



QGIS Server 3.22 User Guide

QGIS Project

04 dic 2023

1	Introduzione	1
2	Come Iniziare	3
2.1	Installazione su sistemi Debian	3
2.1.1	Apache HTTP Server	4
2.1.2	NGINX HTTP Server	6
2.1.3	Xvfb	11
2.2	Installazione su Windows	12
2.3	Servire un progetto	14
2.4	Configura il tuo progetto	15
2.4.1	WMS capabilities	16
2.4.2	WFS capabilities	17
2.4.3	WCS capabilities	17
2.4.4	Messa a punto del tuo OWS	17
2.5	Integrazione con terze parti	17
2.5.1	Integrazione con QGIS Desktop	17
2.5.2	Integrazione con MapProxy	17
2.5.3	Integrazione con QWC2	18
3	Servizi	19
3.1	Di base	19
3.1.1	SERVICE	20
3.1.2	REQUEST	20
3.1.3	MAP	20
3.1.4	FILE_NAME	20
3.1.5	Nome breve	21
3.2	Web Map Service (WMS)	21
3.2.1	GetCapabilities	22
3.2.2	GetMap	22
3.2.3	GetFeatureInfo	31
3.2.4	GetLegendGraphics	34
3.2.5	GetStyle(s)	47
3.2.6	DescribeLayer	47
3.2.7	GetPrint	48
3.2.8	GetProjectSettings	50
3.2.9	GetSchemaExtension	51
3.2.10	Layer WMS esterni	51
3.2.11	Redlining	52
3.3	Web Feature Service (WFS)	54
3.3.1	GetCapabilities	54
3.3.2	GetFeature	55

3.3.3	DescribeFeatureType	61
3.3.4	Transazione	62
3.4	Web Coverage Service (WCS)	64
3.4.1	GetCapabilities	64
3.4.2	DescribeCoverage	65
3.4.3	GetCoverage	67
3.5	Web Map Tile Service (WMTS)	68
3.5.1	GetCapabilities	68
3.5.2	GetTile	69
3.5.3	GetFeatureInfo	70
3.6	OGC API Features	71
3.6.1	Rappresentazione risorse	72
3.6.2	Endpoint	72
3.6.3	Paginazione	76
3.6.4	Filtro delle geometrie	76
3.6.5	Ordinamento elemento	77
3.6.6	Selezione attributo	77
3.6.7	Personalizzare le pagine HTML	78
4	Catalog	79
5	Plugin	83
5.1	Elenco dei plugin	83
5.2	Ubicazione dei plugin	83
5.3	Installazione	83
5.3.1	Manualmente con uno ZIP	83
5.3.2	Con uno strumento a linea di comando	84
5.4	Configura il server HTTP	84
5.4.1	Apache	84
5.5	Come usare un plugin	85
6	Configurazione avanzata	87
6.1	Registrazioni di controllo	87
6.2	Variabili d'ambiente	87
6.3	Riepilogo impostazioni	89
6.4	Connessione a file di servizio	90
6.5	Aggiungere font al server Linux	90
7	Server di sviluppo	91
8	Sviluppo ed esecuzione in container	93
8.1	Semplici immagini docker	93
8.1.1	Prima esecuzione	95
8.1.2	Esempio utilizzabile	95
8.1.3	Pulizia	96
8.2	Stack Docker	96
8.2.1	Swarm/docker-compose	96
8.2.2	Kubernetes	97
8.3	Distribuzione in cloud	101
8.3.1	Caso d'uso AWS	101
9	Domande frequenti	103

Introduzione

QGIS Server è un'implementazione open source di WMS, WFS, OGC API for Features 1.0 (WFS3) e WCS che, inoltre, implementa funzionalità cartografiche avanzate per la mappatura tematica. QGIS Server è un'applicazione FastCGI/CGI (Common Gateway Interface) scritta in C++ che lavora insieme a un server web (ad esempio, Apache, Nginx). Ha un supporto per i plugin Python che permette uno sviluppo veloce ed efficiente e l'implementazione di nuove funzionalità.

QGIS Server utilizza QGIS come backend per la restituzione delle mappe e per le operazioni gis. Inoltre, la libreria Qt viene utilizzata per la grafica e per la programmazione C++ indipendente dalla piattaforma. A differenza di altri server WMS, QGIS Server usa regole cartografiche come linguaggio di configurazione, sia per la configurazione a livello server che per le configurazioni utente.

Come QGIS desktop, QGIS Server utilizza le stesse librerie di visualizzazione e le mappe che vengono pubblicate sulla rete hanno lo stesso aspetto che in desktop GIS.

Nelle sezioni seguenti, forniremo una configurazione di esempio per impostare un server QGIS su Linux (Debian, Ubuntu e derivati) e su Windows. Per maggiori informazioni sullo sviluppo di plugin per server, leggi `server_plugins`.

È concessa l'autorizzazione a copiare, distribuire e/o modificare questo documento secondo i termini della GNU Free Documentation License, versione 1.3 o qualsiasi versione successiva pubblicata dalla Free Software Foundation; senza sezioni non modificabili, senza testi di copertina e senza testi di retrocopertina.

Una copia della licenza è inclusa nella sezione `gnu_fdl`.

2.1 Installazione su sistemi Debian

Daremo un breve e semplice how-to di installazione per una configurazione minima funzionante su sistemi basati su Debian (incluso Ubuntu e derivati). Tuttavia, molte altre distribuzioni e sistemi operativi forniscono package per QGIS Server.

Nota: In Ubuntu puoi usare il tuo normale utente, aggiungendo `sudo` ai comandi che richiedono permessi di admin. In Debian puoi lavorare come admin (`root`), senza usare `sudo`.

I requisiti e i passaggi per aggiungere i repository ufficiali di QGIS per installare QGIS Server su un sistema basato su Debian sono forniti nella pagina [QGIS installers](#). È preferibile che tu installi perlomeno l'ultima Long Term Release.

Una volta configurato il repository della versione di destinazione e installato QGIS Server, puoi testare l'installazione con:

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

Se ottieni il seguente risultato, il server è stato installato correttamente.

Nota: A seconda della versione di QGIS, potresti vedere un output leggermente diverso quando si esegue `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">
```

(continues on next page)

(continua dalla pagina precedente)

```
<ServiceException code="Service configuration error">Service unknown or
↳ unsupported</ServiceException>
</ServiceExceptionReport>
```

Nota: Come riportato di seguito, QGIS segnala un codice di stato 400, che identifica correttamente che la richiesta è fallita perché non c'è una sessione http attiva. Questo non è un bug e indica che il server funziona correttamente.

```
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"}]
```

Aggiungiamo un progetto di esempio. Puoi usare il tuo o uno di [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 .
```

Naturalmente, puoi utilizzare il tuo software GIS preferito per aprire questo file e dare un'occhiata alla configurazione e ai layer disponibili.

Per implementare correttamente il server QGIS è necessario un server HTTP. Le scelte consigliate sono [Apache](#) o [NGINX](#).

2.1.1 Apache HTTP Server

Nota: Quindi, sostituisci `qgis.demo` con il nome o l'indirizzo IP del tuo server.

1. Installa Apache e `mod_fcgid`:

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

2. Puoi eseguire QGIS Server sul sito web predefinito, ma configuriamo un [virtualhost](#) appositamente per questo, come segue.

1. Nella cartella `/etc/apache2/sites-available`, crea un file chiamato `qgis.demo.conf`, con questo contenuto:

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

    DocumentRoot /var/www/html

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

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

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

    # QGIS log
    FcgidInitialEnv QGIS_SERVER_LOG_STDERR 1
    FcgidInitialEnv QGIS_SERVER_LOG_LEVEL 0

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

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

    # Set pg access via pg_service file
    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: Alcune delle opzioni di configurazione di cui sopra sono spiegate nelle sezioni Server *environment variables* e *pg_service* file.

2. Creiamo ora le cartelle che ospiteranno i registri di QGIS Server e il database di autenticazione:

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

Nota: `www-data` è l'utente di Apache sui sistemi basati su Debian e abbiamo bisogno che Apache abbia accesso a queste cartelle o file. I comandi `chown www-data...` cambiano il proprietario delle rispettive cartelle e file in `www-data`.

3. Ora possiamo abilitare l'host virtuale e il mod `"fcgid"`, se non è già stato fatto:

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

4. Ora riavvia Apache per usare la nuova configurazione:

```
systemctl restart apache2
```

5. Ora che Apache sa che deve rispondere alle richieste di <http://qgis.demo>, dobbiamo anche configurare il sistema client in modo che sappia chi è `qgis.demo`. Per farlo, aggiungiamo `127.0.0.1 qgis.demo` nel file `hosts`.

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

Importante: Ricorda che entrambi i file `qgis.demo.conf` e `/etc/hosts` devono essere configurati affinché la configurazione funzioni. Puoi anche testare l'accesso al server QGIS da altri client in rete (ad esempio macchine Windows o MacOS) andando nel file `/etc/hosts` e puntando il nome `qgis.demo` a qualsiasi IP della macchina server sulla rete (non `127.0.0.1` perché è l'IP locale, accessibile solo dalla macchina locale). Sui computer `*nix` il file `hosts` si trova in `/etc`, mentre su Windows si trova nella cartella `C:\Windows\System32\drivers\etc`. In Windows è necessario avviare l'editor di testo con privilegi di amministratore prima di aprire il file `hosts`.

QGIS Server è ora disponibile all'indirizzo <http://qgis.demo>. Per verificare, è sufficiente scrivere in un browser, come nel seguente caso semplice:

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

2.1.2 NGINX HTTP Server

Nota: Quindi, sostituisci `qgis.demo` con il nome o l'indirizzo IP del tuo server.

Puoi utilizzare QGIS Server anche con **NGINX**. A differenza di Apache, NGINX non avvia automaticamente i processi FastCGI. I processi FastCGI devono essere avviati da qualcos'altro.

Installa NGINX:

```
apt install nginx
```

- Come prima opzione, puoi usare **spawn-fcgi** o **fcgiwrap** per avviare e gestire i processi del server QGIS. Per entrambi esistono pacchetti ufficiali Debian. Quando non hai un server X in esecuzione e hai bisogno, per esempio, di stampare, puoi usare *xvfb*.

- Un'altra opzione è quella di affidarsi a **Systemd**, il sistema di init per GNU/Linux che la maggior parte delle distribuzioni Linux utilizza oggi. Uno dei vantaggi di questo metodo è che non richiede altri componenti o processi. È pensato per essere semplice, ma robusto ed efficiente per le distribuzioni di produzione.

Configurazione NGINX

Il `include fastcgi_params`; usato nella configurazione precedente è importante, perché aggiunge i parametri da `/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;
```

Inoltre, puoi utilizzare alcune *Variabili d'ambiente* per configurare QGIS Server. Nel file di configurazione di NGINX, `/etc/nginx/nginx.conf`, devi usare l'istruzione `fastcgi_param` per definire queste variabili, come mostrato di seguito:

```
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;
}
```

Wrapper FastCGI

Avvertimento: `fcgiwrap` è più facile da configurare di `spawn-fcgi`, perché è già confezionato in un servizio Systemd. Ma porta anche a una soluzione molto più lenta rispetto all'uso di `spawn-fcgi`. Con `fcgiwrap`, viene creato un nuovo processo del server QGIS a ogni richiesta, il che significa che il processo di inizializzazione del server QGIS, che include la lettura e l'analisi del file di progetto QGIS, viene eseguito a ogni richiesta. Con `spawn-fcgi`, il processo QGIS Server rimane in vita tra una richiesta e l'altra, con conseguenti migliori prestazioni. Per questo motivo, `spawn-fcgi` è consigliato per l'uso in produzione.

spawn-fcgi

Se vuoi utilizzare `spawn-fcgi`:

1. Il primo passo è installare il package:

```
apt install spawn-fcgi
```

2. Quindi, inserisci il seguente blocco nella configurazione del server NGINX:

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

3. E riavvia NGINX per tenere conto della nuova configurazione:

```
systemctl restart nginx
```

4. Infine, dato che non esiste un file di servizio predefinito per `spawn-fcgi`, devi avviare manualmente QGIS Server nel terminale:

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

QGIS Server è ora disponibile su <http://qgis.demo/qgisserver>.

Nota: Quando si usa `spawn-fcgi`, puoi definire direttamente le variabili d'ambiente prima di eseguire il server. Per esempio: `export QGIS_SERVER_LOG_STDERR=1`

Naturalmente, puoi aggiungere uno script di init per avviare QGIS Server all'avvio o quando vuoi. Ad esempio con **systemd**:

1. Modifica il file `/etc/systemd/system/qgis-server.service` con questo contenuto:

```
[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. Quindi attiva e avvia il servizio:

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

Avvertimento: Con i comandi precedenti, spawn-fcgi genera un solo processo di QGIS Server.

fcgiwrap

L'utilizzo di `fcgiwrap` è molto più semplice da configurare rispetto a `spawn-fcgi`, ma è molto più lento.

1. Devi per prima cosa installare il package corrispondente:

```
apt install fcgiwrap
```

2. Quindi, inserisci il seguente blocco nella configurazione del server 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. Infine, riavvia NGINX e `fcgiwrap` per tenere conto della nuova configurazione:

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

QGIS Server è ora disponibile su <http://qgis.demo/qgisserver>.

Systemd

QGIS Server ha bisogno di un X Server funzionante per essere pienamente utilizzabile, in particolare per la stampa. Nel caso in cui si disponga già di un server X funzionante, puoi utilizzare i servizi systemd.

Questo metodo, per implementare QGIS Server, si basa su due unità Systemd da configurare: un'unità `Socket` e un'unità `Service`.

1. L'unità **QGIS Server Socket** definisce e crea un socket del file system, utilizzato da NGINX per avviare e comunicare con QGIS Server. L'unità Socket deve essere configurata con `Accept=false`, il che significa che le richieste alla chiamata di sistema `accept()` sono delegate al processo creato dall'unità Service. Si trova in `/etc/systemd/system/qgis-server@.socket`, che in realtà è un modello:

```
[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. Ora abilita e avvia i socket:

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

3. L'unità **QGIS Server Service** definisce e avvia il processo QGIS Server. La parte importante è che l'input standard del processo Service sia collegato al socket definito dall'unità Socket. Questo deve essere configurato

usando `StandardInput=socket` nella configurazione dell'unità Service situata 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: Le *environment variables* di QGIS Server sono definite in un file separato, `/etc/qgis-server/env`. Potrebbe apparire come segue:

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

4. Avvia ora il servizio socket:

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

5. Infine, per il server HTTP NGINX, introduciamo la configurazione per questa impostazione:

```
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. Ora riavvia NGINX per tenere conto della nuova configurazione:

```
systemctl restart nginx
```

Grazie a Oslandia per aver condiviso [their tutorial](#).

2.1.3 Xvfb

QGIS Server ha bisogno di un server X funzionante per essere pienamente utilizzabile, in particolare per la stampa. Sui server di solito si raccomanda di non installarlo, quindi si può usare `xvfb` per avere un ambiente X virtuale.

Se stai eseguendo il server in ambiente grafico/X11, non è necessario installare `xvfb`. Maggiori informazioni su <https://www.itopen.it/qgis-server-setup-notes/>.

1. Per installare il package:

```
apt install xvfb
```

2. Crea il file di servizio, `/etc/systemd/system/xvfb.service`, con questo contenuto:

```
[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. Abilita, avvia e verifica lo stato di `xvfb.service`:

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

4. Quindi, a seconda del server HTTP, devi configurare il parametro **DISPLAY** o usare direttamente **xvfb-run**.

- Utilizzando Apache:

1. Aggiungi alla tua configurazione `Fcgid` (vedi *Apache HTTP Server*):

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

2. Riavvia Apache affinché la nuova configurazione venga presa in carico:

```
systemctl restart apache2
```

- Utilizzando NGINX

- Con `spawn-fcgi` usando `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
```

- Con la variabile d'ambiente **DISPLAY** nella configurazione del server HTTP.

