



QGIS Server 3.34 User Guide

QGIS Project

02 mai. 2024

1	Introdução	3
2	Iniciando	5
2.1	Installation on Debian-based systems	5
2.1.1	Servidor HTTP Apache	6
2.1.2	Servidor HTTP NGINX	8
2.1.3	Xvfb	13
2.2	Instalação no Windows	14
2.3	Serve a project	16
2.4	Configurar seu projeto	17
2.4.1	Recursos WMS	19
2.4.2	WMTS capabilities	21
2.4.3	WFS/OAPIF capabilities	22
2.4.4	WCS capabilities	23
2.4.5	Sintonia fina do seu OWS	24
2.5	Integração com terceiros	24
2.5.1	Integração com o QGIS Desktop	24
2.5.2	Integração com MapProxy	24
2.5.3	Integração com QWC2	24
3	Serviços	25
3.1	Fundamentos	25
3.1.1	SERVICE	26
3.1.2	REQUEST	26
3.1.3	MAP	26
3.1.4	FILE_NAME	26
3.1.5	Nome curto	27
3.2	Web Map Service (WMS) Serviço de mapa de web	27
3.2.1	GetCapabilities	28
3.2.2	GetMap	28
3.2.3	GetFeatureInfo	37
3.2.4	GetLegendGraphic	40
3.2.5	GetStyle(s)	55
3.2.6	DescribeLayer	55
3.2.7	GetPrint	56
3.2.8	GetProjectSettings	59
3.2.9	GetSchemaExtension	60
3.2.10	Camadas WMS externas	60
3.2.11	Linha vermelha	61
3.3	Web Feature Service (WFS)	63
3.3.1	GetCapabilities	63

3.3.2	GetFeature	64
3.3.3	DescribeFeatureType	70
3.3.4	Transação	71
3.4	Web Coverage Service (WCS)	74
3.4.1	GetCapabilities	74
3.4.2	DescribeCoverage	75
3.4.3	GetCoverage	77
3.5	Web Map Tile Service (WMTS)	78
3.5.1	GetCapabilities	78
3.5.2	GetTile	79
3.5.3	GetFeatureInfo	80
3.6	Feições OGC API	81
3.6.1	Representação de recursos	82
3.6.2	Endpoints	82
3.6.3	Paginação	86
3.6.4	Filtragem de recursos	86
3.6.5	Feature sorting	87
3.6.6	Seleção de atributo	87
3.6.7	Personalizar as páginas HTML	88
4	Catálogo	89
5	Plugins	93
5.1	Lista de plugins	93
5.2	Localização dos plugins	93
5.3	Instalação	93
5.3.1	Manualmente com um ZIP	93
5.3.2	Com uma ferramenta de linha de comando	94
5.4	Configuração do Servidor HTTP	94
5.4.1	Apache	94
5.5	Como usar um plug-in	95
6	Configuração avançada	97
6.1	Registrando	97
6.2	Variáveis ambientais	97
6.3	Settings summary	102
6.4	Connection to service file	103
6.5	Adicione fontes ao seu servidor linux	104
7	Servidor de desenvolvimento	105
8	Containerized deployment	107
8.1	Imagens simples do estivador	107
8.1.1	Primeira execução	109
8.1.2	Amostra utilizável	109
8.1.3	Limpeza	110
8.2	Pilhas de docas	110
8.2.1	Enxame/doca-composição	110
8.2.2	Kubernetes	111
8.3	Implantação de nuvens	115
8.3.1	AWS usecase	115
9	Perguntas mais frequentes	117

QGIS Server is an open source WMS, WFS, OGC API for Features 1.0 (WFS3) and WCS implementation that, in addition, implements advanced cartographic features for thematic mapping. QGIS Server is a FastCGI/CGI (Common Gateway Interface) application written in C++ that works together with a web server (e.g., Apache, Nginx). It has Python plugin support allowing for fast and efficient development and deployment of new features.

QGIS Server uses QGIS as back end for the GIS logic and for map rendering. Furthermore, the Qt library is used for graphics and for platform-independent C++ programming. In contrast to other WMS software, the QGIS Server uses cartographic rules as a configuration language, both for the server configuration and for the user-defined cartographic rules.

Como o desktop QGIS e o Servidor QGIS usam as mesmas bibliotecas de visualização, os mapas que são publicados na web têm a mesma aparência do desktop GIS.

In the following sections, we will provide a sample configuration to set up a QGIS Server on Linux (Debian, Ubuntu and derivatives) and on Windows. For more information about server plugin development, please read `server_plugins`.

É concedida a permissão para copiar, distribuir e / ou modificar este documento sob os termos da Licença de Documentação Livre GNU, Versão 1.3 ou qualquer versão posterior publicada pela Fundação de Software Livre; sem Secções Invariantes, sem textos de Capa e sem contracapa.

A cópia da licença é incluída na secção `gnu_fdl`.

2.1 Installation on Debian-based systems

We will give a short and simple installation how-to for a minimal working configuration on Debian based systems (including Ubuntu and derivatives). However, many other distributions and OSs provide packages for QGIS Server.

Nota: No Ubuntu, você pode usar seu usuário comum, acrescentando `sudo` aos comandos que requerem permissões de administrador. No Debian você pode trabalhar como administrador (`root`), sem usar `sudo`.

Requirements and steps to add official QGIS repositories to install QGIS Server on a Debian based system are provided in [QGIS installers page](#). You may want to install at least the latest Long Term Release.

Uma vez configurado o repositório da versão desejada e instalado o QGIS Server, você pode testar a instalação com:

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

Se você obtiver a seguinte saída, o servidor está corretamente instalado.

Nota: Dependendo da versão do QGIS, você poderá ver uma saída ligeiramente diferente informada quando executar `qgis_mapserv.fcgi``.

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

Nota: Como visto abaixo, o QGIS informa um código de Status 400, que identifica corretamente que a solicitação falhou porque não há sessão http ativa. Isto não é um bug e indica que o servidor está funcionando corretamente.

```
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"}]
```

Let's add a sample project. You can use your own, or one from [Training demo data](#):

```
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 .
```

Certamente, você pode usar seu software GIS favorito para abrir este arquivo e dar uma olhada na configuração e nas camadas disponíveis.

Para implantar corretamente o servidor QGIS, você precisa de um servidor HTTP. As escolhas recomendadas são [Apache](#) ou [NGINX](#).

2.1.1 Servidor HTTP Apache

Nota: In the following, please replace `qgis.demo` with the name or IP address of your server.

1. Instalar Apache e `mod_fcgid`:

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

2. Você pode executar o QGIS Server em seu site padrão, mas vamos configurar um `virtual host` especificamente para isto, como segue.

1. No diretório `:file:/etc/apache2/sites-available/`, crie um arquivo chamado `:arquivo:qgis.demo.conf`, com este conteúdo:

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
```

(continua na próxima página)

(continuação da página anterior)

```

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
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>

```

Nota: Algumas das opções de configuração acima são explicadas nas seções Servidor *variáveis de ambiente* e *pg_service file*

2. Vamos agora criar os diretórios que irão armazenar os logs e o banco de dados de autenticação do Servidor QGIS:

```

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

```

(continua na próxima página)

(continuação da página anterior)

```
chown www-data:www-data /home/qgis/qgisserverdb
```

Nota: `www-data` is the Apache user on Debian based systems and we need Apache to have access to those locations or files. The `chown www-data...` commands change the owner of the respective directories and files to `www-data`.

3. Agora podemos habilitar o host virtual e o `fcgid` mod, se isso ainda não tiver sido feito:

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

4. Now restart Apache for the new configuration to be taken into account:

```
systemctl restart apache2
```

5. Agora que o Apache sabe que deve responder aos pedidos para <http://qgis.demo> também precisamos configurar o sistema do cliente para que ele saiba quem é `qgis.demo`. Fazemos isso adicionando `127.0.0.1 qgis.demo` no arquivo `hosts`.

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

Importante: Remember that both the `qgis.demo.conf` and `/etc/hosts` files should be configured for your setup to work. You can also test the access to your QGIS Server from other clients on the network (e.g. Windows or macos machines) by going to their `/etc/hosts` file and point the `qgis.demo` name to whatever IP the server machine has on the network (not `127.0.0.1` as it is the local IP, only accessible from the local machine). On `*nix` machines the `hosts` file is located in `/etc`, while on Windows it's under the `C:\Windows\System32\drivers\etc` directory. Under Windows you need to start your text editor with administrator privileges before opening the `hosts` file.

QGIS Server is now available at <http://qgis.demo>. To check, type in a browser, as in the simple case:

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

2.1.2 Servidor HTTP NGINX

Nota: In the following, please replace `qgis.demo` with the name or IP address of your server.

You can also use QGIS Server with **NGINX**. Unlike Apache, NGINX does not automatically spawn FastCGI processes. The FastCGI processes are to be started by something else.

Instalar NGINX:

```
apt install nginx
```

- As a first option, you can use **spawn-fcgi** or **fcgiwrap** to start and manage the QGIS Server processes. Official Debian packages exist for both. When you have no X server running and you need, for example, printing, you can use `xvfb`.
- Another option is to rely on **Systemd**, the init system for GNU/Linux that most Linux distributions use today. One of the advantages of this method is that it requires no other components or processes. It's meant to be simple, yet robust and efficient for production deployments.

Configuração NGINX

The **include fastcgi_params**; used in the previous configuration is important, as it adds the parameters from `/etc/nginx/fastcgi_params`:

```
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;
```

Moreover, you can use some *Variáveis ambientais* to configure QGIS Server. In the NGINX configuration file, `/etc/nginx/nginx.conf`, you have to use `fastcgi_param` instruction to define these variables as shown below:

```
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 wrappers

Aviso: `fcgiwrap` is easier to set up than `spawn-fcgi`, because it's already wrapped in a Systemd service. But it also leads to a solution that is much slower than using `spawn-fcgi`. With `fcgiwrap`, a new QGIS Server process is created on each request, meaning that the QGIS Server initialization process, which includes reading and parsing the QGIS project file, is done on each request. With `spawn-fcgi`, the QGIS Server process remains alive between requests, resulting in much better performance. For that reason, `spawn-fcgi` is recommended for production use.

spawn-fcgi

Se você quiser utilizar `spawn-fcgi`:

1. O primeiro passo é instalar o pacote:

```
apt install spawn-fcgi
```

2. Em seguida, introduzir o seguinte bloco na configuração do seu servidor NGINX:

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

3. E reiniciar o NGINX para levar em conta a nova configuração:

```
systemctl restart nginx
```

4. Finally, considering that there is no default service file for spawn-fcgi, you have to manually start QGIS Server in your terminal:

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

O Servidor QGIS está agora disponível em <http://qgis.demo/qgisserver>.

Nota: When using spawn-fcgi, you may directly define environment variables before running the server. For example: `export QGIS_SERVER_LOG_STDERR=1`

É claro, você pode adicionar um roteiro de inicialização para iniciar o QGIS Server no momento da inicialização ou sempre que quiser. Por exemplo, com **systemd**:

1. Edite o arquivo: `file:/etc/systemd/system/qgis-server.service` com este conteúdo:

```
[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. Em seguida, habilitar e iniciar o serviço:

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

Aviso: With the above commands spawn-fcgi spawns only one QGIS Server process.

fcgiwrap

Utilizar **fcgiwrap** é muito mais fácil de configurar do que **spawn-fcgi** mas é muito mais lento.

1. Primeiro você precisa instalar o pacote correspondente:

```
apt install fcgiwrap
```

2. Em seguida, introduzir o seguinte bloco na configuração do seu servidor NGINX:

```
1 location /qgisserver {
2     gzip                off;
3     include              fastcgi_params;
4     fastcgi_pass          unix:/var/run/fcgiwrap.socket;
5     fastcgi_param         SCRIPT_FILENAME /usr/lib/cgi-bin/qgis_mapserv.fcgi;
6 }
```

3. Finally, restart NGINX and **fcgiwrap** to take into account the new configuration:

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

O Servidor QGIS está agora disponível em <http://qgis.demo/qgisserver>.

Systemd

QGIS Server needs a running X Server to be fully usable, in particular for printing. In the case you already have a running X Server, you can use systemd services.

Este método, para implantar o Servidor QGIS, conta com duas unidades Systemd para configurar: uma **Unidade de bolso** e uma **Unidade de serviço**.

1. The **QGIS Server Socket unit** defines and creates a file system socket, used by NGINX to start and communicate with QGIS Server. The Socket unit has to be configured with `Accept=false`, meaning that the calls to the `accept()` system call are delegated to the process created by the Service unit. It is located in `/etc/systemd/system/qgis-server@.socket`, which is actually a template:

```
[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. Agora ative e inicie as tomadas:

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

3. The **QGIS Server Service unit** defines and starts the QGIS Server process. The important part is that the Service process' standard input is connected to the socket defined by the Socket unit. This has to be configured using `StandardInput=socket` in the Service unit configuration located in `/etc/systemd/system/qgis-server@.service`:

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

[Service]
User=www-data
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
```

Nota: The QGIS Server *environment variables* are defined in a separate file, `/etc/qgis-server/env`. It could look like this:

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

4. Now start socket service:

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

5. Finalmente, para o servidor HTTP NGINX, vamos apresentar a configuração desta instalação:

```
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. Now restart NGINX for the new configuration to be taken into account:

```
systemctl restart nginx
```

Thanks to Oslandia for sharing their tutorial.

2.1.3 Xvfb

QGIS Server needs a running X Server to be fully usable, in particular for printing. On servers it is usually recommended not to install it, so you may use `xvfb` to have a virtual X environment.

If you're running the Server in graphic/X11 environment then there is no need to install `xvfb`. More info at <https://www.itopen.it/qgis-server-setup-notes/>.

1. Para instalar o pacote:

```
apt install xvfb
```

2. Create the service file, `/etc/systemd/system/xvfb.service`, with this content:

```
[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. Ative, inicie e verifique o status do `xvfb.service`:

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

4. Then, according to your HTTP server, you should configure the **DISPLAY** parameter or directly use **xvfb-run**.

- Usando Apache:

1. Adicione à sua configuração *Fcgid* (ver *Servidor HTTP Apache*):

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

2. Reinicie o Apache para que a nova configuração seja considerada:

```
systemctl restart apache2
```

- Usando NGINX

- With spawn-fcgi using `xvfb-run`:

```
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
```

- With the **DISPLAY** environment variable in the HTTP server configuration.

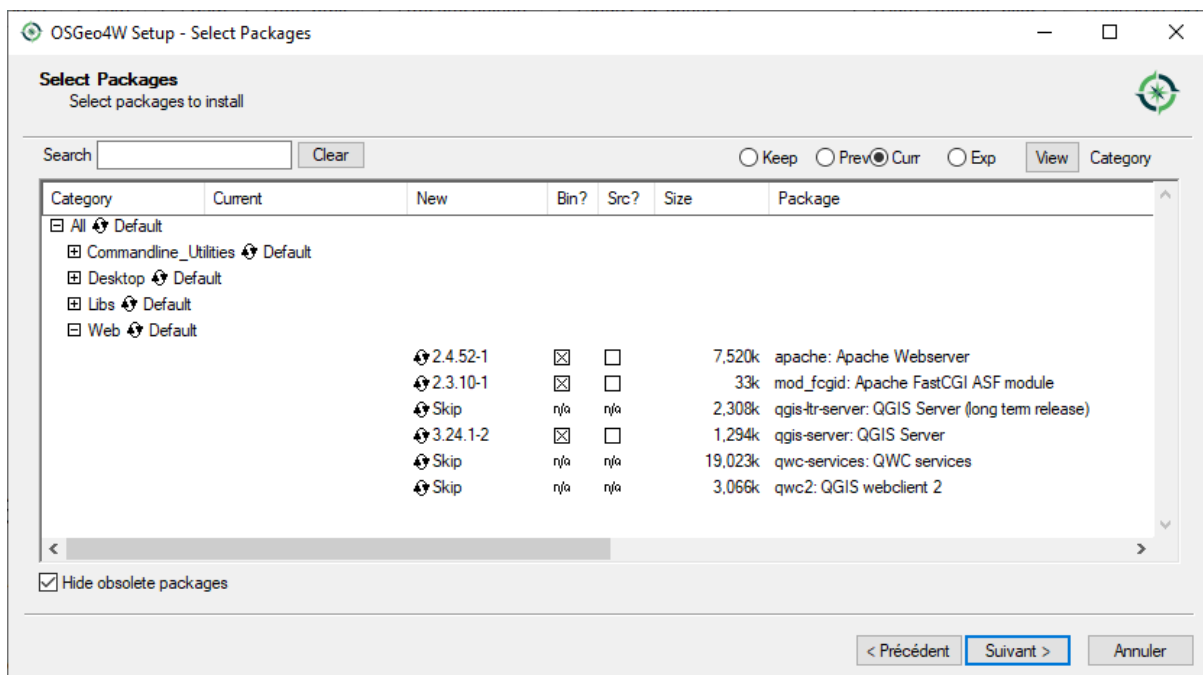
```
fastcgi_param DISPLAY ":99";
```

2.2 Instalação no Windows

O QGIS Server também pode ser instalado em sistemas Windows usando a versão 64 bit do instalador de rede OSGeo4W (<https://qgis.org/en/site/forusers/download.html>).

Um procedimento simples é o seguinte:

1. Baixe e execute o instalador OSGeo4W
2. Siga a “Instalação Avançada” e instale os pacotes **QGIS Desktop**, **QGIS Server apache** e **mod_fcgid**.



3. O Apache não é instalado diretamente como um serviço no Windows. Você precisa fazê-lo:
 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. No console, execute apache-install.bat, que terá saída

```
> 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.
...
```

O serviço está iniciado, como você pode notar no relatório. Mas o servidor pode não funcionar devido à falta de configuração personalizada.

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

Finalidade	Configuração existente	Substituição
(Opcional) Personalize o endereço para ouvir usando um IP e/ou porta, Você pode e adicionar quantas entradas desejar. Indique onde encontrar os arquivos de script	<pre>Listen \${SRVPORT}</pre>	<pre>Listen localhost:8080</pre>
	<pre>ScriptAlias /cgi-bin/ "\$ ↳{SRVROOT}/cgi-bin/"</pre>	<pre>ScriptAlias /cgi-bin/ ↳"C:/OSGeo4W/apps/qgis/ ↳bin/"</pre>
Fornecer permissões na pasta de scripts	<pre><Directory "\${SRVROOT}/ ↳cgi-bin"> AllowOverride None Options None Require all granted </Directory></pre>	<pre><Directory "C:/OSGeo4W/ ↳apps/qgis/bin"> SetHandler cgi- ↳script AllowOverride None Options ExecCGI Require all granted </Directory></pre>
Habilitar extensões de arquivo para uso em arquivos de script. Descomentar e completar	<pre>#AddHandler cgi-script . ↳cgi</pre>	<pre>AddHandler cgi-script . ↳cgi .exe</pre>
Adicionar mais variáveis de configuração personalizadas OSGeo4W	<pre># parse OSGeo4W apache_ ↳conf files IncludeOptional "C:/ ↳OSGeo4W/httpd.d/httpd_ ↳*.conf"</pre>	<pre># 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"</pre>

5. Reiniciar o servidor web Apache

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

6. Abra a janela do navegador para testar uma solicitação de GetCapabilities no Servidor QGIS. Substitua localhost:8080 pelo IP e porta que você definir para ouvir.

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

Um **:arquivo:XML** arquivo com as capacidades deve ser devolvido. Seu servidor está pronto para uso.

2.3 Serve a project

Agora que o Servidor QGIS está instalado e funcionando, só temos que usá-lo.

Obviously, we need a QGIS project to work on. Of course, you can fully customize your project by defining contact information, precise some restrictions on CRS or even exclude some layers. Everything you need to know about that is described later in *Configurar seu projeto*.

But for now, we are going to use a simple project already configured and previously downloaded in `/home/qgis/projects/world.qgs`, as described above.

Ao abrir o projeto e dar uma olhada rápida nas camadas, sabemos que 4 camadas estão disponíveis no momento:

- aeroportos
- lugares
- países
- countries_shapeburst

Você não precisa entender a solicitação completa por enquanto, mas pode recuperar um mapa com algumas das camadas anteriores graças ao Servidor QGIS fazendo algo assim em seu navegador da web para recuperar a camada *países*:

- If you followed the above instructions to install an Apache HTTP Server:

```
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
```

- If you followed the above instructions to install an NGINX HTTP Server:

```
http://qgis.demo/qgisserver?  
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
```

Se você obtiver a próxima imagem, então o Servidor QGIS está rodando corretamente:

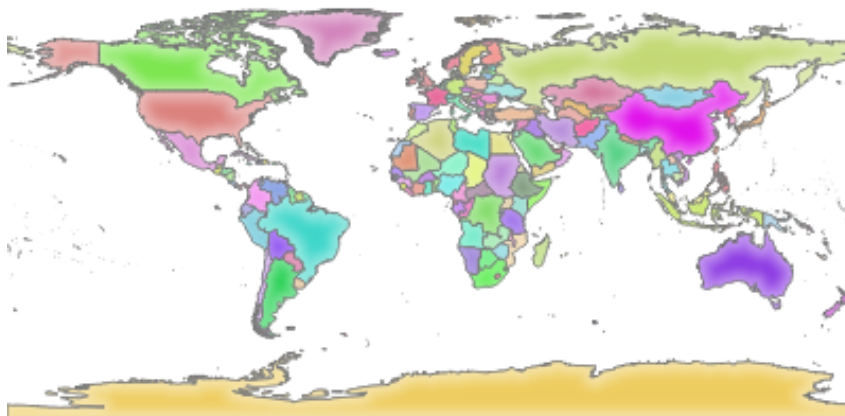


Figura2.1: Server response to a basic GetMap request

Note that you may define **QGIS_PROJECT_FILE** environment variable to use a project by default instead of giving a **MAP** parameter (see *Variáveis ambientais*).

Por exemplo com spawn-fcgi:

```
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 Configurar seu projeto

To provide a new QGIS Server WMS, WFS, OAPIF or WCS, you have to create a QGIS project file with some data or use one of your current project. Define the colors and styles of the layers in QGIS and the project CRS, if not already defined. Then, go to the *QGIS Server* menu of the *Project ► Properties...* dialog and provide some information about the OWS in the *Service Capabilities* tab.

