYLES	No	map *X* のレイヤのスタイル
mapX:SCALE	No	map *X* のレイヤの縮尺
mapX:ROTATION	No	map *X* の角度
mapX:GRID_INTERVAL_X	No	map *X* の x 軸のグリッド間隔
mapX:GRID_INTERVAL_Y	No	map *X* の y 軸のグリッド間隔

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetPrint
&MAP=/home/qgis/projects/world.qgs
&CRS=EPSG:4326
&FORMAT=png
&TEMPLATE=Layout%201
&map0:EXTENT=-180,-90,180,90
&map0:LAYERS=mylayer1,mylayer2,mylayer3
&map0:OPACITIES=125,200,125
&map0:ROTATION=45
```

レイアウトテンプレートには複数のマップが含まれている可能性があることに注意してください。このとき、特定のマップを設定したい場合は、`mapX:` パラメータを使う必要があります。ここで、`X` は **GetProjectSettings** リクエストのおかげで取得できる正の数です。

例:

```
<WMS_Capabilities>
...
<ComposerTemplates xsi:type="wms:_ExtendedCapabilities">
<ComposerTemplate width="297" height="210" name="Druckzusammenstellung 1">
<ComposerMap width="171" height="133" name="map0"/>
<ComposerMap width="49" height="46" name="map1"/></ComposerTemplate>
</ComposerTemplates>
...
</WMS_Capabilities>
```

TEMPLATE

このパラメータを使って、印刷に使うレイアウトテンプレートの名前を指定することができます。

FORMAT

このパラメータは、マップ画像の形式を指定します。使用可能な値は以下の通りです:

- png （デフォルト値）
- image/png
- jpg
- jpeg
- image/jpeg
- svg
- image/svg
- image/svg+xml
- pdf
- application/pdf

もし FORMAT パラメータがこれらの値のいずれとも異なる場合は、例外が返されます。

FORMAT_OPTIONS

This parameter can be used to specify options for the selected format. Only for FORMAT=application/pdf in GetPrint requests. Takes a list of key:value pairs separated by semicolon:

- RASTERIZE_WHOLE_IMAGE: whether the whole pdf should be exported as an image. Default: false.
- FORCE_VECTOR_OUTPUT: whether pdf should be exported as vector. Default: false.
- APPEND_GEOREFERENCE: whether georeference info shall be added to the pdf. Default: true.

- EXPORT_METADATA: whether metadata shall be added to the pdf. Default: true.
- TEXT_RENDER_FORMAT: sets the text render format for pdf export. It can be AlwaysOutlines (default) or AlwaysText.
- SIMPLIFY_GEOMETRY: whether features geometries shall be simplified. Default: true.
- WRITE_GEO_PDF: whether a GeoPDF shall be exported. Default: false.
- USE_ISO_32000_EXTENSION_FORMAT_GEOREFERENCING: whether Iso32000 georeferencing shall be used. Default: false.
- USE_OGC_BEST_PRACTICE_FORMAT_GEOREFERENCING: whether OGC best practice georeferencing shall be used. Default: false.
- EXPORT_THEMES: a comma separated list of map themes to use for a GeoPDF export
- PREDEFINED_MAP_SCALES: a comma separated list of map scales to render the map
- LOSSLESS_IMAGE_COMPRESSION: whether images embedded in pdf must be compressed using a lossless algorithm. Default: false.
- DISABLE_TILED_RASTER_RENDERING: whether rasters shall be untiled in the pdf. Default: false.

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetPrint
&MAP=/home/qgis/projects/world.qgs
&CRS=EPSG:4326
&FORMAT=pdf
&TEMPLATE=Layout%201
&FORMAT_OPTIONS=FORCE_VECTOR_OUTPUT:TRUE;TEXT_RENDER_FORMAT:AlwaysOutlines;PREDEFINED_
↪MAP_SCALES:250
```

ATLAS_PK

このパラメータは、印刷したい地物を指定することで、地図帳レンダリングを有効化することができます。すべての地物を含む地図帳を取得するには、* シンボルを使用することができます（プロジェクト設定で許されている地物の最大数に従います）。

FORMAT が pdf の場合、地物ページを組み合わせた 1 つの PDF ドキュメントが返されます。それ以外の形式の場合は、1 つのページが返されます。

mapX:EXTENT

このパラメータは、レイアウトマップアイテムの範囲を xmin,ymin,xmax,ymax で指定します。

mapX:ROTATION

このパラメータは、マップの角度を度で指定します。

mapX:GRID_INTERVAL_X

このパラメータは、X 方向のグリッド線密度を指定します。

mapX:GRID_INTERVAL_Y

このパラメータは、Y 方向のグリッド線密度を指定します。

mapX:SCALE

このパラメータは、レイアウト地図アイテムの地図縮尺を指定します。これは、クライアントとサーバーで縮尺分母を計算するアルゴリズムが異なる場合でもレイヤとラベルの縮尺に基づく可視性を確保するのに役立ちます。

mapX:LAYERS

このパラメータは、レイアウトマップアイテムのレイヤを指定します。このパラメータの詳細については *GetMap Layers* を参照してください。

mapX:STYLES

このパラメータは、特定のレイアウトマップアイテムに定義されているレイヤのスタイルを指定します。このパラメータの詳細については *GetMap Styles* を参照してください。

3.2.8 GetProjectSettings

このリクエストタイプは *GetCapabilities* と同様に動作しますが、より QGIS Server に特化しており、 *GetCapabilities* の出力では得られない追加情報をクライアントが読み取ることができるようになっています：

- レイヤーの初期可視性
- ベクターの属性とその編集タイプに関する情報

- レイヤーの順序と描画順序に関する情報
- WFS に掲載されたレイヤのリスト
- レイヤツリー内のグループが mutually exclusive であるかどうかを表示します

GetProjectSettings リクエストは次のパラメータをサポートしています:

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**GetProjectSettings**)

3.2.9 GetSchemaExtension

GetSchemaExtension リクエストは、QGIS Server が実装するような WMS サービスのオプションの拡張機能および操作を取得することができます。

パラメータ	必須	説明
MAP	Yes	QGIS プロジェクトファイル
SERVICE	Yes	サービスの名称 (**WMS**)
REQUEST	Yes	リクエストの名称 (**GetSchemaExtension**)

URL の例:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetSchemaExtension
```

XML 文書は次のようになります:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:wms="http://www.opengis.net/wms
↪" xmlns:qgs="http://www.qgis.org/wms" targetNamespace="http://www.qgis.org/wms"␣
↪elementFormDefault="qualified" version="1.0.0">
    <import namespace="http://www.opengis.net/wms" schemaLocation="http://schemas.
↪opengis.net/wms/1.3.0/capabilities_1_3_0.xsd"/>
    <element name="GetPrint" type="wms:OperationType" substitutionGroup="wms:_
↪ExtendedOperation"/>
    <element name="GetStyles" type="wms:OperationType" substitutionGroup="wms:_
↪ExtendedOperation"/>
</schema>
```

3.2.10 外部 WMS レイヤ

QGIS Server では、WMS *GetMap* および WMS *GetPrint* リクエストに外部の WMS サーバーからのレイヤを含めることができます。これは、ウェブクライアントがウェブマップで外部の背景レイヤを使用する場合に特に便利です。パフォーマンス上の理由から、このようなレイヤは Web クライアントから直接リクエストされるべきです（QGIS サーバーを経由してカスケードせずに）。しかし、印刷では、これらのレイヤは、印刷されたマップに表示されるように、QGIS サーバー経由でカスケードされるべきです。

外部レイヤは、LAYERS パラメータに EXTERNAL_WMS: <layername>として追加することができます。外部 WMS レイヤのパラメータ（url, format, dpiMode, crs, layers, styles など）は，後でサービスパラメータ <layername>: <parameter>として与えることができます。*GetMap* リクエストでは、次のようになります:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&LAYERS=EXTERNAL_WMS:basemap,layer1,layer2
&OPACITIES=100,200,255
&STYLES=,,
&basemap:url=http://externalserver.com/wms.fcgi
&basemap:format=image/jpeg
&basemap:dpiMode=7
&basemap:crs=EPSG:2056
&basemap:layers=orthofoto
&basemap:styles=default
```

同様に、外部レイヤーは *GetPrint* リクエストで使用することができます：

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetPrint
&TEMPLATE=A4
&map0:layers=EXTERNAL_WMS:basemap,layer1,layer2
&map0:EXTENT=<minx,miny,maxx,maxy>
&OPACITIES=100,200,255
&basemap:url=http://externalserver.com/wms.fcgi
&basemap:format=image/jpeg
&basemap:dpiMode=7
&basemap:crs=EPSG:2056
&basemap:layers=orthofoto
&basemap:styles=default
```

3.2.11 赤線引き

この機能は *GetMap* と *GetPrint* のリクエストで使用することができます。

赤線引き機能は、サーバーが返す標準的な画像（マップ）の上に重ねるジオメトリとラベルをリクエストに渡すために使用されます。ユーザーはこれにより、標準的なマップにはない領域や場所などを強調したり、コメント（ラベル）を追加したりすることができます。

GetMap リクエストは、以下の形式です：

```
http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetMap
&HIGHLIGHT_GEOM=POLYGON((590000 5647000, 590000 6110620, 2500000 6110620, 2500000␣
↪5647000, 590000 5647000))
&HIGHLIGHT_SYMBOL=<StyledLayerDescriptor><UserStyle><Name>Highlight</Name>
↪<FeatureTypeStyle><Rule><Name>Symbol</Name><LineSymbolizer><Stroke><SvgParameter␣
↪name="stroke">%23ea1173</SvgParameter><SvgParameter name="stroke-opacity">1</
↪SvgParameter><SvgParameter name="stroke-width">1.6</SvgParameter></Stroke></
↪LineSymbolizer></Rule></FeatureTypeStyle></UserStyle></StyledLayerDescriptor>
&HIGHLIGHT_LABELSTRING=Write label here
&HIGHLIGHT_LABELSIZE=16
&HIGHLIGHT_LABELCOLOR=%23000000
&HIGHLIGHT_LABELBUFFERCOLOR=%23FFFFFF
&HIGHLIGHT_LABELBUFFERSIZE=1.5
```

GetPrint <wms_getprint>`と同等の形式です（どのマップに赤線が引かれているかを知るために ``mapX:` パラメータが追加されていることに注意してください）：

```
http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetPrint
&map0:HIGHLIGHT_GEOM=POLYGON((590000 5647000, 590000 6110620, 2500000 6110620,␣
↪2500000 5647000, 590000 5647000))
&map0:HIGHLIGHT_SYMBOL=<StyledLayerDescriptor><UserStyle><Name>Highlight</Name>
↪<FeatureTypeStyle><Rule><Name>Symbol</Name><LineSymbolizer><Stroke><SvgParameter␣
↪name="stroke">%23ea1173</SvgParameter><SvgParameter name="stroke-opacity">1</
↪SvgParameter><SvgParameter name="stroke-width">1.6</SvgParameter></Stroke></
↪LineSymbolizer></Rule></FeatureTypeStyle></UserStyle></StyledLayerDescriptor>
&map0:HIGHLIGHT_LABELSTRING=Write label here
&map0:HIGHLIGHT_LABELSIZE=16
&map0:HIGHLIGHT_LABELCOLOR=%23000000
&map0:HIGHLIGHT_LABELBUFFERCOLOR=%23FFFFFF
&map0:HIGHLIGHT_LABELBUFFERSIZE=1.5
```

上記のリクエストによって出力された画像は次のとおりです、ポリゴンとラベルが通常のマップの上に描

かれています：

図 3.22: 赤線引きパラメータを含む GetMap リクエストに対するサーバーの応答

このリクエストには、赤線引き機能を制御するためのいくつかのパラメータがあることがわかります。その全リストは以下の通りです:

- **highlight_geom**: POINT、MULTILINESTRING、POLYGON などを追加することができます。マルチパートジオメトリをサポートしています。以下はその例です：`HIGHLIGHT_GEOM=MULTILINESTRING((0 0, 0 1, 1 1))`. 座標は GetMap/GetPrint リクエストの CRS にあるはずです。
- **HIGHLIGHT_LABELBUFFERCOLOR**: このパラメーターは、ラベルバッファの色を制御します。
- **HIGHLIGHT_LABELBUFFERSIZE**: このパラメーターは、ラベルバッファのサイズを制御します。
- **HIGHLIGHT_LABELCOLOR**: このパラメーターは、ラベルの色を制御します。
- **HIGHLIGHT_LABEL_DISTANCE**: 地物（例 point 又はライン）とラベルの間の距離を mm で制御します
- **highlight_labelfont**： このパラメータは、ラベルのフォントを制御します（例：Arial)。
- **HIGHLIGHT_LABEL_HORIZONTAL_ALIGNMENT**： ラベルを水平方向に指定された整列（例：'left', 'center', 'right' ）で配置します。

- **HIGHLIGHT_LABEL_ROTATION**: ラベルの角度を度で制御します
- **HIGHLIGHT_LABELSIZE**: このパラメーターは、ラベルのサイズを制御します。
- **HIGHLIGHT_LABELSTRING** ：ラベルのテキストをこのパラメーターに渡すことができます。
- **HIGHLIGHT_LABEL_VERTICAL_ALIGNMENT**： 指定された整列（例：'top', 'half', 'bottom'）でラベルをポイント上に垂直に配置します。
- **HIGHLIGHT_SYMBOL** ：ジオメトリの輪郭をどのように表示するかを制御し、ストロークの幅、色、不透明度を変更できます。

3.3 Web Feature Service (WFS)

QGIS Server に実装されている **1.0.0** 及び **1.1.0** WFS 標準は、QGIS プロジェクトから地理的な地物を照会するための HTTP インターフェイスを提供します。典型的な WFS リクエストは、使用する QGIS プロジェクトとクエリするレイヤを定義します。

サービスのバージョン番号に応じた仕様書:

- WFS 1.0.0
- WFS 1.1.0

QGIS Server が提供する標準リクエスト

リクエスト	説明
GetCapabilities	サーバーに関する情報を含む XML メタデータを返します
GetFeature	選択された地物を返します
DescribeFeatureType	地物の型とプロパティの説明を返します
Transaction	地物の挿入、更新または削除を可能にします

3.3.1 GetCapabilities

OGC WFS 1.0.0 及び 1.1.0 仕様に従った、**GetCapabilities** リクエストの標準パラメータ

パラメーター	必須	説明
SERVICE	はい	サービスの名称 (**WFS**)
REQUEST	はい	リクエストの名称 (**GetCapabilities**)
VERSION	いいえ	サービスのバージョン

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメーター	必須	説明
MAP	はい	QGIS プロジェクトファイル

VERSION

このパラメータでは、使用するサービスのバージョンを指定することができます。VERSION パラメータに指定できる値は以下の通りです:

- 1.0.0
- 1.1.0

リクエストにバージョンが示されなかった場合、1.1.0 がデフォルトとして使われます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
&...
```

3.3.2 GetFeature

OGC WFS 1.0.0 及び 1.1.0 仕様に従った、**GetFeature** リクエストの標準パラメータ:

パラメーター	必須	説明
SERVICE	はい	サービスの名称 (**WFS**)
REQUEST	はい	リクエストの名称 (**GetFeature**)
VERSION	いいえ	サービスのバージョン
TYPENAME	いいえ	レイヤの名前
FEATUREID	いいえ	id で地物をフィルタする
OUTPUTFORMAT	いいえ	出力形式
RESULTTYPE	いいえ	結果の型
PROPERTYNAME	いいえ	返すプロパティの名前
MAXFEATURES	いいえ	返す地物の最大数
SRSNAME	いいえ	座標参照系
FILTER	いいえ	OGC フィルターエンコーディング
BBOX	いいえ	マップの範囲
SORTBY	いいえ	結果を並び替える

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメーター	必須	説明
MAP	はい	QGIS プロジェクトファイル
STARTINDEX	いいえ	ページング
GEOMETRYNAME	いいえ	返すジオメトリの型
EXP_FILTER	いいえ	式フィルタ

TYPENAME

このパラメータでは、レイヤ名を指定することができ、FEATUREID がセットされていないときは必須です。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
&REQUEST=GetFeature
&TYPENAME=countries
```

FEATUREID

このパラメータでは、特定の地物の ID を指定することができ、typename.fid,typename.fid,... のように形成されます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&FEATUREID=countries.0,places.1
```

XML レスポンス:

```
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs" xmlns:ogc="http://www.
↪opengis.net/ogc" xmlns:gml="http://www.opengis.net/gml" xmlns:ows="http://www.
↪opengis.net/ows" xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:qgs="http://www.
↪qgis.org/gml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"␣
↪xsi:schemaLocation="http://www.opengis.net/wfs http://schemas.opengis.net/wfs/1.1.0/
↪wfs.xsd http://www.qgis.org/gml http://192.168.1.15/qgisserver?SERVICE=WFS&
↪VERSION=1.1.0&REQUEST=DescribeFeatureType&TYPENAME=countries,places&
↪OUTPUTFORMAT=text/xml; subtype%3Dgml/3.1.1">
  <gml:boundedBy>
    ...
  </gml:boundedBy>
```

(次のページに続く)

(前のページからの続き)

```
  <gml:featureMember>
    <qgs:countries gml:id="countries.1">
      ...
    </qgs:countries>
  </gml:featureMember>
  <gml:featureMember>
    <qgs:places gml:id="places.1">
      ...
    </qgs:places>
  </gml:featureMember>
</wfs:FeatureCollection>
```

OUTPUTFORMAT

このパラメータを使って、レスポンスの形式を指定することができます。VERSION が 1.1.0 より大きいか等しい場合、GML3 がデフォルトの形式です。それ以外の場合は GML2 が使われます。

利用可能な値は次の通り:

- gml2
- text/xml; subtype=gml/2.1.2
- gml3
- text/xml; subtype=gml/3.1.1
- geojson
- application/vnd.geo+json,
- application/vnd.geo json
- application/geo+json
- application/geo json
- application/json

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&FEATUREID=countries.0
&OUTPUTFORMAT=geojson
```

GeoJSON レスポンス:

```
{
    "type":"FeatureCollection",
    "bbox":[
        -180,
        -90,
        180,
        83.6236
    ],
    "features":[
        {
            "bbox":[
                -61.891113,
                16.989719,
                -61.666389,
                17.724998
            ],
            "geometry":{
                "coordinates":[
                    "..."
                ],
                "type":"MultiPolygon"
            },
            "id":"countries.1",
            "properties":{
                "id":1,
                "name":"Antigua and Barbuda"
            },
            "type":"Feature"
        }
    ]
}
```


RESULTTYPE

このパラメータを使って、返す結果の種類を指定することができます。利用可能な値は以下の通りです:

- `results`: デフォルトの動作
- `hits`: 地物の数だけを返します

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
&REQUEST=GetFeature
```

(次のページに続く)

(前のページからの続き)

```
&RESULTTYPE=hits
&...
```

PROPERTYNAME

このパラメータを使って、特定のプロパティを返すよう指定することができます。プロパティは TYPENAME または FEATUREID と共にマッピングされている必要があります:

有効な URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&PROPERTYNAME=name
&TYPENAME=places
```

対して、次の URL では例外が返されます:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&PROPERTYNAME=name
&TYPENAME=places,countries
```

```
<ServiceExceptionReport xmlns="http://www.opengis.net/ogc" version="1.2.0">
    <ServiceException code="RequestNotWellFormed">There has to be a 1:1 mapping␣
↪between each element in a TYPENAME and the PROPERTYNAME list</ServiceException>
</ServiceExceptionReport>
```

MAXFEATURES

このパラメータは、リクエストによって返される地物の数を制限することができます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&TYPENAME=places
&MAXFEATURES=1000
```

注釈: このパラメータは、下層のベクタレイヤが重い場合にパフォーマンスを向上させるのに有効な場合があります。

SRSNAME

このパラメータは、BBOX CRS と同様にレスポンス出力の空間参照系を指定することができます。パラメータは EPSG:XXXX のように形成する必要があります。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&TYPENAME=places
&SRSNAME=EPSG:32620
```

FILTER

このパラメータは、OGC Filter Encoding standard で定義されている **Filter Encoding** 言語を使ってレスポンスをフィルタすることができます。例えば、以下のようになります:

```
http://localhost/qgisserver?
SERVICE=WFS&
REQUEST=GetFeature&
TYPENAME=places&
FILTER=<Filter><PropertyIsEqualTo><PropertyName>name</PropertyName><Literal>Paris</
↪Literal></PropertyIsEqualTo></Filter>
```

型名が複数ある場合、フィルターは括弧で囲む必要があります:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&TYPENAME=places,countries
&FILTER=(<Filter><PropertyIsEqualTo><PropertyName>name</PropertyName><Literal>Paris</
↪Literal></PropertyIsEqualTo></Filter>)(<Filter><PropertyIsEqualTo><PropertyName>name
↪</PropertyName><Literal>France</Literal></PropertyIsEqualTo></Filter>)
```

ポリゴンと交差する地物をフィルタする:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&VERSION=1.1.0
&TYPENAME=places
&FILTER=<Filter xmlns="http://www.opengis.net/ogc">
           <Intersects>
               <PropertyName>geometry</PropertyName>
               <Polygon xmlns="http://www.opengis.net/gml" srsName="EPSG:4326">
```

(次のページに続く)

(前のページからの続き)

```
                <exterior>
                    <LinearRing>
                        <posList>
                            -0.6389 42.5922
                            10.2683 51.9106
                            14.5196 41.0320
                            -0.6389 42.5922
                        </posList>
                    </LinearRing>
                </exterior>
            </Polygon>
        </Intersects>
    </Filter>
```

BBOX

このパラメータを使用すると、現在の CRS に従った単位でマップの範囲を指定することができます。座標はカンマで区切る必要があります。

SRSNAME パラメータは、範囲の CRS を指定することができます。指定しない場合、レイヤの CRS が使用されます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&TYPENAME=places
&BBOX=-11.84,42.53,8.46,50.98
```

FEATUREID パラメータは BBOX と共に使用することはできません。使用しようとすると、例外が発生します:

```
<ServiceExceptionReport xmlns="http://www.opengis.net/ogc" version="1.2.0">
  <ServiceException code="RequestNotWellFormed">FEATUREID FILTER and BBOX parameters␣
↪are mutually exclusive</ServiceException>
</ServiceExceptionReport>
```

SORTBY

このパラメータは、プロパティ値に従って結果の地物をソートすることができます。パラメータは propertyname SORTRULE のように形成する必要があります。

降順にソートする場合に SORTRULE に指定できる値は次の通りです:

- D
- +D
- DESC
- +DESC

昇順にソートする場合に、SORTRULE に指定できる値は次の通りです:

- A
- +A
- ASC
- +ASC

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&TYPENAME=places
&PROPERTYNAME=name
&MAXFEATURES=3
&SORTBY=name DESC
```

対応する結果です:

```
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs" xmlns:ogc="http://www.
↪opengis.net/ogc" xmlns:gml="http://www.opengis.net/gml" xmlns:ows="http://www.
↪opengis.net/ows" xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:qgs="http://www.
↪qgis.org/gml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"␣
↪xsi:schemaLocation="http://www.opengis.net/wfs http://schemas.opengis.net/wfs/1.1.0/
↪wfs.xsd http://www.qgis.org/gml http://192.168.1.15/qgisserver?SERVICE=WFS&
↪VERSION=1.1.0&REQUEST=DescribeFeatureType&TYPENAME=places&OUTPUTFORMAT=text/xml;␣
↪subtype%3Dgml/3.1.1">
    <gml:boundedBy>
        ...
    </gml:boundedBy>
    <gml:featureMember>
        <qgs:places gml:id="places.90">
            <qgs:name>Zagreb</qgs:name>
        </qgs:places>
```

(次のページに続く)

(前のページからの続き)

```
    </gml:featureMember>
    <gml:featureMember>
        <qgs:places gml:id="places.113">
            <qgs:name>Yerevan</qgs:name>
        </qgs:places>
    </gml:featureMember>
    <gml:featureMember>
        <qgs:places gml:id="places.111">
            <qgs:name>Yaounde</qgs:name>
        </qgs:places>
    </gml:featureMember>
</wfs:FeatureCollection>
```

GEOMETRYNAME

このパラメータは、返す地物のジオメトリの種類を指定するために使用できます。利用可能な値は以下のとおりです：

- extent
- centroid
- none

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
&REQUEST=GetFeature
&GEOMETRYNAME=centroid
&...
```

STARTINDEX

このパラメータは WFS 2.0 では標準ですが、WFS 1.0.0 では拡張です。

実際には、結果セットの一部の地物をスキップするために使用することができ、MAXFEATURES と組み合わせることで、結果をページ送りする機能を提供します。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
```

(次のページに続く)

(前のページからの続き)

```
&REQUEST=GetFeature
&STARTINDEX=2
&...
```

EXP_FILTER

このパラメータを使用すると、QGIS 式でレスポンスをフィルタすることができます。型名が複数ある場合、; 文字でフィルタを区切ることができます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&REQUEST=GetFeature
&TYPENAME=places,countries
&EXP_FILTER="name"='Paris';"name"='France'
```

3.3.3 DescribeFeatureType

OGC WFS 1.0.0 及び 1.1.0 仕様に準拠した **DescribeFeatureType** リクエストの標準パラメータ:

パラメーター	必須	説明
SERVICE	はい	サービスの名称 (**WFS**)
REQUEST	はい	リクエストの名称 (**DescribeFeatureType**)
VERSION	いいえ	サービスのバージョン
OUTPUTFORMAT	いいえ	レスポンスの形式
TYPENAME	いいえ	レイヤの名前

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメーター	必須	説明
MAP	はい	QGIS プロジェクトファイル

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
&REQUEST=DescribeFeatureType
&TYPENAME=countries
```

出力レスポンス:

```
<schema xmlns:ogc="http://www.opengis.net/ogc" xmlns:xsd="http://www.w3.org/2001/
↪XMLSchema" xmlns="http://www.w3.org/2001/XMLSchema" xmlns:qgs="http://www.qgis.org/
↪gml" xmlns:gml="http://www.opengis.net/gml" targetNamespace="http://www.qgis.org/gml
↪" version="1.0" elementFormDefault="qualified">
  <import schemaLocation="http://schemas.opengis.net/gml/3.1.1/base/gml.xsd"␣
↪namespace="http://www.opengis.net/gml"/>
  <element type="qgs:countriesType" substitutionGroup="gml:_Feature" name="countries"/
↪>
  <complexType name="countriesType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element minOccurs="0" type="gml:MultiPolygonPropertyType" maxOccurs="1"␣
↪name="geometry"/>
          <element type="long" name="id"/>
          <element nillable="true" type="string" name="name"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

3.3.4 Transaction

このリクエストでは、XML 文書を使用して、1 つまたは複数の地物を更新、削除、または追加することができます。**削除** の操作は POST リクエストと *OPERATION* パラメータで行うことができ、**追加** と **更新** の操作は POST リクエストでのみ行うことができます。

OGC WFS 1.0.0 及び 1.1.0 仕様に準拠した **Transaction** リクエストの標準パラメータ:

パラメーター	必須	説明
SERVICE	はい	サービスの名称 (**WFS**)
REQUEST	はい	リクエストの名称 (**Transaction**)
VERSION	いいえ	サービスのバージョン
FILTER	いいえ	OGC フィルターエンコーディング
BBOX	いいえ	マップの範囲
FEATUREID	いいえ	id で地物をフィルタする
TYPENAME	いいえ	レイヤの名前

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメーター	必須	説明
MAP	はい	QGIS プロジェクトファイル
OPERATION	いいえ	操作を指定します
EXP_FILTER	いいえ	式フィルタ

OPERATION

このパラメータを使用すると、専用の XML 文書を使用した POST リクエストを使用せずに、地物を削除することができます。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
&REQUEST=Transaction
&OPERATION=DELETE
&FEATUREID=24
```

注釈: FEATUREID、BBOX、FILTER の各パラメータは相互に排他的で、この順番で優先されます。

地物を追加する

POST リクエストの例:

```
wget --post-file=add.xml "http://localhost/qgisserver?SERVICE=WFS&REQUEST=Transaction"
```

使用する *add.xml* 文書:

```
<?xml version="1.0" encoding="UTF-8"?>
<wfs:Transaction service="WFS" version="1.0.0" xmlns:wfs="http://www.opengis.net/wfs"␣
↪xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ogc="http://www.opengis.
↪net/ogc" xmlns="http://www.opengis.net/wfs" updateSequence="0" xmlns:xlink="http://
↪www.w3.org/1999/xlink" xsi:schemaLocation="http://www.opengis.net/wfs http://
↪schemas.opengis.net/wfs/1.0.0/WFS-capabilities.xsd" xmlns:gml="http://www.opengis.
↪net/gml"  xmlns:ows="http://www.opengis.net/ows">
  <wfs:Insert idgen="GenerateNew">
    <qgs:places>
      <qgs:geometry>
        <gml:Point srsDimension="2" srsName="http://www.opengis.net/def/crs/EPSG/0/
↪4326">
```

(次のページに続く)

(前のページからの続き)

```
          <gml:coordinates decimal="." cs="," ts=" ">-4.6167,48.3833</gml:coordinates>
        </gml:Point>
      </qgs:geometry>
      <qgs:name>Locmaria-Plouzané</qgs:name>
    </qgs:places>
  </wfs:Insert>
</wfs:Transaction>
```

地物を更新する

POST リクエストの例:

```
wget --post-file=update.xml "http://localhost/qgisserver?SERVICE=WFS&
↪REQUEST=Transaction"
```

使用する *update.xml* 文書:

```
<?xml version="1.0" encoding="UTF-8"?>
<wfs:Transaction service="WFS" version="1.0.0" xmlns:wfs="http://www.opengis.net/wfs"␣
↪xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ogc="http://www.opengis.
↪net/ogc" xmlns="http://www.opengis.net/wfs" updateSequence="0" xmlns:xlink="http://
↪www.w3.org/1999/xlink" xsi:schemaLocation="http://www.opengis.net/wfs http://
↪schemas.opengis.net/wfs/1.0.0/WFS-capabilities.xsd" xmlns:gml="http://www.opengis.
↪net/gml"  xmlns:ows="http://www.opengis.net/ows">
    <wfs:Update typeName="places">
      <wfs:Property>
          <wfs:Name>name</wfs:Name>
          <wfs:Value>Lutece</wfs:Value>
      </wfs:Property>
      <ogc:Filter>
          <ogc:FeatureId fid="24"/>
      </ogc:Filter>
    </wfs:Update>
</wfs:Transaction>
```

地物を削除する

POST リクエストの例:

```
wget --post-file=delete.xml "http://localhost/qgisserver?SERVICE=WFS&
↪REQUEST=Transaction"
```

使用する *delete.xml* 文書

```
<?xml version="1.0" encoding="UTF-8"?>
<wfs:Transaction service="WFS" version="1.0.0" xmlns:wfs="http://www.opengis.net/wfs"␣
↪xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ogc="http://www.opengis.
↪net/ogc" xmlns="http://www.opengis.net/wfs" updateSequence="0" xmlns:xlink="http://
↪www.w3.org/1999/xlink" xsi:schemaLocation="http://www.opengis.net/wfs http://
↪schemas.opengis.net/wfs/1.0.0/WFS-capabilities.xsd" xmlns:gml="http://www.opengis.
↪net/gml"  xmlns:ows="http://www.opengis.net/ows">
    <wfs:Delete typeName="places">
        <ogc:Filter>
            <ogc:FeatureId fid="24"/>
        </ogc:Filter>
    </wfs:Delete>
</wfs:Transaction>
```

3.4 Web Coverage Service (WCS)

QGIS Server で実装されている **1.0.0** および **1.1.1** WCS 標準は、QGIS プロジェクトからの **カバレッジ** と呼ばれるラスターデータにアクセスする HTTP インターフェースを提供します。

仕様:

- WCS 1.0.0
- WCS 1.1.1

QGIS Server が提供する標準リクエスト

リクエスト	説明
GetCapabilities	サーバーに関する情報を含む XML メタデータを返します
DescribeCoverage	カバレッジに関する追加情報についての XML 文書を取得します
GetCoverage	カバレッジを取得します

3.4.1 GetCapabilities

OGC WCS 1.1.1 仕様に従った **GetCapabilities** リクエストの標準パラメータ:

パラメーター	必須	説明
SERVICE	はい	サービスの名称 (**WCS**)
REQUEST	はい	リクエストの名称 (**GetCapabilities**)
VERSION	いいえ	サービスのバージョン

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメーター	必須	説明
MAP	はい	QGIS プロジェクトファイル

URL の例:

```
http://localhost/qgisserver?
SERVICE=WCS
&VERSION=1.1.1
&REQUEST=GetCapabilities
```

WCS サービスの QGIS プロジェクトで、単独のラスタレイヤ（名前 `T20QPD_20171123T144719_TCI` ）を公開した場合の XML 文書の例:

```
<WCS_Capabilities xmlns="http://www.opengis.net/wcs" xmlns:xlink="http://www.w3.org/
↪1999/xlink" xmlns:gml="http://www.opengis.net/gml" xmlns:xsi="http://www.w3.org/
↪2001/XMLSchema-instance" version="1.0.0" updateSequence="0" xsi:schemaLocation=
↪"http://www.opengis.net/wcs http://schemas.opengis.net/wcs/1.0.0/wcsCapabilities.xsd
↪">
  <Service>
    ...
  </Service>
  <Capability>
    ...
  </Capability>
  <ContentMetadata>
    <CoverageOfferingBrief>
      <name>T20QPD_20171123T144719_TCI</name>
      <label>T20QPD_20171123T144719_TCI</label>
      <lonLatEnvelope srsName="urn:ogc:def:crs:OGC:1.3:CRS84">
        <gml:pos>-61.585973 16.331189</gml:pos>
        <gml:pos>-61.52537 16.400376</gml:pos>
      </lonLatEnvelope>
    </CoverageOfferingBrief>
  </ContentMetadata>
</WCS_Capabilities>
```

VERSION

このパラメータは、使用するサービスのバージョンを指定することができます。現在、このバージョン値は内部的には使用されておらず、常に `1.1.1` にフォールバックします。

3.4.2 DescribeCoverage

このリクエストは、基礎となるデータソースのフォーマット、バンドの数など、カバレッジに関する追加情報を取得することができます。OGC WCS 1.1.1 仕様に基づく **DescribeCoverage** リクエストの標準パラメータです：

パラメーター	必須	説明
SERVICE	はい	サービスの名称 (**WCS**)
REQUEST	はい	リクエストの名称 (**DescribeCoverage**)
VERSION	いいえ	サービスのバージョン
COVERAGE	いいえ	カバレッジレイヤを指定する (WCS 1.0.0)
IDENTIFIER	いいえ	カバレッジレイヤを指定します (WCS 1.1.1)

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメーター	必須	説明
MAP	はい	QGIS プロジェクトファイル

3 バンドの GeoTIFF ラスタレイヤの XML 文書は、次のようになります:

```
<CoverageDescription xmlns:xlink="http://www.w3.org/1999/xlink" xmlns="http://www.
↪opengis.net/wcs" xmlns:gml="http://www.opengis.net/gml" xmlns:xsi="http://www.w3.
↪org/2001/XMLSchema-instance" updateSequence="0" version="1.0.0" xsi:schemaLocation=
↪"http://www.opengis.net/wcs http://schemas.opengis.net/wcs/1.0.0/describeCoverage.
↪xsd">
  <CoverageOffering>
    <name>T20QPD_20171123T144719_TCI</name>
    <label>T20QPD_20171123T144719_TCI</label>
    <lonLatEnvelope srsName="urn:ogc:def:crs:OGC:1.3:CRS84">
      ...
    </lonLatEnvelope>
    <domainSet>
      ...
    </domainSet>
    <rangeSet>
      <RangeSet>
        <name>Bands</name>
        <label>Bands</label>
```

(次のページに続く)

(前のページからの続き)

```
        <axisDescription>
          <AxisDescription>
            <name>bands</name>
            <label>bands</label>
            <values>
              <singleValue>1</singleValue>
              <singleValue>2</singleValue>
              <singleValue>3</singleValue>
            </values>
          </AxisDescription>
        </axisDescription>
      </RangeSet>
    </rangeSet>
    <supportedCRSs>
    ...
    </supportedCRSs>
    <supportedFormats nativeFormat="raw binary">
      <formats>GeoTIFF</formats>
    </supportedFormats>
  </CoverageOffering>
</CoverageDescription>
```

COVERAGE

このパラメータは WCS 1.0.0 で定義されており、追加情報を照会するレイヤーを指定することができます。名前はカンマで区切る必要があります。

また、QGIS Server では、レイヤをその短い名前で選択するオプションが導入されました。レイヤの短い名前は、レイヤメニューの **プロパティ　メタデータ** で設定することができます。短い名前が定義されている場合、それがレイヤ名の代わりにデフォルトで使用されます:

```
http://localhost/qgisserver?
SERVICE=WCS
&REQUEST=DescribeCoverage
&COVERAGE=mylayer1name,mylayer2shortname
```

注釈: IDENTIFIER が設定されていない場合、COVERAGE は必須です。

IDENTIFIER

このパラメータは、WCS 1.1.1 の *COVERAGE* パラメータを置き換えるものです。しかし、QGIS Server は VERSION パラメータによるフィルタリングを行わないため、 IDENTIFIER と COVERAGE は同じ効果を持ちます。

注釈: COVERAGE が設定されていない場合、IDENTIFIER は必須です。IDENTIFIER と COVERAGE の両方のパラメータが定義されている場合、常に COVERAGE が優先的に使用されます。

3.4.3 GetCoverage

このリクエストによって、範囲や CRS のような特定の制約に従って、カバレッジを取得することができます。OGC WCS 1.1.1 仕様に基づく **DescribeCoverage** リクエストの標準パラメータです:

パラメーター	必須	説明
SERVICE	はい	サービスの名称 (**WCS**)
REQUEST	はい	リクエストの名称 (**GetCoverage**)
VERSION	いいえ	サービスのバージョン
COVERAGE	いいえ	カバレッジレイヤを指定する (WCS 1.0.0)
IDENTIFIER	いいえ	カバレッジレイヤを指定します (WCS 1.1.1)
WIDTH	はい	ピクセルによるレスポンスの幅
HEIGHT	はい	ピクセルによるレスポンスの高さ
BBOX	はい	CRS 単位によるマップの範囲
CRS	はい	範囲の座標参照系
RESPONSE_CRS	いいえ	レスポンスの座標参照系

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメーター	必須	説明
MAP	はい	QGIS プロジェクトファイル

BBOX

このパラメータは、現在の CRS の単位でマップの範囲を指定することができます。座標はカンマで区切らなければなりません。BBOX パラメータは minx,miny,maxx,maxy のような形になります。

URL の例:

```
http://localhost/qgisserver?
SERVICE=WCS
&REQUEST=GetCoverage
```

(次のページに続く)

(前のページからの続き)

```
&IDENTIFIER=T20QPD_20171123T144719_TCI
&BBOX=647533,1805950,660987,1813940
&CRS=EPSG:32620
```

CRS

このパラメータは BBOX パラメータの空間参照系を示すことができ、EPSG:XXXX のような形式をとる必要があります。

RESPONSE_CRS

このパラメータは、出力するレスポンスの空間参照系を示すことができ、``EPSG:XXXX``のような形式をとる必要があります。デフォルトでは、対応するカバレッジレイヤの CRS が使用されます。

WIDTH

このパラメータでは、出力画像の幅をピクセル単位で指定することができます。レスポンス画像の解像度は、この値に依存します。

HEIGHT

このパラメータでは、出力画像の高さをピクセル単位で指定することができます。レスポンス画像の解像度は、この値に依存します。

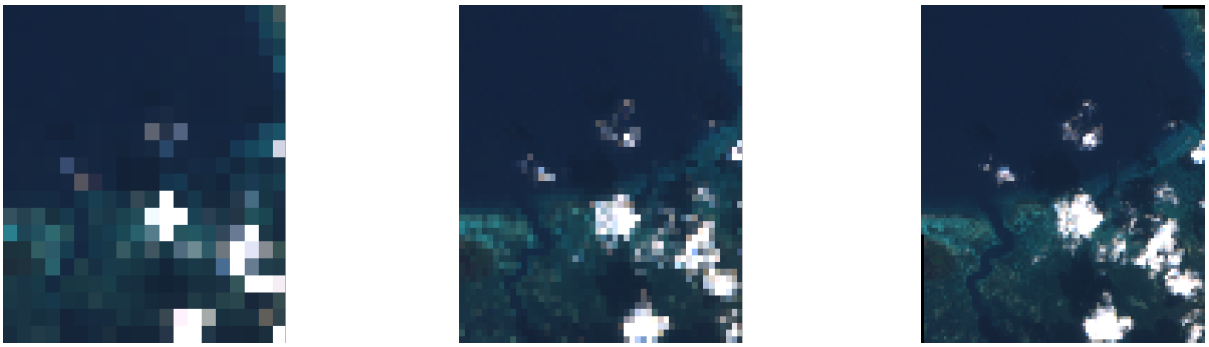


図 3.23: 左から、WIDTH=20&HEIGHT=20、WIDTH=50&HEIGHT=50、WIDTH=100&HEIGHT=100

3.5 ウェブマップタイルサービス (WMTS)

QGIS サーバに実装された **1.0.0** WMTS 標準は、QGIS プロジェクトから作成されたタイル地図画像を要求するための HTTP インターフェースを提供します。典型的な WMTS リクエストでは、使用する QGIS プロジェクト、レンダリングするレイヤなどの一部の WMS パラメータ、およびタイルパラメータを定義しました。

サービスの仕様書:

- WMTS 1.0.0

QGIS Server が提供する標準リクエスト

リクエスト	説明
GetCapabilities	サーバーに関する情報を含む XML メタデータを返します
GetTile	タイルを返します
GetFeatureInfo	ピクセル位置のデータ（ジオメトリと値）を取得します

3.5.1 GetCapabilities

OGC WMTS 1.0.0 仕様に従った **GetCapabilities** リクエストの標準パラメータ:

パラメータ	必須	説明
SERVICE	はい	サービスの名称 (**WMTS**)
REQUEST	はい	リクエストの名称 (**GetCapabilities**)

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメータ	必須	説明
MAP	はい	QGIS プロジェクトファイル

URL の例:

```
http://localhost/?
SERVICE=WMTS
&REQUEST=GetCapabilities
&MAP=/home/qgis/projects/world.qgs
```

3.5.2 GetTile

OGC WMTS 1.0.0 仕様に従った **GetTile** リクエストの標準パラメータ:

パラメータ	必須	説明
SERVICE	はい	サービスの名称 (**WMTS**)
REQUEST	はい	リクエストの名称 (**GetTile**)
LAYER	はい	レイヤ識別
FORMAT	はい	タイルの出力フォーマット
TILEMATRIXSET	はい	ピラミッドの名称
TILEMATRIX	はい	メッシュを作る
TILEROW	はい	メッシュの行座標
TILECOL	はい	メッシュの列座標

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメータ	必須	説明
MAP	はい	QGIS プロジェクトファイル

URL の例:

```
http://localhost/?
SERVICE=WMTS
&REQUEST=GetTile
&MAP=/home/qgis/projects/world.qgs
&LAYER=mylayer
&FORMAT=image/png
&TILEMATRIXSET=EPSG:4326
&TILEROW=0
&TILECOL=0
```

FORMAT

このパラメータを使ってタイル画像のフォーマットを指定することができます。使用できる値は：

- `jpg`
- `jpeg`
- `image/jpeg`
- `image/png`

`FORMAT` パラメータがこれらのいずれとも異なる場合、代わりに既定のフォーマット PNG が使われます。

TILEMATRIXSET

このパラメータは、基礎となるピラミッドを計算するときに使用する CRS を定義します。フォーマット: `EPSG:XXXX`。

TILEMATRIX

このパラメータは出力タイルで使用する格子を定義することができます。

TILEROW

このパラメータは格子から取得するタイルの行を選択することができます。

TILECOL

このパラメータは格子から取得するタイルの列を選択することができます。

3.5.3 GetFeatureInfo

OGC WMTS 1.0.0 仕様に従った **GetFeatureInfo** リクエストの標準パラメータ：

パラメータ	必須	説明
SERVICE	はい	サービスの名称 (**WMTS**)
REQUEST	はい	リクエストの名称 (**GetFeatureInfo**)
LAYER	はい	レイヤ識別
INFOFORMAT	いいえ	出力形式
I	いいえ	ピクセルの X 座標
J	いいえ	ピクセルの Y 座標
TILEMATRIXSET	はい	ピラミッドの名称
TILEMATRIX	メッシュを作る	
TILEROW	はい	メッシュの行座標
TILECOL	はい	メッシュの列座標

標準のものに加えて、QGIS サーバーでは以下の追加パラメータをサポートしています：

パラメータ	必須	説明
MAP	はい	QGIS プロジェクトファイル

URL の例:

```
http://localhost/?
SERVICE=WMTS
&REQUEST=GetFeatureInfo
&MAP=/home/qgis/projects/world.qgs
&LAYER=mylayer
&INFOFORMAT=image/html
&I=10
&J=5
```

INFOFORMAT

このパラメータでは、結果の出力形式を定義することができます。使用できる値は：

- `text/xml`
- `text/html`
- `text/plain`
- `application/vnd.ogc.gml`

既定値は `text/plain` です。

I

このパラメータは、基礎情報を取得したいピクセルの X 座標を定義します。

J

このパラメータは、基礎情報を取得したいピクセルの Y 座標を定義します。

3.6 OGC API 機能

OGC API Features（OAPIF）は、新世代の OGC プロトコルの最初の実装です。これは OGC API - Features - Part 1：コア ドキュメントで説明されています。

一般的なインストールでは、API には *http://localhost/qgisserver/wfs3* からアクセスできます。

以下は、よく知られた WFS プロトコルと OAPIF の最も重要な相違点を簡単にまとめたものです：

- OAPIF は REST API を基にしています
- OAPIF は OPENAPI 仕様に従わなければなりません

- OAPIF は複数の出力形式をサポートしていますが、どの形式も指定することはありません（QGIS OAPIF で現在利用できるのは GeoJSON と HTML のみです）。また、content negotiation を使用して、クライアントに提供する形式を決定します。
- JSON と HTML は OAPIF の一級市民です
- OAPIF は（/api エンドポイントを通して）自己文書化されています。
- OAPIF は（リンクを通じて）完全に誘導可能性があり、ブラウズ可能です

重要: QGIS の OGC API Features の実装では、プロジェクトファイルを指定するために MAP パラメータを活用することができますが、OPENAPI 仕様では余分なクエリパラメータを使用することはできません。このため、MAP を URL で公開せず、他の方法（例えば、ウェブサーバの書き換えルールで QGIS_PROJECT_FILE を環境に設定する）でプロジェクトファイルを指定することを強く推奨します。

注釈: **API** エンドポイントは、サポートされているすべてのパラメータとサービスの出力形式に関する包括的なドキュメントを提供します。以下の段落では、最も重要なもののみを説明します。

3.6.1 リソース表現

QGIS Server における OGC API Features の実装は、現在、以下のリソース表現（出力）形式をサポートしています：

- HTML
- JSON

実際に提供される形式はコンテント・ネゴシエーションに依存しますが、エンドポイントにフォーマット指定子を付加することで、特定の形式を明示的に要求することができます。

サポートされているフォーマット指定子の拡張子：

- .json
- .html

追加のフォーマット指定子エイリアスは、特定のエンドポイントによって定義されることができます：

- .openapi: **API** エンドポイントがサポートする .json のエイリアス
- .geojson: **Features** 及び **Feature** エンドポイントがサポートする .json のエイリアス

3.6.2 エンドポイント

API は、クライアントが取得できるエンドポイントのリストを提供します。システムは、すべてのレスポンスが、提供されたすべてのリソースをナビゲートするためのリンクのセットを提供するように設計されています。

QGIS の実装によって提供されるエンドポイント・ポイントは以下のものです：

名前	パス	説明
Landing Page	/	サービスに関する一般的な情報と、利用可能なすべてのエンドポイントへのリンクを提供します。
Conformance	/conformance	サービスの規格適合性に関する情報
API	/api	サービスによって提供されるエンドポイントの完全な説明と、返された文書の構造
Collections	/collections	サービスが提供するすべてのコレクション（すなわち「ベクタレイヤ」）のリスト
Collection	/collections/{collectionId}	コレクションに関する情報（名前、メタデータ、範囲など）
Features	/collections/{collectionId}/items	コレクションが提供する地物のリスト
Feature	/collections/{collectionId}/items/{featureId}	単独地物に関する情報

WFS-T (transactional Web Feature Service) と同様に、地物の追加、更新、削除 (CRUD) が可能です。それぞれのリクエストは "/api" に記述されています。

Landing Page

主なエンドポイントは **ランディングページ** です。このページから、利用可能なすべてのサービス・エンドポイントに移動することができます。**ランディング・ページ** は、以下へのリンクを提供しなければなりません

- API 定義（パス /api リンクリレーション service-desc 及び service-doc）、
- 適合性宣言（パス /conformance、リンクリレーション conformance）、及び
- コレクション（パス /collections, リンクリレーション data）。

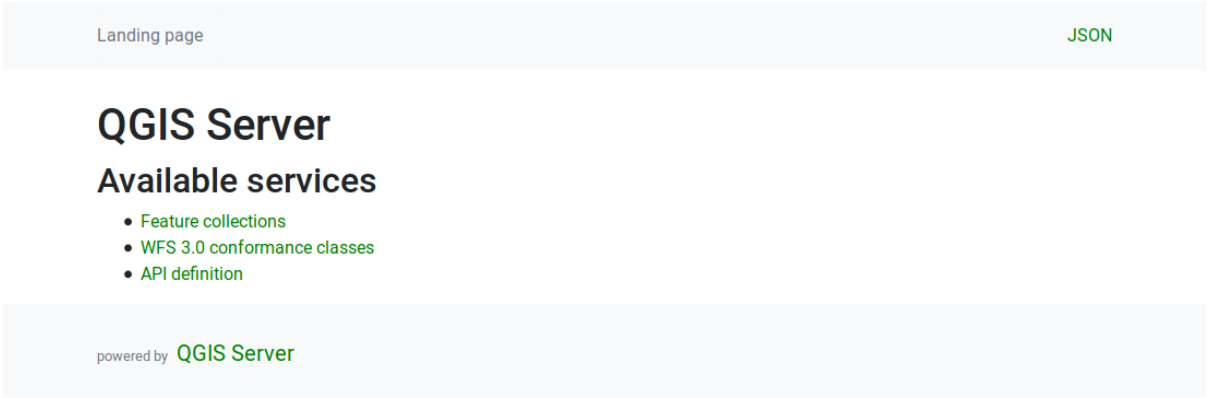


図 3.24: サーバーの OAPIF ランディングページ

API 定義

API 定義 は、サービスが提供する API の OPENAPI 準拠の記述です。HTML 表現においては、すべてのエンドポイントとそのレスポンスフォーマットが正確にリスト化、文書化されたブラウズ可能なページです。このエンドポイントのパスは `/api` です。

API 定義は、サポートされる全てのパラメータと返されるフォーマットを含む、サービスの包括的で権威ある文書を提供します。

注釈: このエンドポイントは WFS の `GetCapabilities` に類似しています

コレクションのリスト

コレクションエンドポイントは、サービスで利用可能な全てのコレクションのリストを提供します。そのサービスはひとつの QGIS プロジェクトに「サービスを提供する」ため、コレクションは現在のプロジェクトのベクタレイヤになります（プロジェクトのプロパティで WFS として公開されている場合）。このエンドポイントのパスは `/collections/` です。

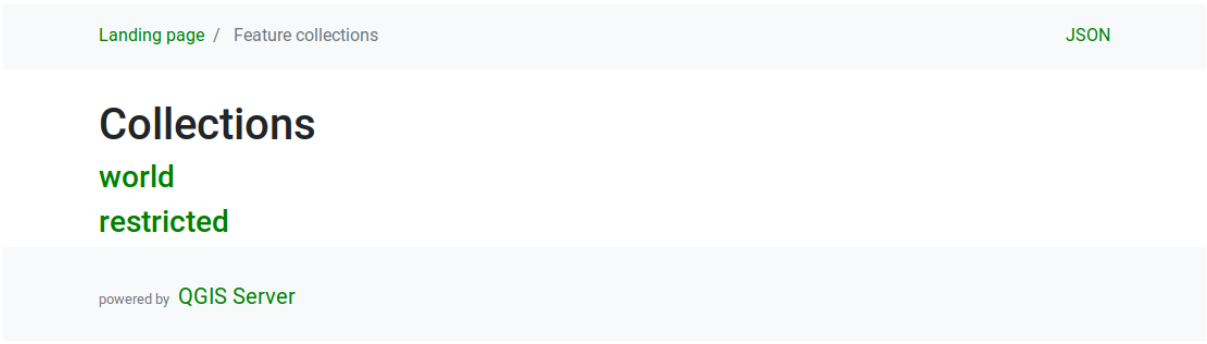


図 3.25: サーバーの OAPIF コレクションリストページ

コレクションの詳細

コレクションエンドポイントは利用可能な各コレクションの詳細情報を提供しませんが、その情報は``/collections/{collectionId}` エンドポイントで利用可能です。典型的な情報には、範囲、説明、CRS、その他のメタデータが含まれます。

また、その HTML 表現は、利用可能な地物を持った閲覧可能な地図も提供されます。

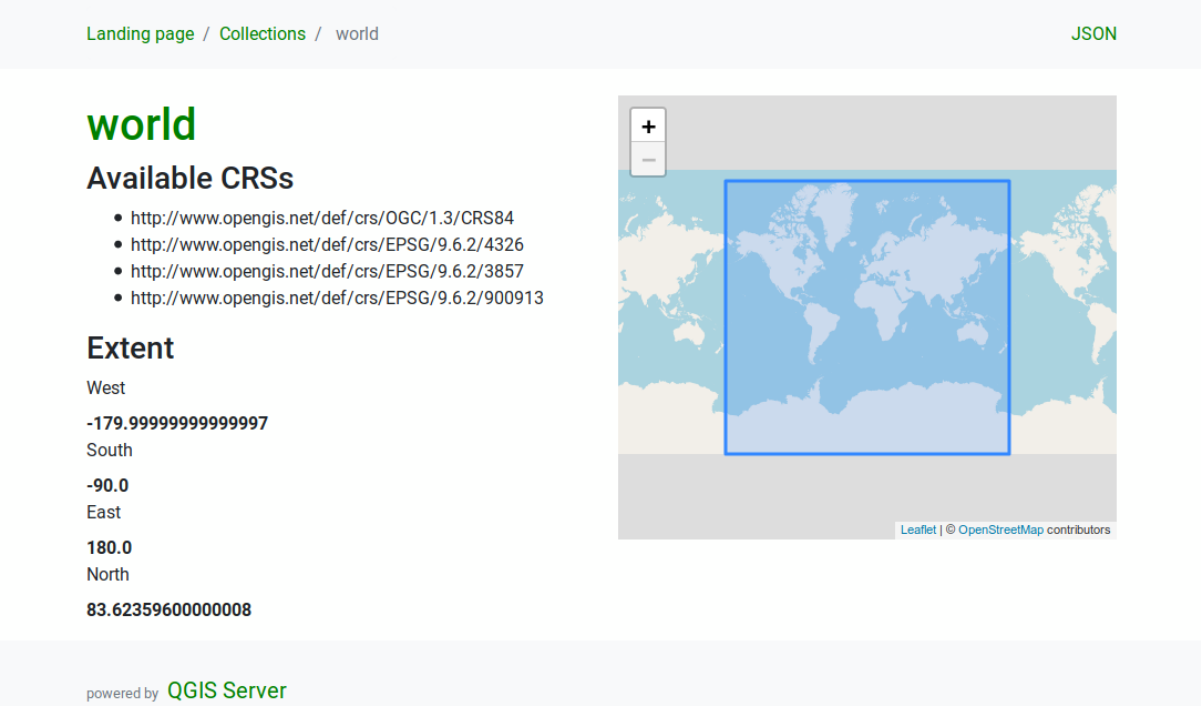


図 3.26: サーバーの OAPIF コレクションの詳細ページ

地物のリスト

このエンドポイントは、コレクション ID がわかっているコレクション内の全ての地物のリストを提供します。このエンドポイントのパスは `/collections/{collectionId}/items` です。

また、その HTML 表現は、利用可能な地物を持った閲覧可能な地図も提供されます。

注釈: このエンドポイントは、WFS 1 と WFS 2 の `GetFeature` に類似しています。

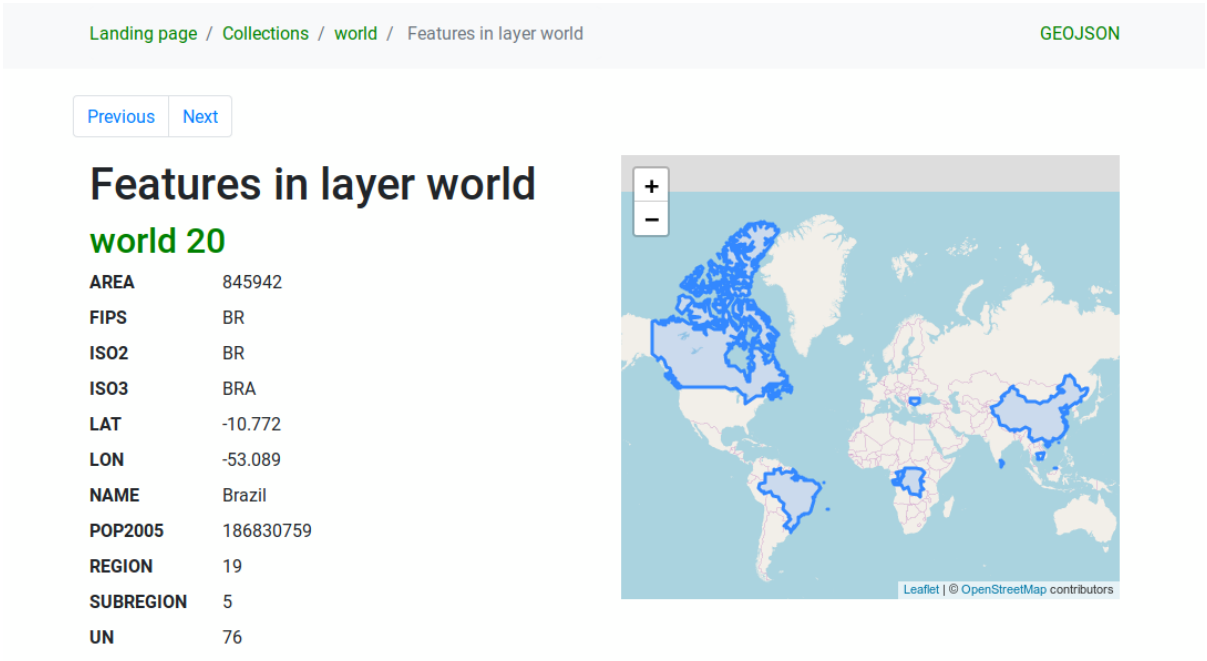


図 3.27: サーバーの OAPIF 地物のリストページ

地物の詳細

このエンドポイントは、地物の属性やジオメトリなど、1 つの地物に関する全ての情報を提供します。このエンドポイントのパスは `/collections/{collectionId}/items/{itemId}` です。

また、その HTML 表現は、地物ジオメトリを含むブラウズ可能な地図も提供されます。

図 3.28: サーバーの OAPIF 地物の詳細ページ

3.6.3 ページ付け

地物の長いリストのページ付けは OGC API で `next` と `prev` のリンクを通して実装されており、QGIS サーバーはクエリ文字列のパラメータとして `limit` と `offset` を追加することでこれらのリンクを構築します。

URL の例:

```
http://localhost/qgisserver/wfs3/collection_one/items.json?offset=10&limit=10
```

注釈: `limit` の最大許容値は `QGIS_SERVER_API_WFS3_MAX_LIMIT` サーバー設定で設定することができます（qgis-server-envar を参照してください）。

3.6.4 地物フィルタリング

コレクションで利用可能な地物は、1 つ以上のフィルタを指定してフィルタ/検索できます。

日付と時間フィルタ

クエリ文字列の引数に `datetime` を指定することで、日付や日付時刻属性を持つコレクションをフィルタすることができます。デフォルトでは、最初の date/datetime フィールドがフィルタに使用されます。この動作は、レイヤプロパティダイアログの *QGIS* **サーバー** *Dimension* セクションで "Date" または "Time" 次元を設定することで設定できます。

日付と時刻のフィルタ構文は *API* **定義** に完全に記述されており、単一の値だけでなく範囲 (開始値と終了値が含まれます) も対応しています。

URL の例:

日付次元が `2019-01-01` に一致する地物のみを返します

```
http://localhost/qgisserver/wfs3/collection_one/items.json?datetime=2019-01-01
```

日付時刻次元が `2019-01-01T01:01:01` に一致する地物のみを返します

```
http://localhost/qgisserver/wfs3/collection_one/items.json?datetime=2019-01-
↪01T01:01:01
```

日付時刻次元が `2019-01-01T01:01:01` - `2019-01-01T12:00:00` の範囲に含まれる地物のみを返します

```
http://localhost/qgisserver/wfs3/collection_one/items.json?datetime=2019-01-
↪01T01:01:01/2019-01-01T12:00:00
```

バウンディングボックスフィルタ

bbox パラメータでバウンディングボックス空間フィルタを指定することができます：

コンマで区切られた要素の順序は:

- 左下隅、WGS 84 経度
- 左下隅、WGS 84 緯度
- 右上隅、WGS 84 経度
- 右上隅、WGS 84 緯度

注釈: OGC 仕様では、3 番目と 6 番目の項目が Z 成分となる 6 項目の bbox 指定子も認められていますが、QGIS サーバーではまだサポートされていません。

URL の例:

```
http://localhost/qgisserver/wfs3/collection_one/items.json?bbox=-180,-90,180,90
```

バウンディングボックスの *CRS* が WGS 84 でない場合、オプションのパラメータ `bbox-crs` を使って別の CRS を指定することができます。CRS 形式識別子は OGC URI 形式でなければなりません：

URL の例:

```
http://localhost/qgisserver/wfs3/collection_one/items.json?bbox=913191,5606014,913234,
↪5606029&bbox-crs=http://www.opengis.net/def/crs/EPSG/9.6.2/3857
```

属性フィルタ

属性フィルタはバウンディングボックスフィルタと組み合わせることができ、一般的な形式は：`<attribute name> = <attribute value>`。複数のフィルタは `AND` 演算子を使って組み合わせることができます。

URL の例:

属性 `name` が "my value" と等しい全ての地物をフィルタします

```
http://localhost/qgisserver/wfs3/collection_one/items.json?attribute_one=my%20value
```

部分一致も * ("star") 演算子を使うことで対応しています：

URL の例:

属性 `name` の末尾が "value" であるすべての地物をフィルタします

```
http://localhost/qgisserver/wfs3/collection_one/items.json?attribute_one=*value
```

3.6.5 地物の並び替え

クエリパラメータ sortby を使って、フィールド値で結果セットを並べ替えることができます。

結果はデフォルトで昇順に並べ替えられます。結果を降順に並べ替えるには、ブール値のフラグ (sortdesc) を設定します：

```
http://localhost/qgisserver/wfs3/collection_one/items.json?sortby=name&sortdesc=1
```

3.6.6 属性の選択

地物のリスト の呼び出しによって返される地物の属性は、オプションの properties クエリ文字列引数にカンマ区切りの属性名のリストを追加することで制限することができます。

URL の例:

name 属性だけを返します

```
http://localhost/qgisserver/wfs3/collection_one/items.json?properties=name
```

3.6.7 HTML ページをカスタマイズする

HTML 表現は HTML テンプレートのセットを使ってレスポンスを生成します。テンプレートは inja と呼ばれるテンプレートエンジンによって解析されます。テンプレートはそれらをオーバーライドすることでカスタマイズすることができます (参照: **テンプレートのオーバーライド**)。テンプレートは JSON 表現と同じデータにアクセスすることができ、さらにいくつかの関数を利用することができます：

カスタムテンプレート関数

- path_append(path): 現在の url にディレクトリパスを追加します
- path_chomp(n): 現在の url パスから指定された "n" 個のディレクトリ構成要素を取り除きます
- json_dump( ): テンプレートに渡された JSON データを表示します
- static(path): 指定された静的パスの完全な URL を返します。例えば "static("/style/black.css")" でルートパスが "http://localhost/qgisserver/wfs3" の場合、"http://localhost/qgisserver/wfs3/static/style/black.css" を返します。
- links_filter(links, key, value): リンクリストからフィルタされたリンクを返します
- content_type_name(content_type): content type から短い名前を返します。例 "text/html" は "HTML" を返します
- nl2br(text): 入力テキストから改行をすべて "<br>" タグに置き換えたものを返します
- starts_with(string, prefix): 文字列が指定された文字列プレフィックスで始まっていれば真を、そうでなければ偽を返します

テンプレートのオーバーライド

テンプレートと静的アセットは QGIS サーバーのデフォルトの API リソースディレクトリのサブディレクトリ (Linux システムの場合 /usr/share/qgis/resources/server/api/) に保存され、ベースディレクトリは環境変数 QGIS_SERVER_API_RESOURCES_DIRECTORY を変更することでカスタマイズできます。

一般的な Linux のインスタレーションは次のディレクトリ構造を持っています:

```
/usr/share/qgis/resources/server/api/
└── ogc
    ├── schema.json
    ├── static
    │   ├── jsonFormatter.min.css
    │   ├── jsonFormatter.min.js
    │   └── style.css
    └── templates
        └── wfs3
            ├── describeCollection.html
            ├── describeCollections.html
            ├── footer.html
            ├── getApiDescription.html
            ├── getFeature.html
            ├── getFeatures.html
            ├── getLandingPage.html
            ├── getRequirementClasses.html
            ├── header.html
            ├── leaflet_map.html
            └── links.html
```

テンプレートを上書きするには、ツリー全体を別の場所にコピーし、QGIS_SERVER_API_RESOURCES_DIRECTORY を新しい場所に指定します。

第4章　カタログ

QGIS Server カタログは、QGIS Server が提供する QGIS プロジェクトのリストを表示する簡単なカタログです。基本的なマッピング機能を備え、QGIS プロジェクトを通じて公開されているデータセットをすばやく閲覧できる、ユーザーフレンドリーな完全に閲覧可能なウェブサイトです。

QGIS Server カタログは変数 QGIS_SERVER_LANDING_PAGE_PROJECTS_DIRECTORIES 及び QGIS_SERVER_LANDING_PAGE_PROJECTS_PG_CONNECTIONS を使います（**環境変数** を参照）

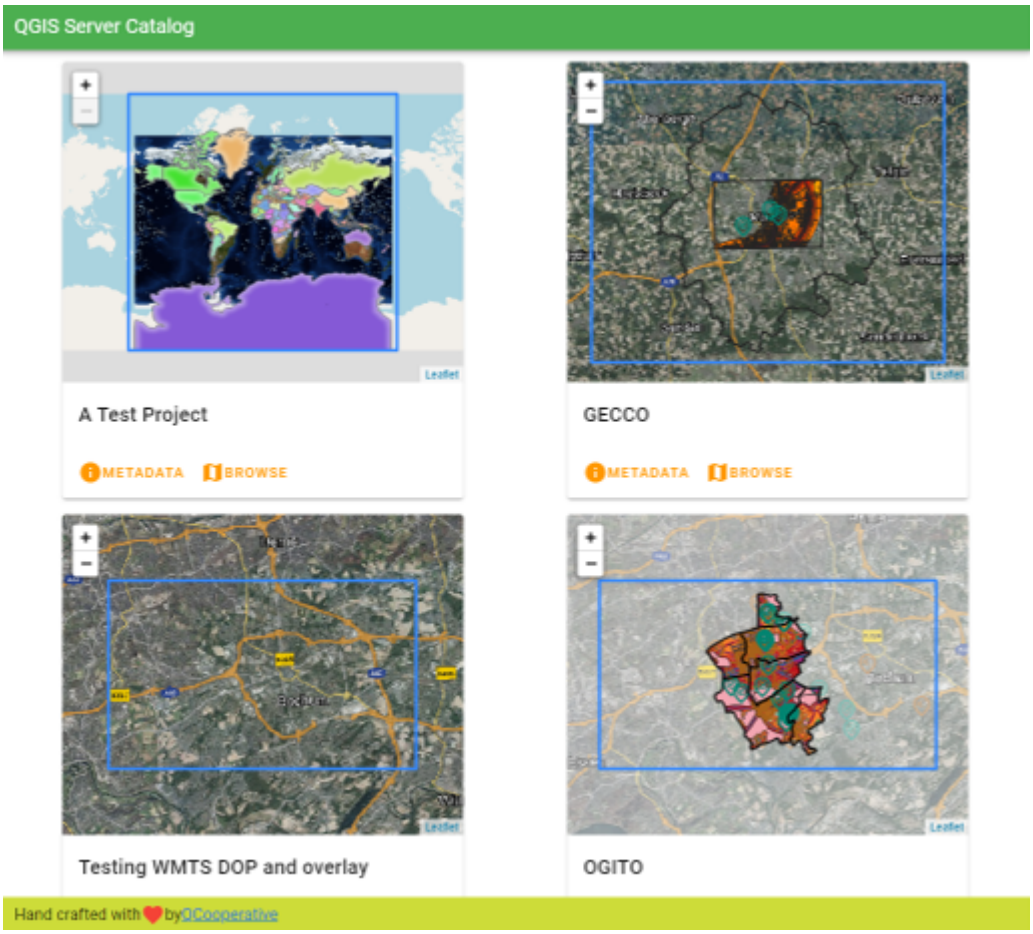


図 4.1: Server カタログプロジェクト一覧ページ

プロジェクトに関連するメタデータや、そのプロジェクトが提供するサービスを参照することができます。これらのサービスへのリンクもあります。

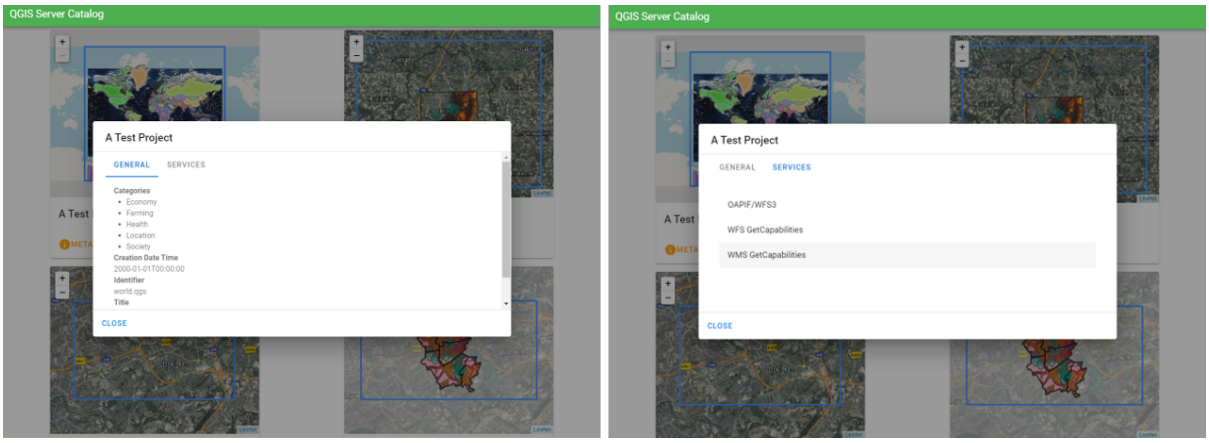


図 4.2: Server カタログ、プロジェクトに関連するメタデータとそれが提供するサービス（へのリンク）。

プロジェクトを閲覧することにより、提供するデータセットの一覧が表示されます。

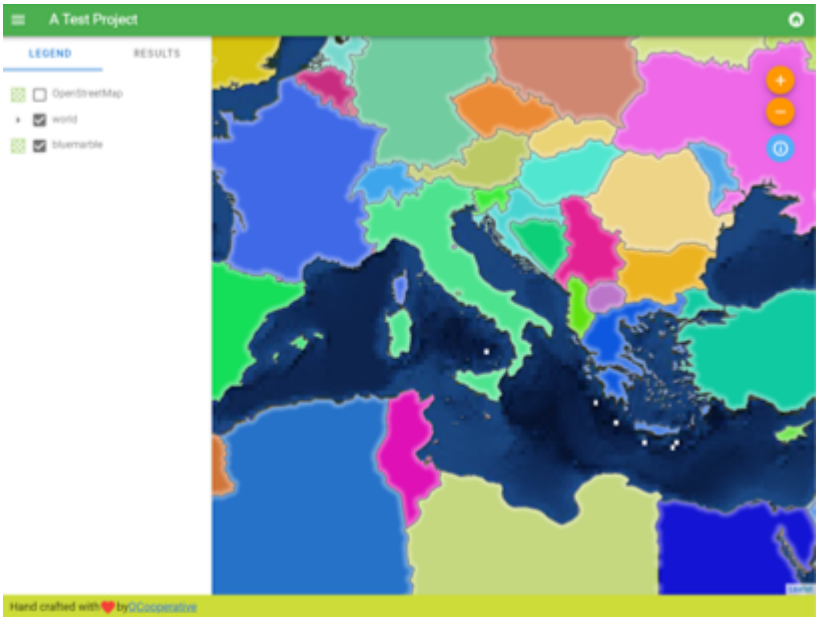


図 4.3: Server カタログにあるプロジェクトが提供するデータセットを閲覧する

レイヤで右クリックすると、それに関連した属性テーブルを表示します

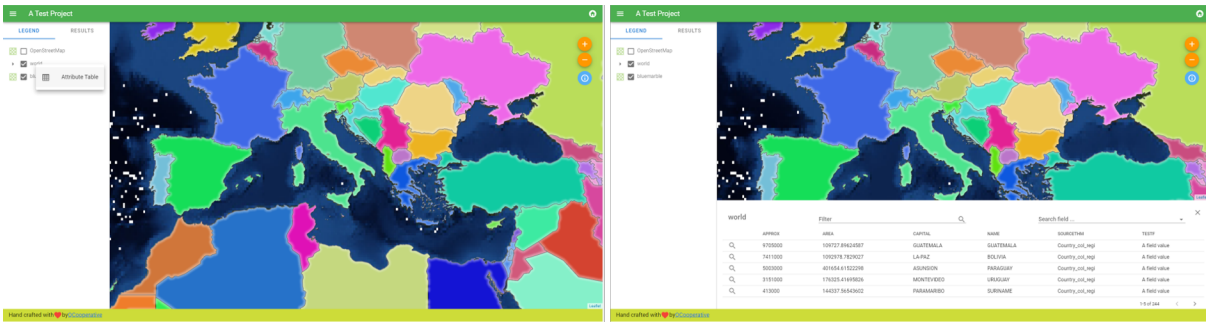


図 4.4: レイヤに関連した属性テーブル

下の画像のように、マップにある要素の情報を参照することができます：

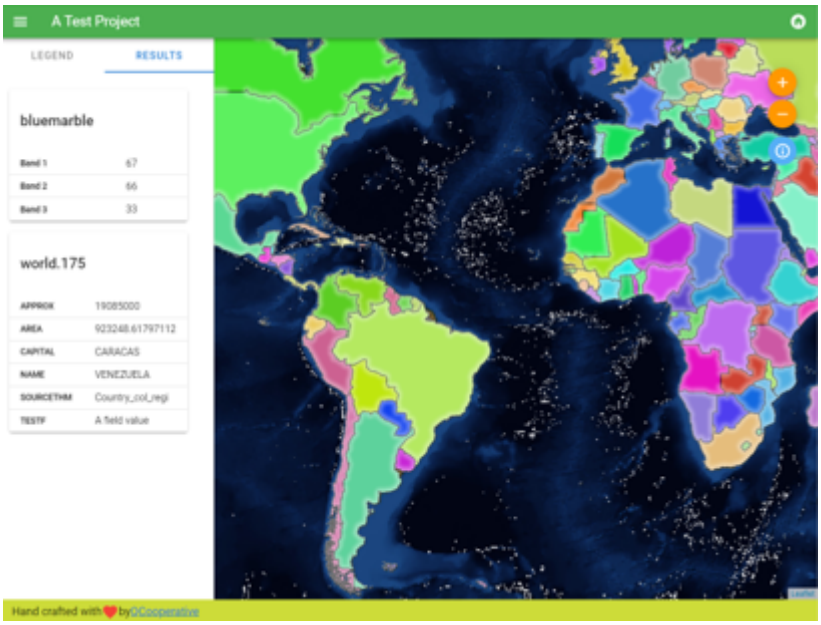


図 4.5: マップ要素の情報を参照する

第5章　プラグイン

5.1 プラグインのリスト

プラグインは QGIS Server にもインストールすることができます。

サーバー向けに設計されたプラグインのいくつかは、公式の QGIS repository に見つかります。

自分の目的に必要なプラグインだけをインストールしてください。QGIS サーバでは、プラグインは QGIS サーバへのフックのようなもので、QGIS サーバの入出力を変更することができます。プラグインがどのように動作するのかを知らないと、予期しない結果をもたらす可能性があります。どのプラグインが役に立つかは、それぞれのドキュメントや QGIS サーバのプラグインを必要とするアプリケーションを参照してください。

5.2 プラグインの場所

デフォルトで、Debian ベースのシステムでは、QGIS Server は `/usr/lib/qgis/plugins` にあるプラグインを探します。デフォルト値は QGIS Server の起動時にログに表示されます。ウェブサーバの設定で環境変数 `QGIS_PLUGINPATH` を定義することで、カスタムパスを設定することができます。

5.3 インストール

5.3.1 ZIP を使って手動で行う

例えば、サーバをテストするために **HelloWorld** プラグインを特定のフォルダを使ってインストールするためには、まずサーバプラグインを格納するディレクトリを作成する必要があります。これは仮想ホストの設定で指定され、環境変数を通してサーバに渡されます：