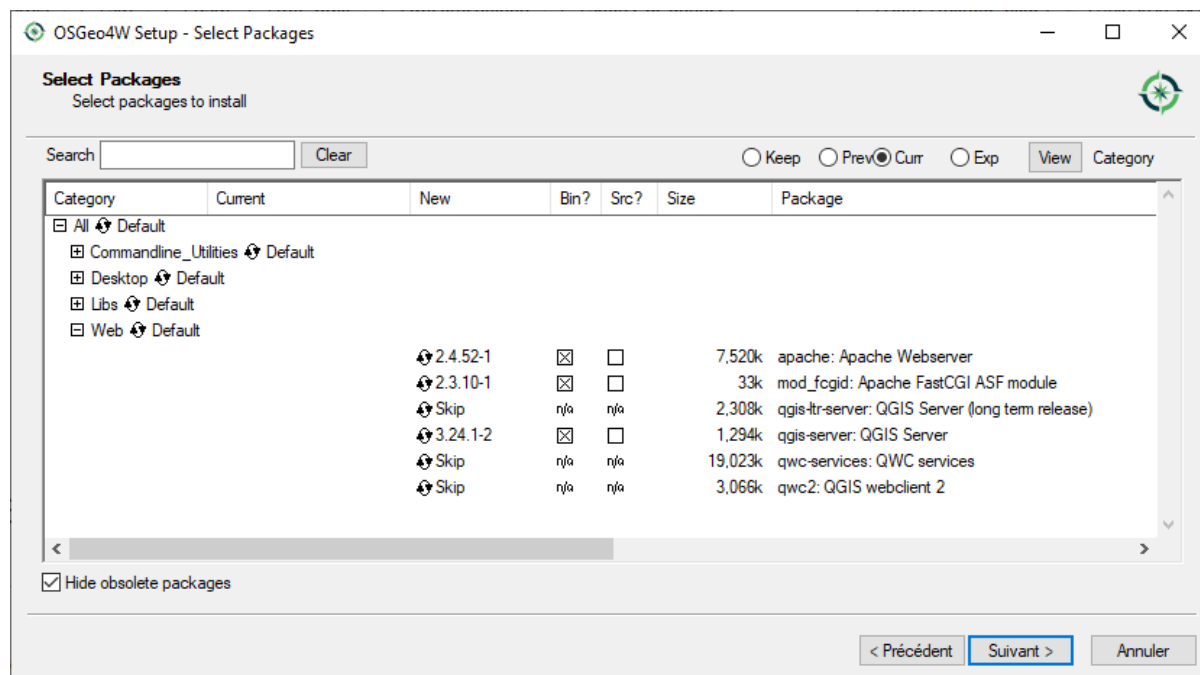
```
fastcgi_param DISPLAY ":99";
```

2.2 Installazione su Windows

QGIS Server può essere installato anche su sistemi Windows utilizzando la versione a 64 bit dell'installatore di rete OSGeo4W (<https://qgis.org/en/site/forusers/download.html>).

Una procedura semplice è la seguente:

1. Scarica ed esegui il programma di installazione di OSGeo4W
2. Procedi con «Installazione avanzata» e installa i pacchetti **QGIS Desktop**, **QGIS Server apache** e **mod_fcgid**.



3. Apache non viene installato direttamente come servizio su Windows. Devi:

1. Fare clic con il tasto destro del mouse sul file OSGeo4W.bat alla radice della cartella C:\OSGeo4W64` (se sono stati utilizzati i percorsi di installazione predefiniti) e selezionare :guilabel:`Esegui come amministratore.
2. Nella console, esegui apache-install.bat, che produrrà il seguente risultato

```
> 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.
...
```

Il servizio è avviato, come puoi notare nel report. Ma il server potrebbe non funzionare a causa di una configurazione personalizzata mancante.

4. Modifica il file C:\OSGeo4w64apps\apacheconf\httpd.conf con le seguenti variazioni (sono possibili altre combinazioni):

Obiettivo	Configurazione esistente	Sostituzione
(Opzionale) Personalizzare l'indirizzo di ascolto utilizzando un IP e/o una porta. Puoi aggiungere tutte le voci che vuoi.	<code>Listen \${SRVPORT}</code>	<code>Listen localhost:8080</code>
Indica dove trovare i file di script	<code>ScriptAlias /cgi-bin/ "\$ ↪{SRVROOT}/cgi-bin/"</code>	<code>ScriptAlias /cgi-bin/ ↪"C:/OSGeo4W64/apps/ ↪qgis/bin/"</code>
Fornisci le autorizzazioni alla cartella degli script	<code><Directory "\${SRVROOT}/ ↪cgi-bin"> AllowOverride None Options None Require all granted </Directory></code>	<code><Directory "C:/ ↪OSGeo4W64/apps/qgis/ ↪bin"> SetHandler cgi- ↪script AllowOverride None Options ExecCGI Require all granted </Directory></code>
Abilita le estensioni dei file da utilizzare per i file di script. Deseleziona e completa	<code>#AddHandler cgi-script . ↪cgi</code>	<code>AddHandler cgi-script . ↪cgi .exe</code>
Aggiungi altre variabili di configurazione personalizzate di OSGeo4W	<code># parse OSGeo4W apache_ ↪conf files IncludeOptional "C:/ ↪OSGeo4W64/httpd.d/ ↪httpd_*.conf"</code>	<code># parse OSGeo4W apache_ ↪conf files IncludeOptional "C:/ ↪OSGeo4W64/httpd.d/ ↪httpd_*.conf" SetEnv GDAL_DATA "C:/ ↪OSGeo4W64/share/gdal" SetEnv QGIS_AUTH_DB_DIR_ ↪PATH "C:/OSGeo4W64/ ↪apps/qgis/resources"</code>

5. Riavvia il server web Apache

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

6. Apri la finestra del browser per testare una richiesta GetCapabilities al server QGIS. Sostituisci localhost:8080 con l'IP e la porta impostati per l'ascolto.

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

Dovrebbe essere restituito un file XML con le capabilities. Il tuo server è pronto per l'uso.

2.3 Servire un progetto

Ora che QGIS Server è installato e funzionante, non resta che utilizzarlo.

Ovviamente, abbiamo bisogno di un progetto QGIS su cui lavorare. Naturalmente, puoi personalizzare completamente il progetto, definendo le informazioni di contatto, specificando alcune restrizioni sul SR o addirittura escludendo alcuni layer. Tutto quello che c'è da sapere a questo proposito è descritto più avanti in [Configura il tuo progetto](#).

Ma per ora useremo un semplice progetto già configurato e precedentemente scaricato in `/home/qgis/projects/world.qgs`, come descritto sopra.

Aprendo il progetto e dando una rapida occhiata ai layer, sappiamo che al momento sono disponibili 4 layer:

- airports
- places
- countries
- countries_shapeburst

Non è necessario che tu capisca la richiesta completa per ora, ma puoi recuperare una mappa con alcuni dei layer precedenti grazie a QGIS Server facendo qualcosa come questo nel tuo browser web per recuperare il layer *countries*:

```
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 ottieni l'immagine seguente, QGIS Server funziona correttamente:

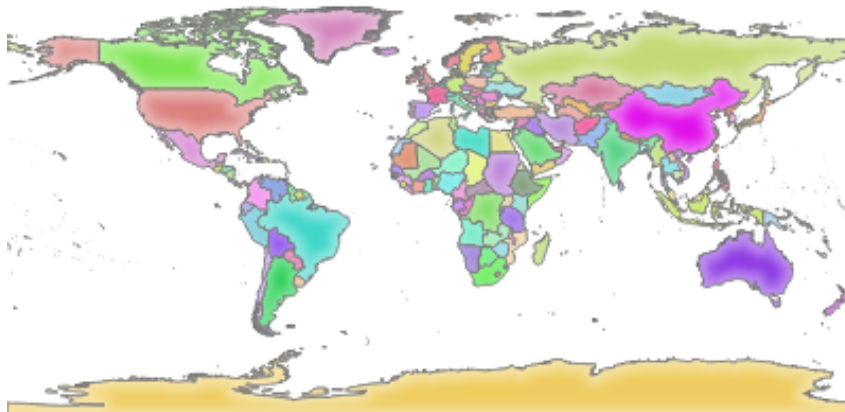


Fig. 2.1: Risposta del server a una richiesta GetMap di base

Nota che puoi definire la variabile d'ambiente **QGIS_PROJECT_FILE** per utilizzare un progetto per impostazione predefinita, invece di fornire un parametro **MAP** (vedi [Variabili d'ambiente](#)).

Per esempio con `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 Configura il tuo progetto

Per realizzare un nuovo QGIS Server WMS, WFS o WCS, devi creare un file di progetto QGIS con alcuni dati o utilizzare uno dei progetti esistenti. Definire i colori e gli stili dei layer in QGIS e il SR del progetto, se non già definiti.

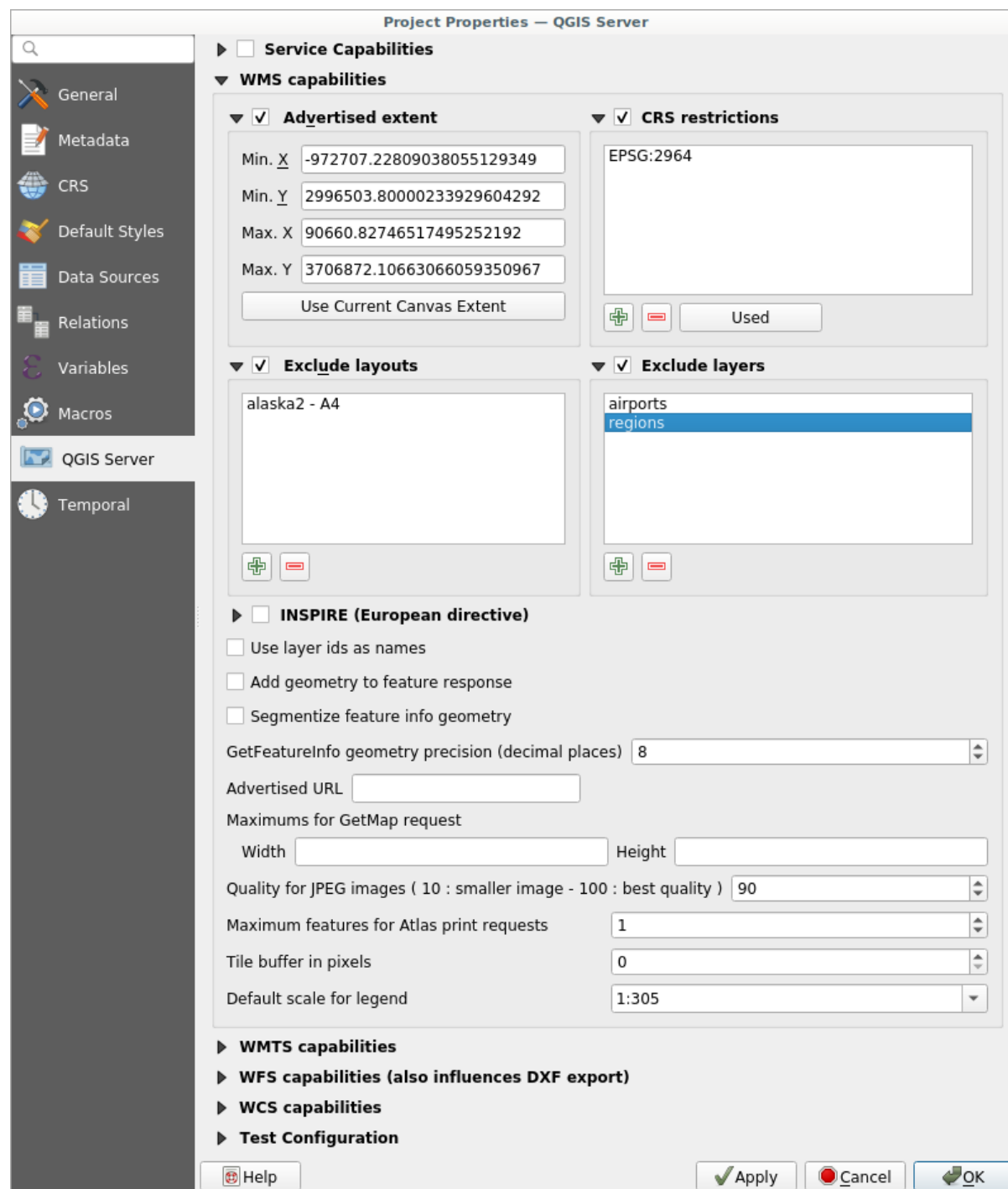


Fig. 2.2: Definizioni per un progetto QGIS Server WMS/WFS/WCS

Quindi, vai al menu *QGIS Server* della finestra di dialogo *Progetto ► Proprietà...* e fornire alcune informazioni sull'OWS nei campi sotto *Service Capabilities*. Queste informazioni appariranno nella risposta *GetCapabilities* di WMS, WFS o WCS. Se non selezioni la casella di controllo *Service capabilities*, QGIS Server utilizzerà le informazioni fornite nel file `wms_metadata.xml` situato nella cartella `cgi-bin`.

2.4.1 WMS capabilities

Nella sezione *WMS capabilities*, puoi definire l'estensione pubblicata nella risposta di WMS GetCapabilities inserendo i valori X e Y minimi e massimi nei campi sotto *Advertised extent*. Facendo clic su *Usa estensione mappa corrente* si impostano questi valori sull'estensione attualmente visualizzata nella area mappa QGIS. Selezionando la casella di controllo *Restrizioni SR*, puoi limitare i sistemi di riferimento delle coordinate (SR) offerti da QGIS Server per la visualizzazione delle mappe. Si consiglia di limitare i SR offerti per ridurre le dimensioni della risposta di WMS GetCapabilities. Usa il pulsante **!simbologyAdd!** qui sotto per selezionare i SR dal selettore del sistema di riferimento di coordinate, oppure fai clic su *Used* per aggiungere all'elenco i SR usati nel progetto QGIS.

Se nel progetto hai definito dei layout di stampa, questi saranno elencati nella risposta *GetProjectSettings* e potranno essere usati dalla richiesta *GetPrint* per creare delle stampe, usando uno dei layout di stampa come modello. Questa è un'estensione specifica di QGIS alla specifica WMS 1.3.0. Se vuoi escludere un layout di stampa dalla pubblicazione da parte del WMS, seleziona la casella di controllo *Escludi layout* e fai clic sul pulsante **!simbologyAdd!** sottostante. Quindi, seleziona un layout di stampa dalla finestra di dialogo *Seleziona layout di stampa* per aggiungerlo all'elenco dei layout esclusi.

Se vuoi escludere un layer o un gruppo di layer dalla pubblicazione da parte del WMS, seleziona la casella di controllo *Escludi layer* e fai clic sul pulsante **!simbologyAdd!** sottostante. Si apre la finestra di dialogo *Selezione di layer e gruppi limitati*, che consente di scegliere i layer e i gruppi che non vuoi vengano pubblicati. Usa i tasti *Shift* o *Ctrl* se si vuoi selezionare più voci. Si consiglia di escludere dalla pubblicazione i layer non necessari, in quanto ciò riduce le dimensioni della risposta di WMS GetCapabilities, con conseguenti tempi di caricamento più rapidi sul lato client.

Se selezioni ☒ *Use layer ids as name*, gli id dei layer saranno usati per fare riferimento ai layer nella risposta *GetCapabilities* o nel parametro *GetMap LAYERS*. In caso contrario, viene utilizzato il nome del layer o il nome breve, se definito (vedi *vectorservermenu*).

Puoi ricevere le *GetFeatureInfo* richieste come testo semplice, XML e GML. L'impostazione predefinita è XML.

Se vuoi, puoi selezionare ☒ *Add geometry to feature response*. Questo includerà il riquadro di delimitazione per ogni elemento nella risposta di *GetFeatureInfo*. Vedi anche il parametro *WITH_GEOMETRY*.

Poiché molti client web non possono visualizzare gli archi circolari nelle geometrie, puoi segmentare la geometria prima di inviarla al client in una risposta *GetFeatureInfo*. Ciò consente a tali client di visualizzare comunque la geometria di un elemento (ad esempio per evidenziare l'elemento). È necessario selezionare la casella di controllo *Segmentize feature info geometry* per attivare l'opzione.

Puoi anche usare l'opzione *GetFeatureInfo geometry precision* per impostare la precisione della geometria *GetFeatureInfo*. Ciò ti consente di risparmiare larghezza di banda quando non è necessaria la massima precisione.

Se vuoi che QGIS Server pubblicizzi URL di richiesta specifici nella risposta di WMS GetCapabilities, inserisci l'URL corrispondente nel campo *Advertised URL*.

Inoltre, puoi limitare la dimensione massima delle mappe restituite dalla richiesta *GetMap* inserendo la larghezza e l'altezza massime nei rispettivi campi sotto *Maximums for GetMap request*.

Puoi modificare il fattore *Quality for JPEG images*. Il fattore di qualità deve essere compreso nell'intervallo 0-100. Specifica 0 per la massima compressione, 100 per nessuna compressione.

Puoi modificare il limite di elementi dell'atlante da stampare in una richiesta impostando il campo *Maximum features for Atlas print requests*.

Quando QGIS Server viene utilizzato in modalità a tasselli (vedi il parametro *TILED*), puoi impostare il parametro *Tile buffer in pixel*. Il valore consigliato è la dimensione del simbolo o della linea più grande del progetto QGIS.

Se uno dei tuoi layer usa la Map Tip display (cioè per mostrare il testo usando le espressioni), questo verrà elencato nell'output di *GetFeatureInfo*. Se il layer utilizza una Mappa dei valori per uno dei suoi attributi, anche questa informazione verrà mostrata nell'output di *GetFeatureInfo*.

2.4.2 WFS capabilities

Nell'area *WFS capabilities* puoi selezionare i layer che vuoi pubblicare come WFS e specificare se consentiranno operazioni di aggiornamento, inserimento e cancellazione. Se inserisci un URL nel campo *Advertised URL* della sezione *WFS capabilities*, QGIS Server pubblicizzerà questo URL specifico nella risposta WFS GetCapabilities.

2.4.3 WCS capabilities

Nell'area *WCS capabilities*, puoi selezionare i livelli che vuoi pubblicare come WCS. Se inserisci un URL nel campo *Advertised URL* della sezione *WCS capabilities*, QGIS Server pubblicizzerà questo URL specifico nella risposta WCS GetCapabilities.

2.4.4 Messa a punto del tuo OWS

Per i layer vettoriali, il menu *Campi* della finestra di dialogo *Layer ► Proprietà del layer* consente di definire per ogni attributo se sarà pubblicato o meno. Per impostazione predefinita, tutti gli attributi sono pubblicati da WMS e WFS. Se non si desidera pubblicare un attributo specifico, deselezionare la casella di controllo corrispondente nella colonna *WMS* o *WFS*.

Puoi sovrapporre watermarks alle mappe prodotte dal WMS aggiungendo annotazioni di testo o SVG al file di progetto. Per istruzioni sulla creazione di annotazioni, vedere la sezione *sec_annotations*. Affinché le annotazioni vengano visualizzate come watermarks nell'output del WMS, la casella di controllo *Posizione fissa mappa* nella finestra di dialogo *Testo dell'annotazione* deve essere deselezionata. Si può accedere a questa casella facendo doppio clic sull'annotazione mentre è attivo uno degli strumenti di annotazione. Per le annotazioni SVG, è necessario impostare il progetto in modo che salvi percorsi assoluti (nel menu *Generale* della finestra di dialogo *Progetto ► Proprietà...*) o modificare manualmente il percorso dell'immagine SVG in modo che rappresenti un percorso relativo valido.

2.5 Integrazione con terze parti

QGIS Server fornisce servizi web standard OGC come *WMS*, *WFS*, *etc.* e può quindi essere utilizzato da un'ampia gamma di strumenti per l'utente finale.

2.5.1 Integrazione con QGIS Desktop

QGIS Desktop è il designer di mappe e QGIS Server è il server di mappe. Le mappe o i progetti QGIS saranno serviti da QGIS Server per fornire gli standard OGC. Questi progetti QGIS possono essere file o voci di un database (utilizzando *:menuselection: Progetto -> Salva in -> PostgreSQL* in QGIS Desktop).

Inoltre, è necessario stabilire un flusso di lavoro di aggiornamento dedicato per aggiornare un progetto utilizzato da un server QGIS (cioè copiare i file del progetto nella posizione del server e riavviare il server QGIS). Per il momento, i processi automatizzati (come il ricaricamento del server tramite il servizio di coda di messaggi) non sono ancora stati implementati.

2.5.2 Integrazione con MapProxy

MapProxy è un server di cache di tasselli e, poiché può leggere e servire qualsiasi server di mappe WMS/WMTS, può essere collegato direttamente ai servizi web del server QGIS e migliorare l'esperienza dell'utente finale.

2.5.3 Integrazione con QWC2

QWC2 è un'applicazione web reattiva dedicata a QGIS Server. Aiuta a costruire un visualizzatore di mappe altamente personalizzato con selezione dei layer, informazioni su elementi, ecc. Sono disponibili anche molti plugin, come il servizio di autenticazione o di stampa; l'elenco completo è disponibile in questo [repository](#).

QGIS è in grado di gestire i dati secondo i protocolli standard come descritto da **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

Ulteriori parametri e richieste sono supportati in aggiunta allo standard originale che migliora notevolmente le possibilità di personalizzazione grazie al motore di visualizzazione QGIS.

3.1 Di base

Questa sezione descrive concetti e parametri condivisi dai servizi. Alcuni di questi sono standard e definiti nelle specifiche OGC, mentre altri sono molto specifici di QGIS Server.

Concetti standard:

Concetti	Descrizione
<i>SERVICE</i>	Nome del servizio
<i>REQUEST</i>	Nome della richiesta

Concetti del fornitore:

Concetti	Descrizione
<i>MAP</i>	file di progetto QGIS
<i>FILE_NAME</i>	Nome del file scaricato
<i>Short name</i>	Definizione del nome breve

3.1.1 SERVICE

Questo parametro standard consente di specificare il nome del servizio da utilizzare per una specifica *request* e deve essere formato come `SERVICE=NAME`.

Esempio di URL per il servizio **WMS**:

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

Nota: Non disponibile per i servizi basati su REST come *WFS3 (OGC API Features)*.

3.1.2 REQUEST

Questo parametro standard consente di specificare il nome della richiesta da eseguire per uno specifico *service* e deve essere formato come `REQUEST=NomeRichiesta`.

Esempio di URL per la richiesta **GetCapabilities**:

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

Nota: Non disponibile per i servizi basati su REST come *WFS3 (OGC API Features)*.

3.1.3 MAP

Questo parametro del fornitore consente di definire il file di progetto QGIS da utilizzare. Può essere un percorso assoluto o un percorso relativo alla posizione dell'eseguibile del server `qgis_mapserv.fcgi`. `MAP` è obbligatorio per impostazione predefinita, perché una richiesta ha bisogno di un progetto QGIS per funzionare. Tuttavia, la variabile d'ambiente **QGIS_PROJECT_FILE** può essere usata per definire un progetto QGIS predefinito. In questo caso specifico, `MAP` non è più un parametro richiesto. Per ulteriori informazioni, puoi consultare il capitolo *Configurazione avanzata*.

Esempio di URL:

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

3.1.4 FILE_NAME

Se questo parametro del fornitore è impostato, la risposta del server sarà inviata al client come file allegato con il nome del file specificato.

Esempio di URL per salvare un documento XML **GetCapabilities**:

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


Nota: Non disponibile per i servizi basati su REST come [WFS3 \(OGC API Features\)](#).

3.1.5 Nome breve

Alcuni elementi hanno sia un **nome breve** che un **titolo**. Il nome abbreviato è una stringa di testo utilizzata per la comunicazione da macchina a macchina, mentre il titolo è a beneficio dell'uomo. Ad esempio, un set di dati potrebbe avere il titolo descrittivo «*Maximum Atmospheric Temperature*» ed essere richiesto con il nome abbreviato «*ATMAX*». Puoi impostare titolo, nome abbreviato e abstract per:

- **Layers:** fai clic con il tasto destro del mouse su un layer e scegli *Proprietà...* ► *Server QGIS* ► *Descrizione*.
- **Groups:** fai clic con il pulsante destro del mouse su un gruppo e selezionare *Imposta dati WMS gruppo*.
- **Project:** vai su *Progetto* ► *Proprietà...* ► *Server QGIS* ► *Service Capabilities*.

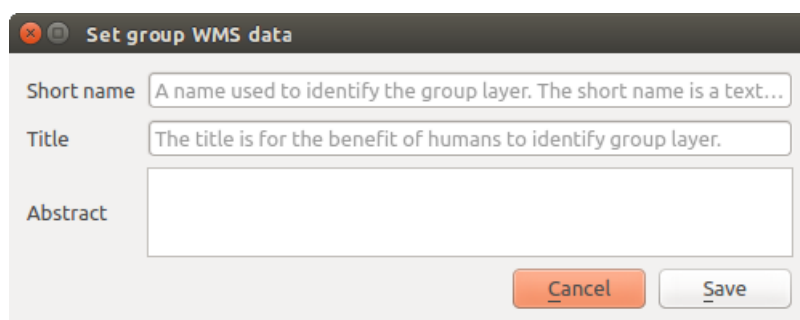


Fig. 3.1: Impostazione della finestra di dialogo WMS dei dati del gruppo

Pertanto, il nome breve può essere utilizzato per identificare questi elementi quando si interagisce con QGIS Server. Ad esempio con il parametro standard `LAYERS`:

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

3.2 Web Map Service (WMS)

Gli standard **1.1.1** e **1.3.0** WMS implementati in QGIS Server forniscono un'interfaccia HTTP per richiedere immagini di mappe o legende generate da un progetto QGIS. Una tipica richiesta WMS definisce il progetto QGIS da usare, i layer da visualizzare e il formato dell'immagine da generare. Il supporto di base è disponibile anche per **Styled Layer Descriptor (SLD)**.

Specifiche:

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

Richieste standard fornite da QGIS Server:

Request	Descrizione
<i>GetCapabilities</i>	Restituisce metadati XML con informazioni sul server
<i>GetMap</i>	Restituisce una mappa
<i>GetFeatureInfo</i>	Recupera i dati (geometria e valori) per una posizione di pixel
<i>GetLegendGraphics</i>	Restituisce i simboli della legenda
<i>GetStyle(s)</i>	Restituisce un documento XML con la descrizione dello stile in SLD
<i>DescribeLayer</i>	Restituisce informazioni sulla disponibilità di WFS e WCS rispettivamente per i layer vettoriali e raster.

Richieste del fornitore date da QGIS Server:

Request	Descrizione
<i>GetPrint</i>	Restituisce un layout QGIS
<i>GetProjectSettings</i>	Restituisce informazioni specifiche su QGIS Server
<i>GetSchemaExtension</i>	Restituisce metadati XML sulle capabilities estese opzionali.

3.2.1 GetCapabilities

Parametri standard per la richiesta **GetCapabilities** in base alle specifiche OGC WMS 1.1.1 e 1.3.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetCapabilities)
<i>VERSION</i>	No	Versione del servizio

La richiesta **GetCapabilities** supporta anche i seguenti parametri del fornitore:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di URL:

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

3.2.2 GetMap

Parametri standard per la richiesta **GetMap** secondo le specifiche OGC WMS 1.1.1 e 1.3.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetMap)
<i>VERSION</i>	Sì	Versione del servizio
<i>LAYERS</i>	No	Layer da visualizzare
<i>STYLES</i>	No	Stile del layers
<i>SRS / CRS</i>	Sì	Sistema di riferimento delle Coordinate
<i>BBOX</i>	Sì	Estensione Mappa
<i>WIDTH</i>	Sì	Larghezza dell'immagine in pixel
<i>HEIGHT</i>	Sì	Altezza dell'immagine in pixel
<i>FORMAT</i>	No	Formato immagine
<i>TRANSPARENT</i>	No	Sfondo trasparente
<i>SLD</i>	No	URL di un SLD da usare per lo stile
<i>SLD_BODY</i>	No	SLD in linea (XML) da usare per lo stile

Oltre a quelli standard, QGIS Server supporta *redlining*, *external WMS layers* e i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS
<i>BGCOLOR</i>	No	Specifica del colore di sfondo
<i>DPI</i>	No	Specifica della risoluzione in uscita
<i>IMAGE_QUALITY</i>	No	Compressione JPEG
<i>OPACITIES</i>	No	Opacità per il layer o il gruppo
<i>FILTER</i>	No	Sottoinsieme di elementi
<i>SELECTION</i>	No	Elementi principali
<i>FILE_NAME</i>	No	Nome del file scaricato Solo per <i>FORMAT=application/dxf</i>
<i>FORMAT_OPTIONS</i>	No	Opzioni del formato di file specificato Solo per <i>FORMAT=application/dxf</i>
<i>TILED</i>	No	Lavorare in <i>modalità tassellata</i>

Esempio di URL:

```
http://localhost/qgis_server?
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

Questo parametro permette di specificare la versione del servizio da utilizzare. I valori disponibili per il parametro VERSION sono:

- 1.1.1
- 1.3.0

A seconda del numero di versione, ci si devono aspettare leggere differenze come spiegato più avanti per i prossimi parametri:

- CRS / SRS
- BBOX

LAYERS

Questo parametro permette di specificare i layer da visualizzare sulla mappa. I nomi devono essere separati da una virgola.

Inoltre, QGIS Server ha introdotto alcune opzioni per selezionare i layer:

- un *short name*
- il layer id

L'opzione del progetto che consente di selezionare i layer in base al loro id si trova nel menu *QGIS Server ► WMS Capabilities* della finestra di dialogo *Progetto ► Proprietà...*. Seleziona la casella di controllo *Usa gli id dei layer come nomi* per attivare questa opzione.

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

STYLES

Questo parametro può essere usato per specificare lo stile di un layer per il processo di visualizzazione. Gli stili devono essere separati da una virgola. Il nome dello stile predefinito è `default`.

SRS / CRS

Questo parametro permette di indicare il sistema di riferimento spaziale in uscita dalla mappa in WMS **1.1.1** e deve essere formulato come `EPSG:XXXX`. Nota che SR è supportato anche se la versione corrente è **1.1.1**.

Per WMS **1.3.0**, il parametro SR è preferibile ma è supportato anche SRS.

Nota che se entrambi i parametri SR e SRS sono indicati nella richiesta, allora è la versione corrente indicata nel parametro VERSION che è decisiva.

Nel prossimo caso, il parametro SRS è mantenuto qualunque sia il parametro VERSION perché SR non è indicato:

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

Nel prossimo caso, il parametro SRS viene tenuto al posto di SR a causa del parametro VERSION:

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

Nel prossimo caso, il parametro SR viene tenuto al posto di SRS a causa del parametro VERSION:

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

BBOX

Questo parametro permette di specificare l'estensione della mappa con unità secondo il SR corrente. Le coordinate devono essere separate da una virgola.

Il parametro BBOX è formato come `min_a,min_b,max_a,max_b` ma la definizione degli assi a e b è diversa secondo il parametro corrente VERSION:

- in WMS **1.1.1**, la disposizione degli assi è sempre est/nord
- in WMS **1.3.0**, la disposizione degli assi dipende dall'autorità del SR

Per esempio nel caso di EPSG:4326 e WMS **1.1.1**, a è la longitudine (est) e b` la latitudine (nord), portando a una richiesta come:

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

Ma nel caso di WMS **1.3.0**, l'ordine degli assi definito nel database EPSG è nord/est quindi a è la latitudine e b` la longitudine:

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

WIDTH

Questo parametro permette di specificare la larghezza in pixel dell'immagine in uscita.

HEIGHT

Questo parametro permette di specificare l'altezza in pixel dell'immagine in uscita.

FORMAT

Questo parametro può essere usato per specificare il formato dell'immagine della mappa. I valori disponibili sono:

- jpg
- jpeg
- image/jpeg
- image/png
- image/png; mode=1bit
- image/png; mode=8bit
- image/png; mode=16bit
- application/dxf: solo i layer che hanno accesso in lettura al servizio WFS vengono esportati in formato DXF.

Esempio di URL:

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

TRANSPARENT

Questo parametro booleano può essere usato per specificare la trasparenza dello sfondo. I valori disponibili sono (non sensibili alle maiuscole):

- TRUE
- FALSE

Tuttavia, questo parametro viene ignorato se il formato dell'immagine indicato con `FORMAT` è diverso da PNG.

BGCOLOR

Questo parametro permette di indicare un colore di sfondo per l'immagine della mappa. Tuttavia non può essere combinato con il parametro `TRANSPARENT` in caso di immagini PNG (la trasparenza ha la priorità). Il colore può essere letterale o in notazione esadecimale.

Esempio di URL con la notazione letterale:

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

Esempio di URL con la notazione esadecimale:

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

DPI

Questo parametro può essere usato per specificare la risoluzione in uscita richiesta.

IMAGE_QUALITY

Questo parametro è usato solo per le immagini JPEG. Per impostazione predefinita, la compressione JPEG è -1.

Puoi cambiare il default per ogni progetto QGIS nel menu *OWS Server ► WMS capabilities* della finestra di dialogo *Progetto ► Proprietà....* Se vuoi sovrascriverlo in una richiesta `GetMap` puoi farlo usando il parametro `IMAGE_QUALITY`.

OPACITIES

Elenco separato da virgole di valori di opacità. L'opacità può essere impostata a livello di layer o di gruppo. I valori consentiti vanno da 0 (completamente trasparente) a 255 (completamente opaco).

Esempio di URL:

```
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
```



Fig. 3.2: Sulla sinistra OPACITIES=255, 0 e sulle destra OPACITIES=255, 255

FILTER

Un sottoinsieme di layer può essere selezionato con il parametro `FILTER`. La sintassi è fondamentalmente la stessa della stringa del sottoinsieme di QGIS. Tuttavia, ci sono alcune restrizioni per evitare introduzioni SQL nei database tramite QGIS Server. Se viene trovata una stringa pericolosa nel parametro, QGIS Server restituirà l'errore successivo:

```
<ServiceExceptionReport>
  <ServiceException code="Security">The filter string XXXXXXXXX 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>
```

Esempio di URL:

```
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'
```




Fig. 3.3: Risposta del server a una richiesta GetMap con il parametro FILTER

In questo esempio, lo stesso filtro `"name" = 'France'` viene applicato ai layer `countries` e `countries_shapeburst`, mentre il filtro `"name" = 'Paris'` viene applicato solo a `places`.

Nota: È possibile fare ricerche di attributi tramite `GetFeatureInfo` e omettere il parametro `X/Y` se c'è un `FILTER`. QGIS Server restituisce quindi le informazioni sugli elementi corrispondenti e genera un perimetro di delimitazione combinato nell'XML in uscita.

SELECTION

Il parametro `SELECTION` può evidenziare elementi da uno o più layer. Gli elementi vettoriali possono essere selezionati passandogli liste separate da virgole con gli id degli elementi.

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

L'immagine seguente presenta la risposta di una richiesta `GetMap` utilizzando l'opzione `SELECTION`, ad esempio `http://myserver.com/...&SELECTION=countries:171,65`.

Poiché questi id degli elementi corrispondono nell'insieme dei dati di origine a **France** e **Romania** sono evidenziati in giallo.