Project Properties — QGIS Server

Services Capabilities | WMS | WMTS | WFS/OAPIF | WCS

☒ **Enable Service Capabilities**

These parameters are used to generate the GetCapabilities document and shall be chosen carefully to avoid interoperability and security issues.

Short name: A name used to identify the root layer. The short name is a text ...

Title: The title should be brief yet descriptive enough to identify this ...

Organization: The name of the service provider.

Online resource: The web site URL of the service provider.

Person: The contact person name for the service.

Position:

E-Mail: The contact person e-mail for the service.

Phone: The contact person phone for the service.

Abstract:

Fees: Conditions Unknown

Access constraints: None

Keyword list: List of keywords separated by comma to help catalog searching.

Test Configuration

Launch

Help Apply Cancel OK

Figura2.2: Definitions for a QGIS Server WMS/WFS/WCS project

You have to *Enable Service Capabilities* first, if it is deactivated. This will appear in the GetCapabilities response of the WMS, WFS or WCS. If you don't check ☒ *Enable Service capabilities*, QGIS Server will use the information given in the `wms_metadata.xml` file located in the `cgi-bin` folder.

2.4.1 Recursos WMS

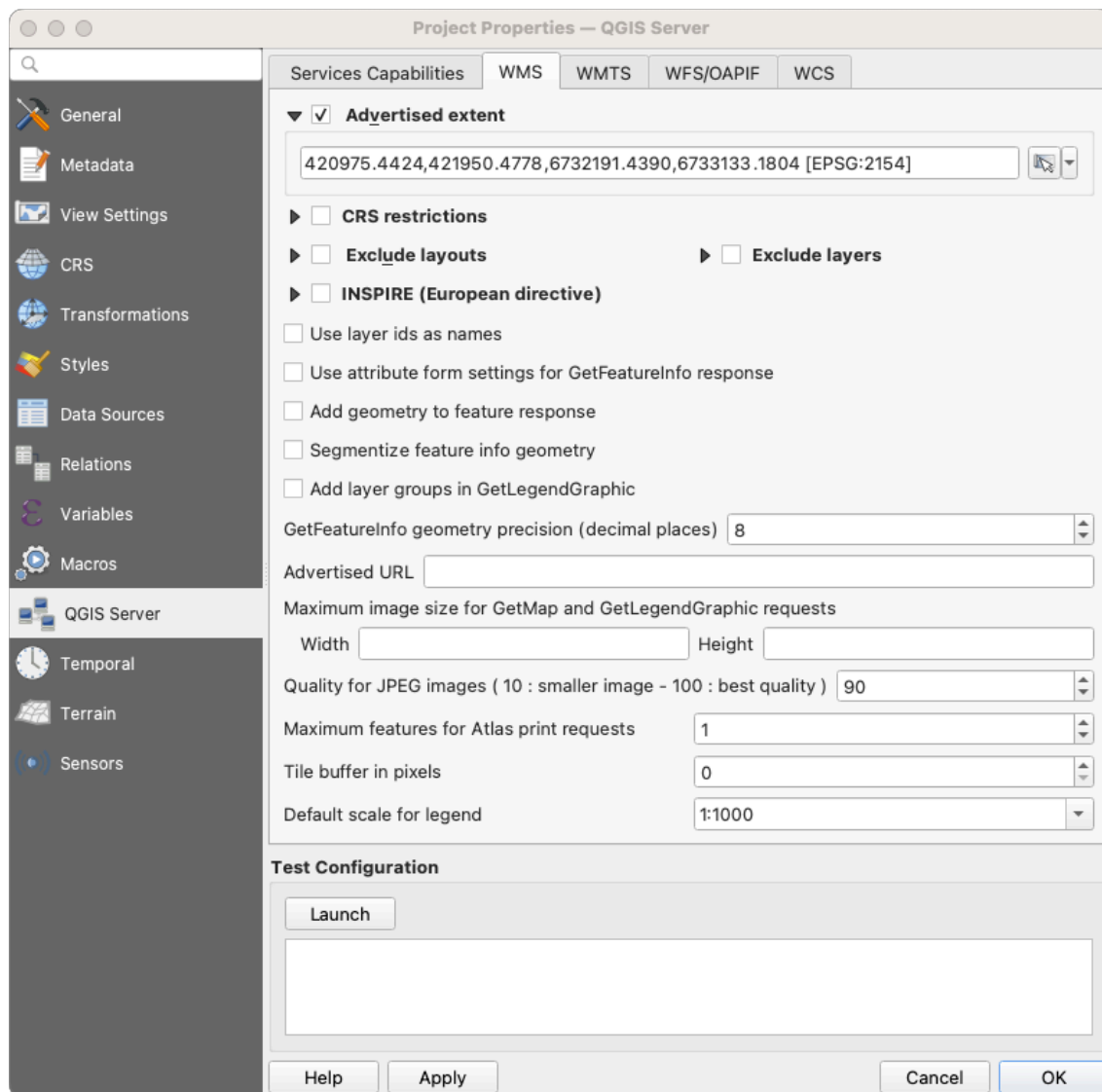





Figura2.3: Definitions in the WMS tab


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 *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.


If you have print layouts defined in your project, they will be listed in the *GetProjectSettings* response, and they can be used by the GetPrint request to create prints, using one of the print layouts as a template. This is a QGIS-specific extension to the WMS 1.3.0 specification. If you want to exclude any print layout from being published by the WMS, check ☒ *Exclude layouts* and click the  button below. Then, select a print layout from the *Select print layout* dialog in order to add it to the excluded layouts list.


If you want to exclude any layer or layer group from being published by the WMS, check ☒ *Exclude Layers* and


click the  button below. This opens the *Select restricted layers and groups* dialog, which allows you to choose the layers and groups that you don't want to be published. Use the `Shift` or `Ctrl` key if you want to select multiple entries. It is recommended that you exclude from publishing the layers that you don't need as this reduces the size of the WMS GetCapabilities response which leads to faster loading times on the client side.

If you check  *Use layer ids as name*, layer ids will be used to reference layers in the GetCapabilities response or GetMap `LAYERS` parameter. If not, layer name or short name if defined (see `vectorservermenu`) is used.

You can receive requested GetFeatureInfo as plain text, XML and GML. The default is XML.

If you wish, you can check  *Add geometry to feature response*. This will include the bounding box for each feature in the GetFeatureInfo response. See also the `WITH_GEOMETRY` parameter.

As many web clients can't display circular arcs in geometries you have the option to segmentize the geometry before sending it to the client in a GetFeatureInfo response. This allows such clients to still display a feature's geometry (e.g. for highlighting the feature). You need to check the  *Segmentize feature info geometry* to activate the option.

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.

You can also use the *GetFeatureInfo geometry precision* option to set the precision of the GetFeatureInfo geometry. This enables you to save bandwidth when you don't need the full precision.

Se você quiser que o QGIS Server anuncie URLs de solicitação específicas na resposta do WMS GetCapabilities, digite a URL correspondente no campo *URL anunciada*

Além disso, você pode restringir o tamanho máximo dos mapas devolvidos pelo pedido GetMap inserindo a largura e altura máximas nos respectivos campos sob :guilabel: Solicitação máxima do GetMap'.

Você pode mudar o fator *Qualidade para imagens JPEG*. O fator de qualidade deve estar na faixa de 0 a 100. Especifique 0 para compressão máxima, 100 para nenhuma compressão.

You can change the limit for atlas features to be printed in one request by setting the *Maximum features for Atlas print requests* field.

When QGIS Server is used in tiled mode (see *TILED parameter*), you can set the *Tile buffer in pixels*. The recommended value is the size of the largest symbol or line width in your QGIS project.

If one of your layers uses the Map Tip display (i.e. to show text using expressions) this will be listed inside the GetFeatureInfo output. If the layer uses a Value Map for one of its attributes, this information will also be shown in the GetFeatureInfo output.

2.4.2 WMTS capabilities

In the *WMTS capabilities* tab you can select the layers you want to publish as WMTS and specify if you want to publish as PNG or JPEG.

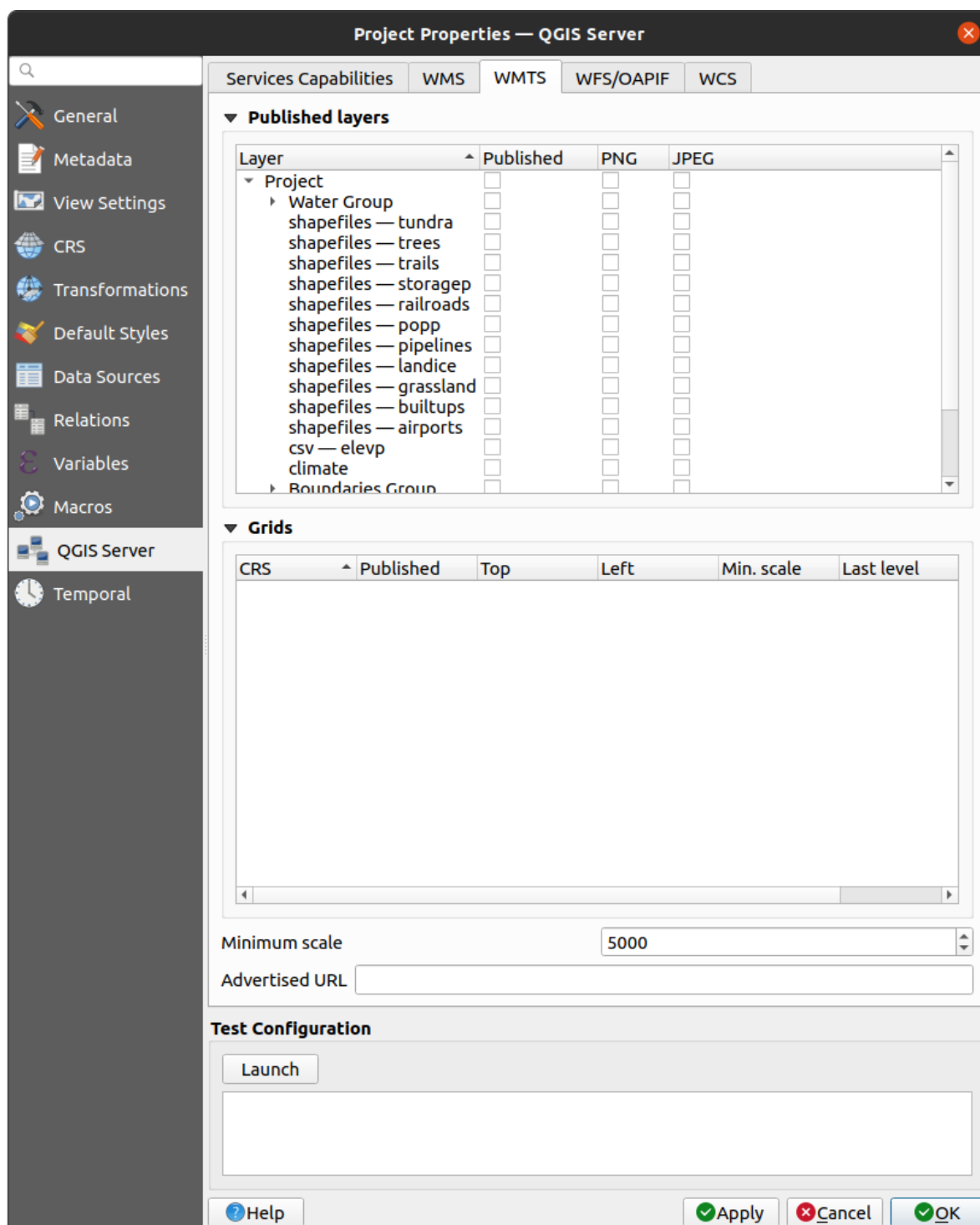


Figura2.4: Definitions in the WMTS tab

If you enter a URL in the *Advertised URL* field of the *WMTS capabilities* section, QGIS Server will advertise this specific URL in the WMTS GetCapabilities response.

2.4.3 WFS/OAPIF capabilities

In the *WFS/OAPIF capabilities* tab, you can select the layers you want to publish as WFS or OAPIF, and specify if they will allow update, insert and delete operations.

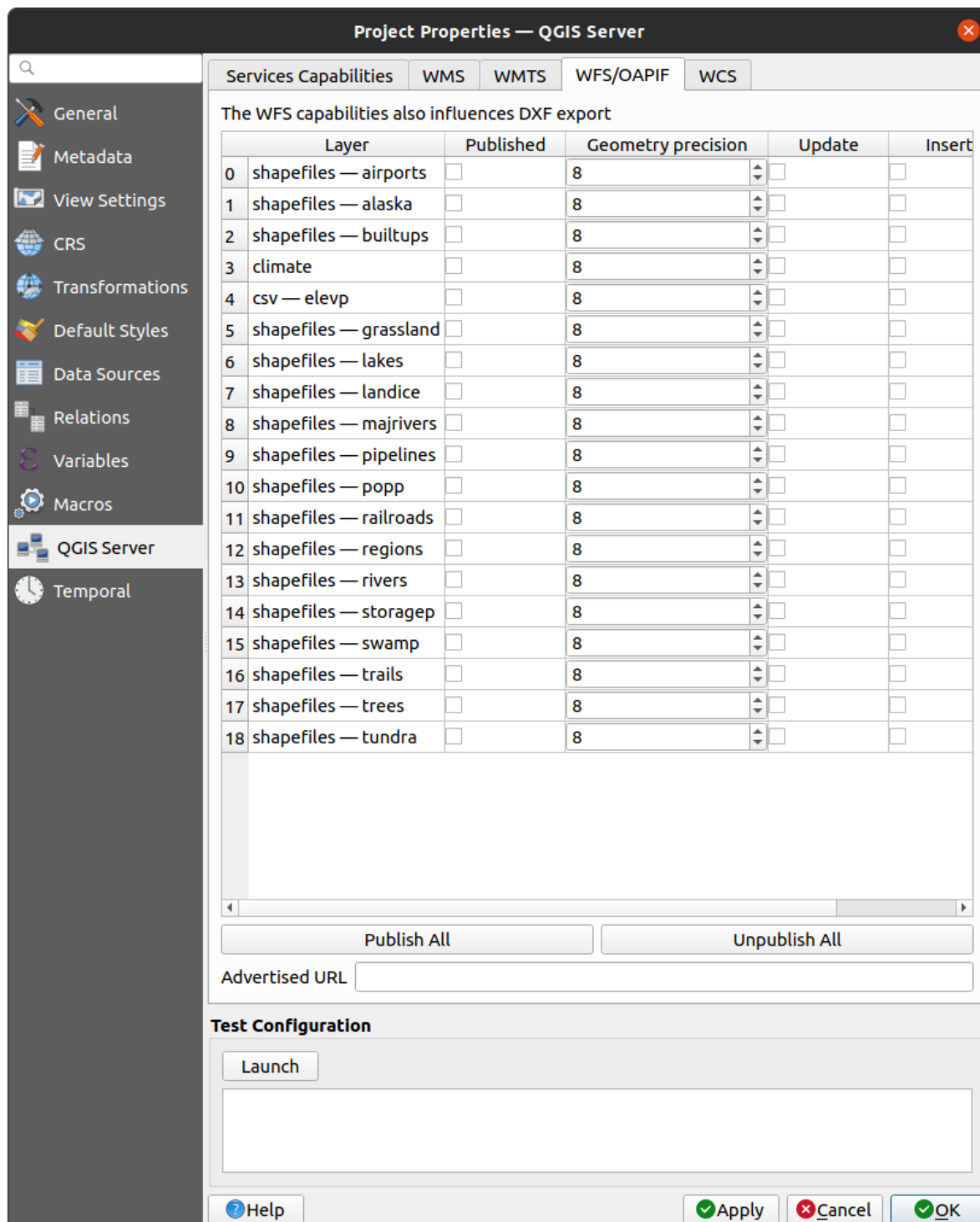


Figura2.5: Definitions in the WFS/OAPIF tab

If you enter a URL in the *Advertised URL* field of the *WFS capabilities* section, QGIS Server will advertise this specific URL in the WFS GetCapabilities response.

2.4.4 WCS capabilities

In the *WCS capabilities* tab, you can select the layers that you want to publish as WCS.

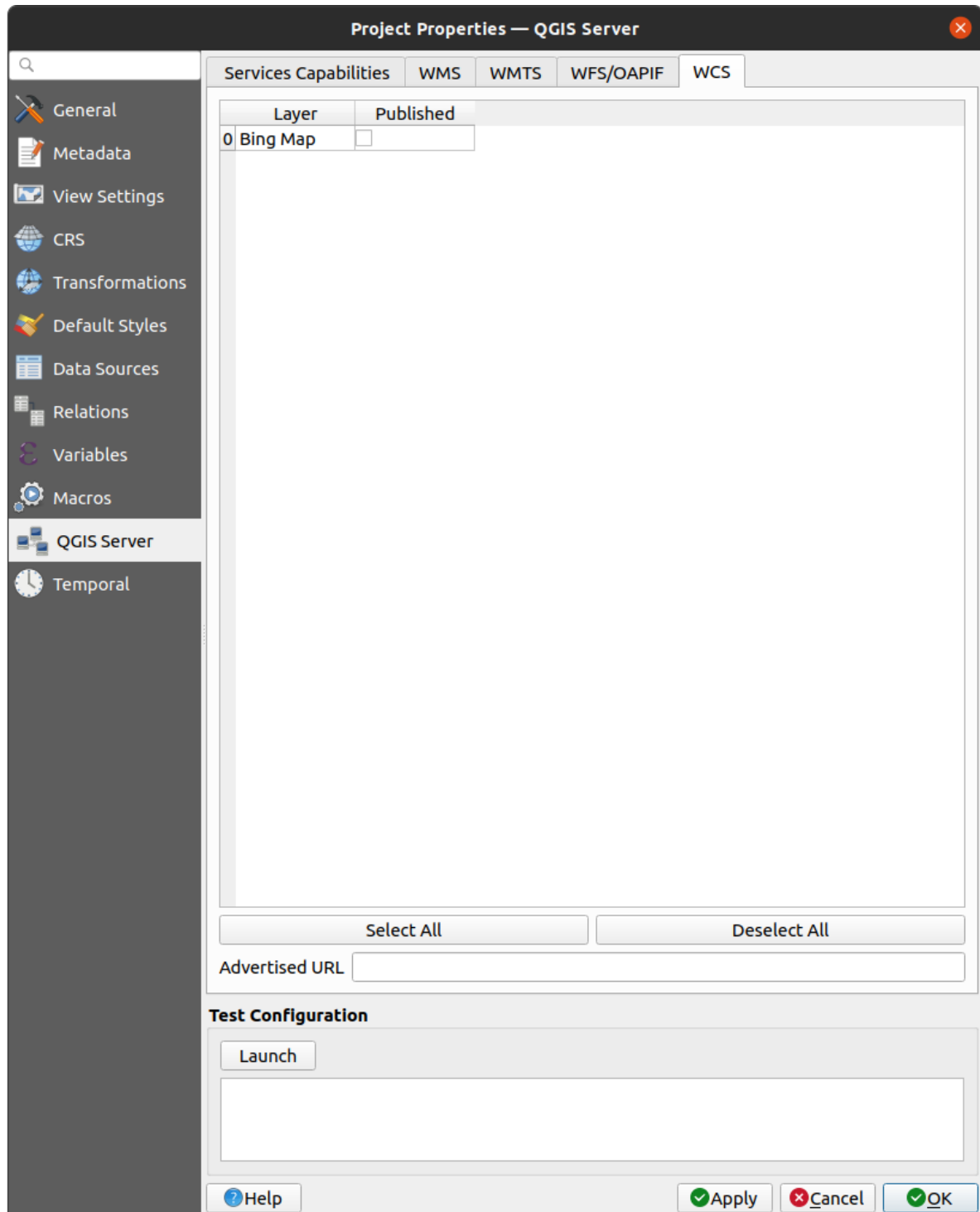


Figura2.6: Definitions in the WCS tab

If you enter a URL in the *Advertised URL* field of the *WCS capabilities* section, QGIS Server will advertise this specific URL in the WCS GetCapabilities response.