```
mkdir -p /var/www/qgis-server/plugins
cd /var/www/qgis-server/plugins
wget https://github.com/elpaso/qgis-helloserver/archive/master.zip
unzip master.zip
mv qgis-helloserver-master HelloServer
```

> **警告:** 説明文 によると、HelloServer プラグインは開発およびデモ用に設計されています。このプラグインが不要な場合は、本番環境用に使用しないでください。

5.3.2 コマンドラインツールを使う

QGIS プラグインリポジトリに保存されているプラグインをインストールしたり、定期的にアップグレードする必要がある場合、QGIS-Plugin-Manager を使うことができます。これはコマンドラインからプラグインを管理するためのツールです。

インストールには pip を使用します。仮想環境にインストールするのは良い方法ですが、必須ではありません：

```
pip3 install qgis-plugin-manager
```

ツールを更新するには:

```
pip3 install --upgrade qgis-plugin-manager
```

その後、コマンドラインから qgis-plugin-manager 実行ファイルを使うことができます：

```
cd /var/www/qgis-server/plugins
qgis-plugin-manager list

QGIS server version 3.19.0
List all plugins in /var/www/qgis-server/plugins

-------------------------------------------------------------------------------------------
↪-----------------------------------------------------------------------
|  Folder                    |  Name              |  Version  |  Experimental  | ␣
↪QGIS min  |  QGIS max  |  Author                    |  Action !          |
-------------------------------------------------------------------------------------------
↪-----------------------------------------------------------------------
|wfsOutputExtension          |wfsOutputExtension  |1.6.2      |                |3.0
      |            |3Liz                          |                    |
|qgis_server_render_geojson  |GeoJson Renderer    |v0.4       |                |3.4
      |            |Matthias Kuhn (OPENGIS.ch)    |                    |
|DataPlotly                  |Data Plotly         |3.7.1      |                |3.4
      |3.98        |Matteo Ghetta (Faunalia)      |Upgrade to 3.8.1    |
-------------------------------------------------------------------------------------------
↪-----------------------------------------------------------------------
```

このツールでプラグインをインストールしたりアップグレードしたりする方法については、readme ファイル にある完全なドキュメントを読むことをお勧めします。

5.4 HTTP サーバー構成

5.4.1 Apache

サーバー プラグインが使えるようにするには、FastCGI がどこを見るか知る必要があります。そのため、FastCGI に QGIS_PLUGINPATH 環境変数を示すように Apache 設定ファイルを変更する必要があります：

```
FcgidInitialEnv QGIS_PLUGINPATH "/var/www/qgis-server/plugins"
```

さらに、以前に紹介した HelloWorld プラグインで遊ぶためには、基本的な HTTP 認証が必要です。そこで、最後に Apache の設定ファイルを更新する必要があります：

```
# Needed for QGIS HelloServer plugin HTTP BASIC auth
<IfModule mod_fcgid.c>
    RewriteEngine on
    RewriteCond %{HTTP:Authorization} .
    RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]
</IfModule>
```

それから、Apache を再起動します:

```
systemctl restart apache2
```

5.5 プラグインの使用方法

HelloWorld のプラグインでサーバーをテストします：

```
wget -q -O - "http://localhost/cgi-bin/qgis_mapserv.fcgi?SERVICE=HELLO"
 HelloServer!
```

QGIS サーバーのデフォルトの GetCapabilities をご覧ください：

```
http://localhost/cgi-bin/qgis_mapserv.fcgi?SERVICE=WMS&VERSION=1.3.0&
↪REQUEST=GetCapabilities
```

第6章　高度な構成

6.1 ログを記録する

サーバーに送られたリクエストを記録するには、次の環境変数を設定しなければなりません：

- *QGIS_SERVER_LOG_STDERR*

次の変数によって、ログの記録を更にカスタマイズすることができます：

- *QGIS_SERVER_LOG_LEVEL*
- *QGIS_SERVER_LOG_PROFILE*

6.2 環境変数

環境変数 を設定することによって QGIS Server のいくつかの側面を設定することができます。

HTTP サーバーと QGIS Server の実行方法に応じて、これらの変数を定義する方法がいくつかあります。詳しくは *Apache HTTP* **サーバー** を参照してください。

名前	説明	デフォルト	サービス
QGIS_OPTIONS_PATH	設定のあるディレクトリへのパスを指定します。QGIS アプリケーションの `--optionspath` オプションと同じように動作します。これは `<QGIS_OPTIONS_PATH>/QGIS/QGIS3.ini` にある設定ファイルを探します。	''	All
QGIS_PLUGINPATH	サーバーに Python プラグインを使用している場合に便利です。Python プラグインを検索するフォルダを設定します。	''	All

次のページに続く

表 6.1 – 前のページからの続き

名前	説明	デフォルト	サービス
QGIS_PROJECT_FILE	プロジェクトファイル `.qgs` または `.qgz` は、通常はクエリ文字列のパラメータとして渡されます (*MAP* を使用します) が、環境変数として設定することもできます (例えば、`mod_rewrite` Apache モジュールを使用します)。 Note that you may also indicate a project stored in PostgreSQL, e.g. `postgresql://localhost:5432?sslmode=disable&dbname=mydb&schema` or inside a geopackage file, e.g. `geopackage:/path/to/geopackage/file.gpkg?projectName=myProjectName`.	"	All
QGIS_SERVER_ALLOWED_E	地物フィルタの構成要素として使用できる追加の SQL トークンを表す、カンマで区切られた文字列のリスト。	"	WMS
QGIS_SERVER_API_RESOUR	すべての OGC API（OAPIF/WFS3 など）の静的リソース（HTML テンプレート、CSS、JS など）のベースディレクトリ	パッケージに依存します	OAPIF/WFS3
QGIS_SERVER_APPLICATIO	Name of the application to be used, for instance when connecting to a database to identify the QGIS server instance connected	QGIS3 server	All
QGIS_SERVER_API_WFS3_M	OAPIF/WFS3 地物リクエストにおける `limit` の最大値。	10000	OAPIF/WFS3
QGIS_SERVER_CACHE_DIRE	ファイルシステム上のネットワークキャッシュディレクトリを指定します。	プロファイルディレクトリの `cache`	All
QGIS_SERVER_CACHE_SIZE	ネットワークキャッシュの大きさを MB で設定します。	50 MB	All
QGIS_SERVER_CAPABILITIE	The maximum number of project capabilities to cache.	40	All
QGIS_SERVER_DISABLE_GE	これは、レイアウトの読み込みを無効にすることで、プロジェクトの読み込み時間を短縮するためのプロジェクトレベルのオプションです。 このオプションを有効にすると、QGIS WMS GetPrint 要求が無効になります。レイアウトを読み込まないようにするには、この QGIS プロジェクトフラグを設定します。	false	WMS

次のページに続く

表 6.1 – 前のページからの続き

名前	説明	デフォルト	サービス
QGIS_SERVER_FORCE_REAI	全てのレイヤを読み取り専用モードで開くよう QGIS Server に強制します	false	All
QGIS_SERVER_IGNORE_BAI	「不良」レイヤとは、読み込むことができないレイヤのことです。QGIS Server のデフォルトの動作では、不良レイヤが含まれている場合、そのプロジェクトは利用できないとみなします。 この変数を 1 または `true` に設定することで、デフォルトの動作をオーバーライドすることができます。この場合、「不良」レイヤは無視され、プロジェクトは有効で利用可能であるとみなされます。	false	All
QGIS_SERVER_LANDING_PA	ランディングページのベース URL のパス要素のプレフィックス	""	All
QGIS_SERVER_LANDING_PA	ランディングページサービスが.qgs 及び.qgz プロジェクトを見つけるのに使うディレクトリ群	""	All
QGIS_SERVER_LANDING_PA	ランディングページサービスがプロジェクトを見つけるのに使う PostgreSQL 接続文字列	""	All

次のページに続く

表 6.1 – 前のページからの続き

名前	説明	デフォルト	サービス
QGIS_SERVER_LOG_FILE	パスとファイル名を指定します。サーバーがファイルへの書き込みに適切なパーミッションを持っていることを確認してください。ファイルは自動的に作成されるはずなので、サーバーに要求を送るだけです。ファイルがない場合は、パーミッションを確認してください。 **警告:** QGIS_SERVER_LOG_FILE は QGIS 3.4 以降で非推奨です。代わりに QGIS_SERVER_LOG_STDERR を使用してください。ファイルロギングのサポートは	"	All

表 6.1 – 前のページからの続き

名前	説明	デフォルト	サービス
QGIS_SERVER_LOG_LEVEL	必要なログレベルを指定します。使用可能な値は以下のとおりです： • 0 又は INFO（全ての要求をログに記録します） • 1 又は WARNING • 2 又は CRITICAL（致命的なエラーだけをログに記録するので、製品版に適しています）	0	All
QGIS_SERVER_LOG_PROFIL	詳細なプロファイル情報をログに追加します。QGIS_SERVER_LOG_LEVEL=0 のときだけ有効です	false	All
QGIS_SERVER_LOG_STDERI	stderr へのロギングを有効にします。この変数は QGIS_SERVER_LOG_FILE が設定されている場合には効果がありません。 • 0 又は false（大文字小文字は区別しない） • 1 又は true（大文字小文字を区別する）	false	All
QGIS_SERVER_MAX_THREA	並列レンダリングが有効な場合に使用するスレッド数。値が -1 の場合、プロセッサのコア数を使用します。	-1	All
QGIS_SERVER_OVERRIDE_S	QGIS server で使用する LOCALE を設定します。デフォルト値は空です（上書きしない）。 例: de_CH.utf8	"	All
QGIS_SERVER_PARALLEL_F	WMS GetMap リクエストの並列レンダリングを有効にします。デフォルトでは無効 (false) です。利用可能な値は以下のとおりです： • 0 又は false（大文字小文字は区別しない） • 1 又は true（大文字小文字を区別する）	false	WMS
QGIS_SERVER_PROJECT_CA	Controls the periodic strategy interval for cache invalidation, in milliseconds. Defaults to 0 which selects the legacy File system watcher.		All

次のページに続く

表 6.1 – 前のページからの続き

名前	説明	デフォルト	サービス
QGIS_SERVER_PROJECT_CA	プロジェクト・キャッシュを無効にする方法を定義します。利用可能な戦略は以下のとおりです： • filesystem: ファイルシステムウォッチャー戦略を取ります • periodic: プロジェクトの設定の変更をチェックするために、プロジェクトの最終更新値を使用します。NFS のような非定型のファイルシステムや、プロジェクトファイルが PostgreSQL のようなデータベースシステムに保存されている場合に便利です。 • off：内部キャッシュの無効化を完全に無効にします	filesystem	All
QGIS_SERVER_SERVICE_UR	サービス URL がプロジェクトに存在しない場合に設定するオプションです。サービス URL は、（優先順位の高い順に）以下から定義されます： • サービスごとにプロジェクトに定義された値 • QGIS_SERVER_<service>_SERVICE 環境変数 • QGIS_SERVER_SERVICE_URL 環境変数 • X-Qgis-<service>-Service-Url ヘッダ • X-Qgis-Service-Url ヘッダ • Forwarded ヘッダから組み立てる • X-Forwarded-Host 及び X-Forwarded-Proto ヘッダから組み立てる • Host ヘッダとサーバープロトコルから組み立てる • サーバー名とサーバープロトコルから組み立てる 最後の 4 つのケースでは、得られるサービス URL はクエリ文字列で提供された MAP パラメータと、受信したパスリクエストに基づいています。	"	All

次のページに続く

表 6.1 – 前のページからの続き

名前	説明	デフォルト	サービス
QGIS_SERVER_SHOW_GROU	(GetFeatureInfo レスポンスなどで) 数値にグループセパレータ (3 桁ごとの区切りなど) を使用するかどうかを定義します。デフォルト値は `0` です。 • `0` 又は `false`（大文字小文字は区別しない） • `1` 又は `true`（大文字小文字を区別する）	false	WMS
QGIS_SERVER_TRUST_LAYI	これは、プロジェクトのメタデータで定義されているベクタレイヤ範囲を使用し、PostgreSQL/PostGIS レイヤの主キーの一意性チェックを無効にすることで、プロジェクトの読み込み時間を改善する、プロジェクトレベルのオプションです。 この変数を `1` または `true` に設定することで、レイヤのメタデータを信頼させることができます。ベクタレイヤの範囲はプロジェクトで定義されたものになり、データソースで定義された PostgreSQL/PostGIS レイヤの主キーはチェックなしで一意であるとみなされます。プロジェクトの使用中にレイヤの範囲が固定されない場合は使用しないでください。	false	All
QGIS_SERVER_WCS_SERVIC	これはサービス URL がプロジェクトに存在しない場合に設定するオプションです。詳しくは *QGIS_SERVER_SERVICE_URL* を参照してください。	"	WCS
QGIS_SERVER_WFS_SERVIC	これはサービス URL がプロジェクトに存在しない場合に設定するオプションです。詳しくは *QGIS_SERVER_SERVICE_URL* を参照してください。	"	WFS
QGIS_SERVER_WMS_MAX_I / QGIS_SERVER_WMS_MAX_	WMS リクエストの最大高さ/幅。この値とプロジェクトの値の間で最も保守的な値が使用されます。値が -1 の場合は、最大値が設定されていないことを意味します。	-1	WMS

次のページに続く

表 6.1 – 前のページからの続き

名前	説明	デフォルト	サービス
QGIS_SERVER_WMS_SERVI(	これはサービス URL がプロジェクトに存在しない場合に設定するオプションです。詳しくは *QGIS_SERVER_SERVICE_URL* を参照してください。	"	WMS
QGIS_SERVER_WMTS_SERV	これはサービス URL がプロジェクトに存在しない場合に設定するオプションです。詳しくは *QGIS_SERVER_SERVICE_URL* を参照してください。	"	WMTS

QUERY_STRING	通常ウェブサーバから渡されるクエリ文字列。この変数はコマンドラインから QGIS サーバのバイナリをテストする際に役立ちます。
	例えば、pg_service.conf ファイルで定義された PostgreSQL 接続も必要とするプロジェクトに対して、コマンドラインで GetCapabilities リクエストをテストする場合：
	`PGSERVICEFILE=/etc/pg_service.conf \`
	`QUERY_STRING="MAP=/home/projects/world.qgs&SERVICE=WMS&REQUEST=GetCapabilities" \`
	`/usr/lib/cgi-bin/qgis_mapserv.fcgi`
	結果は、GetCapabilities レスポンスの内容か、何か問題がある場合はエラーメッセージのいずれかになります。
		"	All

6.3 設定の要約

QGIS Server を起動すると、環境変数により設定可能なパラメータの概要が表示されます。さらに、現在使用されている値と出所も表示されます。

spawn-fcgi を使う例：