Fig. 3.4: Risposta del server a una richiesta GetMap con il parametro SELECTION

FORMAT-OPTIONS

Questo parametro può essere usato per specificare le opzioni per il formato selezionato. Solo per `FORMAT=application/dxf`. Una lista di coppie chiave:valore separate da punto e virgola:

- **SCALE:** da usare per le regole della simbologia, i filtri e gli stili (non la scalatura effettiva dei dati - i dati rimangono nella scala originale).
- **MODE:** corrisponde alle opzioni di esportazione offerte nella finestra di dialogo di esportazione DXF di QGIS Desktop. I valori possibili sono `NOSYMBOLLOGY`, `FEATURESYMBOLLOGY` e `SYMBOLLAYERSYMBOLLOGY`.
- **LAYERSATTRIBUTES:** specifica un campo che contiene valori per i nomi dei layer DXF - se non specificato, vengono usati i nomi originali dei layer QGIS.
- **USE_TITLE_AS_LAYERNAME:** se abilitato, il titolo del layer sarà usato come nome del layer.
- **CODEC:** specifica un codec da usare per la codifica. Il valore predefinito è `ISO-8859-1`, verificare la finestra di dialogo di esportazione DXF di QGIS desktop per i valori validi.
- **NO_MTEXT:** usa `TEXT` invece di `MTEXT` per le etichette.
- **FORCE_2D:** Forza il risultato in 2D. Questo è richiesto per la larghezza della polilinea.

TILED

Per ragioni di performance, QGIS Server può essere usato in modalità tiled. In questa modalità, il client richiede diverse piccoli tasselli di dimensioni fisse, e le assembla per formare l'intera mappa. Facendo questo, i simboli al o vicino al confine tra due tasselli possono apparire tagliati, perché sono presenti solo in uno dei tasselli.

Imposta il parametro `TILED` a `TRUE` per dire a QGIS Server di lavorare in modalità *tiled*, e di applicare il *Tassello buffer* configurato nel progetto QGIS (vedi [Configura il tuo progetto](#)).

Quando `TILED` è `TRUE` e quando un Tassello buffer non nullo è configurato nel progetto QGIS, gli elementi al di fuori dell'estensione del tassello sono disegnati per evitare simboli di taglio ai confini del tassello.

Per default, `TILED` è impostato su `FALSE`.

3.2.3 GetFeatureInfo

Parametri standard per la richiesta **GetFeatureInfo** secondo le specifiche OGC WMS 1.1.1 e 1.3.0:

Parametro	Richiesto	Descrizione
<code>SERVICE</code>	Sì	Nome del servizio (WMS)
<code>REQUEST</code>	Sì	Nome della richiesta (GetFeatureInfo)
<code>VERSION</code>	No	Versione del servizio
<code>QUERY_LAYERS</code>	Sì	Layer su cui fare la query
<code>LAYERS</code>	Sì	Layer da visualizzare (identico a <code>QUERY_LAYERS</code>)
<code>STYLES</code>	No	Stile del layers
<code>SRS</code> / <code>CRS</code>	Sì	Sistema di riferimento delle Coordinate
<code>BBOX</code>	No	Estensione Mappa
<code>WIDTH</code>	Sì	Larghezza dell'immagine in pixel
<code>HEIGHT</code>	Sì	Altezza dell'immagine in pixel
<code>TRANSPARENT</code>	No	Sfondo trasparente
<code>INFO_FORMAT</code>	No	Formato in uscita
<code>FEATURE_COUNT</code>	No	Numero massimo di elementi da restituire
<code>I</code>	No	Colonna pixel del punto da interrogare
<code>X</code>	No	Come il parametro <code>I</code> , ma in WMS 1.1.1
<code>J</code>	No	Riga pixel del punto da interrogare
<code>Y</code>	No	Come il parametro <code>J</code> , ma in WMS 1.1.1
<code>WMS_PRECISION</code>	No	La precisione (numero di cifre) da usare quando si restituisce la geometria (vedi how to add geometry to feature response). Il valore predefinito è -1 che significa che viene usata la precisione definita nel progetto.

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<code>MAP</code>	Sì	File di progetto QGIS
<code>FILTER</code>	No	Sottoinsieme di elementi
<code>FI_POINT_TOLERANCE</code>	No	Tolleranza in pixel per i layer punto
<code>FI_LINE_TOLERANCE</code>	No	Tolleranza in pixel per i layer linea
<code>FI_POLYGON_TOLERANCE</code>	No	Tolleranza in pixel per i layer poligono
<code>FILTER_GEOM</code>	No	Filtrare geometria
<code>WITH_MAPTIP</code>	No	Aggiungere suggerimenti mappa al risultato
<code>WITH_GEOMETRY</code>	No	Aggiungere la geometria al risultato

Esempio di URL:

```
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

Questo parametro può essere usato per specificare il formato del risultato. I valori disponibili sono:

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

QUERY_LAYERS

Questo parametro specifica i layer da visualizzare sulla mappa. I nomi sono separati da una virgola.

Inoltre, QGIS Server introduce opzioni per selezionare i layer in base a:

- nome breve
- layer id

Per ulteriori informazioni, vedi il parametro `LAYERS` definito in *GetMap*.

FEATURE_COUNT

Questo parametro specifica il numero massimo di elementi per layer da restituire. Per esempio se `QUERY_LAYERS` è impostato su `layer1, layer2` e `FEATURE_COUNT` è impostato su 3 allora saranno restituite un massimo di 3 elementi dal layer1. Allo stesso modo verrà restituito un massimo di 3 elementi dal layer2.

Per impostazione predefinita, viene restituito solo 1 elemento per layer.

I

Questo parametro, definito in WMS 1.3.0, permette di specificare la colonna pixel del punto di interrogazione.

X

Stesso parametro di **I**, ma definito in WMS 1.1.1.

J

Questo parametro, definito in WMS 1.3.0, permette di specificare la riga pixel del punto di interrogazione.

Y

Stesso parametro di **J**, ma definito in WMS 1.1.1.

FI_POINT_TOLERANCE

Questo parametro specifica la tolleranza in pixel per i layer punto.

FI_LINE_TOLERANCE

Questo parametro specifica la tolleranza in pixel per i layer linea.

FI_POLYGON_TOLERANCE

Questo parametro specifica la tolleranza in pixel per i layer poligono.

FILTER_GEOM

Questo parametro specifica una geometria WKT con cui gli elementi devono intersecarsi.

WITH_MAPTIP

Questo parametro specifica se aggiungere suggerimenti mappa al risultato.

I valori disponibili sono (non sensibili alle maiuscole):

- TRUE
- FALSE

WITH_GEOMETRY

Questo parametro specifica se aggiungere geometrie al risultato. Per usare questa funzione devi prima abilitare l'opzione *Aggiungi la geometria alla risposta dell'oggetto* nel progetto QGIS. Vedi *Configure your project*.

I valori disponibili sono (non sensibili alle maiuscole):

- TRUE
- FALSE

3.2.4 GetLegendGraphics

Parametri standard per la richiesta **GetLegendGraphics** in base alle specifiche OGC WMS 1.1.1 e 1.3.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetLegendGraphics)
<i>VERSION</i>	No	Versione del servizio
<i>LAYERS</i>	Sì	Layer da visualizzare
<i>STYLES</i>	No	Stile del layers
<i>SRS / CRS</i>	No	Sistema di riferimento delle Coordinate
<i>BBOX</i>	No	Estensione Mappa
<i>WIDTH</i>	No	Larghezza dell'immagine in pixel
<i>HEIGHT</i>	No	Altezza dell'immagine in pixel
<i>FORMAT</i>	No	Formato legenda
<i>TRANSPARENT</i>	No	Sfondo trasparente

Oltre a quelli standard, QGIS Server supporta parametri aggiuntivi per modificare le dimensioni degli elementi della legenda o le proprietà dei caratteri per i titoli dei layer e le etichette degli elementi:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS
<i>SRCWIDTH</i>	No	Larghezza Mappa
<i>SRCHEIGHT</i>	No	Altezza Mappa
<i>SHOWFEATURECOUNT</i>	No	Aggiungere il conteggio degli elementi
<i>RULE</i>	No	Regola simbolo da visualizzare
<i>RULELABEL</i>	No	Etichette degli elementi da visualizzare
<i>BOXSPACE</i>	No	Spazio tra la cornice della legenda e il contenuto (mm)
<i>LAYERSPACE</i>	No	Spazio verticale tra i layer (mm)
<i>LAYERTITLESPACE</i>	No	Spazio verticale tra il titolo del layer e gli elementi (mm)
<i>SYMBOLSPACE</i>	No	Spazio verticale tra simbolo e elementi (mm)
<i>ICONLABELSPACE</i>	No	Spazio orizzontale tra simbolo ed etichetta (mm)
<i>SYMBOLWIDTH</i>	No	Larghezza dell'anteprima del simbolo (mm)
<i>SYMBOLHEIGHT</i>	No	Altezza dell'anteprima del simbolo (mm)
<i>LAYERTITLE</i>	No	Visualizzazione del titolo del layer
<i>LAYERFONTFAMILY</i>	No	Famiglia carattere Layer
<i>LAYERFONTBOLD</i>	No	Visualizzazione in grassetto del titolo del layer
<i>LAYERFONTSIZE</i>	No	Dimensione carattere del titolo del layer (pt)
<i>LAYERFONTITALIC</i>	No	Titolo del layer in corsivo
<i>LAYERFONTCOLOR</i>	No	Colore titolo del layer
<i>ITEMFONTFAMILY</i>	No	Famiglia carattere elemento
<i>ITEMFONTBOLD</i>	No	Visualizzazione in grassetto etichetta elemento
<i>ITEMFONTSIZE</i>	No	Dimensione carattere etichetta elemento (pt)
<i>ITEMFONTITALIC</i>	No	Visualizzazione in corsivo etichetta elemento
<i>ITEMFONTCOLOR</i>	No	Colore etichetta elemento

BBOX

Questo parametro può essere utilizzato per specificare l'area geografica per la quale deve essere costruita la legenda (il suo formato è descritto [here](#)), ma non può essere combinato con il parametro RULE. Il parametro SRS/CRS diventa obbligatorio quando si utilizza il parametro BBOX.

Esempio di URL:

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

Nota: Quando il parametro BBOX è definito, la legenda viene definita come *legenda basata sul contenuto*.

WIDTH

Questo parametro non è utilizzato di default, ma diventa obbligatorio quando viene impostato il parametro RULE. In questo caso, consente di specificare la larghezza in pixel dell'immagine in uscita.

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

HEIGHT

Questo parametro non è utilizzato di default, ma diventa obbligatorio quando viene impostato il parametro RULE. In questo caso, consente di specificare l'altezza in pixel dell'immagine in uscita.

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

FORMAT

Questo parametro può essere utilizzato per specificare il formato dell'immagine della legenda. I valori disponibili sono:

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

Per JSON, i simboli sono codificati con Base64 e la maggior parte delle altre opzioni relative al layout o ai font non sono prese in considerazione, perché la legenda deve essere costruita sul lato client. Il parametro "RULE" non può essere combinato con questo formato.

Esempio di URL con l'output JSON corrispondente:

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

E il corrispondente output JSON:

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

SRCWIDTH

Questo parametro può essere definito quando è impostato il parametro `RULE`. In questo caso, il valore `SRCWIDTH` viene inoltrato alla richiesta `GetMap` sottostante come parametro `WIDTH`, mentre il parametro `WIDTH` di `GetLegendGraphics` viene usato per la dimensione della legenda dell'immagine.

SRCHEIGHT

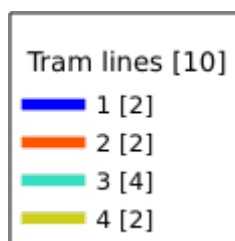
Questo parametro può essere definito quando il parametro `RULE` è impostato. In questo caso, il valore `SRCHEIGHT` viene inoltrato alla richiesta `GetMap` sottostante come parametro `HEIGHT`, mentre il parametro `HEIGHT` di `GetLegendGraphics` viene usato per la dimensione della legenda dell'immagine.

SHOWFEATURECOUNT

Questo parametro può essere utilizzato per attivare il conteggio degli elementi nella legenda. I valori disponibili sono (senza distinzione tra maiuscole e minuscole):

- TRUE
- FALSE

Per esempio:



RULE

Questo parametro è disponibile sui layer con restituzione *Rule-based* e consente di costruire una legenda con il solo simbolo della regola indicata. Non può essere combinato con il parametro BBOX né con il formato JSON.

Esempio di URL:

```
http://localhost/qgisserver?
SERVICE=WMS
&REQUEST=GetLegendGraphics
&LAYERS=mylayer,
&RULE=myrulename
```

RULELABEL

This parameter allows to control the item label rendering. Available values are (not case sensitive):

- TRUE: visualizza etichetta elemento
- FALSE: nasconde etichetta elemento
- AUTO: nasconde l'etichetta dell'elemento per i layer con visualizzazione *Singolo simbolo*.

Esempio di URL:

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



Fig. 3.5: Visualizzazione della legenda senza etichetta per i layer simbolo singolo

BOXSPACE

Questo parametro consente di specificare lo spazio tra la cornice della legenda e il contenuto in millimetri. Per impostazione predefinita, il valore dello spazio è 2 mm.

Esempio di URL:

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



Fig. 3.6: A sinistra BOXSPACE=0 e a destra BOXSPACE=15.

LAYERSPACE

Questo parametro consente di specificare lo spazio verticale tra i layer in millimetri. Per impostazione predefinita, il valore dello spazio è 3 mm.

Esempio di URL:

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



Fig. 3.7: A sinistra LAYERSPACE=0 e a destra LAYERSPACE=10.

LAYERTITLESPACE

Questo parametro consente di specificare lo spazio verticale tra il titolo del layer e gli elementi successivi, in millimetri. Per impostazione predefinita, il valore dello spazio è 3 mm.

Esempio di URL:

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

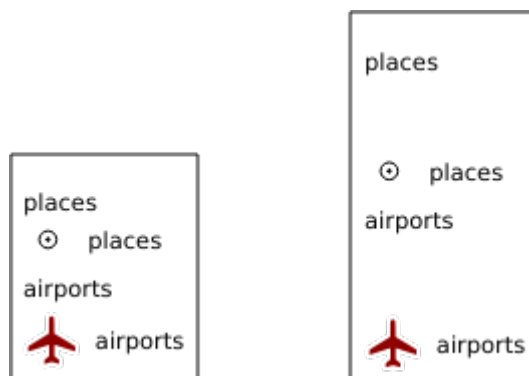


Fig. 3.8: A sinistra LAYERTITLESPACE=0 e a destra LAYERTITLESPACE=10.

SYMBOLSPACE

Questo parametro consente di specificare lo spazio verticale in millimetri tra il simbolo e l'elemento successivo. Per impostazione predefinita, il valore dello spazio è 2 mm.

Esempio di URL:

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

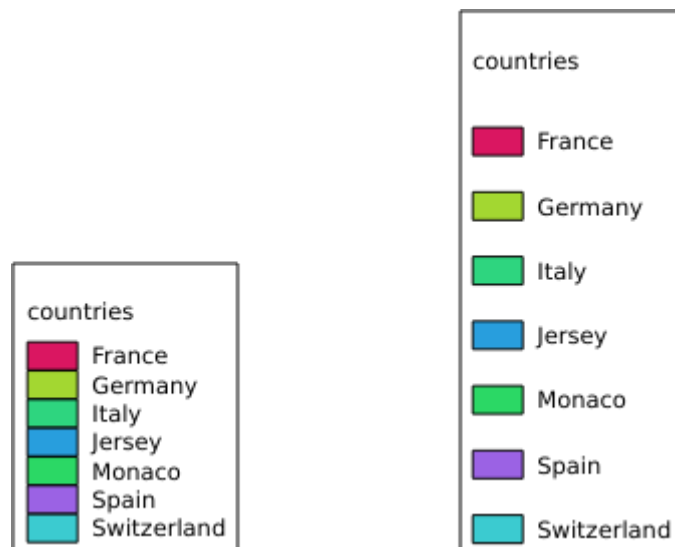


Fig. 3.9: A sinistra SYMBOLSPACE=0 e a destra SYMBOLSPACE=5.

ICONLABELSPACE

Questo parametro consente di specificare lo spazio orizzontale tra il simbolo e il testo dell'etichetta in millimetri. Per impostazione predefinita, il valore dello spazio è 2 mm.

Esempio di URL:

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



Fig. 3.10: A sinistra ICONLABELSPACE=0 e a destra ICONLABELSPACE=10.

SYMBOLWIDTH

Questo parametro consente di specificare la larghezza dell'anteprima del simbolo in millimetri. Per impostazione predefinita, il valore della larghezza è 7 mm.

Esempio di URL:

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



Fig. 3.11: A sinistra SYMBOLWIDTH=2 e a destra SYMBOLWIDTH=20.

SYMBOLHEIGHT

Questo parametro consente di specificare l'altezza dell'anteprima del simbolo in millimetri. Per impostazione predefinita, il valore dell'altezza è 4 mm.

Esempio di URL:

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



Fig. 3.12: A sinistra SYMBOLHEIGHT=2 e a destra SYMBOLHEIGHT=6.

LAYERTITLE

Questo parametro specifica se visualizzare il titolo del layer.

I valori disponibili sono (non sensibili alle maiuscole):

- TRUE (valore predefinito)
- FALSE

LAYERFONTFAMILY

Questo parametro specifica la famiglia di caratteri da utilizzare per visualizzare il titolo del layer.

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

LAYERFONTBOLD

Questo parametro specifica se il titolo del layer viene visualizzato in grassetto. I valori disponibili sono (senza distinzione tra maiuscole e minuscole):

- TRUE
- FALSE

Esempio di URL:

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



Fig. 3.13: Legenda con LAYERFONTBOLD=TRUE

LAYERFONTSIZE

Questo parametro specifica la dimensione carattere per visualizzare il titolo del layer in punti.

Esempio di URL:

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

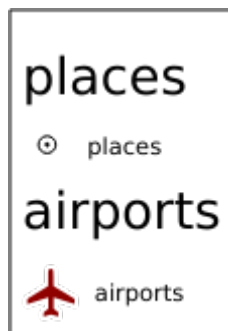


Fig. 3.14: Legenda con LAYERFONTSIZE=20

LAYERFONTITALIC

Questo parametro specifica se il titolo del layer viene visualizzato in corsivo. I valori disponibili sono (senza distinzione tra maiuscole e minuscole):

- TRUE
- FALSE

Esempio di URL:

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

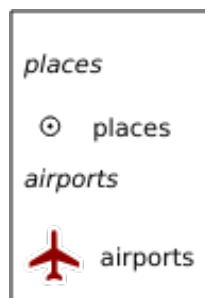


Fig. 3.15: Legenda con LAYERFONTITALIC=TRUE

LAYERFONTCOLOR

Questo parametro specifica il colore del titolo del layer. Il colore può essere letterale (red, green, ..) o in notazione esadecimale (0xFF0000, 0x00FF00, ...).

Esempio di URL:

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

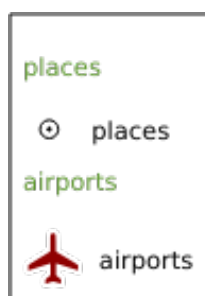


Fig. 3.16: Legenda con LAYERFONTCOLOR=0x5f9930

ITEMFONTFAMILY

Questo parametro specifica la famiglia di caratteri da utilizzare per la visualizzazione dell'etichetta dell'elemento.

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


ITEMFONTBOLD

Questo parametro specifica se l'etichetta dell'elemento viene visualizzata in grassetto. I valori disponibili sono (senza distinzione tra maiuscole e minuscole):

- TRUE
- FALSE

Esempio di URL:

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

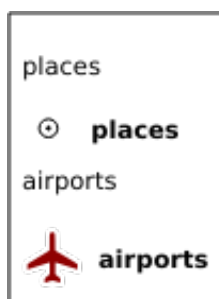


Fig. 3.17: Legenda con ITEMFONTBOLD=TRUE

ITEMFONTSIZE

Questo parametro specifica la dimensione carattere per visualizzare il titolo del layer in punti.

Esempio di URL:

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



Fig. 3.18: Legenda con ITEMFONTSIZE=30

ITEMFONTITALIC

Questo parametro specifica se l'etichetta dell'elemento viene visualizzata in corsivo. I valori disponibili sono (senza distinzione tra maiuscole e minuscole):

- TRUE
- FALSE

Esempio di URL:

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

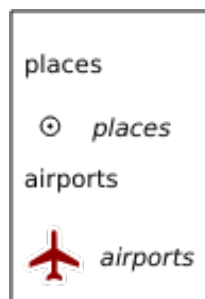


Fig. 3.19: Legenda con ITEMFONTITALIC=TRUE

ITEMFONTCOLOR

Questo parametro specifica il colore dell'etichetta dell'elemento. Il colore può essere letterale (red, green, ..) o in notazione esadecimale (0xFF0000, 0x00FF00, ...).

Esempio di URL:

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

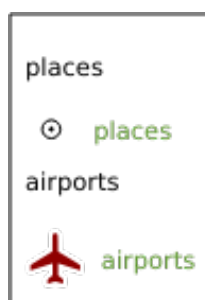


Fig. 3.20: Legenda con ITEMFONTCOLOR=0x5f9930

3.2.5 GetStyle(s)

Parametri standard per la richiesta **GetStyle** (o **GetStyles**) in base alle specifiche OGC WMS 1.1.1:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della request (GetStyle o GetStyles)
<i>LAYERS</i>	Sì	Layer su cui fare la query

La request **GetStyle** supporta anche i seguenti parametri del fornitore:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di URL:

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

3.2.6 DescribeLayer

Parametri standard per la richiesta **DescribeLayer** secondo le specifiche OGC WMS 1.1.1 e 1.3.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della request (DescribeLayer)
<i>LAYERS</i>	Sì	Layer da descrivere
<i>SLD_VERSION</i>	Sì	Versione SLD

La richiesta **DescribeLayer** supporta anche i seguenti parametri del fornitore:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di URL:

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

Il documento XML si presenta come:

```
<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>
  <LayerDescription>
    <owsType>wfs</owsType>
    <se:OnlineResource xlink:href="http://localhost/qgis_server" xlink:type=
↪"simple"/>
  </LayerDescription>
</DescribeLayerResponse>
```

(continues on next page)

```

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

```

SLD_VERSION

Questo parametro consente di specificare la versione di SLD. È disponibile solo il valore 1.1.0.

3.2.7 GetPrint

QGIS Server ha la capacità di creare output di layout di stampa in formato pdf o pixel. Le finestre di layout di stampa nel progetto pubblicato sono usate come modelli. Nella richiesta **GetPrint**, il client ha la possibilità di specificare i parametri delle mappe e delle etichette del layout contenuto.

La richiesta **GetPrint** supporta *redlining*, *external WMS layers* come pure i seguenti parametri:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetPrint)
<i>VERSION</i>	No	Versione del servizio
<i>LAYERS</i>	No	Layer da visualizzare
<i>TEMPLATE</i>	Sì	Modello di layout da utilizzare
<i>SRS / CRS</i>	Sì	Sistema di riferimento delle Coordinate
<i>FORMAT</i>	No	Formato in uscita
<i>ATLAS_PK</i>	No	Caratteristiche dell'Atlante
<i>STYLES</i>	No	Stile del layers
<i>TRANSPARENT</i>	No	Sfondo trasparente
<i>OPACITIES</i>	No	Opacità per il layer o il gruppo
<i>SELECTION</i>	No	Elementi principali
<i>mapX:EXTENT</i>	No	Estensione della mappa "X"
<i>mapX:LAYERS</i>	No	Layer della mappa "X"
<i>mapX:STYLES</i>	No	Stile dei layer della mappa "X"
<i>mapX:SCALE</i>	No	Scala dei layer della mappa "X"
<i>mapX:ROTATION</i>	No	Rotazione della mappa "X"
<i>mapX:GRID_INTERVAL_X</i>	No	Intervallo del reticolo sull'asse x della mappa "X"
<i>mapX:GRID_INTERVAL_Y</i>	No	Intervallo del reticolo sull'asse y della mappa "Y"

Esempio di URL:

```

http://localhost/qgisserver?
SERVICE=WMS
&VERSION=1.3.0
&REQUEST=GetPrint
&MAP=/home/qgis/projects/world.qgs

```

(continues on next page)

(continua dalla pagina precedente)

```
&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
```

Nota che il modello di layout può contenere più di una mappa. In questo modo, se vuoi configurare una mappa specifica, devi usare i parametri mappaX: dove X è un numero positivo che puoi recuperare grazie alla richiesta **GetProjectSettings**.

Per esempio:

```
<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

Questo parametro può essere utilizzato per specificare il nome di un modello di layout da utilizzare per la stampa.

FORMAT

Questo parametro specifica il formato dell'immagine della mappa. I valori disponibili sono:

- png (default value)
- image/png
- jpg
- jpeg
- image/jpeg
- svg
- image/svg
- image/svg+xml
- pdf
- application/pdf

Se il parametro **FORMAT** è diverso da uno di questi valori, allora viene restituita una eccezione.

ATLAS_PK

Questo parametro permette di attivare la visualizzazione dell'atlante indicando quali caratteristiche vogliamo stampare. Per ottenere un atlante con tutte gli elementi, si può usare il simbolo * (secondo il numero massimo di elementi consentito nella configurazione del progetto).

Quando il `FORMAT` è `pdf`, viene restituito un singolo documento PDF che combina le pagine degli elementi. Per tutti gli altri formati, viene restituita una singola pagina.

mapX:EXTENT

Questo parametro specifica l'estensione per un oggetto della mappa di layout come `xmin,ymin,xmax,ymax`.

mapX:ROTATION

Questo parametro specifica la rotazione della mappa in gradi.

mapX:GRID_INTERVAL_X

Questo parametro specifica la densità delle linee del reticolo nella direzione X.

mapX:GRID_INTERVAL_Y

Questo parametro specifica la densità delle linee del reticolo nella direzione Y.

mapX:SCALE

Questo parametro specifica la scala della mappa per un oggetto mappa del layout. Questo è utile per assicurare la visibilità dei layer e delle etichette in base alla scala, anche se client e server possono avere algoritmi diversi per calcolare il denominatore della scala.

mapX:LAYERS

Questo parametro specifica i layer per un elemento layout mappa. Per ulteriori informazioni su questo parametro, vedere *GetMap Layers*.

mapX:STYLES

Questo parametro specifica gli stili dei layer definiti in uno specifico elemento della mappa di layout. Per ulteriori informazioni su questo parametro, vedere *GetMap Styles*.

3.2.8 GetProjectSettings

Questo tipo di richiesta funziona in modo simile a *GetCapabilities*, ma è più specifico per QGIS Server e consente a un client di leggere informazioni aggiuntive che non sono disponibili nell'output *GetCapabilities*:

- visibilità iniziale dei layer
- informazione sugli attributi vettoriali e i loro tipi di edit
- informazione sull'ordine dei layer e l'ordine di rappresentazione
- elenco dei layer pubblicati in WFS
- mostra se un gruppo nell'albero dei layer è mutuamente esclusiva

La richiesta **GetProjectSettings** supporta i seguenti parametri:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetProjectSettings)

3.2.9 GetSchemaExtension

La richiesta **GetSchemaExtension** consente di recuperare capabilities e operazioni estese opzionali del servizio WMS, come quelle implementate da QGIS Server.

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS
<i>SERVICE</i>	Sì	Nome del servizio (WMS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetSchemaExtension)

Esempio di URL:

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

Il documento XML si presenta come:

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

3.2.10 Layer WMS esterni

QGIS Server consente di includere layer da server WMS esterni nelle richieste WMS *GetMap* e WMS *GetPrint*. Ciò è particolarmente utile se un client web utilizza un layer di sfondo esterno nella mappa web. Per ragioni di prestazioni, tali layer dovrebbero essere richiesti direttamente dal client web (non in cascata tramite il server QGIS). Per la stampa, invece, questi layer devono essere collegati in cascata tramite il server QGIS per apparire nella mappa stampata.

I layer esterni possono essere aggiunti al parametro **LAYERS** come **EXTERNAL_WMS:<layername>**. I parametri per i layer WMS esterni (ad esempio url, format, dpiMode, crs, layers, styles) possono essere forniti in seguito come parametri di servizio **<layername>:<parameter>**. In una richiesta *GetMap*, questo potrebbe apparire come segue:

```
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
```

(continues on next page)

(continua dalla pagina precedente)

```
&basemap:layers=orthofoto
&basemap:styles=default
```

Allo stesso modo, i layer esterni possono essere utilizzati nelle richieste *GetPrint*:

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

3.2.11 Redlining

Questa funzione è disponibile e può essere utilizzata con le richieste *GetMap* e *GetPrint*.

La funzione redlining può essere usata per passare geometrie ed etichette nella richiesta che vengono sovrapposte dal server sull'immagine standard restituita (mappa). Questo permette all'utente di mettere l'accento o magari aggiungere dei commenti (etichette) ad alcune aree, luoghi ecc. che non sono nella mappa standard.

La richiesta *GetMap* è nel 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
```

L'equivalente di *GetPrint* è nel formato (nota che il parametro mapX: è aggiunto per indicare quale mappa ha il redlining):

```
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>
```

(continues on next page)

(continua dalla pagina precedente)

```
&map0:HIGHLIGHT_LABELSTRING=Write label here
&map0:HIGHLIGHT_LABELSIZE=16
&map0:HIGHLIGHT_LABELCOLOR=%23000000
&map0:HIGHLIGHT_LABELBUFFERCOLOR=%23FFFFFF
&map0:HIGHLIGHT_LABELBUFFERSIZE=1.5
```

Ecco l'immagine prodotta dalla richiesta di cui sopra in cui un poligono e un'etichetta sono disegnati sopra la mappa normale:



Fig. 3.21: Risposta del server a una richiesta GetMap con parametri di redlining

Puoi vedere che ci sono diversi parametri in questa richiesta:

- **HIGHLIGHT_GEOM:** Puoi aggiungere POINT, MULTILINESTRING, POLYGON ecc. Supporta le geometrie multiparte. Ecco un esempio: `HIGHLIGHT_GEOM=MULTILINESTRING((0 0 0, 0 1, 1 1))`. Le coordinate dovrebbero essere nel SR della richiesta GetMap/GetPrint.
- **HIGHLIGHT_SYMBOL:** Questo controlla come la geometria è evidenziata e si può cambiare la larghezza del tratto, il colore e l'opacità.
- **HIGHLIGHT_LABELSTRING:** Puoi passare il tuo testo di etichettatura a questo parametro.
- **HIGHLIGHT_LABELSIZE:** Questo parametro controlla la dimensione dell'etichetta.
- **HIGHLIGHT_LABELFONT:** Questo parametro controlla il carattere dell'etichetta (ad esempio Arial)
- **HIGHLIGHT_LABELCOLOR:** Questo parametro controlla il colore dell'etichetta.

- **HIGHLIGHT_LABELBUFFERCOLOR**: Questo parametro controlla il colore del buffer dell'etichetta.
- **HIGHLIGHT_LABELBUFFERSIZE**: Questo parametro controlla la dimensione del buffer dell'etichetta.

3.3 Web Feature Service (WFS)

Gli standard **1.0.0** e **1.1.0** WFS implementati in QGIS Server forniscono un'interfaccia HTTP per interrogare gli elementi geografici da un progetto QGIS. Una tipica richiesta WFS definisce il progetto QGIS da usare e il layer da interrogare.

Documento delle specifiche secondo il numero di versione del servizio:

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

Richieste standard fornite da QGIS Server:

Request	Descrizione
<i>GetCapabilities</i>	Restituisce metadati XML con informazioni sul server
<i>GetFeature</i>	Restituisce una selezione di elementi
<i>DescribeFeatureType</i>	Restituisce una descrizione dei tipi di elementi e delle proprietà
<i>Transaction</i>	Permette l'inserimento, l'aggiornamento o la cancellazione di elementi

3.3.1 GetCapabilities

Parametri standard per la richiesta **GetCapabilities** in conformità alle specifiche OGC WFS 1.0.0 e 1.1.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WFS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetCapabilities)
<i>VERSION</i>	No	Versione del servizio

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

VERSION

Questo parametro permette di specificare la versione del servizio da utilizzare. I valori disponibili per il parametro **VERSION** sono:

- 1.0.0
- 1.1.0

Se nessuna versione è indicata nella richiesta, allora 1.1.0 è usato per default.

Esempio di URL:

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

3.3.2 GetFeature

Parametri standard per la richiesta **GetFeature** secondo le specifiche OGC WFS 1.0.0 e 1.1.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WFS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetFeature)
<i>VERSION</i>	No	Versione del servizio
<i>TYPENAME</i>	No	Nome dei layer
<i>FEATUREID</i>	No	Filtrare gli elementi per id
<i>OUTPUTFORMAT</i>	No	Formato risultato
<i>RESULTTYPE</i>	No	Tipo di risultato
<i>PROPERTYNAME</i>	No	Nome delle proprietà da restituire
<i>MAXFEATURES</i>	No	Numero massimo di elementi da restituire
<i>SRSNAME</i>	No	Sistema di riferimento delle Coordinate
<i>FILTER</i>	No	Codifica filtro OGC
<i>BBOX</i>	No	Estensione mappa
<i>SORTBY</i>	No	Ordinare i risultati

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS
<i>STARTINDEX</i>	No	Paginazione
<i>GEOMETRYNAME</i>	No	Tipo di geometria da restituire
<i>EXP_FILTER</i>	No	Espressione di filtraggio

TYPENAME

Questo parametro consente di specificare i nomi dei layer ed è obbligatorio se *FEATUREID* non è impostato.

Esempio di URL:

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

FEATUREID

Questo parametro consente di specificare l'ID di un elemento specifico e ha una forma del tipo `typename.fid`, `typename.fid, ...`.

Esempio di URL:

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

Risposta 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

Questo parametro può essere usato per specificare il formato della risposta. Se `VERSION` è maggiore o uguale a `1.1.0`, il formato predefinito è GML3. Altrimenti viene utilizzato GML2.

Valori possibili sono:

- 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

Esempio di URL:

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

Risposta GeoJSON:

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

(continues on next page)

(continua dalla pagina precedente)

```

-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

Questo parametro può essere usato per specificare il tipo di risultato da restituire. I valori disponibili sono:

- **results**: il risultato predefinito
- **hits**: restituisce solo un conteggio degli elementi

Esempio di URL:

```

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

```

PROPERTYNAME

Questo parametro può essere utilizzato per specificare una proprietà specifica da restituire. Una proprietà deve essere mappata con un **TYPENAME** o un **FEATUREID**:

Esempio di URL valido:

```

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

```

Al contrario, il seguente URL restituirà un'eccezione:

```
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

Questo parametro consente di limitare il numero di elementi restituiti dalla richiesta.

Nota: Questo parametro può essere utile per migliorare le prestazioni quando i layer vettoriali sottostanti sono pesanti.

SRSNAME

Questo parametro consente di indicare il sistema di riferimento spaziale in uscita dalla risposta e anche il SR BBOX e deve essere formato come EPSG:XXXX.

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

FILTER

Questo parametro consente di filtrare la risposta con il linguaggio **Filter Encoding** definito dallo standard **OGC Filter Encoding**.

Esempio di URL:

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

BBOX

Questo parametro permette di specificare l'estensione della mappa con unità secondo il SR corrente. Le coordinate devono essere separate da una virgola.

Il parametro SRSNAME può specificare il SR dell'estensione. Se non viene specificato, viene utilizzato il SR del layer.

Esempio di URL:

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

Il parametro FEATUREID non può essere utilizzato con la BBOX. Qualsiasi tentativo provocherà un'eccezione:

```
<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

Questo parametro consente di ordinare gli elementi risultanti in base ai valori delle proprietà e deve essere formato come `propertyname SORTRULE`.