2.4.5 Sintonia fina do seu OWS

For vector layers, the *Fields* menu of the *Layer ► Layer Properties* dialog allows you to define for each attribute if it will be published or not. By default, all the attributes are published by your WMS and WFS. If you don't want a specific attribute to be published, uncheck the corresponding checkbox in the *WMS* or *WFS* column.

You can overlay watermarks over the maps produced by your WMS by adding text annotations or SVG annotations to the project file. See the *sec_annotations* section for instructions on creating annotations. For annotations to be displayed as watermarks on the WMS output, the *Fixed map position* checkbox in the *Annotation text* dialog must be unchecked. This can be accessed by double clicking the annotation while one of the annotation tools is active. For SVG annotations, you will need either to set the project to save absolute paths (in the *General* menu of the *Project ► Properties...* dialog) or to manually modify the path to the SVG image so that it represents a valid relative path.

2.5 Integração com terceiros

O Servidor QGIS fornece serviços web padrão OGC como *WMS*, *WFS*, etc. assim ele pode ser utilizado por uma grande variedade de ferramentas para o usuário final.

2.5.1 Integração com o QGIS Desktop

QGIS Desktop is the map designer where QGIS Server is the map server. The maps or QGIS projects will be served by the QGIS Server to provide OGC standards. These QGIS projects can either be files or entries in a database (by using *Project ► Save to ► PostgreSQL* in QGIS Desktop).

Além disso, deve ser estabelecido um fluxo de trabalho dedicado de atualização para atualizar um projeto utilizado por um Servidor QGIS (ou seja, copiar arquivos de projeto para a localização do servidor e reiniciar o Servidor QGIS). Por enquanto, os processos automatizados (como serviço de recarga de mensagens do servidor em fila de espera) ainda não estão implementados.

2.5.2 Integração com MapProxy

MapProxy is a tile cache server and as it can read and serve any WMS/WMTS map server, it can be directly connected to QGIS server web services and improve end user experience.

2.5.3 Integração com QWC2

QWC2 <<https://github.com/qgis/qwc2>>`_ é uma aplicação web responsiva e dedicada ao Servidor QGIS. Ele ajuda você a construir um visualizador de mapas altamente personalizado com seleção de camadas, informações de características, etc... Também estão disponíveis muitos plugins como autenticação ou serviço de impressão, a lista completa está disponível neste `repositório <<https://github.com/qwc-services>>>`_.

QGIS Server is able to serve data according to standard protocols as described by the **Open Geospatial Consortium (OGC)**:

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

Extra vendor parameters and requests are supported in addition to the original standard that greatly enhance the possibilities of customizing its behavior thanks to the QGIS rendering engine.

3.1 Fundamentos

This section describes concepts and parameters mutually shared by services. Some of these are standard and defined in OGC specifications while others are very specific to QGIS Server.

Conceitos padrão:

Conceito	Descrição
<i>SERVICE</i>	Nome do serviço
<i>REQUEST</i>	Nome da requisição

Conceitos de fornecedor:

Conceito	Descrição
<i>MAP</i>	arquivo de projeto QGIS
<i>FILE_NAME</i>	Nome do arquivo do arquivo baixado
<i>Nome curto</i>	Definição de nome curto

3.1.1 SERVICE

This standard parameter allows to specify the name of the service to use for a specific *request* and has to be formed like `SERVICE=NAME`.

URL example for the **WMS** service:

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

Nota: Not available for REST based services like *WFS3 (OGC API Features)*.

3.1.2 REQUEST

This standard parameter allows to specify the name of the request to execute for a specific *service* and has to be formed like `REQUEST=RequestName`.

URL example for the **GetCapabilities** request:

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

Nota: Not available for REST based services like *WFS3 (OGC API Features)*.

3.1.3 MAP

This vendor parameter allows to define the QGIS project file to use. It may be an absolute path or a path relative to the location of the server executable `qgis_mapserv.fcgi`. MAP is mandatory by default because a request needs a QGIS project to actually work. However, the **QGIS_PROJECT_FILE** environment variable may be used to define a default QGIS project. In this specific case, MAP is no longer a required parameter. For further information you may refer to the *Configuração avançada* chapter.

Exemplo de URL:

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

3.1.4 FILE_NAME

Se este parâmetro do fornecedor for definido, a resposta do servidor será enviada ao cliente como um anexo de arquivo com o nome do arquivo especificado.

URL example to save an XML **GetCapabilities** document:

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

Nota: Not available for REST based services like *WFS3 (OGC API Features)*.

3.1.5 Nome curto

A number of elements have both a **short name** and a **title**. The short name is a text string used for machine-to-machine communication while the title is for the benefit of humans. For example, a dataset might have the descriptive title “*Maximum Atmospheric Temperature*” and be requested using the abbreviated short name “*ATMAX*”. You can set title, short name and abstract for:

- **Layers:** right-click on a layer and choose *Properties... ► QGIS Server ► Description*.
- **Groups:** right-click on a group and select *Set Group WMS data*
- **Project:** go to *Project ► Properties... ► QGIS Server ► Service Capabilities*.

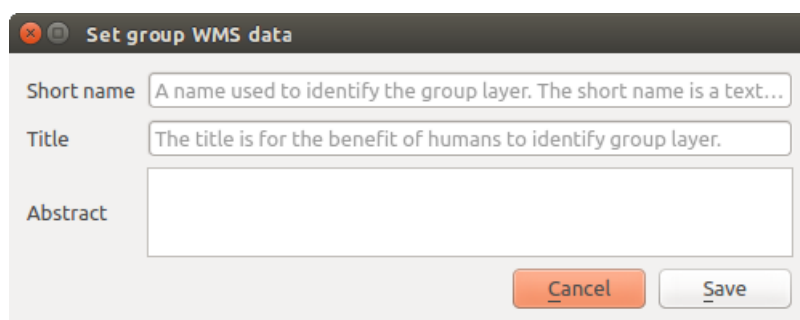


Figura3.1: Set group WMS data dialog

Assim, o nome abreviado pode ser usado para identificar esses itens ao interagir com o Servidor QGIS. Por exemplo, com o parâmetro padrão CAMADAS:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&LAYERS=shortname1,shortname2
&...
```

3.2 Web Map Service (WMS) Serviço de mapa de web

Os padrões WMS **1.1.1** e **1.3.0** implementados no servidor do QGIS fornecem uma interface HTTP para solicitar imagens de mapas ou legendas geradas a partir de um projeto QGIS. Uma solicitação WMS típica define o projeto QGIS a ser usado, as camadas a serem renderizadas, bem como o formato de imagem a ser gerado. O suporte básico também está disponível para **Styled Layer Descriptor (SLD)**

Specifications:

- [WMS 1.1.1](#)
- [WMS 1.3.0](#)
- [SLD 1.1.0 WMS profile](#)

Standard requests provided by QGIS Server:

Request	Description
<i>GetCapabilities</i>	Retorna metadados XML com informações sobre o servidor
<i>GetMap</i>	Retorna um mapa
<i>GetFeatureInfo</i>	Recupera dados (geometria e valores) para uma localização de pixel
<i>GetLegend-Graphic</i>	Retorna os símbolos das legendas
<i>GetStyle(s)</i>	Retorna documento XML com descrição de estilo no SLD
<i>DescribeLayer</i>	Retorna informações sobre a disponibilidade de WFS e WCS, respectivamente, para camadas vetoriais e raster

Pedidos vendor fornecidos pelo QGIS Server:

Request	Description
<i>GetPrint</i>	Retorna um layout QGIS
<i>GetProjectSettings</i>	Retorna informação específica sobre o servidor do QGIS
<i>GetSchemaExtension</i>	Retorna metadados XML sobre recursos estendidos opcionais

3.2.1 GetCapabilities

Parâmetros padrões para o pedido **GetCapabilities** de acordo com as especificações OGC WMS 1.1.1 e 1.3.0:

Parameter	Required	Description
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Name of the request (GetCapabilities)
<i>VERSION</i>	No	Version of the service

O pedido do **GetCapabilities** também suporta os seguintes parâmetro:

Parameter	Required	Description
<i>MAP</i>	Yes	QGIS project file

URL example:

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

3.2.2 GetMap

Parâmetros padrões para o pedido de ****GetMap*** que esteja consoante às especificações OGC WMS 1.1.1 e 1.3.0

Parameter	Required	Description
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Nome do pedido (** GetMap**)
<i>VERSION</i>	Yes	Version of the service
<i>LAYERS</i>	No	Camadas para mostrar
<i>STYLES</i>	No	Estilo das camadas
<i>SRS / CRS</i>	Yes	Sistema de referência de coordenadas
<i>BBOX</i>	Yes	Extensão do mapa
<i>WIDTH</i>	Yes	Largura da imagem em pixels
<i>HEIGHT</i>	Yes	Altura da imagem em pixels
<i>FORMAT</i>	No	Formato de imagem
<i>TRANSPARENT</i>	No	Plano de fundo transparente
<i>SLD</i>	No	URL do SLD para ser usado para estilização
<i>SLD_BODY</i>	No	SLD em linha (XML) para ser usado para estilização

Em adição àqueles que são padrões, o servidor do QGIS suporta *redlining*, *external WMS layers* como também as seguintes parâmetros extras:

Parameter	Re-quired	Description
<i>MAP</i>	Yes	QGIS project file
<i>BGCOLOR</i>	No	Especificar a cor de fundo
<i>DPI</i>	No	Especificar a resolução de saída
<i>IMAGE_QUALITY</i>	No	Compressão JPEG
<i>OPACITIES</i>	No	Opacidade para camada ou grupo
<i>FILTER</i>	No	Subconjunto de características
<i>SELECTION</i>	No	Recursos de destaque
<i>FILE_NAME</i>	No	File name of the downloaded file Somente para o <i>FORMAT=application/dxf</i>
<i>FORMAT_OPTIONS</i>	No	Opções do formato de arquivo especificado Somente para o <i>FORMAT=application/dxf</i>
<i>TILED</i>	No	Trabalhando no <i>tiled mode</i>

URL example:

```
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

Este parâmetro permite especificar a versão do serviço a ser utilizada. Os valores disponíveis para o parâmetro `VERSÃO` são:

- 1.1.1
- 1.3.0

De acordo com o número da versão, diferenças sutis têm de ser esperadas como explicado depois para os próximos parâmetros:

- CRS / SRS
- BBOX

LAYERS

Este parâmetro permite especificar as camadas a serem exibidas no mapa. Os nomes devem ser separados por vírgula.

Em adição, o servidor do QGIS introduziu algumas opções para selecionar camadas por:

- 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.
- a *short name*

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

STYLES

Este parâmetro pode ser usado para especificar o estilo de uma camada para a etapa de renderização. Os estilos devem ser separados por vírgula. O nome do estilo padrão é `padrão`.

SRS / CRS

Esse parâmetro permite indicar a saída do sistema de referência espacial do mapa no WMS **1.1.1** e tem que ser formado como `EPSG:XXXX`. Note que CRS é também suportado se versão atual é **1.1.1**

Para o WMS **1.3.0**, o parâmetro CRS é preferível, mas SRS é também suportado.

Note que se ambos os parâmetros `CRS` e `SRS` são indicados no pedidos, então é a versão atual que é indicada no parâmetro `VERSION`, o qual é decisivo.

No próximo caso, o parâmetro `SRS` é mantido independente de qual seja o parâmetro `VERSION`, porque o `CRS` não é indicado.

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

No próximo caso, o parâmetro `SRS` é mantido ao invés do `CRS`, devido ao parâmetro `VERSION`.

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.1.1
&CRS=EPSG:4326
&SRS=EPSG:2854
&...
```

No próximo caso, o parâmetro CRS é mantido ao invés do SRS devido ao VERSION. Parâmetro:

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

BBOX

Este parâmetro permite especificar a extensão do mapa com unidades de acordo com o SRC atual. As coordenadas devem ser separadas por vírgula.

O parâmetro BBOX é formado como `min_a, min_b, max_a, max_b` e a definição do eixo b é diferente de acordo com a versão atual do parâmetro VERSION.

- No WMS **1.1.1**, a ordenação do eixo é sempre leste/norte.
- no WMS **1.3.0**, a ordenação do eixo depende da autoridade do CRS.

Por exemplo, no caso do EPSG:4326 e WMS **1.1.1**, a é a longitude (leste) e b é a latitude (norte), levando a um pedido como:

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

Mas no do WMS **1.3.0**, a ordenação do eixo definida no banco de dados do EPSG é norte/leste, então a é a latitude e b é a longitude.

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetMap
&VERSION=1.3.0
&CRS=epsg:4326
&BBOX=-90,-180,90,180
&...
```

WIDTH

Este parâmetro permite especificar a largura em pixels da imagem de saída.

HEIGHT

Este parâmetro permite especificar a altura em pixels da imagem de saída.

FORMAT

Este parâmetro pode ser usado para especificar o formato da imagem do mapa. Os valores disponíveis são:

- jpg
- jpeg
- image/jpeg
- image/png
- image/png; mode=1bit
- image/png; mode=8bit
- image/png; mode=16bit
- image/webp
- application/dxf: somente camadas que possuem acesso no serviço WFS são exportadas no formato DXF

URL example:

```
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:SYMBOLLAYERSYMBOLOLOGY;SCALE:250  
&FILE_NAME=plan.dxf
```

TRANSPARENT

Este parâmetro booleano pode ser usado para especificar a transparência do plano de fundo. Os valores disponíveis são (não diferenciam maiúsculas de minúsculas):

- TRUE
- FALSE

No entanto, este parâmetro é ignorado se o formato da imagem indicado com `FORMATO` for diferente de PNG.

BGCOLOR

Esse parâmetro permite indicar uma cor de fundo para a imagem do mapa. De qualquer forma, não pode ser combinado com o parâmetro `TRANSPARENTE` no caso de imagens PNG (a transparência ganha prioridade). A cor pode ser literal ou na notação hexadecimal.

Exemplo de URL com a notação literal:

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

Exemplo de URL com a notação hexadecimal:

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

DPI

Este parâmetro pode ser usado para especificar a resolução de saída solicitada.

IMAGE_QUALITY

Este parâmetro é usado apenas para imagens JPEG. Por padrão, a compressão JPEG é -1.

Você pode mudar o padrão para cada projeto de QGIS no menu *OWS Server ► WMS capabilities* da caixa de diálogo *Project ► Properties...*. Se você quer sobrepor em um pedido *GetMap*, você pode fazê-lo usando o parâmetro `IMAGE_QUALITY`.

OPACITIES

A vírgula separa a lista de valores de opacidade. A opacidade pode ser configurada na camada ou a nível de grupo. Permitindo valores que vão desde de 0 (completamente transparente) para 255 (completamente opaco)

URL example:

```
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
```



Figura3.2: À esquerda OPACIDADES=255, 0 e à direita OPACIDADES=255, 255

FILTER

Um subconjunto de camadas pode ser selecionado com o parâmetro `FILTRO`. A sintaxe é basicamente a mesma para a linha de subconjuntos do QGIS. No entanto, há algumas restrições para evitar injeções de SQL no banco de dados via servidor SQL. Se uma linha de código perigosa é achada no parâmetro, o servidor do QGIS retornará o seguinte erro:

```
<ServiceExceptionReport>
  <ServiceException code="Security">The filter string XXXXXXXXXX has been rejected_
  ↳because of security reasons.
  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 example:

```
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'
```



Figura3.3: Resposta do servidor para um pedido GetMap com parâmetro FILTER

Neste exemplo, o mesmo filtro "name" = 'Paris' é aplicado para camadas countries e countries_shapeburst, enquanto o filtro "name" = 'Paris' é somente aplicável para "places".

Nota: É possível fazer pesquisas de atributos vi GetFeatureInfo e omitir o parâmetro X/Y se um FILTER está lá. O servidor do QGIS então retorna informações sobre características correspondentes e gera uma caixa limitadora combinada na saída do XML

SELECTION

O parâmetro SELECTION pode destacar características de uma ou mais camadas. Atributos de vetor podem ser selecionados por uma vírgula separando listas com o atributo ids

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

A seguinte imagem apresenta a resposta para um pedido GetMap usando a opção SELECTION e.g. `http://myserver.com/...&SELECTION=countries:171,65`.

Como esses id's de recurso correspondem no banco de dados fonte para **France** e **Romania** eles são destacados em amarelo



Figura3.4: A resposta do servidor para pedido GetMap com o parâmetro SELECTION.

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:** para ser usada para regras de simbologia, filtros e estilos (Não é a escala real dos dados - o dado continua na escala original)
- **MODE:** corresponde as opções de exportação oferecidas na caixa de diálogo QGIS Desktop DXF. Possíveis valores são `NOSYMBOLGY``, ```FEATURESYMBOLGY`` e `SYMBOLLAYERSYMBOLGY``.
- **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:** se habilitado, o título da camada será usa como o nome da camada.
- **CODE:** especifique um codec para ser usado para codificação. O padrão é `ISO-8859-1`. Marque a caixa de diálogo QGIS desktop DXF para valores válidos.
- **NO_MTEXT:** Usa texto em vez de MTEXT para rótulos.
- **FORCE_2D:** Saída Force 2D. Isso é requerido para a largura do polyline.

TILED

Por motivos de performance, o servidor do QGIS pode ser usado em modo lado a lado (tiled). Nesse modo, o cliente pede vários pequenos ladrilhos fixos e então junta-os para formar o mapa inteiro. Fazendo isso, símbolos em cima ou perto do limite entre dois ladrilhos talvez apareçam cortados porque eles estão presentes somente em um dos ladrilhos

Configure o parâmetro `TILED` para `TRUE` para dizer ao servidor do QGIS para trabalhar em modo *tiled* e para aplicar o *Tile bufer* configurado no projeto do QGIS (veja ref: *Creating wms from project*).

Quando `TILED` é `TRUE` e quando um não zero *Tile buffer* é configurado no projeto do QGIS, características fora do ladrilho estendido são desenhadas para evitar simbologias cortadas nos limites do ladrilho.

`TILED` padronizado em `FALSE`.

3.2.3 GetFeatureInfo

Parâmetros padrão para o pedido **GetFeatureInfo** de acordo com as especificações do OGC WMS 1.1.1 e 1.3.0:

Parameter	Require	Description
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Nome do pedido (GetFeatureInfo)
<i>VERSION</i>	No	Version of the service
<i>QUERY_LAYERS</i>	Yes	Camadas para consulta
<i>STYLES</i>	Yes	Camadas para mostrar (idênticas ao <i>QUERY_LAYERS</i>)
<i>SRS</i> / <i>CRS</i>	No	Estilo das camadas
<i>BBOX</i>	Yes	Sistema de referência de coordenadas
<i>WIDTH</i>	No	Extensão do mapa
<i>HEIGHT</i>	Yes	Largura da imagem em pixels
<i>TRANSPARENT</i>	Yes	Altura da imagem em pixels
<i>INFO_FORMAT</i>	No	Plano de fundo transparente
<i>FEATURE_COUNT</i>	No	Formato de saída
<i>I</i>	No	Número máximo de feições a serem retornadas
<i>X</i>	No	Coluna de pixels do ponto para consultar
<i>J</i>	No	Igual ao parâmetro <i>I</i> , mas em WMS 1.1.1
<i>Y</i>	No	Linhas de pixels do ponto para consultar.
<i>WMS_PR</i>	No	Igual ao parâmetro <i>J</i> , mas em WMS 1.1.1
	No	A precisão (números de dígitos) para ser usado quando retornando uma geometria (veja <i>how para adicionar geometria à resposta do do recurso</i>). O valor padrão é <code>-1</code> o que significa que a precisão definida no projeto é usada.

In addition to the standard ones, QGIS Server supports the following extra parameters:

Parameter	Required	Description
<i>MAP</i>	Yes	QGIS project file
<i>FILTER</i>	No	Subconjunto de características
<i>FI_POINT_TOLERANCE</i>	No	Tolerância em pixels para camadas de ponto
<i>FI_LINE_TOLERANCE</i>	No	Tolerância em pixels para camadas de linha
<i>FI_POLYGON_TOLERANCE</i>	No	Tolerância em pixels para camadas de polígono
<i>FILTER_GEOM</i>	No	Filtragem de geometria
<i>WITH_DISPLAY_NAME</i>	No	Add the feature display name to the output
<i>WITH_MAPTIP</i>	No	Adicione dicas de mapa na saída.
<i>WITH_GEOMETRY</i>	No	Adicionar geometria à saída

URL example:

```
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

Este parâmetro pode ser usado para especificar o formato do resultado. Os valores disponíveis são:

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

QUERY_LAYERS

Este parâmetro especifica as camadas a serem exibidas no mapa. Os nomes são separados por uma vírgula.