```
export QGIS_OPTIONS_PATH=/home/user/.local/share/QGIS/QGIS3/profiles/default/
export QGIS_SERVER_LOG_STDERR=1
export QGIS_SERVER_LOG_LEVEL=2
spawn-fcgi -f /usr/lib/cgi-bin/qgis_mapserv.fcgi -s /tmp/qgisserver.sock -U www-data -
```

(次のページに続く)

(前のページからの続き)

```
↪G www-data -n

QGIS Server Settings:

  - QGIS_OPTIONS_PATH / '' (Override the default path for user configuration): '/
↪home/user/.local/share/QGIS/QGIS3/profiles/default/' (read from ENVIRONMENT_
↪VARIABLE)

  - QGIS_SERVER_PARALLEL_RENDERING / '/qgis/parallel_rendering' (Activate/Deactivate␣
↪parallel rendering for WMS getMap request): 'true' (read from INI_FILE)

  - QGIS_SERVER_MAX_THREADS / '/qgis/max_threads' (Number of threads to use when␣
↪parallel rendering is activated): '4' (read from INI_FILE)

  - QGIS_SERVER_LOG_LEVEL / '' (Log level): '2' (read from ENVIRONMENT_VARIABLE)

  - QGIS_SERVER_LOG_STDERR / '' (Activate/Deactivate logging to stderr): '1' (read␣
↪from ENVIRONMENT_VARIABLE)

  - QGIS_PROJECT_FILE / '' (QGIS project file): '' (read from DEFAULT_VALUE)

  - MAX_CACHE_LAYERS / '' (Specify the maximum number of cached layers): '100' (read␣
↪from DEFAULT_VALUE)

  - QGIS_SERVER_CACHE_DIRECTORY / '/cache/directory' (Specify the cache directory):
↪'/root/.local/share/QGIS/QGIS3/profiles/default/cache' (read from DEFAULT_VALUE)

  - QGIS_SERVER_CACHE_SIZE / '/cache/size' (Specify the cache size): '52428800'␣
↪(read from INI_FILE)

Ini file used to initialize settings: /home/user/.local/share/QGIS/QGIS3/profiles/
↪default/QGIS/QGIS3.ini
```

この特定のケースでは、**QGIS_SERVER_MAX_THREADS** と **QGIS_SERVER_PARALLEL_RENDERING** 値は、**QGIS_OPTIONS_PATH** ディレクトリにある ini ファイル(環境変数により定義されます)から読み込まれることが分かります。ini ファイルの対応するエントリは **/qgis/max_threads** と **/qgis/parallel_rendering** で、それらの値は **true** と **4** スレッドです。

6.4 サービスファイルへの接続

PostgreSQL のサービス・ファイルの Apache が認識できるようにするために（PG サービスファイル セクションを参照） *.conf ファイルのようなを見て、確認する必要があり:

```
SetEnv PGSERVICEFILE /home/web/.pg_service.conf

<Directory "/home/web/apps2/bin/">
  AllowOverride None
.....
```

6.5 お使いの Linux サーバーにフォントを追加

他のマシン上のデフォルトでは存在しないかもしれないフォントを指す QGIS プロジェクトを使用するかもしれないことに注意してください。これは、プロジェクトを共有する場合（フォントがターゲットマシン上に存在しない場合)、それは他のマシン上で異なって見える可能性があることを意味します。

これが起こらないようにするために、単にターゲットマシン上で不足しているフォントをインストールする必要があります。デスクトップシステム上でこれを行うのは通常は自明です（フォントをダブルクリック)。

Linux に対しては、デスクトップ環境をインストールしていない（または、コマンドラインを好む）場合、する必要があります：

- Debian のベースのシステム:

```
sudo su
mkdir -p /usr/local/share/fonts/truetype/myfonts && cd /usr/local/share/fonts/
↪truetype/myfonts

# copy the fonts from their location
cp /fonts_location/* .

chown root *
cd .. && fc-cache -f -v
```

- Fedora のベースのシステム:

```
sudo su
mkdir /usr/share/fonts/myfonts && cd /usr/share/fonts/myfonts

# copy the fonts from their location
cp /fonts_location/* .

chown root *
cd .. && fc-cache -f -v
```

第7章　開発用サーバ

QGIS Server の実稼働環境へのインストールとデプロイメントには通常、クライアントからの HTTP 要求を QGIS Server FastCGI バイナリアプリケーションに転送できる Web サーバーコンポーネント（Apache または Nginx など）のセットアップが必要です。

完全なウェブサーバスタックを設定およびインストールすることなく、ローカルマシン上で QGIS Server を迅速にテストしたい場合は、QGIS 開発用スタンドアロンサーバを使うことができます。

これは独立したアプリケーションで、あなたのプロジェクトファイルを提供するための非常にシンプルなウェブサーバを提供します。

> **警告:** スタンドアロン開発用サーバは、実稼働での使用を目的に開発されたものではありません。セキュリティの脆弱性や、一般に公開されたサーバで通常発生するその他のストレス条件についてはチェックされていません。

サーバを起動するには:

```
$ qgis_mapserver
```

開発用サーバがリッスンするデフォルトのポートは 8000 です。出力例：

```
QGIS Development Server listening on http://localhost:8000
CTRL+C to exit
127.0.0.1 [lun gen 20 15:16:41 2020] 5140 103ms "GET /wfs3/?MAP=/tests/testdata/qgis_
↪server/test_project.qgs HTTP/1.1" 200
127.0.0.1 [lun gen 20 15:16:41 2020] 3298 2ms "GET /wfs3/static/jsonFormatter.min.js␣
↪HTTP/1.1" 200
127.0.0.1 [lun gen 20 15:16:41 2020] 1678 3ms "GET /wfs3/static/jsonFormatter.min.css␣
↪HTTP/1.1" 200
127.0.0.1 [lun gen 20 15:16:41 2020] 1310 5ms "GET /wfs3/static/style.css HTTP/1.1"␣
↪200
127.0.0.1 [lun gen 20 15:16:43 2020] 4285 13ms "GET /wfs3/collections?MAP=/tests/
↪testdata/qgis_server/test_project.qgs HTTP/1.1" 200
```

サーバにはコマンドライン引数として渡せるオプションがいくつかあります。これらのオプションは -h でサーバを起動することで見ることができます。

```
Usage: qgis_mapserver [options] [address:port]
QGIS Development Server
```

(次のページに続く)

(前のページからの続き)

```
Options:
-h, --help          Displays this help.
-v, --version       Displays version information.
-l <logLevel>       Sets log level (default: 0)
                      0: INFO
                      1: WARNING
                      2: CRITICAL
-p <projectPath>    Path to a QGIS project file (*.qgs or *.qgz),
                      if specified it will override the query string MAP argument
                      and the QGIS_PROJECT_FILE environment variable

Arguments:
addressAndPort      Listen to address and port (default: "localhost:8000")
                      address and port can also be specified with the environment
                      variables QGIS_SERVER_ADDRESS and QGIS_SERVER_PORT
```

第8章　コンテナによるデプロイメント

コンテナ化されたアプリケーションを使用するには、最も単純なもの（単純な Docker イメージ）から高度なもの（Kubernetes など）まで、さまざまな方法があります。

注釈: この種類のデプロイメントでは、 docker アプリケーション をインストールして実行する必要があります。この チュートリアル を確認してください。

ヒント: Docker は、パッケージ化されたアプリケーション（別名イメージ）を実行します。これは、ビルドするソース（Dockerfile およびリソース）として取得するか、レジストリ（プライベートまたはパブリック）から既にビルドできます。

注釈: QGIS Debian-Ubuntu パッケージのダウンロードには有効な gpg 認証キーが必要です。インストールページ を参照して、以下の Dockerfile を更新してください。

8.1 シンプルな docker イメージ

Docker イメージがパブリックレジストリに存在しないため。ビルドする必要があります。そのためには、ディレクトリ `qgis-server` をそのディレクトリ内に作成します。

- この内容で `Dockerfile` ファイルを作ります：

```
FROM debian:bookworm-slim

ENV LANG=en_EN.UTF-8


RUN apt-get update \
    && apt-get install --no-install-recommends --no-install-suggests --allow-
↪unauthenticated -y \
        gnupg \
        ca-certificates \
        wget \
        locales \
```

(次のページに続く)

(前のページからの続き)

```
    && localedef -i en_US -f UTF-8 en_US.UTF-8 \
    # Add the current key for package downloading
    # Please refer to QGIS install documentation (https://www.qgis.org/fr/site/
↪forusers/alldownloads.html#debian-ubuntu)
    && mkdir -m755 -p /etc/apt/keyrings \
    && wget -O /etc/apt/keyrings/qgis-archive-keyring.gpg https://download.qgis.org/
↪downloads/qgis-archive-keyring.gpg \
    # Add repository for latest version of qgis-server
    # Please refer to QGIS repositories documentation if you want other version␣
↪(https://qgis.org/en/site/forusers/alldownloads.html#repositories)
    && echo "deb [signed-by=/etc/apt/keyrings/qgis-archive-keyring.gpg] https://qgis.
↪org/debian bookworm main" | tee /etc/apt/sources.list.d/qgis.list \
    && apt-get update \
    && apt-get install --no-install-recommends --no-install-suggests --allow-
↪unauthenticated -y \
        qgis-server \
        spawn-fcgi \
        xauth \
        xvfb \
    && apt-get remove --purge -y \
        gnupg \
        wget \
    && rm -rf /var/lib/apt/lists/*

RUN useradd -m qgis

ENV TINI_VERSION v0.19.0
ADD https://github.com/krallin/tini/releases/download/${TINI_VERSION}/tini /tini
RUN chmod +x /tini

ENV QGIS_PREFIX_PATH /usr
ENV QGIS_SERVER_LOG_STDERR 1
ENV QGIS_SERVER_LOG_LEVEL 2

COPY cmd.sh /home/qgis/cmd.sh
RUN chmod -R 777 /home/qgis/cmd.sh
RUN chown qgis:qgis /home/qgis/cmd.sh

USER qgis
WORKDIR /home/qgis

ENTRYPOINT ["/tini", "--"]

CMD ["/home/qgis/cmd.sh"]
```

- この内容で cmd.sh ファイルを作ります：

```
#!/bin/bash

[[ $DEBUG == "1" ]] && env

exec /usr/bin/xvfb-run --auto-servernum --server-num=1 /usr/bin/spawn-fcgi -p 5555 -n␣
↪-d /home/qgis -- /usr/lib/cgi-bin/qgis_mapserv.fcgi
```

- イメージを組み立てます：

```
docker build -f Dockerfile -t qgis-server ./
```

8.1.1 最初の実行

サーバーを実行するには QGIS プロジェクトファイルが必要です。自分のものを使うことも、このサンプル を使うこともできます。

これを行うには、ディレクトリ qgis-server 内にディレクトリ data を作成し、その中にファイルをコピーします。以下の説明に準拠するため、名前を osm.qgs に変更します。

注釈: GetCapabilites が壊れている場合、**プロジェクト　プロパティ** の *QGIS Server* タブの下に広告 URL を追加する必要があるかもしれません。例えば、サーバーが 8080 番ポートで公開されている場合、広告 URL として http://localhost:8080/qgis-server/ を指定します。詳しくは **プロジェクトを設定する** セクション以降を参照してください。

これでサーバーを走らせることができます：

```
docker network create qgis
docker run -d --rm --name qgis-server --net=qgis --hostname=qgis-server \
              -v $(pwd)/data:/data:ro -p 5555:5555 \
              -e "QGIS_PROJECT_FILE=/data/osm.qgs" \
              qgis-server
```

使われたオプション:

- **-d**: バックグラウンドで動かします
- **--rm**: 停止させられたとき、コンテナを削除します
- **--name**: 作成されるコンテナの名前
- **--net**: （前回作成された）サブネットワーク
- **--hostname**: 後で参照するための、コンテナのホスト名
- **-v**: コンテナ内でマウントされるローカルのデータディレクトリ
- **-p**: ホスト／コンテナ・ポートマッピング

- **-e**: コンテナ内で使われる環境変数

確認するには、`docker ps | grep qgis-server` と入力すると、**qgis-server** という 1 行が表示されます:

```
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          ␣
↪PORTS                     NAMES
4de8192da76e   qgis-server    "/tini -- /home/qgis···"   3 seconds ago    Up 2 seconds    ␣
↪0.0.0.0:5555->5555/tcp    qgis-server
```

8.1.2 使用可能なサンプル

サーバは fastcgi 接続しか受け付けないので、このプロトコルを扱う HTTP サーバが必要です。そのためには、簡単な Nginx 設定ファイルを作成し、Nginx イメージを起動する必要があります。

この内容でカレントディレクトリにファイル `nginx.conf` を作成します：

```
server {
  listen 80;
  server_name _;
  location / {
    root  /usr/share/nginx/html;
    index index.html index.htm;
  }
  location /qgis-server {
    proxy_buffers 16 16k;
    proxy_buffer_size 16k;
    gzip off;
    include fastcgi_params;
    fastcgi_pass qgis-server:5555;
  }
}
```

そしてこのコマンドを入力します:

```
docker run -d --rm --name nginx --net=qgis --hostname=nginx \
              -v $(pwd)/nginx.conf:/etc/nginx/conf.d/default.conf:ro -p 8080:80 \
              nginx:1.13
```

能力の有無を確認するには、ブラウザで http://localhost:8080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities と入力してください

8.1.3 クリーンアップ

動作中のイメージをクリーンアップするには次を入力します:

```
docker stop qgis-server nginx
```

8.2 Docker スタック

これまでの方法は、スクリプトを書くことはできますが、簡単にパッケージ化したり、標準化したり、簡単に管理したりすることはできません。

Docker イメージセットを操作するには、オーケストレーターが管理する Docker スタックを使用できます。スタックでは、イメージは同じプライベートネットワークで動作しており、スタック全体を開始/停止したり、スタックを他のワーカーにデプロイしたりできます。たとえば、Swarm、Kubernetes、Mesos など、多くのオーケストレーターがいます。

以下では、テスト用の簡単な設定を紹介します。本番用には適していません。

8.2.1 Swarm/docker-compose

Docker に独自のオーケストレーター Swarm が追加されました（docker-compose ファイルと互換性があります)。 これを有効にする 必要があります（Mac バージョンは Linux でも動作します)。

スタックの説明

Swarm が動作するようになったので、サービスファイルを作成します（Deploy to Swarm を参照）`qgis-stack.yaml`：

```
version: '3.7'

services:
  qgis-server:
    # Should use version with utf-8 locale support:
    image: qgis-server:latest
    volumes:
      - REPLACE_WITH_FULL_PATH/data:/data:ro
    environment:
      - LANG=en_EN.UTF-8
      - QGIS_PROJECT_FILE=/data/osm.qgs
      - QGIS_SERVER_LOG_LEVEL=0  # INFO (log all requests)
      - DEBUG=1                  # display env before spawning QGIS Server

  nginx:
```

(次のページに続く)

(前のページからの続き)

```
    image: nginx:1.13
    ports:
      - 8080:80
    volumes:
      - REPLACE_WITH_FULL_PATH/nginx.conf:/etc/nginx/conf.d/default.conf:ro
    depends_on:
      - qgis-server
```

スタックをデプロイ（または更新）するには、次を入力します：

```
docker stack deploy -c qgis-stack.yaml qgis-stack
```

replicas 列に **1/1** が表示されるまで、スタックのデプロイメント状態を確認してください：

```
docker stack services qgis-stack
```

次のようになります:

```
ID                  NAME                MODE                REPLICAS            IMAGE
              PORTS
gmx7ewlvwsqt        qgis_nginx          replicated          1/1                 ␣
↪nginx:1.13          *:8080->80/tcp
l0v2e7cl43u3        qgis_qgis-server    replicated          1/1                 qgis-
↪server:latest
```

WMS の機能を確認するには、ウェブブラウザに http://localhost:8080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities を入力します

クリーンアップ

クリーンアップには、次を入力します:

```
docker stack rm qgis-stack
```

8.2.2 Kubernetes

インストール

もしあなたが **Docker Desktop** をインストールしているなら、Kubernetes (別名 k8s) を使うのはとても簡単です: enable k8s。

そうでない場合は、minikube tutorial または microk8s for Ubuntu に従ってください。

Kubernetes のインストールは本当に複雑なので、このデモで使用される側面にのみ焦点を当てます。さらに詳しい情報については、公式ドキュメント を参照してください。

microk8s

microk8s では追加の手順が必要です。Kubernetes で作成されたイメージを検出するには、レジストリを有効にして qgis-server イメージにタグを付ける必要があります。

まずレジストリを有効にします:

```
microk8s enable dashboard dns registry
```

そして、タグを付け、新しく作成したレジストリにイメージをプッシュします：

```
docker tag qgis-server 127.0.0.1:32000/qgis-server && docker push 127.0.0.1:32000/
↪qgis-server
```

最後に、レジストリ **127.0.0.1:32000** が **insecure-registries** フィールドにリストされるように、`/etc/docker/daemon.json` を追加または補完します：

```
{
  "insecure-registries": ["127.0.0.1:32000"]
}
```

マニフェストを作る

Kubernetes は、yaml マニフェストにデプロイするオブジェクトを記述します。さまざまな種類がありますが、デプロイメント（ポッドを処理する、つまり Docker イメージ）とサービスのみを使用して、デプロイメントを内部または外部の目的に公開します。

デプロイメントマニフェスト

ファイル `deployments.yaml` を次の内容で作成します:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: qgis-server
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      myLabel: qgis-server
  template:
    metadata:
      labels:
        myLabel: qgis-server
```

(次のページに続く)

(前のページからの続き)

```
    spec:
      containers:
        - name: qgis-server
          image: localhost:32000/qgis-server:latest
          imagePullPolicy: Always
          env:
            - name: LANG
              value: en_EN.UTF-8
            - name: QGIS_PROJECT_FILE
              value: /data/osm.qgs
            - name: QGIS_SERVER_LOG_LEVEL
              value: "0"
            - name: DEBUG
              value: "1"
          ports:
            - containerPort: 5555
          volumeMounts:
            - name: qgis-data
              mountPath: /data/
      volumes:
        - name: qgis-data
          hostPath:
            path: REPLACE_WITH_FULL_PATH/data

---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: qgis-nginx
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      myLabel: qgis-nginx
  template:
    metadata:
      labels:
        myLabel: qgis-nginx
    spec:
      containers:
        - name: qgis-nginx
          image: nginx:1.13
          ports:
            - containerPort: 80
```

(次のページに続く)

(前のページからの続き)

```
          volumeMounts:
            - name: nginx-conf
              mountPath: /etc/nginx/conf.d/
      volumes:
        - name: nginx-conf
          configMap:
            name: nginx-configuration

---
kind: ConfigMap
apiVersion: v1
metadata:
  name: nginx-configuration
data:
  nginx.conf: |
    server {
      listen 80;
      server_name _;
      location / {
        root  /usr/share/nginx/html;
        index index.html index.htm;
      }
      location /qgis-server {
        proxy_buffers 16 16k;
        proxy_buffer_size 16k;
        gzip off;
        include fastcgi_params;
        fastcgi_pass qgis-server:5555;
        }
      }
```

サービスマニフェスト

ファイル services.yaml を次の内容で作成します:

```
apiVersion: v1
kind: Service
metadata:
  name: qgis-server
  namespace: default
spec:
  type: ClusterIP
  selector:
    myLabel: qgis-server
```

(次のページに続く)

(前のページからの続き)

```
  ports:
    - port: 5555
      targetPort: 5555
---
apiVersion: v1
kind: Service
metadata:
  name: qgis-nginx
  namespace: default
spec:
  type: NodePort
  selector:
    myLabel: qgis-nginx
  ports:
    - port: 80
      targetPort: 80
      nodePort: 30080
```

マニフェストをデプロイする

Kubernetes にイメージとサービスをデプロイするには、ダッシュボード（右上の + をクリック）またはコマンドラインを使用します。

注釈: microk8s でコマンドラインを使う場合、各コマンドの前に *microk8s* を付ける必要があります。

マニフェストをデプロイまたは更新する:

```
kubectl apply -f ./
```

現在デプロイされているものを確認する:

```
kubectl get pods,services,deployment
```

このようになる筈です:

```
NAME                              READY   STATUS    RESTARTS   AGE
pod/qgis-nginx-54845ff6f6-8skp9   1/1     Running   0          27m
pod/qgis-server-75df8ddd89-c7t7s  1/1     Running   0          27m

NAME                        TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)        ␣
↪AGE
service/Kubernetes          ClusterIP   10.152.183.1     <none>        443/TCP        ␣
↪5h51m
```

(次のページに続く)

(前のページからの続き)

```
service/qgis-exec-server   ClusterIP   10.152.183.218   <none>        5555/TCP       ␣
↪35m
service/qgis-nginx         NodePort    10.152.183.234   <none>        80:30080/TCP   ␣
↪27m
service/qgis-server        ClusterIP   10.152.183.132   <none>        5555/TCP       ␣
↪27m

NAME                           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/qgis-nginx     1/1     1            1           27m
deployment.apps/qgis-server    1/1     1            1           27m
```

nginx/qgis ログを読むには次を入力します:

```
kubectl logs -f POD_NAME
```

WMS の機能を確認するには、ウェブブラウザに次のように入力します http://localhost:30080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities

クリーンアップ

クリーンアップには、次を入力します:

```
kubectl delete service/qgis-server service/qgis-nginx deployment/qgis-nginx␣
↪deployment/qgis-server configmap/nginx-configuration
```

8.3 クラウドデプロイメント

コンテナ化されたアプリケーションのデプロイを処理するために、独自のサーバークラスタを管理するのは複雑な仕事です。ハードウェア、帯域幅、セキュリティなど、さまざまな問題をさまざまなレベルで処理しなければなりません。

クラウド・デプロイメント・ソリューションは、インフラの管理に注力したくない場合に有効な選択肢となります。

クラウドデプロイメントでは独自のメカニズムを使用することができますが、先に説明した段階（*docker images* と *stack management*）と互換性があります。

8.3.1 AWS ユースケース

Amazon AWS では、ECS (Elastic Container Service) の機能により、docker-compose や Kubernetes 互換のラッパーを使ってスタックを管理することができます。カスタムイメージにアクセスするためには image registry を作成する必要があります。

docker-compose 同様の機能を使うには、**ecs-cli** クライアントをインストールし、 適切なパーミッション/ロール を持っている必要があります。その後、ecs-cli compose コマンドの助けを借りて、 *stack description* を再利用することができます。

Kubernetes を使うには、AWS の Web コンソールかコマンドラインツール eksctl を使い、適切なパーミッション/ロールを設定します。その後、うまく設定された kubectl 環境で、 *Kubernetes manifests* を再利用することができます。

第9章　よくある質問

- ***QGIS Desktop* と *QGIS Server* の違いは何ですか？**

  QGIS Desktop はグラフィカル・ユーザー・インタフェースを持ち、マップの作成と修正を行うことができます。QGIS Server は、WMS、WFS などの OGC Web サービスを介してエンドユーザーアプリケーションに QGIS プロジェクトファイルを提供するサーバーアプリケーションです。<https://www.ogc.org/docs/is>`_ 。

- ***OGC* とは何ですか*?***

  OGC（Open Geospatial Consortium）<https://www.ogc.org/>`_ は、世界の地理空間コミュニティのために質の高いオープンスタンダードを作ることを約束する国際的な非営利組織です。

- **他のウェブマッピングサーバーを教えてください**

  ArcGIS server, Geoserver, Mapserver, Mapnik など

- ***QGIS* サーバと他のウェブマッピングサーバを比較すると？** *(2021/01/01)*

QGIS の主な機能	QGIS Server	GeoServer	ArcGIS Server
いつから	2006	2001	1999
許諾	GPL	GPL	商用
商用サポート	複数の企業	複数の企業	ESRI とそのベンダーネットワーク
技術	C++/python	Java	C++
タイルのキャッシュ	ある	あ　る（GeoWebCache 経由）	ある
3D	ない	ない	ある
クエリ	FES (2.0) 及び OGC (1.0) フィルタ	CQL 及び OGC フィルタ	OGC フィルタ
リポートの生成	ある	ある	ある
サーバ管理	サードパーティ経由 (LizMap, QWC2, etc.)	web + API REST	web + API REST
GIS プロジェクトレイヤ／シンボロジ	専用 GUI により完結	web インタフェースにより簡素	専用 GUI により完結

- ***QGIS* サーバに実装されている *OGC* 仕様のバージョンは、他の *Web* マッピングサーバと比較してどうなっていますか？** *(2021/01/01)*

OGC 標準	QGIS Server	GeoServer	ArcGIS Server
WMS (Web Map Service)	1.3.0 - 1.1.1	1.3.0 - 1.1.1	1.3.0 - 1.1.1
WFS (Web Feature Service)	1.1.0 - 1.0.0	2.0.0 - 1.0.0	2.0.0 - 1.0.0
OAPIF (別名 WFS3)	1.0.0	なし	なし
WMTS (Web Map Tile Service)	1.0.0	1.0.0	1.0.0
WCS (Web Coverage Service)	1.0.0	2.0.1 - 1.0.0	2.0.1 - 1.0.0
WPS (Web Processing Service)	なし	1.0.0	1.0.0
CSW (Catalogue Service for the Web)	なし	2.0.2	なし
SLD (Styled Layer Descriptor)	ある	ある	ある

- **タイルキャッシュとは何ですか***?*

  マップは多くの場合静的です。ほとんどのマッピングクライアントは、照会されるたびに WMS (Web Map Service) データをレンダリングするため、不必要な処理や待ち時間の増加につながります。

  タイルキャッシュは、要求された地図画像やタイルを保存（キャッシュ）することで、事実上、クライアント（OpenLayers や Google Maps など）とサーバー（WMS 準拠のサーバー）の間のプロキシとして機能し、エクスペリエンスを最適化します。新しい地図やタイルが要求されると、QGIS サーバーはその呼び出しを捕え、保存されていればレンダリング済みのタイルを返し、そうでなければ必要に応じて QGIS エンジンを呼び出して新しいタイルをレンダリングします。このように、一度タイルが保存されると、地図のレンダリング速度は何倍にもなり、ユーザーエクスペリエンスが大幅に向上します。

- *PostgreSQL* **とは何ですか***?*

  PostgreSQL は、QGIS のための強力なオープンソースのオブジェクトリレーショナルデータベースです。

- *PostGIS* **とは何ですか***?*

  PostGIS は、PostgreSQL オブジェクトリレーショナルデータベース用の空間データベース拡張です。地理オブジェクトのサポートを追加し、SQL で位置クエリを実行できるようにします。