Valori possibili per SORTRULE in caso di ordinamento discendente:

- D
- +D
- DESC
- +DESC

Valori possibili per SORTRULE in caso di ordinamento ascendente:

- A
- +A
- ASC
- +ASC

Esempio di URL:

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

Il risultato ottenuto:

```
<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>
```

(continues on next page)

(continua dalla pagina precedente)

```
</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

Questo parametro può essere usato per specificare il tipo di geometria da restituire per gli elementi. I valori disponibili sono:

- extent
- centroid
- none

Esempio di URL:

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

STARTINDEX

Questo parametro è standard in WFS 2.0, ma è un'estensione per WFS 1.0.0.

In pratica, può essere usato per saltare alcuni elementi dell'insieme dei risultati e, in combinazione con MAXFEATURES, fornisce la possibilità di sfogliare i risultati.

Esempio di URL:

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


EXP_FILTER

Questo parametro consente di filtrare la risposta con le espressioni di QGIS.

Esempio di URL:

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

3.3.3 DescribeFeatureType

Parametri standard per la richiesta **DescribeFeatureType** secondo le specifiche OGC WFS 1.0.0 e 1.1.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WFS)
<i>REQUEST</i>	Sì	Nome della richiesta (DescribeFeatureType)
<i>VERSION</i>	No	Versione del servizio
<i>OUTPUTFORMAT</i>	No	Formato della risposta
<i>TYPENAME</i>	No	Nome dei layer

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di URL:

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

Risposta in uscita:

```
<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 Transazione

Questa richiesta consente di aggiornare, cancellare o aggiungere una o più elementi grazie a un documento XML. L'azione *delete* può essere realizzata con una richiesta POST e con il parametro *OPERATION*, mentre le operazioni *add* e *update* possono essere realizzate solo tramite richiesta POST.

Parametri standard per la richiesta **Transaction** secondo le specifiche OGC WFS 1.0.0 e 1.1.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WFS)
<i>REQUEST</i>	Sì	Nome della richiesta (Transazione)
<i>VERSION</i>	No	Versione del servizio
<i>FILTER</i>	No	Codifica filtro OGC
<i>BBOX</i>	No	Estensione mappa
<i>FEATUREID</i>	No	Filtrare gli elementi per id
<i>TYPENAME</i>	No	Nome dei layer

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS
<i>OPERATION</i>	No	Specificare l'operazione
<i>EXP_FILTER</i>	No	Espressione di filtraggio

OPERATION

Questo parametro consente di eliminare un elemento senza utilizzare una richiesta POST con un documento XML dedicato.

Esempio di URL:

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

Nota: I parametri *FEATUREID*, *BBOX* e *FILTER* si escludono a vicenda e sono prioritari in questo ordine.

Aggiungere elementi

Esempio richiesta POST:

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

con il documento *add.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">
```

(continues on next page)

(continua dalla pagina precedente)

```

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

```

Aggiornare elementi

Esempio richiesta POST:

```

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

```

con il documento *update.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>

```

Eliminare elementi

Esempio richiesta POST:

```

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

```

con il documento *delete.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">

```

(continues on next page)

(continua dalla pagina precedente)

```

<ogc:Filter>
  <ogc:FeatureId fid="24"/>
</ogc:Filter>
</wfs:Delete>
</wfs:Transaction>

```

3.4 Web Coverage Service (WCS)

Gli standard WCS 1.0.0 e 1.1.1 implementati in QGIS Server forniscono un'interfaccia HTTP per accedere ai dati raster, denominati *coverage*, provenienti da un progetto QGIS.

Specifiche:

- WCS 1.0.0
- WCS 1.1.1

Richieste standard fornite da QGIS Server:

Request	Descrizione
<i>GetCapabilities</i>	Restituisce metadati XML con informazioni sul server
<i>DescribeCoverage</i>	Recupera il documento XML relativo alle informazioni aggiuntive sulle coverages.
<i>GetCoverage</i>	Recupera coverage

3.4.1 GetCapabilities

Parametri standard per la richiesta **GetCapabilities** in conformità alle specifiche OGC WCS 1.1.1:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WCS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetCapabilities)
<i>VERSION</i>	No	Versione del servizio

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di URL:

```

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

```

Esempio di documento XML quando un singolo layer raster (denominato T20QPD_20171123T144719_TCI) viene pubblicato nel progetto QGIS per il servizio WCS:

```

<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>
    ...

```

(continues on next page)

(continua dalla pagina precedente)

```

</Service>
<Capability>
...
</Capability>
<ContentMetadata>
  <CoverageOfferingBrief>
    <name>T20QPD_20171123T144719_TCI</name>
    <label>T20QPD_20171123T144719_TCI</label>
    <lonLatEnvelope srsName="urn:ogc:def:crs:OGC:1.3:CRS84">
      <gml:pos>-61.585973 16.331189</gml:pos>
      <gml:pos>-61.52537 16.400376</gml:pos>
    </lonLatEnvelope>
  </CoverageOfferingBrief>
</ContentMetadata>
</WCS_Capabilities>

```

VERSION

Questo parametro consente di specificare la versione del servizio da utilizzare. Attualmente, i valori della versione non sono utilizzati internamente e sono sempre di riferimento a 1.1.1.

3.4.2 DescribeCoverage

Questa richiesta consente di recuperare informazioni aggiuntive sulle coperture, come il formato della fonte di dati sottostante, il numero di bande, ... Parametri standard per la richiesta **DescribeCoverage** secondo le specifiche OGC WCS 1.1.1:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WCS)
<i>REQUEST</i>	Sì	Nome della richiesta (DescribeCoverage)
<i>VERSION</i>	No	Versione del servizio
<i>COVERAGE</i>	No	Specificare i layer coverage (WCS 1.0.0)
<i>IDENTIFIER</i>	No	Specificare layer coverage (WCS 1.1.1)

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Il documento XML per un layer raster GeoTIFF a 3 bande si presenta come segue:

```

<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>
  </CoverageOffering>

```

(continues on next page)

```

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

```

COVERAGE

Questo parametro, definito in WCS 1.0.0, consente di specificare i layer da interrogare per ottenere informazioni aggiuntive. I nomi devono essere separati da una virgola.

Inoltre, QGIS Server ha introdotto un'opzione per selezionare i layer in base al loro nome breve. Il nome breve di un layer può essere configurato tramite :menuselection: *Proprietà* → *Metadati* nel menu dei layer. Se il nome breve è definito, viene utilizzato per impostazione predefinita al posto del nome del layer:

```

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

```

Nota: COVERAGE è obbligatorio se IDENTIFIER non è impostato.

IDENTIFIER

Questo parametro sostituisce il parametro *COVERAGE* in WCS 1.1.1. Ma QGIS Server non filtra in base al parametro VERSION, quindi IDENTIFIER e COVERAGE hanno lo stesso effetto.

Nota: IDENTIFIER è obbligatorio se COVERAGE non è impostato. Se sono definiti entrambi i parametri IDENTIFIER e COVERAGE, COVERAGE è sempre usato in modo prioritario.

3.4.3 GetCoverage

Questa richiesta consente di recuperare la copertura in base a vincoli specifici come l'estensione o il SR. Parametri standard per la richiesta **DescribeCoverage** secondo le specifiche OGC WCS 1.1.1:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WCS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetCoverage)
<i>VERSION</i>	No	Versione del servizio
<i>COVERAGE</i>	No	Specificare i layer coverage (WCS 1.0.0)
<i>IDENTIFIER</i>	No	Specificare layer coverage (WCS 1.1.1)
<i>WIDTH</i>	Sì	Larghezza della risposta in pixel
<i>HEIGHT</i>	Sì	Altezza della risposta in pixel
<i>BBOX</i>	Sì	Estensione della mappa in unità SR
<i>CRS</i>	Sì	Sistema di riferimento delle coordinate dell'estensione
<i>RESPONSE_CRS</i>	No	Sistema di riferimento delle coordinate della risposta

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

BBOX

Questo parametro consente di specificare l'estensione della mappa nelle unità del SR corrente. Le coordinate devono essere separate da una virgola. Il parametro BBOX è formato come `minx,miny,maxx,maxy`.

Esempio di URL:

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

CRS

Questo parametro permette di indicare il sistema di riferimento spaziale del parametro BBOX e deve essere formato come `EPSG:XXXX`.

RESPONSE_CRS

Questo parametro consente di indicare il sistema di riferimento spaziale della risposta in uscita e deve essere formato come `EPSG:XXXX`. Per impostazione predefinita, viene utilizzato il SR del layer coverage corrispondente.

WIDTH

Questo parametro consente di specificare la larghezza in pixel dell'immagine in uscita. La risoluzione dell'immagine risultante dipende da questo valore.

HEIGHT

Questo parametro consente di specificare l'altezza in pixel dell'immagine in uscita. La risoluzione dell'immagine risultante dipende da questo valore.

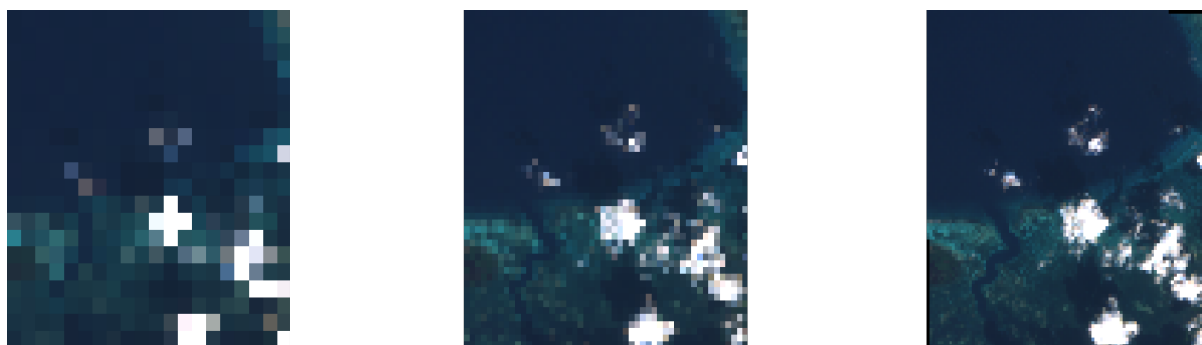


Fig. 3.22: Da sinistra a destra: WIDTH=20&HEIGHT=20, WIDTH=50&HEIGHT=50, WIDTH=100&HEIGHT=100

3.5 Web Map Tile Service (WMTS)

Lo standard **1.0.0** WMTS implementato in QGIS Server fornisce un'interfaccia HTTP per richiedere immagini di mappe tassellate generate da un progetto QGIS. Una tipica richiesta WMTS definisce il progetto QGIS da usare, alcuni parametri WMS come i layer da rappresentare, così come i parametri dei tasselli.

Documento di specifiche del servizio:

- [WMTS 1.0.0](#)

Richieste standard fornite da QGIS Server:

Request	Descrizione
<i>GetCapabilities</i>	Restituisce metadati XML con informazioni sul server
<i>GetTile</i>	Restituisce un tassello
<i>GetFeatureInfo</i>	Recupera i dati (geometria e valori) per una posizione di pixel

3.5.1 GetCapabilities

Parametri standard per la richiesta **GetCapabilities** secondo le specifiche OGC WMTS 1.0.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMTS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetCapabilities)

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di URL:

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

3.5.2 GetTile

Parametri standard per la richiesta **GetTile** secondo le specifiche OGC WMTS 1.0.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMTS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetTile)
<i>LAYER</i>	Sì	Identificatore layer
<i>FORMAT</i>	Sì	Formato in uscita del tassello
<i>TILEMATRIXSET</i>	Sì	Nome della piramide
<i>TILEMATRIX</i>	Sì	Meshing
<i>TILEROW</i>	Sì	Coordinata riga nella mesh
<i>TILECOL</i>	Sì	Coordinata colonna nella mesh

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di 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
```

FORMAT

Questo parametro può essere usato per specificare il formato dell'immagine del tassello. I valori disponibili sono:

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

Se il parametro `FORMAT` è diverso da uno di questi valori, allora viene usato il formato predefinito PNG.

TILEMATRIXSET

Questo parametro definisce il SR da utilizzare nel calcolo della piramide sottostante. Formato: EPSG:XXXX.

TILEMATRIX

Questo parametro permette di definire la matrice da utilizzare per il tassello in uscita.

TILEROW

Questo parametro permette di selezionare la riga del tassello da ottenere all'interno della matrice.

TILECOL

Questo parametro permette di selezionare la colonna del tassello da ottenere all'interno della matrice.

3.5.3 GetFeatureInfo

Parametri standard per la richiesta **GetFeatureInfo** secondo la specifica OGC WMTS 1.0.0:

Parametro	Richiesto	Descrizione
<i>SERVICE</i>	Sì	Nome del servizio (WMTS)
<i>REQUEST</i>	Sì	Nome della richiesta (GetFeatureInfo)
<i>LAYER</i>	Sì	Identificatore layer
<i>INFOFORMAT</i>	No	Formato in uscita
<i>I</i>	No	Coordinata X di un pixel
<i>J</i>	No	Coordinata Y di un pixel
<i>TILEMATRIXSET</i>	Sì	Nome della piramide
<i>TILEMATRIX</i>	Meshing	
<i>TILEROW</i>	Sì	Coordinata riga nella mesh
<i>TILECOL</i>	Sì	Coordinata colonna nella mesh

Oltre a quelli standard, QGIS Server supporta i seguenti parametri extra:

Parametro	Richiesto	Descrizione
<i>MAP</i>	Sì	File di progetto QGIS

Esempio di URL:

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

INFOFORMAT

Questo parametro permette di definire il formato in uscita del risultato. I valori disponibili sono:

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

Il valore predefinito è `text/plain`.

I

Questo parametro permette di definire la coordinata X del pixel per il quale vogliamo recuperare l'informazione sottostante.

J

Questo parametro permette di definire la coordinata Y del pixel per il quale vogliamo recuperare l'informazione sottostante.

3.6 OGC API Features

OGC API Features (OAPIF) è la prima implementazione della nuova generazione di protocolli OGC. È descritto dal documento [OGC API - Features - Part 1: Core](#).

Ecco un rapido riassunto informale delle differenze più importanti tra il noto protocollo WFS e OAPIF:

- OAPIF è basato su un'API [REST](#)
- OAPIF deve seguire le specifiche di [OPENAPI](#).
- OAPIF supporta diversi formati di output, ma non ne impone nessuno (solo GeoJSON e HTML sono attualmente disponibili in QGIS OAPIF) e utilizza la [content negotiation](#) per determinare quale formato deve essere servito al client.
- JSON e HTML sono considerati di prima classe in OAPIF
- OAPIF è auto-documentata (attraverso l'endpoint `/api`)
- L'OAPIF è completamente navigabile (tramite link) e sfogliabile

Importante: Mentre l'implementazione di OGC API Features in QGIS può utilizzare il parametro `MAP` per specificare il file di progetto, le specifiche OPENAPI non consentono di utilizzare parametri di interrogazione aggiuntivi. Per questo motivo si raccomanda vivamente di non esporre `MAP` nell'URL e di specificare il file di progetto nell'ambiente con altri mezzi (ad esempio impostando `QGIS_PROJECT_FILE` nell'ambiente tramite una regola di riscrittura del server web).

Nota: L'endpoint `API` fornisce una documentazione completa di tutti i parametri supportati e dei formati in uscita del tuo servizio. I paragrafi seguenti descriveranno solo i più importanti.

3.6.1 Rappresentazione risorse

L'implementazione di OGC API Features in QGIS Server supporta attualmente i seguenti formati di rappresentazione delle risorse (output):

- HTML
- JSON

Il formato che viene effettivamente reso dipenderà dalla negoziazione del contenuto, ma un formato specifico può essere richiesto esplicitamente aggiungendo uno specifico formato agli endpoint.

Le estensioni di specifica del formato supportate sono:

- .json
- .html

Ulteriori alias di specifica di formato possono essere definiti da endpoint specifici:

- .openapi: alias per .json supportato dall'endpoint **API**
- .geojson: alias per .json supportato dagli endpoint **Features** e **Feature**

3.6.2 Endpoint

L'API fornisce una lista di endpoint che i client possono richiamare. Il sistema è progettato in modo tale che ogni risposta fornisce un insieme di link per navigare attraverso tutte le risorse fornite.