Em adição, o servidor do QGIS introduz opções para selecionar camadas por:

- nome curto
- camada id

Veja o parâmetro `LAYERS` definido no [GetMap](#) para mais informações.

FEATURE_COUNT

This parameter specifies the maximum number of features per layer to return. For example if `QUERY_LAYERS` is set to `layer1, layer2` and `FEATURE_COUNT` is set to 3 then a maximum of 3 features from `layer1` will be returned. Likewise a maximum of 3 features from `layer2` will be returned.

Por padrão, apenas 1 feição por camada é retornada.

I

Este parâmetro, definido no WMS 1.3.0, permite especificar a coluna de pixel do ponto de consulta.

X

Mesmo parâmetro que `I`, mas definido no WMS 1.1.1.

J

Este parâmetro, definido no WMS 1.3.0, permite especificar a linha de pixel do ponto de consulta.

Y

Mesmo parâmetro que `J`, mas definido no WMS 1.1.1.

FI_POINT_TOLERANCE

Este parâmetro especifica a tolerância em pixels para camadas de pontos.

FI_LINE_TOLERANCE

Este parâmetro especifica a tolerância em pixels para camadas de linha.

FI_POLYGON_TOLERANCE

Este parâmetro especifica a tolerância em pixels para camadas de polígono.

FILTER_GEOM

Esse parâmetro especifica uma geometria WKT na qual cada característica tem que se intersectar

WITH_DISPLAY_NAME

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

Valores disponíveis são (não em caso sensíveis):

- TRUE
- FALSE

WITH_MAPTIP

Esse parâmetro especifica se será ou não adicionadas dicas de mapa na saída.

Valores disponíveis são (não em caso sensíveis):

- TRUE
- FALSE

WITH_GEOMETRY

Esse parâmetro especifica se adiciona-se ou não as geometrias na saída. Para usar esse recurso você deve primeiro habilitar a seguinte opção no projeto do QGIS :guilabel: *Adicionar geometria para a resposta do recurso* para adicionar a geometria à resposta do recurso. Veja *Configure your project*.

Valores disponíveis são (não em caso sensíveis):

- TRUE
- FALSE

3.2.4 GetLegendGraphic

Standard parameters for the **GetLegendGraphic** request according to the OGC WMS 1.1.1 and 1.3.0 specifications:

Parameter	Required	Description
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Name of the request (GetLegendGraphic)
<i>VERSION</i>	No	Version of the service
<i>LAYERS</i>	Yes	Camadas para mostrar
<i>STYLES</i>	No	Estilo das camadas
<i>SRS / CRS</i>	No	Sistema de referência de coordenadas
<i>BBOX</i>	No	Extensão do mapa
<i>WIDTH</i>	No	Largura da imagem em pixels
<i>HEIGHT</i>	No	Altura da imagem em pixels
<i>FORMAT</i>	No	Formato da legenda
<i>TRANSPARENT</i>	No	Plano de fundo transparente

Em adição aos padrões, o servidor do QGIS suporta parâmetros extras para alterar o tamanho dos elementos da legenda ou a propriedade das fontes por título de camada e rótulos de itens:

Parameter	Required	Description
<i>MAP</i>	Yes	QGIS project file
<i>SRCWIDTH</i>	No	Largura do mapa
<i>SRCHEIGHT</i>	No	Altura do mapa
<i>SHOWFEATURECOUNT</i>	No	Adicionar recurso de contador de recursos
<i>RULE</i>	No	Regra símbolo para renderizar.
<i>RULELABEL</i>	No	Renderização do item “rótulos”
<i>BOXSPACE</i>	No	Espaço entre o quadro da legenda e o conteúdo (mm)
<i>LAYERSPACE</i>	No	Espaço vertical entre camadas (mm)
<i>LAYERTITLESPACE</i>	No	Espaço vertical entre o título da camada e os itens (mm)
<i>SYMBOLSPACE</i>	No	Espaço vertical entre o símbolo e os itens (mm)
<i>ICONLABELSPACE</i>	No	Espaço horizontal entre o símbolo e o rótulo (mm)
<i>SYMBOLWIDTH</i>	No	Largura da pré-visualização do símbolo (mm)
<i>SYMBOLHEIGHT</i>	No	Altura da visualização do símbolo (mm)
<i>LAYERTITLE</i>	No	Renderização da camada “título”
<i>LAYERFONTFAMILY</i>	No	Família de fontes da camada
<i>LAYERFONTBOLD</i>	No	Renderização em negrito do título da camada
<i>LAYERFONTSIZE</i>	No	Tamanho da fonte do título da camada (pt)
<i>LAYERFONTITALIC</i>	No	Renderização em itálico do título da camada
<i>LAYERFONTCOLOR</i>	No	Cor do título da camada
<i>ITEMFONTFAMILY</i>	No	Família de fontes do item
<i>ITEMFONTBOLD</i>	No	Renderização em negrito do rótulo do item
<i>ITEMFONTSIZE</i>	No	Tamanho da fonte do rótulo do item (pt)
<i>ITEMFONTITALIC</i>	No	Renderização em itálico do item “rótulo”
<i>ITEMFONTCOLOR</i>	No	Cor do rótulo do item
<i>ADDLAYERGROUPS</i>	No	Adds the layer groups to JSON output

BBOX

Esse parâmetro pode ser usado para especificar a área geográfica para qual a legenda deve construir (seu formato é descrito como [here](#)) mas não pode ser combinado com o parâmetro REGRA . Os parâmetros SRS/CRS se tornam mandatórios quando usando o parâmetro BBOX.

URL example:

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

Nota: Quando o parâmetro BBOX é definido, a legenda é referida como *Legenda baseada em conteúdo*

WIDTH

Esse parâmetro não é usado por padrão, mas se torna mandatório quando o parâmetro `RULE` é definido. Nesse caso, permite especificar a largura em pixels da saída da imagem

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

HEIGHT

Esse parâmetro não é usado por padrão, mas se torna mandatório quando o parâmetro `RULE` é definido. Nesse caso, permite especificar a altura em pixels da saída da imagem

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

FORMAT

Este parâmetro pode ser usado para especificar o formato da imagem da legenda. Os valores disponíveis são:

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

Para JSON, símbolos são encodados com Base64 e a maioria das outras opções relacionadas ao layout ou fontes não são levadas em consideração porque a legenda deve ser construída do lado do cliente. O parâmetro `REGRA` não pode ser combinado com esse formato.

Exemplo de URL com a saída JSON correspondente:

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

E a saída JSON correspondente:

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

SRCWIDTH

This parameter may be defined when the `RULE` parameter is set. In this case, the `SRCWIDTH` value is forwarded to the underlying `GetMap` request as the `WIDTH` parameter while the `WIDTH` parameter of `GetLegendGraphic` is used for the image legend size.

SRCHEIGHT

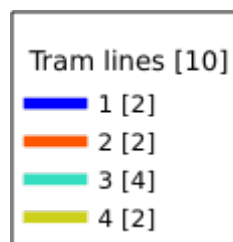
This parameter may be defined when the `RULE` parameter is set. In this case, the `SRCHEIGHT` value is forwarded to the underlying `GetMap` request as the `HEIGHT` parameter while the `HEIGHT` parameter of `GetLegendGraphic` is used for the image legend size.

SHOWFEATURECOUNT

Esse parâmetro pode ser usado para ativar o recurso de contagem na legenda. Valores disponíveis são (não são case sensitives):

- `TRUE`
- `FALSE`

Por exemplo:



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 example:

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

RULELABEL

Esse parâmetro permite controlar a renderização do rótulo do item. Valores disponíveis são (não case sensitives):

- `VERDADEIRO`: exibe o rótulo do item
- `FALSO`: oculta o rótulo do item
- `AUTO`: esconde o rótulo do item para camadas com renderização *Simbolo único*

URL example:

```
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
```



Figura3.5: Renderizar legenda sem rótulo para camadas de símbolo único.

BOXSPACE

Este parâmetro permite especificar o espaço entre o quadro da legenda e o conteúdo em milímetros. Por padrão, o valor do espaço é 2 mm.

URL example:

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




Figura3.6: Para a esquerda BOXSPACE=0 e para a direita BOXSPACE=15

LAYERSPACE

Este parâmetro permite especificar o espaço vertical entre as camadas em milímetros. Por padrão, o valor do espaço é 3 mm.

URL example:

```
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
```



Figura3.7: Para a esquerda LAYERSPACE=0 e para a direita LAYERSPACE=10

LAYERTITLESPACE

Este parâmetro permite especificar o espaço vertical entre o título da camada e os itens seguintes em milímetros. Por padrão, o valor do espaço é 3 mm.

URL example:

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

(continua na próxima página)

(continuação da página anterior)

```
&TRANSPARENT=TRUE
&LAYERTITLESPACE=0
```

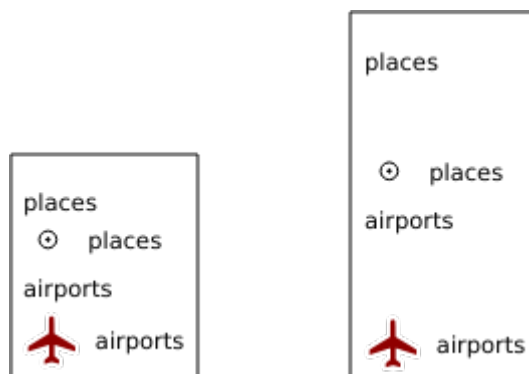


Figura3.8: Para a esquerda LAYERTITLESPACE=0 e para a direita LAYERTITLESPACE=10

SYMBOLSPACE

Este parâmetro permite especificar o espaço vertical entre o símbolo e o item seguinte em milímetros. Por padrão, o valor do espaço é 2 mm.

URL example:

```
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
```

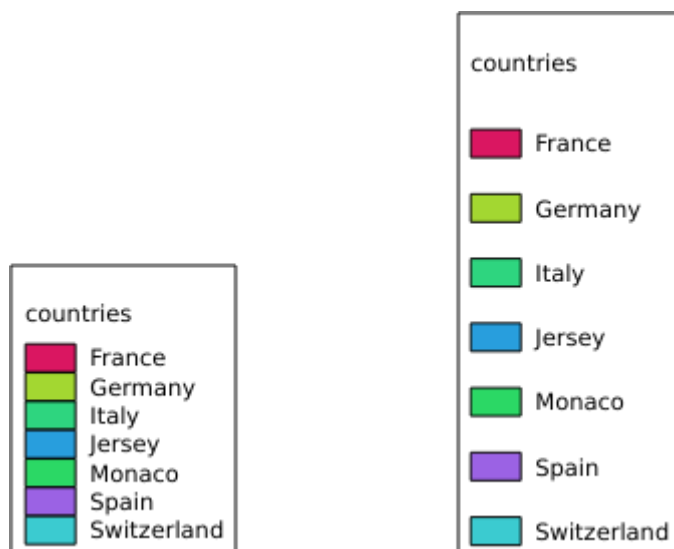


Figura3.9: Para a esquerda SYMBOLSPACE=0 e para a direita SYMBOLSPACE=5

ICONLABELSPACE

Este parâmetro permite especificar o espaço horizontal entre o símbolo e o texto do rótulo em milímetros. Por padrão, o valor do espaço é 2 mm.

URL example:

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



Figura3.10: Para a esquerda ICONLABELSPACE=0 e para a direita ICONLABELSPACE=10

SYMBOLWIDTH

Este parâmetro permite especificar a largura da visualização do símbolo em milímetros. Por padrão, o valor da largura é 7 mm.

URL example:

```
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
```



Figura3.11: Para a esquerda SYMBOLWIDTH=2 e para a direita SYMBOLWIDTH=20

SYMBOLHEIGHT

Este parâmetro permite especificar a altura da visualização do símbolo em milímetros. Por padrão, o valor da altura é 4 mm.

URL example:

```
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
```



Figura3.12: Para a esquerda SYMBOLHEIGHT=2 e para a direita SYMBOLHEIGHT=6

LAYERTITLE

Esse parâmetro especifica se renderiza ou não o título da camada

Valores disponíveis são (não em caso sensíveis):

- VERDADEIRO (valor padrão)
- FALSE

LAYERFONTFAMILY

Esse parâmetro especifica a família da fonte para a ser usada para renderizar o título da camada

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

LAYERFONTBOLD

Esse parâmetro especifica se o título da camada será renderizado em negrito. Valores disponíveis são (não case sensitives):

- TRUE
- FALSE

URL example:

```
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
```



Figura3.13: legenda com LAYERFONTBOLD=TRUE

LAYERFONTSIZE

Esse parâmetro especifica o tamanho da fonte para renderizar o título da camada em ponto

URL example:

```
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
```

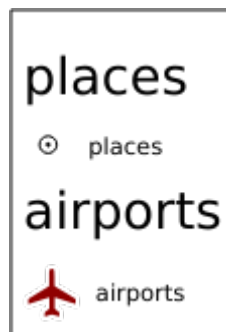


Figura3.14: Legenda com LAYERFONTSIZE=20

LAYERFONTITALIC

Esse parâmetro especifica se o título da camada é renderizado em itálico. Valores disponíveis são (não case sensitives):

- TRUE
- FALSE

URL example:

```
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
```

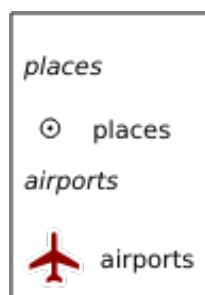


Figura3.15: Legenda com LAYERFONTITALIC=TRUE

LAYERFONTCOLOR

Esse parâmetro especifica a cor do título da camada. A cor pode ser literal (red, green, ...) ou in notação hexadecimal (0xFF0000, 0x00FF00, ...).

URL example:

```
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
```



Figura3.16: Legenda com LAYERFONTCOLOR=0x5f9930

ITEMFONTFAMILY

Esse parâmetro especifica a família da fonte a ser usada para renderizar o rótulo do item

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

ITEMFONTBOLD

Esse parâmetro especifica se o rótulo do item é renderizado em negrito. Valores disponíveis são (não case sensitive):

- TRUE
- FALSE

URL example:

```
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
```

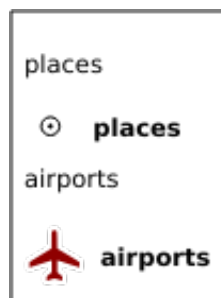


Figura3.17: Legenda com ITEMFONTBOLD=TRUE

ITEMFONTSIZE

Esse parâmetro especifica o tamanho da fonte para renderizar o título da camada em ponto

URL example:

```
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
```

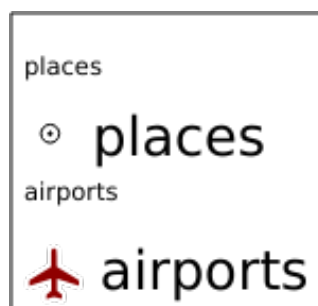


Figura3.18: Legenda com ITEMFONTSIZE=30

ITEMFONTITALIC

Esse parâmetro especifica se o rótulo do item é renderizado em itálico. Valores disponíveis são (não case sensitives):

- TRUE
- FALSE

URL example:

```
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
```




Figura3.19: Legenda com ITEMFONTITALIC=TRUE

ITEMFONTCOLOR

Esse parâmetro especifica a cor do rótulo do item. A cor pode ser literal (red, green, ..) ou em notação hexadecimal (0xFF0000, 0x00FF00, ...).

URL example:

```
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
```



Figura3.20: Legenda com ITEMFONTCOLOR=0x5f9930

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

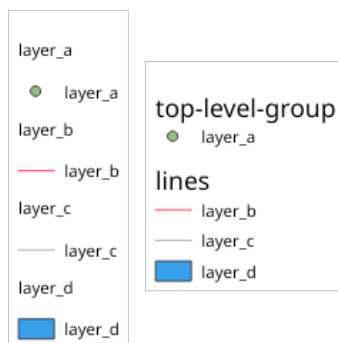


Figura3.21: Legend without (left) and with (right) layer groups labels display

The corresponding JSON output showing groups name would look like:

```
{
  "nodes":
  [ {
    "nodes":
    [ {
      "icon":
      ↪ "iVBORw0KGgoAAAANSUhEUgAAABQAAAAUCAYAAACNiR0NAAACXBIWXMAABYlAAAWJQFJUiTAAAAUklEQVQ4jWNgGAXDHZ
      ↪ 37l28evBKKisr/
      ↪ 0+a7IMiNi93C15DcUpgM4wYQ5nWuZAcMIQNvHv3LuO83C0kG0hysoFZRrJNo2AYAQC87BpkGQj1fwAAAABJRU5Erk
      ↪ Jggg==",
      "title": "layer_a", "type": "layer"
    },
    {
      "nodes":
      [ {
        "icon":
        ↪ "iVBORw0KGgoAAAANSUhEUgAAABMAAAAUCAyAAABvVQZ0AAAAACXBIWXMAABYlAAAWJQFJUiTAAAAHULEQVQ4jWNgGAWjY
        ↪ C",
        "title": "layer_b", "type": "layer"
      },
      {
        "icon":
        ↪ "iVBORw0KGgoAAAANSUhEUgAAABMAAAAUCAyAAABvVQZ0AAAAACXBIWXMAABYlAAAWJQFJUiTAAAAHULEQVQ4jWNgGAWjY
        ↪ C8aaEeMglEwYAAAAIoCzTtn5XoAAAAASUVORK5CYII=",
        "title": "layer_c", "type": "layer"
      }
    ],
    "title": "lines", "type": "group"
  },
  {
    "icon":
    ↪ "iVBORw0KGgoAAAANSUhEUgAAABMAAAATCAyAAABYUDbMAAAACXBIWXMAABYlAAAWJQFJUiTAAAAKklEQVQ4jWNUVlX/
    ↪ z0AlwMLAwMDAnXmQKoYxUcWUUCNGDRs1bNSwYWYYACXDAsvQaTuVAAAAAE1FTkSuQmCC",
    "title": "layer_d", "type": "layer"
  }
  ],
  "title": "top-level-group", "type": "group"
},
"title": ""
}
```

3.2.5 GetStyle(s)

Parâmetros padrão para o pedido **GetStyle** (ou **GetStyles**) de acordo com as especificações do OGC WMS 1.1.1:

Parameter	Required	Description
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Nome do pedido (GetStyle ou GetStyles)
<i>LAYERS</i>	Yes	Camadas para consulta

O pedido **GetStyle** também suporta os seguintes parâmetros de fornecimento:

Parameter	Required	Description
<i>MAP</i>	Yes	QGIS project file

URL example:

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

3.2.6 DescribeLayer

Parâmetros padrão para o pedido **DescribeLayer** de acordo com as especificações OGC WMS 1.1.1 e 1.3.0:

Parameter	Required	Description
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Nome do pedido (DescribeLayer)
<i>LAYERS</i>	Yes	Camadas para descrever
<i>SLD_VERSION</i>	Yes	Versão SLD

O pedido **DescribeLayer** também suporta o seguinte parâmetros fornecidos:

Parameter	Required	Description
<i>MAP</i>	Yes	QGIS project file

URL example:

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

O documento XML parece assim:

```
<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>
```

(continua na próxima página)

```

<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

Este parâmetro permite especificar a versão do SLD. Apenas o valor 1.1.0 está disponível.

3.2.7 GetPrint

O servidor do QGIS tem a capacidade de criar uma saída de impressão de layout em formato PDF ou pixel. Janelas de layout de impressão no projeto publicado são usados como templates. No pedido **GetPrint**, o cliente tem a possibilidade de especificar os parâmetros dos rótulos e mapas contidos no layout.

O pedido **GetPrint** suporta *redlining*, *external WMS layers* assim como os seguintes parâmetros:

Parameter	Re-quired	Description
<i>MAP</i>	Yes	QGIS project file
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Nome do pedido (GetPrint)
<i>VERSION</i>	No	Version of the service
<i>LAYERS</i>	No	Camadas para mostrar
<i>TEMPLATE</i>	Yes	Template de layout para usar
<i>SRS / CRS</i>	Yes	Sistema de referência de coordenadas
<i>FORMAT</i>	No	Formato de saída
<i>FORMAT_OPTIONS</i>	No	Opções do formato de arquivo especificado Only for FORMAT=application/pdf
<i>ATLAS_PK</i>	No	Feições Atlas
<i>STYLES</i>	No	Estilo das camadas
<i>TRANSPARENT</i>	No	Plano de fundo transparente
<i>OPACITIES</i>	No	Opacidade para camada ou grupo
<i>SELECTION</i>	No	Recursos de destaque
<i>mapX:EXTENT</i>	No	Extensão do mapa 'X'
<i>mapX:LAYERS</i>	No	Camadas do mapa 'X'
<i>mapX:STYLES</i>	No	Estilo das camadas do mapa 'X'
<i>mapX:SCALE</i>	No	Camadas de escala do mapa 'X'
<i>mapX:ROTATION</i>	No	Rotação do mapa 'X'
<i>mapX:GRID_INTERVAL</i>	No	Intervalo da grade no eixo y do mapa 'X'
<i>mapX:GRID_INTERVAL</i>	No	Intervalo da grade no eixo y do mapa 'X'

URL example:

```
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
```

Note o layout do template talvez contenha mais de um mapa. Dessa maneira, se você quer configurar um mapa específico, você tem que usar parâmetros mapX: onde X é um número positivo que você consegue recuperar graças ao pedido **GetProjectSettings**

Por exemplo:

```
<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

Esse parâmetro pode ser usado para especificar o nome de modelo compositor para imprimir

FORMAT

Este parâmetro especifica o formato da imagem do mapa. Os valores disponíveis são:

- png (valor padrão)
- image/png
- jpg
- jpeg
- image/jpeg
- svg
- image/svg
- image/svg+xml
- pdf
- application/pdf

Se o parâmetro FORMATO for diferente de um desses valores, uma exceção será retornada.

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 example:

```
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

Este parâmetro permite ativar a renderização do Atlas indicando quais feições queremos imprimir. Para recuperar um atlas com todas as feições, pode-se utilizar o símbolo * (de acordo com o número máximo de feições permitidas na configuração do projeto)..

Quando `FORMAT` é `pdf`, um único documento PDF combinando as páginas de recursos é retornado. Para todos os outros formatos, uma única página é retornada.

mapX:EXTENT

Este parâmetro especifica a extensão de um item de mapa de layout como xmin,ymin,xmax,ymax.

mapX:ROTATION

Este parâmetro especifica a rotação do mapa em graus.

mapX:GRID_INTERVAL_X

Este parâmetro especifica a densidade da linha de grade na direção X.

mapX:GRID_INTERVAL_Y

Este parâmetro especifica a densidade da linha de grade na direção Y.

mapX:SCALE

Este parâmetro especifica a escala do mapa para um item de mapa de compositor. Isso é útil para garantir a visibilidade baseada em escala de camadas e rótulos, mesmo que o cliente e o servidor possam ter algoritmos diferentes para calcular o denominador de escala.

mapX:LAYERS

Esse parâmetro especifica as camadas para um item de compositor de mapa. Veja [GetMap Layers](#) para mais informações desse parâmetro.

mapX:ESTILOS

Esse parâmetro especifica os estilos das camadas definidas em um item compositor específico de mapa. Veja [GetMap Styles](#) para mais informações desse parâmetro.

3.2.8 GetProjectSettings

Esse tipo de pedido funciona similarmente ao [GetCapabilities](#), mas é mais específico para o servidor do QGIS e permite que um cliente leia informação adicional aos quais não são disponíveis na saída [GetCapabilities](#):

- visibilidade inicial das camadas
- informações sobre atributos vetoriais e seus tipos de edição
- informações sobre ordem de camada e ordem de desenho
- Lista de camadas publicadas em WFS.
- Mostra se um grupo na camada árvore é mutuamente exclusivo

O pedido ****GetProjectSettings**** suporta os seguintes parâmetros:

Parameter	Required	Description
<i>MAP</i>	Yes	QGIS project file
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Nome do pedido (GetProjectSettings)

3.2.9 GetSchemaExtension

A solicitação **GetSchemaExtension** permite recuperar recursos estendidos opcionais e operações do serviço WMS, como implementado pelo QGIS Server.

Parameter	Required	Description
<i>MAP</i>	Yes	QGIS project file
<i>SERVICE</i>	Yes	Nome do serviço (WMS)
<i>REQUEST</i>	Yes	Nome do pedido (GetSchemaExtension)

URL example:

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

O documento XML parece assim:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:wms="http://www.opengis.net/
↪wms" xmlns:ggs="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 Camadas WMS externas

O QGIS Server permite incluir camadas de servidores WMS externos em solicitações WMS :ref:`GetMap` e WMS :ref:`GetPrint`. Isso é especialmente útil se um cliente da web usar uma camada de plano de fundo externa no mapa da web. Por motivos de desempenho, essas camadas devem ser solicitadas diretamente pelo cliente da Web (não em cascata via servidor QGIS). No entanto, para impressão, essas camadas devem ser colocadas em cascata via servidor QGIS para aparecer no mapa impresso.

Camadas externas podem ser adicionadas ao parâmetro LAYERS como EXTERNAL_WMS:. Os parâmetros para as camadas WMS externas (por exemplo, url, formato, dpiMode, crs, camadas, estilos) podem ser fornecidos posteriormente como parâmetros de serviço :. Em uma solicitação :ref:`GetMap`, isso pode se parecer com isto:

```
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
```

Similarmente, camadas externas podem ser usadas nos pedidos no *GetPrint*:

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

(continua na próxima página)

(continuação da página anterior)

```
&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 Linha vermelha

Esse atributo está disponível e pode ser usado com pedidos *GetMap* e *GetPrint*.

O recurso de linha vermelha pode ser usado para passar geometrias e rótulos na solicitação que são sobrepostos pelo servidor sobre a imagem padrão retornada (mapa). Isso permite que o usuário coloque ênfase ou talvez adicione alguns comentários (rótulos) a algumas áreas, locais etc. que não estão no mapa padrão.

O pedido *GetMap* está no formato:

```
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
```

O equivalente ao *GetPrint* está no formato (note que o parâmetro mapX: é adicionado para falar qual mapa tem linha vermelha):

```
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
```

Aqui está a imagem gerada pela solicitação acima, na qual um polígono e um rótulo são desenhados no topo do mapa normal:



Figura3.22: Resposta do servidor a uma solicitação GetMap com parâmetros de linha vermelha

Você pode ver que há vários parâmetros nesta solicitação para controlar o recurso de linha vermelha. A lista completa inclui:

- **HIGHLIGHT_GEOM:** Você pode adicionar POINT, MULTILINESTRING, POLYGON etc. Suporta geometrias multipartes. Aqui está um exemplo: `HIGHLIGHT_GEOM=MULTILINESTRING((0 0, 0 1, 1 1))`. As coordenadas devem estar no CRS da solicitação GetMap/GetPrint.
- **HIGHLIGHT_LABELBUFFERCOLOR:** Esse parâmetro controla o rótulo da cor do buffer.
- **HIGHLIGHT_LABELBUFFERSIZE:** Esse parâmetro controla o rótulo do tamanho do buffer.
- **HIGHLIGHT_LABELCOLOR:** Esse parâmetro controla a cor do rótulo.
- **HIGHLIGHT_LABEL_DISTANCE:** controla a distância entre os recursos (e.g. ponto ou linha) e o rótulo em mm
- **HIGHLIGHT_LABELFONT:** Esse parâmetro controla a fonte do rótulo (e.g. Arial)
- **HIGHLIGHT_LABEL_HORIZONTAL_ALIGNMENT:** posiciona horizontalmente o rótulo em um ponto usando o alinhamento especificado (e.g. 'left', 'center', 'right')
- **HIGHLIGHT_LABEL_ROTATION:** controla a rotação do rótulo em graus.
- **HIGHLIGHT_LABELSIZE:** Esse parâmetro controla o tamanho do rótulo.
- **HIGHLIGHT_LABELSTRING:** Você pode passar o seu rotulamento de texto para esse parâmetro
- **HIGHLIGHT_LABEL_VERTICAL_ALIGNMENT:** posiciona o rótulo verticalmente em um ponto usando o alinhamento específico (e.g. 'top', 'half', 'bottom')

- **HIGHLIGHT_SYMBOL**: Isso controla como a geometria é tirada da linha e você pode mudar a largura do contorno, cor e opacidade.

3.3 Web Feature Service (WFS)

The **1.0.0** and **1.1.0** WFS standards implemented in QGIS Server provide a HTTP interface to query geographic features from a QGIS project. A typical WFS request defines the QGIS project to use and the layer to query.

Documento de especificações de acordo com o número da versão do serviço:

- [WFS 1.0.0](#)
- [WFS 1.1.0](#)

Requisitos padrões fornecidos pelo Servidor QGIS:

Solicitação	Descrição
<i>GetCapabilities</i>	Retorna metadados XML com informações sobre o servidor
<i>GetFeature</i>	Retorna uma seleção de feições
<i>DescribeFeatureType</i>	Retorna uma descrição dos tipos de características e propriedades
<i>Transaction</i>	Permite que feições sejam inseridas, atualizadas ou excluídas

3.3.1 GetCapabilities

Standard parameters for the **GetCapabilities** request according to the OGC WFS 1.0.0 and 1.1.0 specifications:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Nome do serviço (WFS)
<i>REQUEST</i>	Sim	Nome da requisição (GetCapabilities)
<i>VERSION</i>	Não	Versão do serviço

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	arquivo de projeto QGIS

VERSION

Este parâmetro permite especificar a versão do serviço a ser utilizada. Os valores disponíveis para o parâmetro **VERSÃO** são:

- 1.0.0
- 1.1.0

Se nenhuma versão for indicada na solicitação, 1.1.0 será usado por padrão.

Exemplo de URL:

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

3.3.2 GetFeature

Parâmetros padrão para a solicitação **GetFeature** de acordo com as especificações OGC WFS 1.0.0 e 1.1.0:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Nome do serviço (WFS)
<i>REQUEST</i>	Sim	Nome da requisição (GetFeature)
<i>VERSION</i>	Não	Versão do serviço
<i>TYPENAME</i>	Não	Nome das camadas
<i>FEATUREID</i>	Não	Filtrar as feições por ids
<i>OUTPUTFORMAT</i>	Não	Formato de Saída
<i>RESULTTYPE</i>	Não	Tipo do resultado
<i>PROPERTYNAME</i>	Não	Nome das propriedades a serem retornadas
<i>MAXFEATURES</i>	Não	Número máximo de feições a serem retornadas
<i>SRSNAME</i>	Não	Sistema de referência de coordenadas
<i>FILTER</i>	Não	Codificação de filtros OGC
<i>BBOX</i>	Não	Extensão do Mapa
<i>SORTBY</i>	Não	Classificar os resultados

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	arquivo de projeto QGIS
<i>STARTINDEX</i>	Não	Paginação
<i>GEOMETRYNAME</i>	Não	Tipo de geometria para retornar
<i>EXP_FILTER</i>	Não	Filtragem de expressão

TYPENAME

Este parâmetro permite especificar nomes de camadas e é obrigatório se `ID DA FEIÇÃO` não estiver definido.

Exemplo de URL:

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

FEATUREID

Este parâmetro permite especificar a identificação de uma característica específica e é formado como `type-name.fid,typename.fid,...`.

Exemplo de URL:

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

Resposta em XML:

```

<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=gml/3.1.1">
  <gml:boundedBy>
    ...
  </gml:boundedBy>
  <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

Este parâmetro pode ser usado para especificar o formato da resposta. Se “VERSÃO” for maior ou igual a “1.1.0”, GML3 é o formato padrão. Caso contrário, é utilizado GML2.

Os valores disponíveis são:

- 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

Exemplo de URL:

```

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

```

Resposta de GeoJSON:

```

{
  "type": "FeatureCollection",
  "bbox": [
    -180,

```

(continua na próxima página)

```

-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

Este parâmetro pode ser usado para especificar o tipo de resultado a ser retornado. Os valores disponíveis são:

- resultados: o comportamento padrão
- “acertos”: retorna apenas uma contagem de recursos

Exemplo de URL:

```

http://localhost/qgisserver?
SERVICE=WFS
&VERSION=1.1.0
&REQUEST=GetFeature
&RESULTTYPE=hits
&...

```

PROPERTYNAME

This parameter may be used to specify a specific property to return. A property needs to be mapped with a TYPE-NAME or a FEATUREID:

Exemplo de URL válido:

```

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

```

Pelo contrário, a próxima URL retornará uma exceção:

```
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

Este parâmetro permite limitar o número de feições devolvidas pelo pedido.

Exemplo de URL:

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

Nota: Este parâmetro pode ser útil para melhorar o desempenho quando as camadas vetoriais subjacentes são pesadas.

SRSNAME

This parameter allows to indicate the response output Spatial Reference System as well as the BBOX CRS and has to be formed like EPSG:XXXX.

Exemplo de URL:

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

FILTER

This parameter allows to filter the response with the **Filter Encoding** language defined by the [OGC Filter Encoding standard](#). For example:

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

In case of multiple typenames, filters have to be enclosed in parentheses:

```
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>)
```

Filter features that intersect with a polygon:

```
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">
      <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

Este parâmetro permite especificar a extensão do mapa com unidades de acordo com o SRC atual. As coordenadas devem ser separadas por vírgula.

The SRSNAME parameter may specify the CRS of the extent. If not specified, the CRS of the layer is used.

Exemplo de URL:

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

The FEATUREID parameter cannot be used with the BBOX. Any attempt will result in an exception:

```
<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

Este parâmetro permite ordenar as características resultantes de acordo com os valores de propriedade e deve ser formado como `nomPropriedade REGRA DE ORGANIZAÇÃO`.

Valores disponíveis para REGRA DE ORGANIZAÇÃO em caso de triagem descendente:

- D
- +D
- DESC
- +DESC

Valores disponíveis para REGRA DE ORGANIZAÇÃO em caso de triagem ascendente:

- A
- +A
- ASC
- +ASC

Exemplo de URL:

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

O resultado correspondente:

```
<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>
  </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

Este parâmetro pode ser usado para especificar o tipo de geometria a ser retornada para as feições. Os valores disponíveis são:

- extent
- centroid
- none

Exemplo de URL:

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

STARTINDEX

Este parâmetro é padrão no WFS 2.0, mas é uma extensão para o WFS 1.0.0.

Na verdade, ele pode ser utilizado para pular algumas características no conjunto de resultados e, em combinação com “MAXFEIÇÕES”, ele fornece a capacidade de paginar através de resultados.

Exemplo de URL:

```
http://localhost/qgisserver?  
SERVICE=WFS  
&VERSION=1.1.0  
&REQUEST=GetFeature  
&STARTINDEX=2  
&...
```

EXP_FILTER

This parameter allows to filter the response with QGIS expressions. The ; character is used to separate filters in case of multiple typenames.

Exemplo de URL:

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

3.3.3 DescribeFeatureType

Standard parameters for the **DescribeFeatureType** request according to the OGC WFS 1.0.0 and 1.1.0 specifications:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Nome do serviço (WFS)
<i>REQUEST</i>	Sim	Name of the request (DescribeFeatureType)
<i>VERSION</i>	Não	Versão do serviço
<i>OUTPUTFORMAT</i>	Não	Formato da resposta
<i>TYPENAME</i>	Não	Nome das camadas

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	arquivo de projeto QGIS

Exemplo de URL:

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

Resposta de saída:

```
<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 Transação

Esta solicitação permite atualizar, apagar ou adicionar uma ou várias características graças a um documento XML. A ação *deletar* pode ser realizada com uma solicitação POST, bem como com o parâmetro *OPERAÇÃO* enquanto que as operações *adicionar* e a *atualizar* podem ser realizadas apenas através de uma solicitação POST.

Parâmetros padrão para a solicitação da **Transação** de acordo com as especificações OGC WFS 1.0.0 e 1.1.0:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Nome do serviço (WFS)
<i>REQUEST</i>	Sim	Nome da solicitação (Transação)
<i>VERSION</i>	Não	Versão do serviço
<i>FILTER</i>	Não	Codificação de filtros OGC
<i>BBOX</i>	Não	Extensão do Mapa
<i>FEATUREID</i>	Não	Filtrar as feições por ids
<i>TYPENAME</i>	Não	Nome das camadas

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	arquivo de projeto QGIS
<i>OPERATION</i>	Não	Especifique a operação
<i>EXP_FILTER</i>	Não	Filtragem de expressão

OPERATION

Este parâmetro permite excluir um recurso sem utilizar um pedido POST com um documento XML dedicado.

Exemplo de URL:

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

Nota: FEATUREID, BBOX and FILTER parameters are mutually exclusive and prioritized in this order.

Adicionar feições

Exemplo de solicitação de POST:

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

com o documento *adiciona.xml*:

```
<?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">
          <gml:coordinates decimal="." cs="," ts=" ">-4.6167,48.3833</
```

(continua na próxima página)

(continuação da página anterior)

```

<gml:coordinates>
  </gml:Point>
</qgs:geometry>
<qgs:name>Locmaria-Plouzané</qgs:name>
</qgs:places>
</wfs:Insert>
</wfs:Transaction>

```

Atualizar feições

Exemplo de solicitação de POST:

```

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

```

com o documento *atualiza.xml*:

```

<?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>

```

Excluir feições

Exemplo de solicitação de POST:

```

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

```

com o documento *deleta.xml*:

```

<?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)

The **1.0.0** and **1.1.1** WCS standards implemented in QGIS Server provide a HTTP interface to access raster data, referred to as *coverage*, coming from a QGIS project.

Especificações:

- WCS 1.0.0
- WCS 1.1.1

Solicitações padrões providas pelo servidor do QGIS:

Solicitação	Descrição
<i>GetCapabilities</i>	Retorna metadados XML com informações sobre o servidor
<i>DescribeCoverage</i>	Retrieves XML document about additional information about coverages
<i>GetCoverage</i>	Retrieves coverage

3.4.1 GetCapabilities

Standard parameters for the **GetCapabilities** request according to the OGC WCS 1.1.1 specifications:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Name of the service (WCS)
<i>REQUEST</i>	Sim	Nome do pedido (GetCapabilities)
<i>VERSION</i>	Não	Versão do serviço

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	Arquivo de projeto QGIS

Exemplo de URL:

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

XML document example when a single raster layer (named T20QPD_20171123T144719_TCI) is published in the QGIS project for the WCS service:

```
<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>
```

(continua na próxima página)

(continuação da página anterior)

```

<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

Este parâmetro permite especificar a versão do serviço a ser utilizada. Atualmente, os valores de versão não são usados internamente e sempre retornam para 1.1.1.

3.4.2 DescribeCoverage

This request allows to retrieve additional information about coverages like the format of the underlying datasource, the number of bands, ... Standard parameters for the **DescribeCoverage** request according to the OGC WCS 1.1.1 specifications:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Name of the service (WCS)
<i>REQUEST</i>	Sim	Name of the request (DescribeCoverage)
<i>VERSION</i>	Não	Versão do serviço
<i>COVERAGE</i>	Não	Specify coverage layers (WCS 1.0.0)
<i>IDENTIFIER</i>	Não	Especificar camadas de cobertura (WCS 1.1.1)

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	Arquivo de projeto QGIS

O documento XML para uma camada raster GeoTIFF de 3 bandas se parece com:

```

<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>
        <axisDescription>

```

(continua na próxima página)

```
<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

Este parâmetro, definido no WCS 1.0.0, permite especificar as camadas a serem consultadas por informações adicionais. Os nomes devem ser separados por vírgula.