Punti endpoint forniti dall'implementazione di QGIS sono:

Nome	Percorso	Descrizione
Pagina di destinazione	/	Informazioni generali sul servizio e fornisce collegamenti a tutti gli endpoint disponibili
Conformità	/conformance	Informazioni sulla conformità del servizio agli standard
API	/api	Descrizione completa degli endpoint forniti dal servizio e della struttura dei documenti restituiti
Collezioni	/collections	Elenco di tutte le collezioni (cioè » layer vettoriali») fornite dal servizio
Collezione	/collections/{collectionId}	Informazioni su una collezione (nome, metadati, estensione ecc.)
Caratteristiche	/collections/{collectionId}/items	Elenco degli elementi forniti dalla collezione
Elemento	/collections/{collectionId}/items/{featureId}	Informazioni su un singolo elemento

Pagina di destinazione

L'endpoint principale è la **Landing Page**. Da questa pagina è possibile navigare verso tutti gli endpoint di servizio disponibili. La **Landing Page** deve fornire collegamenti a

- la definizione dell'API (percorso /api link delle relazioni service-desc e service-doc),
- la dichiarazione di Conformità (percorso /conformance, relazione di collegamento conformance), e
- le Collezioni (percorso /collections, relazione di collegamento data).

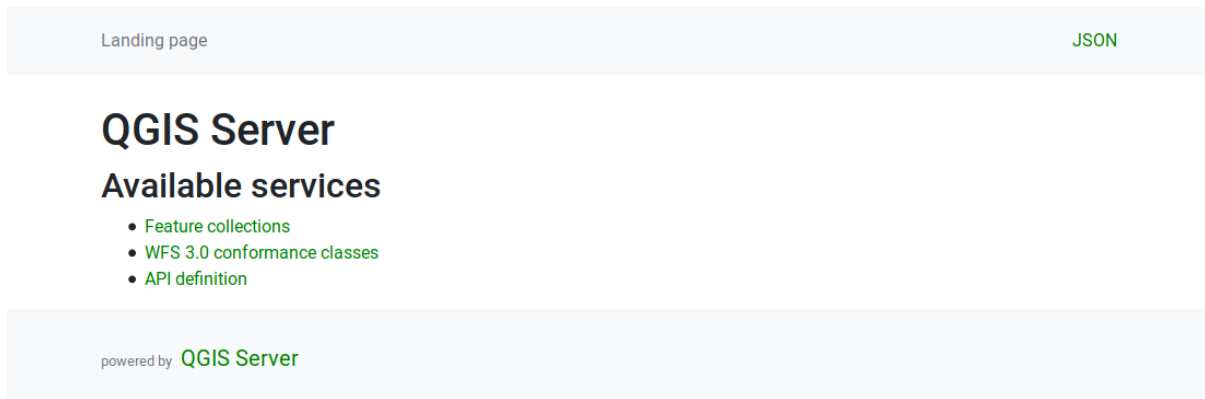


Fig. 3.23: Server OAPIF landing page

Definizione API

La **Definizione dell'API** è una descrizione conforme a OPENAPI dell'API fornita dal servizio. Nella sua rappresentazione HTML è una pagina navigabile dove tutti gli endpoint e i loro formati di risposta sono accuratamente elencati e documentati. Il percorso di questo endpoint è `/api`.

La definizione API fornisce una documentazione completa e autorevole del servizio, compresi tutti i parametri supportati e i formati restituiti.

Nota: Questo endpoint è analogo a `GetCapabilities` di WFS

Lista Collezioni

L'endpoint collezioni fornisce una lista di tutte le collezioni disponibili nel servizio. Poiché il servizio «serve» un singolo progetto QGIS, le collezioni sono i layer vettoriali del progetto corrente (se sono stati pubblicati come WFS nelle proprietà del progetto). Il percorso di questo endpoint è `/collections/`.

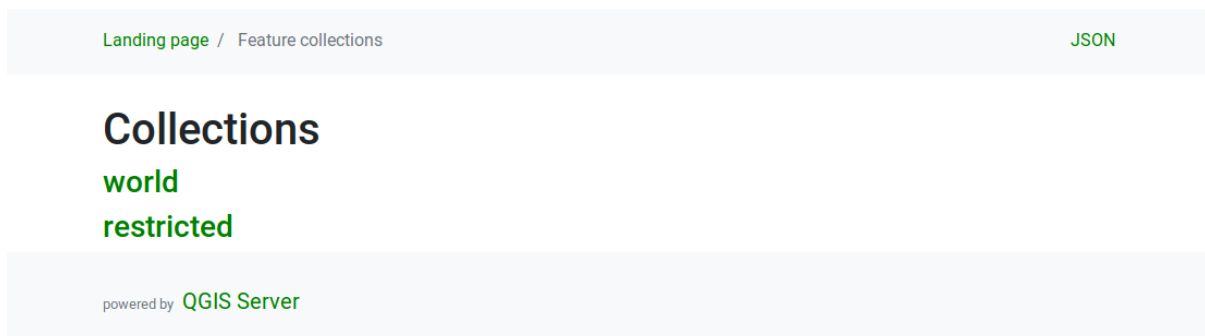


Fig. 3.24: Server OAPIF collections list page

Dettaglio collezione

Mentre l'endpoint delle collezioni non fornisce informazioni dettagliate su ogni collezione disponibile, queste informazioni sono disponibili negli endpoint `/collections/{collectionId}`. Le informazioni tipiche includono l'estensione, una descrizione, i SR e altri metadati.

La rappresentazione HTML fornisce anche una mappa navigabile con gli elementi disponibili.

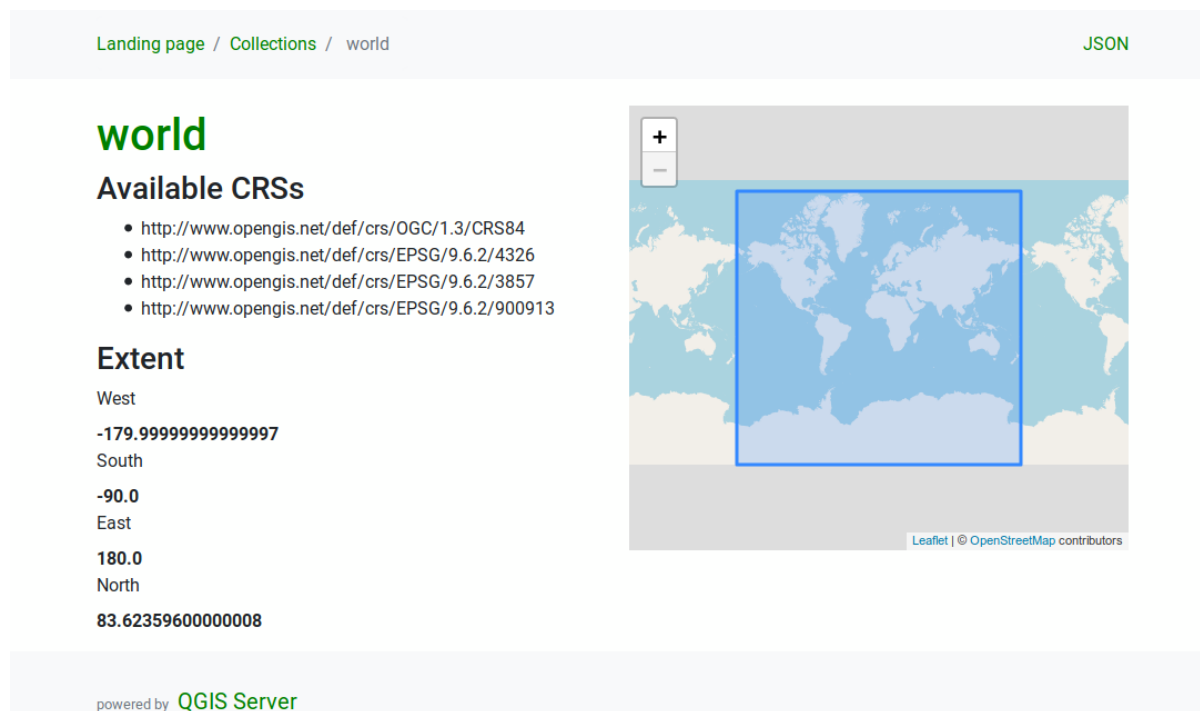


Fig. 3.25: Server OAPIF collection detail page

Lista elementi

Questo endpoint fornisce un elenco di tutti gli elementi in una collezione conoscendo l'ID della collezione. Il percorso di questo endpoint è `/collections/{collectionId}/items`.

La rappresentazione HTML fornisce anche una mappa navigabile con gli elementi disponibili.

Nota: Questo endpoint è analogo a `GetFeature` in WFS 1 e WFS 2.

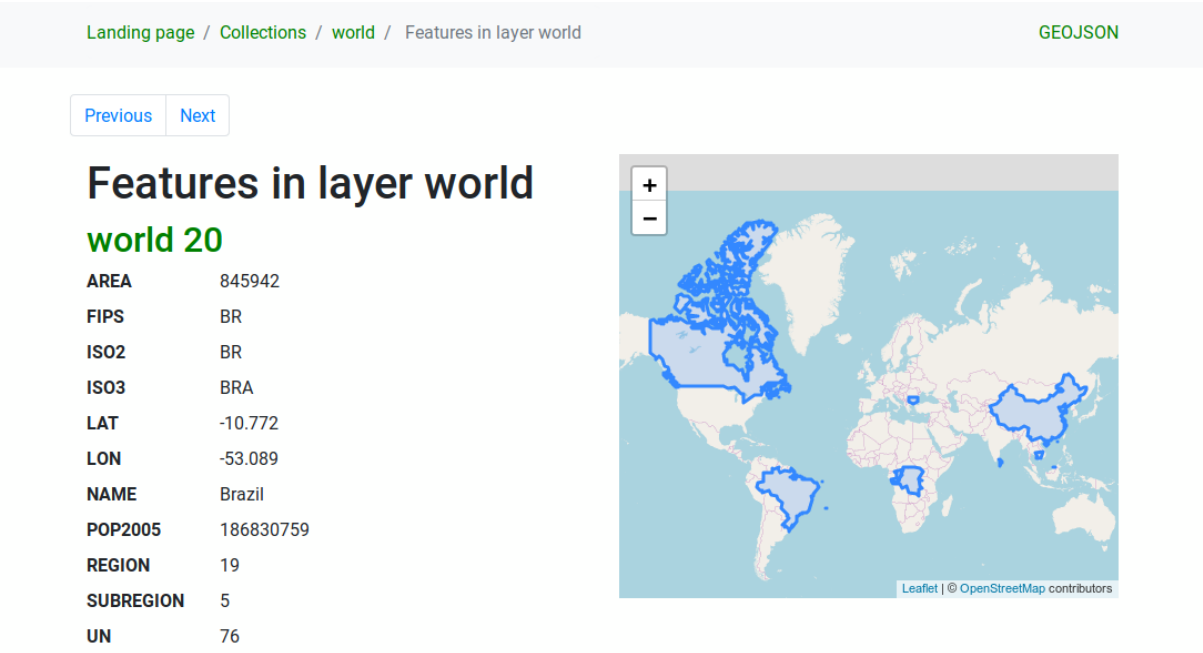


Fig. 3.26: Server OAPIF features list page

Dettaglio elemento

Questo endpoint fornisce tutte le informazioni disponibili su un singolo elemento, compresi gli attributi dell'elemento e la sua geometria. Il percorso di questo endpoint è /collections/{collectionId}/items/{itemId}. La rappresentazione HTML fornisce anche una mappa navigabile con la geometria degli elementi.



Fig. 3.27: Server OAPIF feature detail page

3.6.3 Paginazione

La paginazione di una lunga lista di elementi è implementata nell'API OGC attraverso i link `next`` e `prev`, il server QGIS costruisce questi link aggiungendo `limit` e `offset` come parametri della query string.

Esempio di URL:

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

Nota: Il valore massimo accettabile per `limit` può essere configurato con l'impostazione di configurazione del server `QGIS_SERVER_API_WFS3_MAX_LIMIT` (vedi: `qgis-server-envvar`).

3.6.4 Filtro delle geometrie

Gli elementi disponibili in una collezione possono essere filtrati/ricercati specificando uno o più filtri.

Filtro data e ora

Le collezioni con attributi data e/o datetime possono essere filtrate specificando un argomento `datetime` nella stringa della query. Per impostazione predefinita, il primo campo data/ora è usato per il filtraggio. Questo comportamento può essere configurato impostando una dimensione «Date» o «Time» nella sezione *QGIS Server* ► *Dimension* della finestra di dialogo delle proprietà del layer.

La sintassi del filtraggio di data e ora è completamente descritta nella *Definizione API* e supporta anche intervalli (i valori iniziali e finali sono inclusi) oltre a valori singoli.

Esempi URL:

Restituisce solo gli elementi con una dimensione di data corrispondente a 2019-01-01.

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

Restituisce solo gli elementi con dimensione datetime che corrisponde a 2019-01-01T01:01:01.

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

Restituisce solo gli elementi con dimensione datetime nell'intervallo 2019-01-01T01:01:01 - 2019-01-01T12:00:00.

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

Filtro rettangolo di delimitazione

Un filtro spaziale bounding box può essere specificato con il parametro `bbox`:

L'ordine degli elementi separati da virgola è:

- Angolo inferiore sinistro, longitudine WGS 84
- Angolo inferiore sinistro, latitudine WGS 84
- Angolo superiore destro, longitudine WGS 84
- Angolo superiore destro, latitudine WGS 84

Nota: Le specifiche OGC permettono anche uno specifico bbox a 6 elementi dove il terzo e il sesto elemento sono i componenti Z, questo non è ancora supportato dal server QGIS.

Esempio di URL:

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

Se il SR del riquadro di delimitazione non è [WGS 84](#), un CRS diverso può essere specificato usando il parametro opzionale `bbox-crs`. L'identificatore del formato SR deve essere nel formato [OGC URI](#):

Esempio di URL:

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

Filtro Attributo

I filtri degli attributi possono essere combinati con il filtro bounding box e sono nella forma generale: `<attribute name>=<attribute value>`. Filtri multipli possono essere combinati usando l'operatore `AND``.

Esempio di URL:

filtra tutti gli elementi in cui l'attributo `name` è uguale a «my value»

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

Le corrispondenze parziali sono supportate anche usando un operatore `*` («star»):

Esempio di URL:

filtra tutti gli elementi in cui l'attributo `name` finisce con «value»

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

3.6.5 Ordinamento elemento

È possibile ordinare l'insieme dei risultati per valore di campo usando il parametro della query `sortby`.

I risultati sono ordinati in ordine crescente per impostazione predefinita. Per ordinare i risultati in ordine decrescente, si può impostare un flag booleano (`sortdesc`):

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

3.6.6 Selezione attributo

Gli attributi elementelementi restituiti da una chiamata [Lista elementi](#) possono essere limitati aggiungendo una lista separata da virgole di nomi di attributi nell'argomento opzionale `properties` della query string.

Esempio di URL:

restituisce solo l'attributo `name`.

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

3.6.7 Personalizzare le pagine HTML

La rappresentazione HTML utilizza un insieme di modelli HTML per generare la risposta. Il template viene analizzato da un motore di template chiamato *inja*. I template possono essere personalizzati sovrascrivendoli (vedere: *Sovrascritture modello*). Il template ha accesso agli stessi dati disponibili per la rappresentazione JSON e alcune funzioni aggiuntive sono disponibili per il template:

Funzioni personalizzate modello

- `path_append( path )`: aggiunge un percorso di cartella all'url corrente
- `path_chomp( n )`: rimuove il numero specificato «n» di componenti di cartella dal percorso url corrente
- `json_dump( )`: stampa i dati JSON passati al modello
- `static( path )`: restituisce l'URL completo del percorso statico specificato. Ad esempio: «static(«/style/black.css»)» con un percorso radice «http://localhost/qgisserver/oapif» restituirà «http://localhost/qgisserver/oapif/static/style/black.css».
- `links_filter( links, key, value )`: Restituisce i link filtrati da un elenco di link
- `content_type_name( content_type )`: Restituisce un nome breve di un tipo di contenuto, per esempio «text/html» restituirà «HTML»
- `nl2br( text )`: Restituisce il testo in ingresso con tutte le linee nuove sostituite dal tag «
».

Sovrascritture modello

I modelli e le risorse statiche sono memorizzati nelle sottocartelle della cartella delle risorse API di default del server QGIS (/usr/share/qgis/resources/server/api/ su un sistema Linux), la cartella di base può essere personalizzata cambiando la variabile di ambiente QGIS_SERVER_API_RESOURCES_DIRECTORY.

Una tipica installazione di Linux avrà il seguente albero di cartelle:

```
/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
```

Per sovrascrivere i modelli puoi copiare l'intero albero in un'altra posizione e assegnare QGIS_SERVER_API_RESOURCES_DIRECTORY alla nuova posizione.

Catalog

Il QGIS Server Catalog è un semplice catalogo che mostra l'elenco dei progetti QGIS forniti dal QGIS Server. Fornisce un sito web user-friendly completamente navigabile con capacità di mappatura di base per sfogliare rapidamente i dataset presenti in quei progetti QGIS.