In addition, QGIS Server introduced an option to select layers by its short name. The short name of a layer may be configured through *Properties* ► *Metadata* in layer menu. If the short name is defined, then it's used by default instead of the layer's name:

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

Nota: COVERAGE is mandatory if IDENTIFIER is not set.

IDENTIFIER

This parameter replaces the *COVERAGE* parameter in WCS 1.1.1. But QGIS Server does not filter according to the VERSION parameter so IDENTIFIER and COVERAGE have the same effect.

Nota: IDENTIFIER is mandatory if COVERAGE is not set. If both IDENTIFIER and COVERAGE parameters are defined, COVERAGE is always used in priority.

3.4.3 GetCoverage

This request allows to retrieve the coverage according to specific constraints like the extent or the CRS. Standard parameters for the **DescribeCoverage** request according to the OGC WCS 1.1.1 specifications:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Name of the service (WCS)
<i>REQUEST</i>	Sim	Name of the request (GetCoverage)
<i>VERSION</i>	Não	Versão do serviço
<i>COVERAGE</i>	Não	Specify coverage layers (WCS 1.0.0)
<i>IDENTIFIER</i>	Não	Especificar camadas de cobertura (WCS 1.1.1)
<i>WIDTH</i>	Sim	Largura da resposta em pixels
<i>HEIGHT</i>	Sim	Altura da resposta em pixels
<i>BBOX</i>	Sim	Extensão do mapa em unidades SRC
<i>CRS</i>	Sim	Sistema de Referência de Coordenadas da extensão
<i>RESPONSE_CRS</i>	Não	Sistema de referência de coordenadas da resposta

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	Arquivo de projeto QGIS

BBOX

This parameter allows to specify the map extent in the units of the current CRS. Coordinates have to be separated by a comma. The BBOX parameter is formed like `minx,miny,maxx,maxy`.

Exemplo de URL:

```
http://localhost/qgisserver?
SERVICE=WCS
&REQUEST=GetCoverage
&IDENTIFIER=T20QPD_20171123T144719_TCI
&BBOX=647533,1805950,660987,1813940
&CRS=EPSG:32620
```

CRS

This parameter allows to indicate the Spatial Reference System of the BBOX parameter and has to be formed like `EPSG:XXXX`.

RESPONSE_CRS

This parameter allows to indicate the output response Spatial Reference System and has to be formed like `EPSG:XXXX`. The CRS of the corresponding coverage layer is used by default.

WIDTH

Este parâmetro permite especificar a largura em pixels da imagem de saída. A resolução da imagem de resposta depende desse valor.

HEIGHT

Este parâmetro permite especificar a altura em pixels da imagem de saída. A resolução da imagem de resposta depende desse valor.

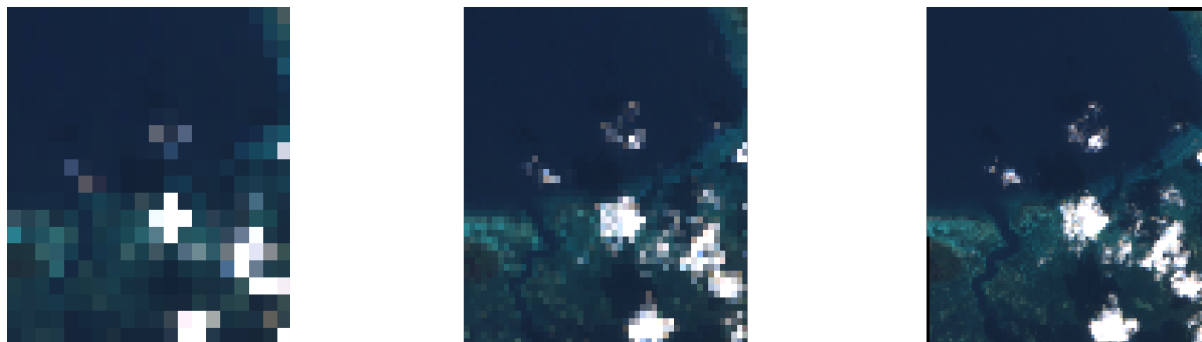


Figura3.23: From left to right: WIDTH=20&HEIGHT=20, WIDTH=50&HEIGHT=50, WIDTH=100&HEIGHT=100

3.5 Web Map Tile Service (WMTS)

The **1.0.0** WMTS standard implemented in QGIS Server provides a HTTP interface to request tiled map images generated from a QGIS project. A typical WMTS request defines the QGIS project to use, some WMS parameters like layers to render, as well as tile parameters.

Documento de especificações do serviço:

- [WMTS 1.0.0](#)

Solicitações padrões providas pelo servidor do QGIS:

Solicitação	Descrição
<i>GetCapabilities</i>	Retorna metadados XML com informações sobre o servidor
<i>GetTile</i>	Retorna um mosaico
<i>GetFeatureInfo</i>	Recupera dados (geometria e valores) para um local de pixel

3.5.1 GetCapabilities

Standard parameters for the **GetCapabilities** request according to the OGC WMTS 1.0.0 specifications:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Name of the service (WMTS)
<i>REQUEST</i>	Sim	Nome do pedido (GetCapabilities)

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	arquivo de projeto QGIS

Exemplo de URL:

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

3.5.2 GetTile

Standard parameters for the **GetTile** request according to the OGC WMTS 1.0.0 specifications:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Name of the service (WMTS)
<i>REQUEST</i>	Sim	Name of the request (GetTile)
<i>LAYER</i>	Sim	Identificador de camada
<i>FORMAT</i>	Sim	Output format of the tile
<i>TILEMATRIXSET</i>	Sim	Nome da pirâmide
<i>TILEMATRIX</i>	Sim	Meshing
<i>TILEROW</i>	Sim	Coordenada de linha na malha
<i>TILECOL</i>	Sim	Coordenada da coluna na malha

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	arquivo de projeto QGIS

Exemplo de URL:

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

FORMATO

Este parâmetro pode ser usado para especificar o formato da imagem do bloco. Os valores disponíveis são:

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

Se o parâmetro `FORMATO` for diferente de um desses valores, então o formato padrão PNG será usado.

TILEMATRIXSET

This parameter defines the CRS to use when computing the underlying pyramid. Format: EPSG:XXXX.

TILEMATRIX

This parameter allows to define the matrix to use for the output tile.

TILEROW

This parameter allows to select the row of the tile to get within the matrix.

TILECOL

This parameter allows to select the column of the tile to get within the matrix.

3.5.3 GetFeatureInfo

Standard parameters for the **GetFeatureInfo** request according to the OGC WMTS 1.0.0 specification:

Parâmetros	Requerido	Descrição
<i>SERVICE</i>	Sim	Name of the service (WMTS)
<i>REQUEST</i>	Sim	Nome do pedido (GetFeatureInfo)
<i>LAYER</i>	Sim	Identificador de camada
<i>INFOFORMAT</i>	Não	Formato de saída
<i>I</i>	Não	Coordenada X de um pixel
<i>J</i>	Não	Coordenada Y de um pixel
<i>TILEMATRIXSET</i>	Sim	Nome da pirâmide
<i>TILEMATRIX</i>	Meshing	
<i>TILEROW</i>	Sim	Coordenada de linha na malha
<i>TILECOL</i>	Sim	Coordenada da coluna na malha

Além dos padrões, o QGIS Server suporta os seguintes parâmetros extras:

Parâmetros	Requerido	Descrição
<i>MAP</i>	Sim	arquivo de projeto QGIS

Exemplo de URL:

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

INFOFORMAT

Este parâmetro permite definir o formato de saída do resultado. Os valores disponíveis são:

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

The default value is `text/plain`.

I

Este parâmetro permite definir a coordenada X do pixel para o qual queremos recuperar a informação subjacente.

J

Este parâmetro permite definir a coordenada Y do pixel para o qual queremos recuperar a informação subjacente.

3.6 Feições OGC API

OGC API Features (OAPIF) is the first implementation of the new generation of OGC protocols. It is described by the [OGC API - Features - Part 1: Core](#) document.

The API can be reached on typical installations via `http://localhost/qgisserver/wfs3`

Aqui está um rápido resumo informal das diferenças mais importantes entre o bem conhecido protocolo WFS e a OAPIF:

- OAPIF é baseado em uma [API REST](#)
- OAPIF deve seguir as [OPENAPI](#) especificações
- OAPIF supports multiple output formats but it does not dictate any (only GeoJSON and HTML are currently available in QGIS OAPIF) and it uses [content negotiation](#) to determine which format is to be served to the client
- JSON e HTML são cidadãos de primeira classe na OAPIF
- OAPIF é autodocumentação (através do endpoint `/api`)
- OAPIF é totalmente navegável (através de links) e buscável

Importante: While the OGC API Features implementation in QGIS can make use of the `MAP` parameter to specify the project file, no extra query parameters are allowed by the OPENAPI specification. For this reason it is strongly recommended that `MAP` is not exposed in the URL and the project file is specified in the environment by other means (i.e. setting `QGIS_PROJECT_FILE` in the environment through a web server rewrite rule).

Nota: The [API](#) endpoint provides comprehensive documentation of all supported parameters and output formats of your service. The following paragraphs will only describe the most important ones.

3.6.1 Representação de recursos

The implementation of OGC API Features in QGIS Server currently supports the following resource representation (output) formats:

- HTML
- JSON

The format that is actually served will depend on content negotiation, but a specific format can be explicitly requested by appending a format specifier to the endpoints.

Supported format specifier extensions are:

- `.json`
- `.html`

Additional format specifier aliases may be defined by specific endpoints:

- `.openapi`: alias for `.json` supported by the **API** endpoint
- `.geojson`: alias for `.json` supported by the **Features** and **Feature** endpoints

3.6.2 Endpoints

The API provides a list of endpoints that the clients can retrieve. The system is designed in such a way that every response provides a set of links to navigate through all the provided resources.

Endpoints provided by the QGIS implementation are:

Nome	Caminho	Descrição
Landing Page	<code>/</code>	General information about the service and provides links to all available endpoints
Conformidade	<code>/conformance</code>	Informações sobre a conformidade do serviço com as normas
API	<code>/api</code>	Full description of the endpoints provided by the service and the returned documents structure
Coleções	<code>/collections</code>	List of all collections (i.e. 'vector layers') provided by the service
Coleção	<code>/collections/{collectionId}</code>	Informações sobre uma coleção (nome, metadados, extensão etc.)
Feições	<code>/collections/{collectionId}/items</code>	Lista das feições fornecidas pela coleção
Feição	<code>/collections/{collectionId}/items/{featureId}</code>	Informações sobre uma feição única

Similar to WFS-T (transactional Web Feature Service), it is possible to add, update and delete features (CRUD). The respective requests are described on `"/api"`.

Landing Page

The main endpoint is the **Landing Page**. From that page it is possible to navigate to all the available service endpoints. The **Landing Page** must provide links to

- the API definition (path `/api` link relations `service-desc` and `service-doc`),
- the Conformance declaration (path `/conformance`, link relation `conformance`), and
- the Collections (path `/collections`, link relation `data`).

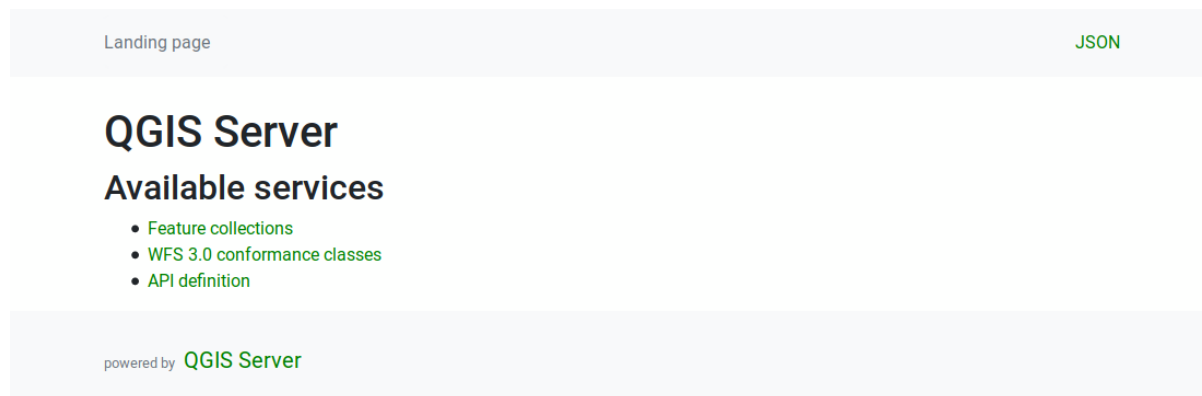


Figura3.24: Página principal do servidor OAPIF

API Definition

The **API Definition** is an OPENAPI-compliant description of the API provided by the service. In its HTML representation it is a browsable page where all the endpoints and their response formats are accurately listed and documented. The path of this endpoint is `/api`.

The API definition provides a comprehensive and authoritative documentation of the service, including all supported parameters and returned formats.

Nota: This endpoint is analogue to WFS's `GetCapabilities`

Lista de coleções

The collections endpoint provides a list of all the collections available in the service. Since the service “serves” a single QGIS project the collections are the vector layers from the current project (if they were published as WFS in the project properties). The path of this endpoint is `/collections/`.

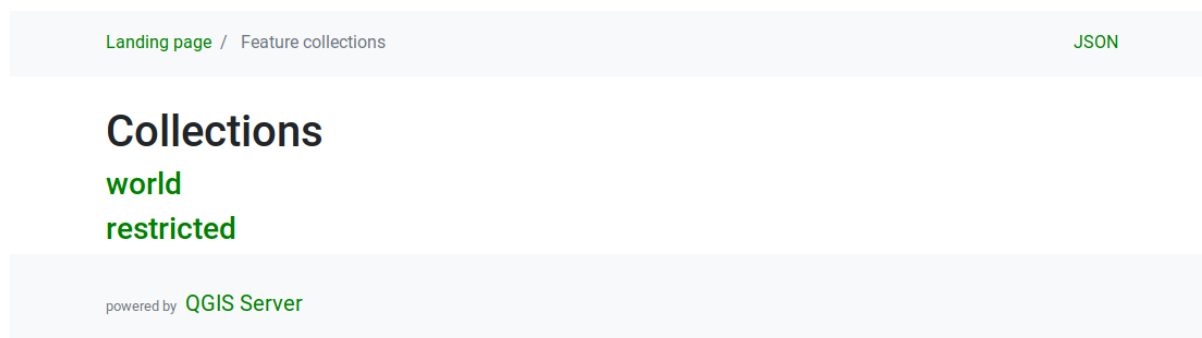


Figura3.25: Server OAPIF collections list page

Collection detail

While the collections endpoint does not provide detailed information about each available collection, that information is available in the `/collections/{collectionId}` endpoints. Typical information includes the extent, a description, CRSs and other metadata.

The HTML representation also provides a browsable map with the available features.

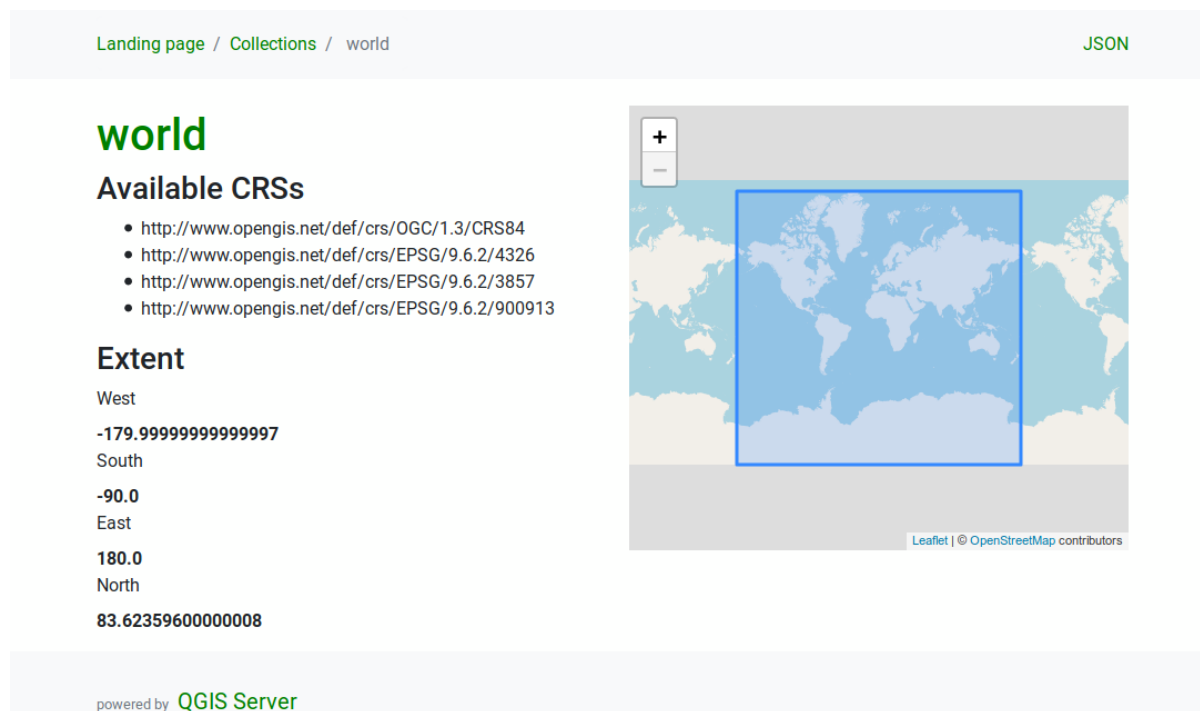


Figura3.26: Página de detalhes da coleta do Servidor OAPIF

Lista de feições

This endpoint provides a list of all features in a collection knowing the collection ID. The path of this endpoint is `/collections/{collectionId}/items`.

The HTML representation also provides a browsable map with the available features.

Nota: This endpoint is analogue to `GetFeature` in WFS 1 and WFS 2.

Landing page / Collections / world / Features in layer world GEOJSON

[Previous](#) [Next](#)

Features in layer world

world 20

AREA	845942
FIPS	BR
ISO2	BR
ISO3	BRA
LAT	-10.772
LON	-53.089
NAME	Brazil
POP2005	186830759
REGION	19
SUBREGION	5
UN	76

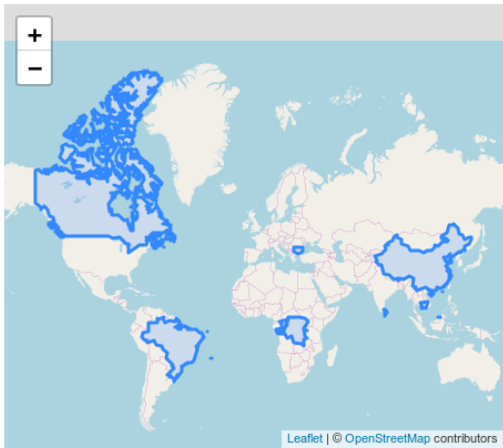


Figura3.27: Página de lista de recursos do servidor OAPIF

Detalhe da feição

This endpoint provides all the available information about a single feature, including the feature attributes and its geometry. The path of this endpoint is `/collections/{collectionId}/items/{itemId}`.

The HTML representation also provides a browsable map with the feature geometry.

Landing page / Collections / world / Items of world / world - feature 20 GEOJSON

world - feature 20

AREA	845942
FIPS	BR
ISO2	BR
ISO3	BRA
LAT	-10.772
LON	-53.089
NAME	Brazil
POP2005	186830759
REGION	19
SUBREGION	5
UN	76



powered by **QGIS Server**

Figura3.28: Página de detalhes do servidor OAPIF

3.6.3 Paginação

Pagination of a long list of features is implemented in the OGC API through `next` and `prev` links, QGIS server constructs these links by appending `limit` and `offset` as query string parameters.

Exemplo de URL:

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

Nota: The maximum acceptable value for `limit` can be configured with the `QGIS_SERVER_API_WFS3_MAX_LIMIT` server configuration setting (see: *Variáveis ambientais*).

3.6.4 Filtragem de recursos

The features available in a collection can be filtered/searched by specifying one or more filters.

Filtro de data e hora

Collections with date and/or datetime attributes can be filtered by specifying a `datetime` argument in the query string. By default the first date/datetime field is used for filtering. This behavior can be configured by setting a “Date” or “Time” dimension in the *QGIS Server* ► *Dimension* section of the layer properties dialog.

The date and time filtering syntax is fully described in the *API Definition* and also supports ranges (begin and end values are included) in addition to single values.

Exemplo de URL:

Returns only the features with date dimension matching 2019-01-01

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

Returns only the features with datetime dimension matching 2019-01-01T01:01:01

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

Returns only the features with datetime dimension in the range 2019-01-01T01:01:01 - 2019-01-01T12:00:00

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

Bounding box filter

A bounding box spatial filter can be specified with the `bbox` parameter:

A ordem dos elementos separados por vírgula é:

- Lower left corner, WGS 84 longitude
- Lower left corner, WGS 84 latitude
- Upper right corner, WGS 84 longitude
- Upper right corner, WGS 84 latitude

Nota: The OGC specifications also allow a 6 item bbox specifier where the third and sixth items are the Z components, this is not yet supported by QGIS server.

Exemplo de URL:

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

If the CRS of the bounding box is not [WGS 84](#), a different CRS can be specified by using the optional parameter `bbox-crs`. The CRS format identifier must be in the [OGC URI](#) format:

Exemplo de URL:

```
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
```

Filtros de atributos

Attribute filters can be combined with the bounding box filter and they are in the general form: `<attribute name>=<attribute value>`. Multiple filters can be combined using the AND operator.

Exemplo de URL:

filters all features where attribute name equals “my value”

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

Partial matches are also supported by using a * (“star”) operator:

Exemplo de URL:

filters all features where attribute name ends with “value”

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

3.6.5 Feature sorting

It is possible to order the result set by field value using the `orderby` query parameter.

The results are sorted in ascending order by default. To sort the results in descending order, a boolean flag (`sortdesc`) can be set:

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

3.6.6 Seleção de atributo

The feature attributes returned by a [Lista de feições](#) call can be limited by adding a comma separated list of attribute names in the optional `properties` query string argument.

Exemplo de URL:

retorna apenas o atributo nome

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

3.6.7 Personalizar as páginas HTML

The HTML representation uses a set of HTML templates to generate the response. The template is parsed by a template engine called *inja*. The templates can be customized by overriding them (see: *Substituição de modelos*). The template has access to the same data that are available to the JSON representation and a few additional functions are available to the template:

Funções de modelo personalizado

- `path_append(path)`: appends a directory path to the current url
- `path_chomp(n)`: removes the specified number “n” of directory components from the current url path
- `json_dump()`: prints the JSON data passed to the template
- `static(path)`: returns the full URL to the specified static path. For example: “static(“/style/black.css”)” with a root path “http://localhost/qgisserver/wfs3” will return “http://localhost/qgisserver/wfs3/static/style/black.css”.
- `links_filter(links, key, value)`: Returns filtered links from a link list
- `content_type_name(content_type)`: Returns a short name from a content type, for example “text/html” will return “HTML”
- `nl2br(texto)`: Retorna o texto de entrada com todas as novas linhas substituídas por “
” tags
- `starts_with(string, prefix)`: returns true if a string begins with the provided string prefix, false otherwise

Substituição de modelos

Templates and static assets are stored in subdirectories of the QGIS server default API resource directory (`/usr/share/qgis/resources/server/api/` on a Linux system), the base directory can be customized by changing the environment variable `QGIS_SERVER_API_RESOURCES_DIRECTORY`.

A typical Linux installation will have the following directory tree:

```
/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
```

To override the templates you can copy the whole tree to another location and point `QGIS_SERVER_API_RESOURCES_DIRECTORY` to the new location.

Catálogo

The QGIS Server Catalog is a simple catalog that shows the list of QGIS projects served by the QGIS Server. It provides a user-friendly fully browsable website with basic mapping capabilities to quickly browse the datasets exposed through those QGIS projects.

O catálogo do QGIS Server usa as variáveis `QGIS_SERVER_LANDING_PAGE_PROJECTS_DIRECTORIES` e `QGIS_SERVER_LANDING_PAGE_PROJECTS_PG_CONNECTIONS` (veja *Variáveis ambientais*)

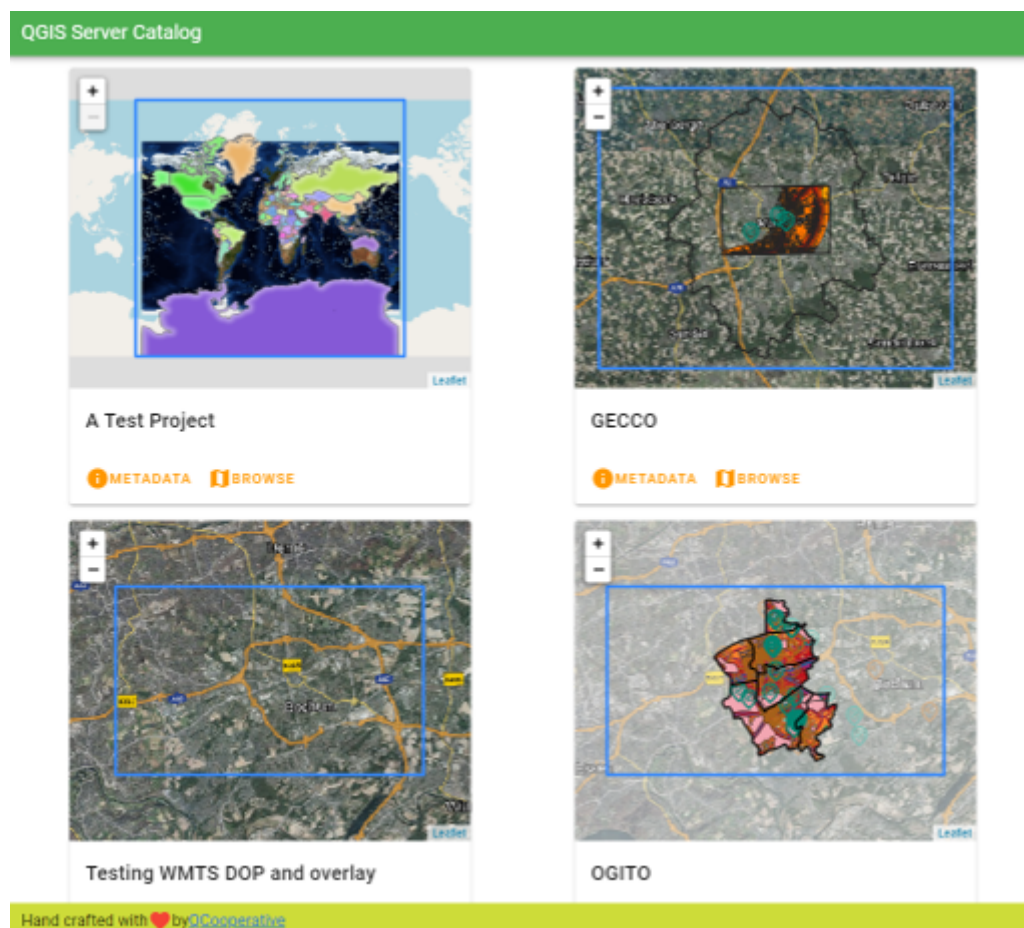


Figura4.1: Server Catalog project list page

Você pode consultar os metadados associados a um projeto e os serviços que ele fornece. Links para esses serviços também são fornecidos.

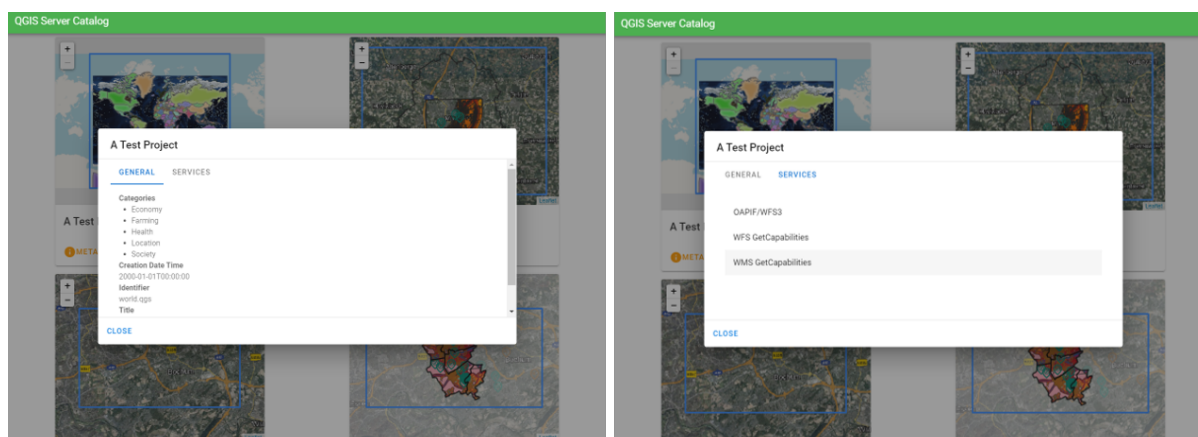


Figura4.2: Server Catalog, metadata associated to a project and services (links to) that it provides.

By browsing a project, it is listed the dataset that it serves.

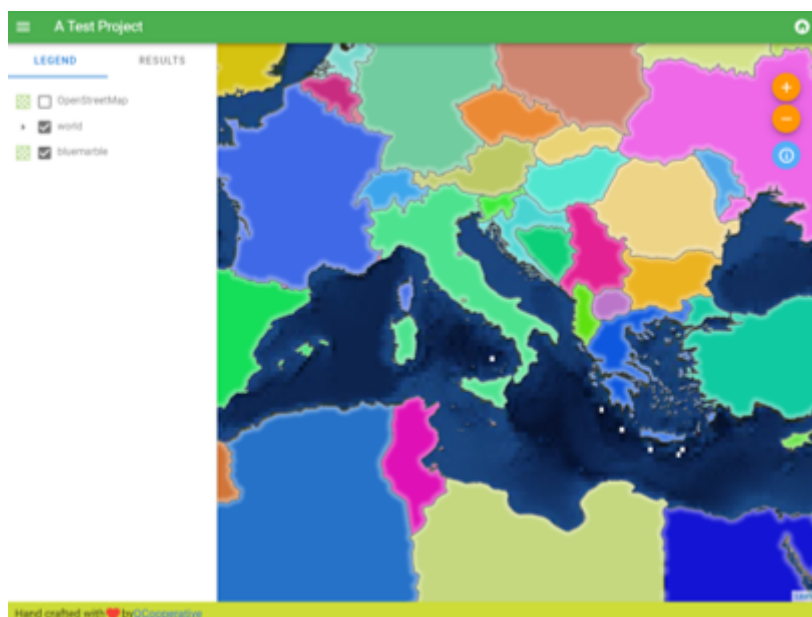


Figura4.3: Browsing a dataset served by a project in the Server Catalog

Use o clique direito sobre uma camada para exibir a tabela de atributos associada a ela.

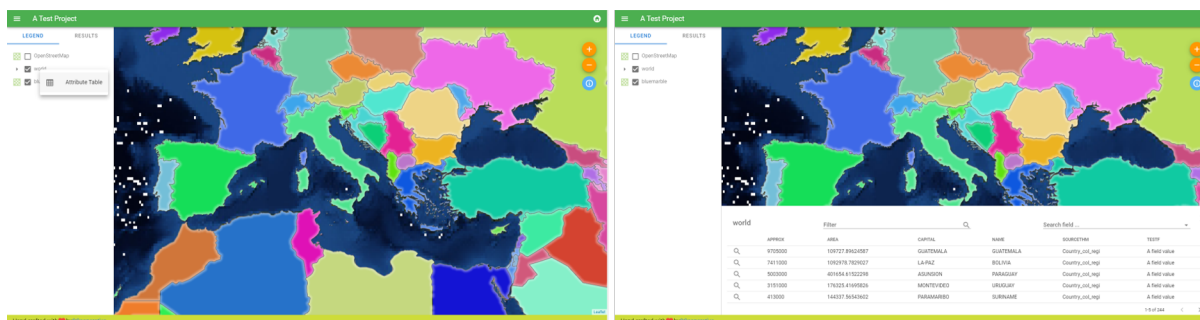


Figura4.4: Tabela de atributos associados a uma camada

É possível consultar as informações dos elementos do mapa, como mostrado na imagem abaixo:

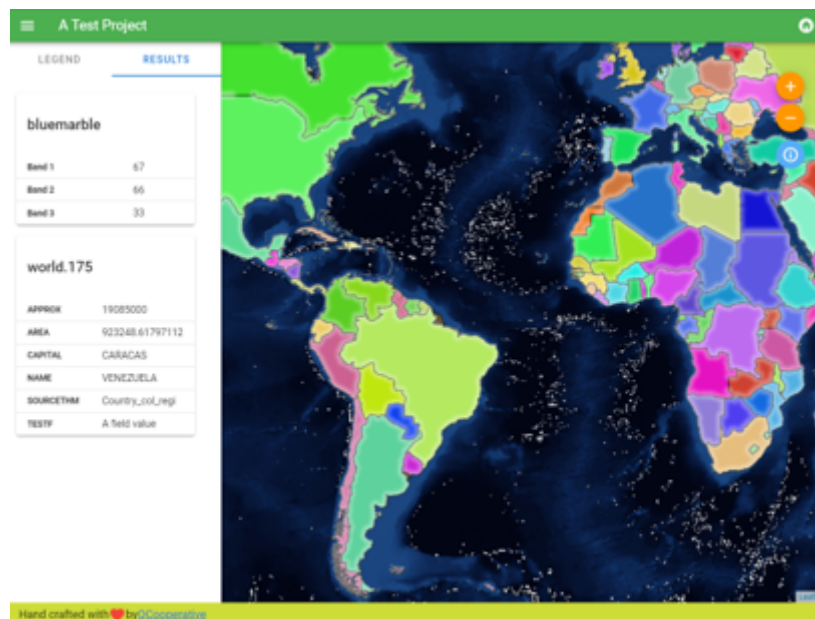


Figura4.5: Consultando informações de um elemento do mapa

5.1 Lista de plugins

Plugins também podem ser instalados no Servidor QGIS.

Alguns plugins projetados para servidor podem ser encontrados no repositório oficial [QGIS](#).

Instale apenas plugins que você precisa para seu próprio propósito. No servidor QGIS, os plugins são como ganchos no servidor QGIS, eles podem alterar entradas ou saídas do servidor QGIS. Eles podem produzir resultados inesperados se você não souber como o plugin funciona. Consulte a respectiva documentação ou o aplicativo que precisa dos plug-ins do servidor QGIS para saber qual plug-in pode ser útil para você.

5.2 Localização dos plugins

Por padrão, nos sistemas baseados em Debian, QGIS Server irá verificar por plugins localizados em `/usr/lib/qgis/plugins`. O valor padrão exibido quando QGIS Server está iniciando, nos logs. É possível configurar um caminho customizado através da alteração da variável de ambiente `QGIS_PLUGINPATH` nas configurações do servidor web.

5.3 Instalação

5.3.1 Manualmente com um ZIP

Como exemplo, para instalar o plugin **HelloWorld** para testar o servidor, usando uma pasta específica, primeiro é necessário criar um diretório para manter os plugins do servidor. Isto será especificado na configuração do host virtual e passado para o servidor através de uma variável de ambiente:

```
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
```

Aviso: De acordo com sua [descrição](#), o plugin HelloServer é projetado para fins de desenvolvimento e demonstração. Não mantenha este plugin em produção se você não precisar dele.

5.3.2 Com uma ferramenta de linha de comando

Se você precisar instalar e atualizar regularmente plugins que são armazenados no repositório de plugins QGIS, você pode utilizar o [QGIS-Plugin-Manager](#). É uma ferramenta para ajudá-lo a gerenciar os plugins a partir da linha de comando.

A instalação está usando pip. Instalar em um ambiente virtual é uma boa prática, mas não é obrigatório:

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

Para atualizar a ferramenta:

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

Então, você pode utilizar o executável `qgis-plugin-manager` a partir da linha de comando:

```
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                |         |           | 3.
| qgis_server_render_geojson | GeoJson Renderer | v0.4    |           | 3.
| 4           | Matthias Kuhn (OPENGIS.ch) |         |           | 3.
| DataPlotly   | Data Plotly         | 3.7.1   |           | 3.
| 4           | 3.98               | Matteo Ghetta (Faunalia) | Upgrade to 3.8.1 | 3.
|-----|-----|-----|-----|-----|
| 4           | 3.98               | Matteo Ghetta (Faunalia) | Upgrade to 3.8.1 | 3.
```

Sugerimos que você leia a documentação completa no [arquivo leia-me](#) para saber como instalar ou atualizar plugins com esta ferramenta.

5.4 Configuração do Servidor HTTP

5.4.1 Apache

Para poder usar plugins de servidor, FastCGI precisa saber onde procurar. Portanto, temos que modificar o arquivo de configuração do Apache para indicar a variável de ambiente `QGIS_PLUGINPATH` para FastCGI:

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

Além disso, uma autorização HTTP básica é necessária para jogar com o plugin HelloWorld introduzido anteriormente. Portanto, temos que atualizar o arquivo de configuração do Apache uma última vez:

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

Em seguida, reinicie o Apache:

```
systemctl restart apache2
```

5.5 Como usar um plug-in

Teste o servidor com o plugin HelloWorld:

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

Você pode dar uma olhada no padrão GetCapabilities do QGIS Server em:

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

Configuração avançada

6.1 Registrando

To log requests sent to the server, you have to set the following environment variable:

- *QGIS_SERVER_LOG_STDERR*

With the following variables the logging can be further customized:

- *QGIS_SERVER_LOG_LEVEL*
- *QGIS_SERVER_LOG_PROFILE*

6.2 Variáveis ambientais

Você pode configurar alguns aspectos do QGIS Server configurando **environment variables**.

According to the HTTP server and how you run QGIS Server, there are several ways to define these variables. This is fully described in *Servidor HTTP Apache*.

Nome	Descrição	Padrão	Serviços
QGIS_OPTIONS_PATH	Specifies the path to the directory with settings. It works the same way as QGIS application <code>--optionspath</code> option. It is looking for settings file in <code><QGIS_OPTIONS_PATH>/QGIS/QGIS3.ini</code> .	"	Tudo
QGIS_PLUGINPATH	Useful if you are using Python plugins for the server, this sets the folder that is searched for Python plugins.	"	Tudo

continua na próxima página

Tabela 6.1 – continuação da página anterior

Nome	Descrição	Padrão	Serviços
QGIS_PROJECT_FILE	The .qgs or .qgz project file, normally passed as a parameter in the query string (with <i>MAP</i>), you can also set it as an environment variable (for example by using <code>mod_rewrite</code> Apache module). Note that you may also indicate a project stored in PostgreSQL, e.g. <code>postgresql://localhost:5432?sslmode=disable&dbname=mydb&project=myproject</code> or inside a geopackage file, e.g. <code>geopackage:/path/to/geopackage/file.gpkg?projectName=myProjectName</code> .	“	Tudo
QGIS_SERVER_ALLOWED_EXTRA_SQL_TOKENS	Comma separated list of strings that represent the allowed extra SQL tokens accepted as components of a feature filter.	“	WMS
QGIS_SERVER_API_RESOURCES	Diretório base para todos os recursos estáticos OGC API (tais como OAPIF/WFS3) (modelos HTML, CSS, JS, ...)	depende do pacote	OAPIF/WFS3
QGIS_SERVER_APPLICATION_NAME	Name of the application to be used, for instance when connecting to a database to identify the QGIS server instance connected	QGIS3 server	Tudo
QGIS_SERVER_API_WFS3_MAX_FEATURES	Maximum value for limit in a OAPIF/WFS3 features request.	10000	OAPIF/WFS3
QGIS_SERVER_CACHE_DIRECTORY	Specifies the network cache directory on the filesystem.	cache in profile directory	Tudo
QGIS_SERVER_CACHE_SIZE	Sets the network cache size in MB.	50 MB	Tudo
QGIS_SERVER_CACHE_MAX_ENTRIES	The maximum number of project capabilities to cache.	40	Tudo
QGIS_SERVER_DISABLE_GET_LAYOUTS	Esta é uma opção no nível do projeto para melhorar o tempo de leitura do projeto desativando o carregamento de layouts. Activating this option disables the QGIS WMS GetPrint request. Set this QGIS project flag to not load layouts.	falso	WMS
QGIS_SERVER_FORCE_READ_ONLY	Force QGIS Server to open all layers in a read only mode	falso	Tudo
QGIS_SERVER_IGNORE_BAD_LAYERS	“Bad” layers are layers that cannot be loaded. The default behavior of QGIS Server is to consider the project as not available if it contains a bad layer. The default behavior can be overridden by setting this variable to 1 or true. In this case, “bad” layers will just be ignored, and the project will be considered valid and available.	falso	Tudo
QGIS_SERVER_LANDING_PAGE_BASE	Prefixo do componente do caminho da URL base da página inicial	“”	Tudo
QGIS_SERVER_LANDING_PAGE_PROJECTS	Directories used by the landing page service to find .qgs and .qgz projects	“”	Tudo
QGIS_SERVER_LANDING_PAGE_CONNECTIONS	PostgreSQL connection strings used by the landing page service to find projects	“”	Tudo

continua na próxima página

Tabela 6.1 – continuação da página anterior

Nome	Descrição	Padrão	Serviços
QGIS_SERVER_LOG_FILE	<p>Especificar o caminho e o nome do arquivo. Certifique-se de que o servidor tenha as permissões adequadas para gravar no arquivo. O arquivo deve ser criado automaticamente, basta enviar algumas requisições para o servidor. Se não estiver lá, verifique as permissões.</p> <div> <p>Aviso:</p> <p>QGIS_SERVER_LOG_FILE is deprecated since QGIS 3.4, use QGIS_SERVER_LOG_STDERR instead. File logging support will be removed in QGIS 4.0.</p> </div>	“	Tudo
QGIS_SERVER_LOG_LEVEL	<p>Especifique o nível de log desejado. Os valores disponíveis são:</p> <ul style="list-style-type: none"> • 0 ou INFO (registrar todas as solicitações) • 1 ou ALERTA • 2 or CRITICAL (log just critical errors, suitable for production purposes) 	0	Tudo
QGIS_SERVER_LOG_PROFILE	Add detailed profile information to the logs, only effective when QGIS_SERVER_LOG_LEVEL=0	falso	Tudo
QGIS_SERVER_LOG_STDERR	<p>Activate logging to stderr. This variable has no effect when QGIS_SERVER_LOG_FILE is set.</p> <ul style="list-style-type: none"> • 0 ou falso (não diferencia maiúsculas de minúsculas) • 0 ou verdadeiro (não diferencia maiúsculas de minúsculas) 	falso	Tudo

continua na próxima página

Tabela 6.1 – continuação da página anterior

Nome	Descrição	Padrão	Serviços
QGIS_SERVER_MAX_THREADS	Number of threads to use when parallel rendering is activated. If value is <code>-1</code> it uses the number of processor cores.	<code>-1</code>	Tudo
QGIS_SERVER_OVERRIDE_LOCALE	Sets LOCALE to be used by QGIS server. The default value is empty (no override). Example: <code>de_CH.utf8</code>	<code>"</code>	Tudo
QGIS_SERVER_PARALLEL_RENDERING	Activates parallel rendering for WMS GetMap requests. It's disabled (<code>false</code>) by default. Available values are: <ul style="list-style-type: none"> <code>0</code> ou <code>falso</code> (não diferencia maiúsculas de minúsculas) <code>0</code> ou <code>verdadeiro</code> (não diferencia maiúsculas de minúsculas) 	<code>falso</code>	WMS
QGIS_SERVER_PROJECT_CACHE_STRATEGY	Controls the periodic strategy interval for cache invalidation, in milliseconds. Defaults to <code>0</code> which selects the legacy File system watcher.		Tudo
QGIS_SERVER_PROJECT_CACHE_STRATEGY	Defines method for invalidating the project cache. Available strategies are: <ul style="list-style-type: none"> <code>filesystem</code>: uses the file system watcher strategy <code>periodic</code>: uses the last modified value of a project for checking changes on project configuration. Convenient on atypical file systems, such as NFS, or when the project file is stored in a database system like PostgreSQL. <code>off</code>: disables internal cache invalidation completely 	<code>filesystem</code>	Tudo

continua na próxima página

Tabela 6.1 – continuação da página anterior

Nome	Descrição	Padrão	Serviços
QGIS_SERVER_SERVICE_URL	<p>This is an option to set the service URL if it is not present in the project.</p> <p>The service URL is defined from (in order of precedence):</p> <ul style="list-style-type: none"> • Value defined in the project per service • The <code>QGIS_SERVER_<service>_SERVICE_URL</code> environment variable • The <code>QGIS_SERVER_SERVICE_URL</code> environment variable • The <code>X-Qgis-<service>-Service-Header</code> header • The <code>X-Qgis-Service-Url</code> header • Build from the <code>Forwarded-Header</code> • Build from the <code>X-Forwarded-Host</code> and <code>X-Forwarded-Proto</code> headers • Build from the <code>Host</code> header and the server protocol • Build from the server name and the server protocol. <p>In the last four cases, the resulting Service URL is based on the <code>MAP</code> parameter provided in the query string and on the incoming path request.</p>	“	Tudo
QGIS_SERVER_SHOW_GROUP_SEPARATOR	<p>Defines whether a group separator (e.g. thousand separator) should be used for numeric values (e.g. in <code>GetFeatureInfo</code> responses). The default value is 0.</p> <ul style="list-style-type: none"> • 0 ou falso (não diferencia maiúsculas de minúsculas) • 1 ou verdadeiro (não diferencia maiúsculas de minúsculas) 	falso	WMS
QGIS_SERVER_TRUST_LAYER_METADATA	<p>This is an option at the project level to improve project read time by using the vector layer extents defined in the project metadata and disabling the check for PostgreSQL/PostGIS layer primary key uniqueness.</p> <p>Trusting layer metadata can be forced by setting this variable to 1 or <code>true</code>. The vector layer's extent will then be the one defined in the project, and the PostgreSQL/PostGIS layer's primary key defined in the data source is considered as unique without a check.</p> <p>Do not use it if layers' extent is not fixed during the project's use.</p>	falso	Tudo

continua na próxima página

Tabela 6.1 – continuação da página anterior

Nome	Descrição	Padrão	Serviços
QGIS_SERVER_WCS_SERVIC	This is an option to set the service URL if it is not present in the project. See QGIS_SERVER_SERVICE_URL for more information.	“	WCS
QGIS_SERVER_WFS_SERVIC	This is an option to set the service URL if it is not present in the project. See QGIS_SERVER_SERVICE_URL for more information.	“	WFS
QGIS_SERVER_WMS_MAX_I / QGIS_SERVER_WMS_MAX_I	Maximum height/width for a WMS request. The most conservative between this and the project one is used. If the value is -1, it means that there is no maximum set.	-1	WMS
QGIS_SERVER_WMS_SERVIC	This is an option to set the service URL if it is not present in the project. See QGIS_SERVER_SERVICE_URL for more information.	“	WMS
QGIS_SERVER_WMTS_SERV	This is an option to set the service URL if it is not present in the project. See QGIS_SERVER_SERVICE_URL for more information.	“	WMTS
QUERY_STRING	<p>The query string, normally passed by the web server. This variable can be useful while testing QGIS server binary from the command line.</p> <p>For example for testing a GetCapabilities request on the command line to a project that also requires a PostgreSQL connection defined in a pg_service.conf file:</p> <pre>PGSERVICEFILE=/etc/pg_service.conf \ QUERY_STRING="MAP=/home/projects/world.qgs&SERVICE=WMS&REQUEST=GetCapabilities" \ /usr/lib/cgi-bin/qgis_mapserv.fcgi</pre> <p>The result should be either the content of the GetCapabilities response or, if something is wrong, an error message.</p>	“	Tudo

6.3 Settings summary

When QGIS Server is starting, you have a summary of all configurable parameters thanks to environment variables. Moreover, the value currently used and the origin is also displayed.

Por exemplo com spawn-fcgi:

```
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 -G www-data -n
```

QGIS Server Settings:

(continua na próxima página)

(continuação da página anterior)

```

- 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

```

In this particular case, we know that **QGIS_SERVER_MAX_THREADS** and **QGIS_SERVER_PARALLEL_RENDERING** values are read from the ini file found in **QGIS_OPTIONS_PATH** directory (which is defined through an environment variable). The corresponding entries in the ini file are **/qgis/max_threads** and **/qgis/parallel_rendering** and their values are **true** and **4** threads.

6.4 Connection to service file

In order to make apache aware of the PostgreSQL service file (see the pg-service-file section) you need to make your *.conf file look like:

```

SetEnv PGSERVICEFILE /home/web/.pg_service.conf

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

```

6.5 Adicione fontes ao seu servidor linux

Tenha em mente que você pode usar projetos QGIS que apontam para fontes que podem não existir por padrão em outras máquinas. Isso significa que se você compartilhar o projeto, ele poderá parecer diferente em outras máquinas (se as fontes não existirem na máquina de destino).

Para garantir que isso não aconteça, você só precisa instalar as fontes ausentes na máquina de destino. Fazer isso em sistemas desktop geralmente é trivial (clique duas vezes nas fontes).

Para linux, se você não tiver um ambiente de desktop instalado (ou preferir a linha de comando), precisará:

- On Debian based systems:

```
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
```

- On Fedora based systems:

```
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
```

Servidor de desenvolvimento

A production installation and deployment of QGIS Server usually involves setting up a web server component (e.g. Apache or Nginx) that can forward the HTTP requests coming from the clients to the QGIS Server FastCGI binary application.

If you want to quickly test QGIS Server on your local machine without configuring and installing a full web server stack you can use the QGIS Development Standalone server.

Esta é uma aplicação independente que fornece um servidor web muito simples, pronto para servir seus arquivos de projeto.

Aviso: The Standalone Development Server has not been developed with the purpose of being used in production, it was not checked for security vulnerabilities or for other stress conditions that normally will occur on a publicly exposed server.

Para lançar o servidor:

```
$ qgis_mapserver
```

A porta padrão que o Servidor de Desenvolvimento escuta é 8000. Exemplo de saída:

```
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
```

O servidor tem algumas opções que podem ser passadas como argumentos de linha de comando. Você pode vê-las todas invocando o servidor com `-h`.

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

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
```

Containerized deployment

There are many ways to use containerized application, from the most simple (simple Docker images) to sophisticated (Kubernetes and so on).

Nota: Este tipo de implantação precisa da aplicação `docker` para ser instalado e executado. Verifique este **tutorial** [<https://www.docker.com/101-tutorial>](https://www.docker.com/101-tutorial)`.

Dica: Docker run pre packaged application (aka images) which can be retrieved as sources (Dockerfile and resources) to build or already built from registries (private or public).

Nota: Os downloads de pacotes QGIS Debian-Ubuntu precisam de uma chave de autenticação gpg válida. Favor consultar as [páginas de instalação](#) para atualizar o seguinte arquivo Docker.

8.1 Imagens simples do estivador

As the docker image does not exist in a public registry. you will need to build it. To do so create a directory `qgis-server` and within its directory:

- create a file `Dockerfile` with this content:

```
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 \
```

(continua na próxima página)

(continuação da página anterior)

```
&& 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"]
```

- criar um arquivo cmd.sh com este conteúdo:

```
#!/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
```

- construir a imagem com:

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


8.1.1 Primeira execução

Para executar o servidor, você precisará de um arquivo de projeto QGIS. Você pode utilizar um dos seus ou escolher [esta amostra](#).

To do so, create a directory `data` within the directory `qgis-server` and copy your file in it. To comply with the following explanations, rename it to `osm.qgs`.

Nota: You may need to add advertised URLs under the *QGIS Server* tab of the *Project ► Properties* if the Get-Capabilities are broken. For example if your server is exposed on port 8080, you will put this for advertised URL `http://localhost:8080/qgis-server/`. More information available in section [Configurar seu projeto](#) and subsequent.

Agora, você pode executar o servidor com:

```
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
```

Opções utilizadas:

- **-d***: rodar em segundo plano
- **--rm**: remover o container quando parar
- **--nome**: nome do container a ser criado
- **--net***: (anteriormente criada) sub-rede
- **--hostname**: nome da hostname do recipiente, para referência posterior
- **-v**: diretório de dados local a ser montado no contêiner
- ****--p**: mapeamento da porta
- **-e***: variável de ambiente a ser utilizada no contêiner

Para verificar, digite `docker ps | grep qgis-server` e você deve ver uma linha com **qgis-server**:

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

8.1.2 Amostra utilizável

As the server is only accepting fastcgi connections, you need an HTTP server that handles this protocol. To do so we have to create a simple Nginx configuration file and start a Nginx image.

Criar um arquivo `nginx.conf` no diretório atual com este conteúdo:

```
server {
    listen 80;
    server_name _;
    location / {
        root /usr/share/nginx/html;
        index index.html index.htm;
    }
    location /qgis-server {
        proxy_buffers 16 16k;
```

(continua na próxima página)

(continuação da página anterior)

```
proxy_buffer_size 16k;
gzip off;
include fastcgi_params;
fastcgi_pass qgis-server:5555;
}
}
```

E digite este comando:

```
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
```

To check capabilities availability, type in a browser <http://localhost:8080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities>

8.1.3 Limpeza

Para limpar as imagens em execução, digite:

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

8.2 Pilhas de docas

O método anterior é fácil de ser roteirizado, mas não pode ser facilmente compactado, nem padronizado ou facilmente administrado.

To work with a docker image set you could use a docker stack managed by an orchestrator. In a stack, the images are working in the same private network, and you can start / stop the whole stack or deploy the stack to other workers. There are many orchestrators, for example Swarm, Kubernetes and Mesos.

A seguir, apresentaremos configurações simples para fins de teste. Elas não são adequadas para produção.

8.2.1 Enxame/doco-composição

Docker now has its own orchestrator: Swarm (compatible with docker-compose files). You have to [enable it](#) (the Mac version will also work with Linux).

Descrição da pilha

Agora que o Swarm está funcionando, crie o arquivo de serviço (veja [Deploy to Swarm](#)) `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
```

(continua na próxima página)

(continuação da página anterior)

```

nginx:
  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

```

Para implantar (ou atualizar) a pilha, digite:

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

Check the stack deployment status until you obtain **1/1** in the **replicas** column:

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

Algo parecido com:

ID	NAME	MODE	REPLICAS	
↪ IMAGE	PORTS			
gmx7ewlvwsqt	qgis_nginx	replicated	1/1	↪
↪ nginx:1.13	*:8080->80/tcp			
10v2e7cl43u3	qgis_qgis-server	replicated	1/1	↪
↪ qgis-server:latest				

To check WMS capabilities, type in a web browser <http://localhost:8080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities>

Limpeza

Para limpeza, digite:

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

8.2.2 Kubernetes

Instalação

If you have a **Docker Desktop** installation, using Kubernetes (aka k8s) is pretty straight forward: [enable k8s](#).

If not, follow the [minikube tutorial](#) or [microk8s for Ubuntu](#).

As Kubernetes installation can be really complex, we will only focus on aspects used by this demo. For further / deeper information, check the [official documentation](#).

microk8s

microk8s needs extra steps: you have to enable the registry and tag the qgis-server image in order to have Kubernetes to find the created images.

Primeiro, habilite o registro:

```
microk8s enable dashboard dns registry
```

Em seguida, etiqüete e passe a imagem para seu registro recém-criado:

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

Finally, add or complete the `/etc/docker/daemon.json` to have your registry **127.0.0.1:32000** listed in the **insecure-registries** field:

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

Criando manifestos

Kubernetes describes the objects to deploy in yaml manifests. There are many different kinds, but we will only use deployments (handle pods, i.e. docker images) and services to expose the deployments to internal or external purposes.

Manifestações de implantação

Criar um arquivo **:arquivo: deployments.yaml** com este conteúdo:

```
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
    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"
```

(continua na próxima página)

(continuação da página anterior)

```

    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
          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;
      }
    }

```

Manifestos de serviço

Criar um arquivo `:arquivo:services.yaml` com este conteúdo:

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

Implantação de manifestos

Para implantar as imagens e serviços em Kubernetes, pode-se usar o painel de controle (clique no **+** no canto superior direito) ou a linha de comando.

Nota: When using the command line with microk8s you will have to prefix each command with *microk8s*.

Para implantar ou atualizar seus manifestos:

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

Para verificar o que está atualmente implantado:

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

Você deveria obter algo como:

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					
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	↪

(continua na próxima página)

(continuação da página anterior)

```
↪ 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

Para ler os logs nginx/qgis, digite:

```
kubectl logs -f POD_NAME
```

To check WMS capabilities, type in a web browser <http://localhost:30080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities>

Limpeza

Para limpar, digite:

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

8.3 Implantação de nuvens

Managing your own cluster of servers to handle the deployment of containerized applications, is a complex job. You have to handle multiple issues, such as hardware, bandwidth and security at different levels.

Cloud deployment solutions can be a good alternative when you do not want to focus on infrastructure management.

A cloud deployment may use proprietary mechanisms, but they are also compatible with the stages explained previously (*docker images* and *stack management*).

8.3.1 AWS usecase

With Amazon AWS, through [ECS \(Elastic Container Service\)](#) functionalities, you can use docker-compose or Kubernetes compatible wrappers to manage your stack. You will have to create an [image registry](#) for your custom images to be accessible.

To use docker-compose alike functionalities, you need to install the **ecs-cli** client and have [proper permissions / roles](#). Then, with the help of the [ecs-cli compose](#) commands, you can reuse the *stack description*.

To use Kubernetes, you can use the AWS web console or the command line tool **eksctl** and have the proper permissions / roles. Then with a well configured kubectl environment, you can reuse the *Kubernetes manifests*.

Perguntas mais frequentes

- *Quais são as diferenças entre o QGIS Desktop e o Servidor QGIS?*

QGIS Desktop has a graphical user interface and allows you to create and modify maps. QGIS Server is a server application serving your QGIS project files to end user applications via OGC web services like [WMS](#), [WFS](#), etc..

- *What is OGC?*

The [OGC \(Open Geospatial Consortium\)](#) is an international not for profit organization committed to making quality open standards for the global geospatial community.

- *Name some other web mapping servers?*

ArcGIS server, Geoserver, Mapserver, Mapnik etc.

- *How to compare QGIS server to other web mapping servers? (2021/01/01)*

Feições	Servidor QGIS	GeoServer	Servidor ArcGIS
Desde	2006	2001	1999
Licença	GPL	GPL	comercial
Suporte comercial	Multiple companies	Multiple companies	ESRI and its vendors network
Tecnologia	C++/python	Java	C++
Tile cache	sim	yes (via GeoWebCache)	sim
3D	Não	Não	Sim
Consultas	FES (2.0) and OGC (1.0) filters	CQL and OGC filters	OGC filters
Report generation	sim	sim	sim
Administração do servidor	yes via third parties (LizMap, QWC2, etc.)	web + API REST	web + API REST
Edição de camada/simbologia do projeto GIS	complete via dedicated GUI	simple via web interface	complete via dedicated GUI

- *What are the OGC specification versions implemented in QGIS server compared to other web mapping servers? (2021/01/01)*

OGC standards	Servidor QGIS	GeoServer	Servidor ArcGIS
WMS (Web Map Service)	1.3.0 - 1.1.1	1.3.0 - 1.1.1	1.3.0 - 1.1.1
SRW (Serviço de Recursos da Web)	1.1.0 - 1.0.0	2.0.0 - 1.0.0	2.0.0 - 1.0.0
OAPIF (aka WFS3)	1.0.0	não	não
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)	não	1.0.0	1.0.0
SCW (Serviço de Catálogo para a Web)	não	2.0.2	não
SLD (Styled Layer Descriptor)	sim	sim	sim

- *What is a tile cache?*

Maps are often static. As most mapping clients render WMS (Web Map Service) data every time they are queried, this can result in unnecessary processing and increased wait times.

The tile cache optimizes this experience by saving (caching) map images, or tiles, as they are requested, in effect acting as a proxy between client (such as OpenLayers or Google Maps) and server (any WMS-compliant server). As new maps and tiles are requested, QGIS server intercepts these calls and returns pre-rendered tiles if stored, or calls the QGIS engine to render new tiles as necessary. Thus, once tiles are stored, the speed of map rendering increases by many times, creating a much improved user experience.

- *What is PostgreSQL?*

PostgreSQL <<https://www.postgresql.org/>>_ é um poderoso e de código aberto companheiro de banco de dados objeto-relacional para QGIS.

- *O que é PostGIS?*

PostGIS is a spatial database extender for PostgreSQL object-relational database. It adds support for geographic objects allowing location queries to be run in SQL.