Il QGIS Server catalog usa le variabili `QGIS_SERVER_LANDING_PAGE_PROJECTS_DIRECTORIES` e `QGIS_SERVER_LANDING_PAGE_PROJECTS_PG_CONNECTIONS` (vedi *[Variabili d'ambiente](#)*)

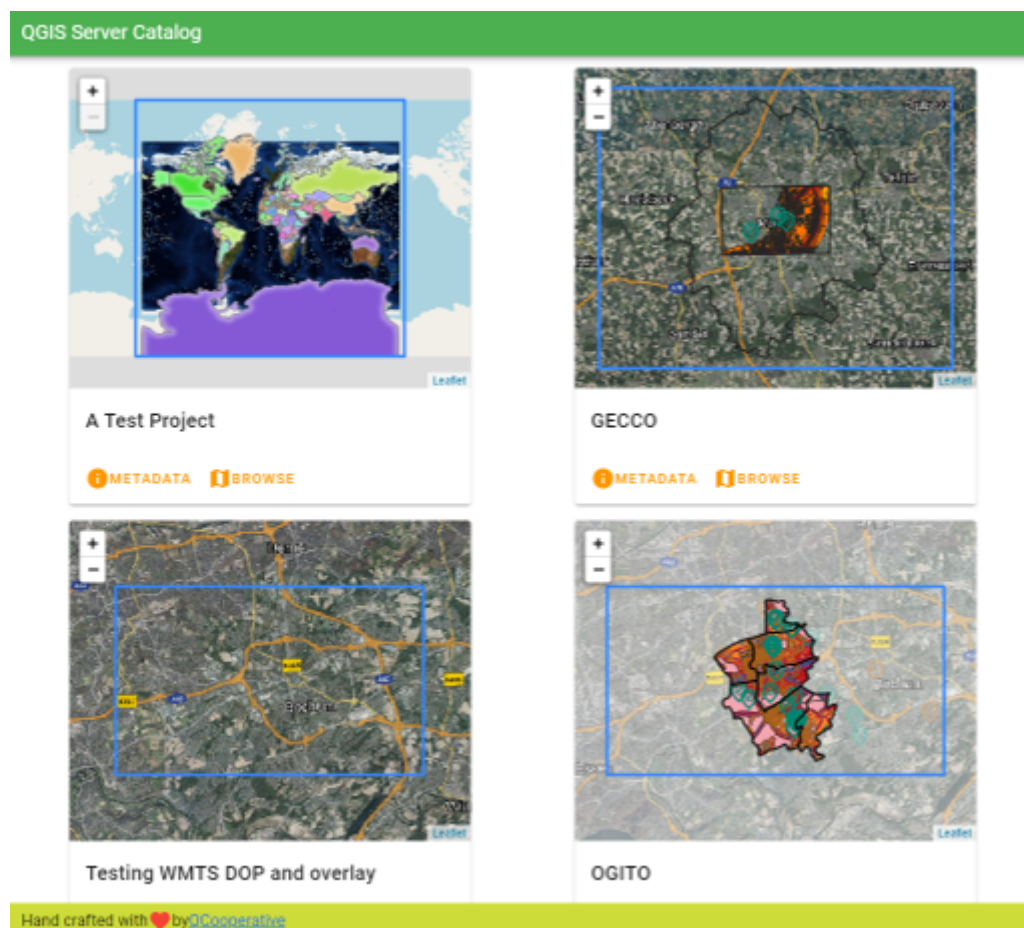


Fig. 4.1: Pagina dell'elenco dei progetti del Server Catalog

Puoi consultare i metadati associati a un progetto e i servizi che fornisce. Vengono forniti anche i link a questi servizi.

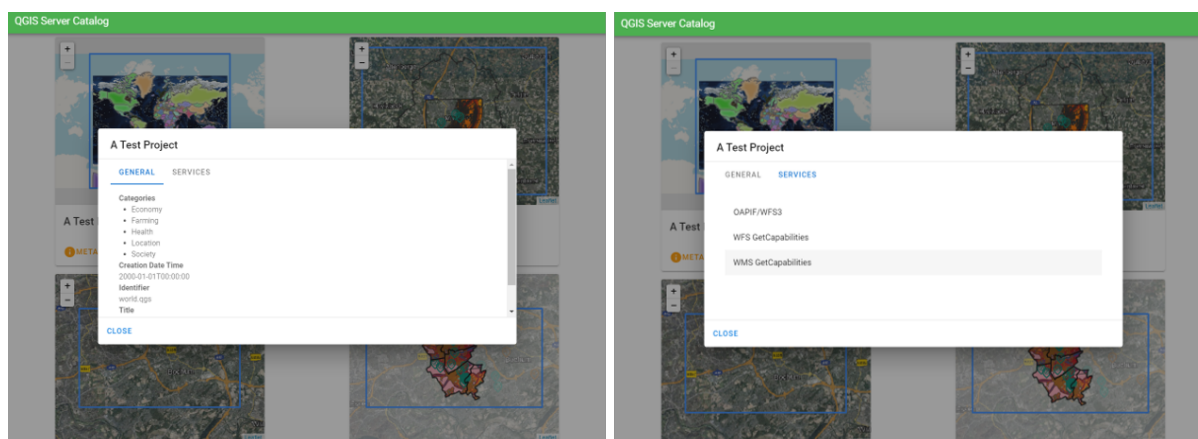


Fig. 4.2: Server Catalog, metadati associati a un progetto e servizi (link a) che fornisce.

Sfogliando un progetto, viene elencato il dataset che serve.

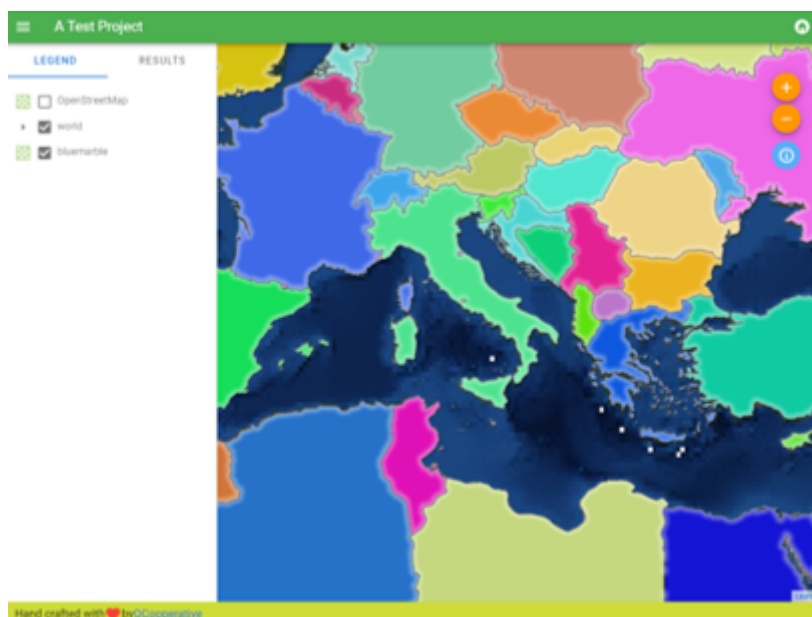


Fig. 4.3: Sfogliare un dataset fornito da un progetto nel catalogo del server

Usa il clic destro su un layer per visualizzare la tabella degli attributi ad esso associati.

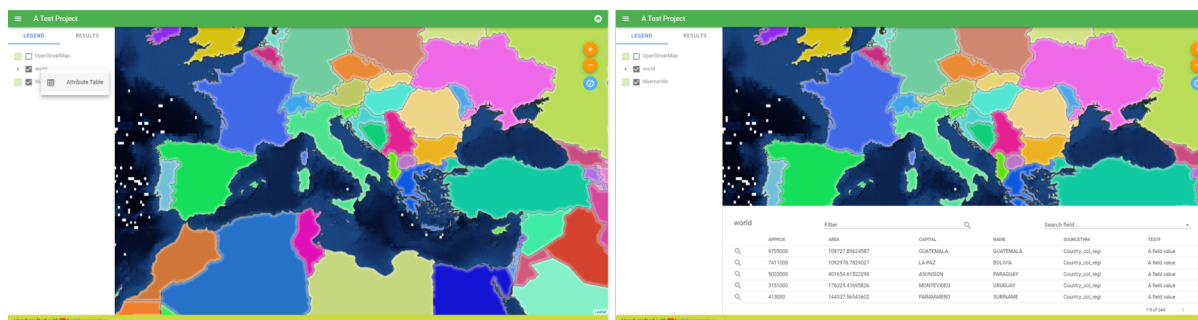


Fig. 4.4: Tabella degli attributi associati a un layer

È possibile consultare le informazioni degli elementi nella mappa come mostrato nell'immagine qui sotto:

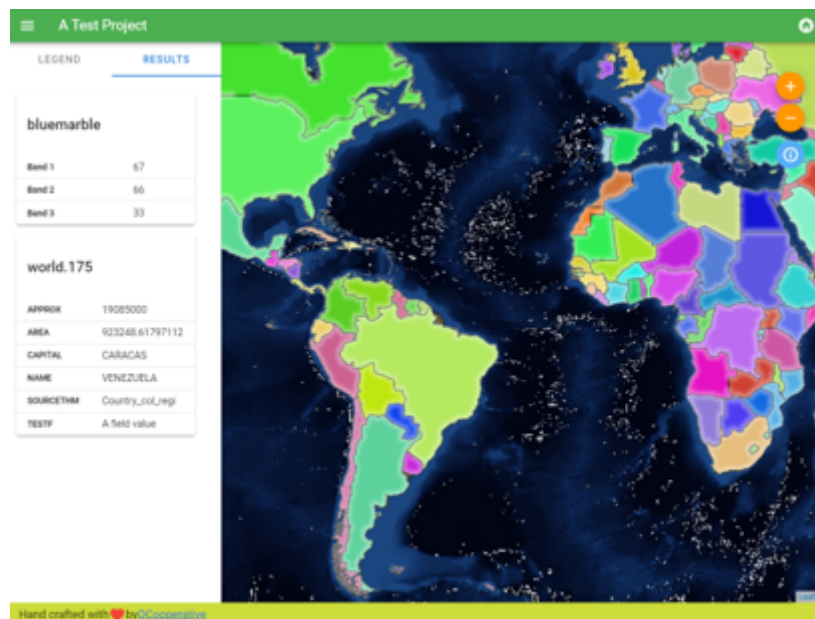


Fig. 4.5: Consultazione delle informazioni di un elemento della mappa

5.1 Elenco dei plugin

I plugin possono essere installati anche su QGIS Server.

Alcuni plugin progettati per il server si trovano nel repository ufficiale di [QGIS](#).

Installa solo i plugin necessari per i tuoi scopi. Sul server QGIS, i plugin sono come dei collegamenti nel server QGIS e possono alterare gli input o gli output del server QGIS. Possono produrre risultati inaspettati se conosci il funzionamento del plugin. Per sapere quale plugin ti può essere utile, consulta la rispettiva documentazione o l'applicazione che necessita dei plugin del server QGIS.

5.2 Ubicazione dei plugin

Per impostazione predefinita, sui sistemi basati su Debian, QGIS Server cercherà i plugin situati in `/usr/lib/qgis/plugins`. Il valore predefinito viene visualizzato all'avvio di QGIS Server, nei log. È possibile impostare un percorso personalizzato definendo la variabile d'ambiente `QGIS_PLUGINPATH` nella configurazione del server web.

5.3 Installazione

5.3.1 Manualmente con uno ZIP

Ad esempio, per installare il plugin **HelloWorld** per testare il server, utilizzando una cartella specifica, devi prima creare una cartella per contenere i plugin del server. Questa sarà specificata nella configurazione dell'host virtuale e passata al server attraverso una variabile d'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
```

Avvertimento: Conformemente alla sua [descrizione](#), il plugin HelloServer è stato progettato per scopi di sviluppo e dimostrativi. Non utilizzare questo plugin per la produzione se non ne hai bisogno.

5.3.2 Con uno strumento a linea di comando

Se hai bisogno di installare e aggiornare regolarmente i plugin che sono memorizzati nel repository dei plugin di QGIS, puoi usare il [QGIS-Plugin-Manager](#). È uno strumento che aiuta a gestire i plugin dalla linea di comando.

L'installazione avviene tramite pip. L'installazione in un ambiente virtuale è una buona pratica, ma non è obbligatoria:

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

Per aggiornare lo strumento:

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

Quindi, puoi utilizzare l'eseguibile `qgis-plugin-manager` dalla linea di 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          |          |
|qgis_server_render_geojson|GeoJson Renderer |v0.4     |          | 3.4
|          |          |Matthias Kuhn (OPENGIS.ch)|          |
|DataPlotly            |Data Plotly       |3.7.1    |          | 3.4
|          |          |3.98      |          |
|          |          |Matteo Ghetta (Faunalia)|Upgrade to 3.8.1|
-----

```

Si consiglia di leggere la documentazione completa nel file [readme](#) per sapere come installare o aggiornare i plugin con questo strumento.

5.4 Configura il server HTTP

5.4.1 Apache

Per poter utilizzare i plugin del server, FastCGI deve sapere dove cercare. Quindi, dobbiamo modificare il file di configurazione di Apache per indicare a FastCGI la variabile d'ambiente `QGIS_PLUGINPATH`:

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

Inoltre, un'autorizzazione HTTP di base è necessaria per lavorare con il plugin HelloWorld precedentemente introdotto. Quindi dobbiamo aggiornare il file di configurazione di Apache un'ultima volta:

```
# Needed for QGIS HelloServer plugin HTTP BASIC auth
<IfModule mod_fcgid.c>
```

(continues on next page)

(continua dalla pagina precedente)

```
RewriteEngine on
RewriteCond %{HTTP:Authorization} .
RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]
</IfModule>
```

Poi, riavvia Apache:

```
systemctl restart apache2
```

5.5 Come usare un plugin

Fai il test il server con il plugin HelloWorld:

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

Puoi dare un'occhiata al GetCapabilities di default del server QGIS su:

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


6.1 Registrazioni di controllo

Per registrare le richieste inviate al server, devi impostare la seguente variabile d'ambiente:

- *QGIS_SERVER_LOG_STDERR*

Con le seguenti variabili è possibile personalizzare ulteriormente la registrazione:

- *QGIS_SERVER_LOG_LEVEL*
- *QGIS_SERVER_LOG_PROFILE*

6.2 Variabili d'ambiente

Puoi configurare alcuni aspetti di QGIS Server impostando le **environment variables**.

A seconda del server HTTP e del modo in cui si esegue QGIS Server, esistono diversi modi per definire queste variabili. Questo è descritto in dettaglio in *Apache HTTP Server*.

Nome	Descrizione	Predefinito	Servizi
QGIS_OPTIONS_PATH	Specifica il percorso della cartella con le impostazioni. Funziona come l'opzione <code>--optionspath</code> dell'applicazione QGIS. Cerca il file delle impostazioni in <code><QGIS_OPTIONS_PATH>/QGIS/QGIS3.ini</code> .	""	All
QGIS_PLUGINPATH	Utile se stai utilizzando i plugin Python per il server, questo imposta la cartella che viene cercata per i plugin Python.	""	All
QGIS_PROJECT_FILE	Il file di progetto <code>.qgs</code> o <code>.qgz</code> , normalmente passato come parametro nella stringa di query (con <i>MAP</i>); lo puoi anche impostare come variabile d'ambiente (per esempio usando il modulo Apache <code>mod_rewrite</code>). Nota che puoi anche indicare un progetto memorizzato in PostgreSQL, ad esempio <code>postgresql://localhost:5432?sslmode=disable&dbname=mydb&schema=myschema&project=myproject</code> .	""	All
QGIS_SERVER_API_RESOURCE_DIRECTORY	Indica il percorso per tutte le risorse statiche OGC API (come OAPIF/WFS3) (modelli HTML, CSS, JS, ...)	dipende dal packaging	OAPIF/WFS3
QGIS_SERVER_API_WFS3_MAXFEATURES	Valore massimo per <code>limit</code> in una richiesta di features OAPIF/WFS3.	10000	OAPIF/WFS3
QGIS_SERVER_CACHE_DIRECTORY	Specifica la cartella della cache di rete sul filesystem.	cache nella cartella del profilo	All
QGIS_SERVER_CACHE_SIZE	Imposta la dimensione della cache di rete in MB.	50 MB	All
QGIS_SERVER_DISABLE_GETPRINT	È un'opzione a livello di progetto per migliorare i tempi di lettura del progetto disabilitando il caricamento dei layout. L'attivazione di questa opzione disabilita la richiesta <code>GetPrint</code> di QGIS WMS. Imposta questo flag del progetto QGIS per non caricare i layout.	false	WMS
QGIS_SERVER_IGNORE_BADLAYERS	I «layer bad» sono layer che non possono essere caricati. Il comportamento predefinito di QGIS Server è quello di considerare il progetto come non disponibile se contiene un layer non valido. Il comportamento predefinito può essere sovrascritto impostando questa variabile a 1 o <code>true</code> . In questo caso, i layer «bad» saranno semplicemente ignorati e il progetto sarà considerato valido e disponibile.	false	All
QGIS_SERVER_LANDING_PAGE_REFERENCE	Componente di percorso dell'URL di base della pagina di destinazione	<>	All
QGIS_SERVER_LANDING_PAGE_PROJECTS_DIRECTORIES	Cartelle utilizzate dal servizio di landing page per trovare i progetti <code>.qgs</code> e <code>.qgz</code>	<>	All
QGIS_SERVER_LANDING_PAGE_PROJECTS_PG_CONNECTIONS	Stringhe di connessione PostgreSQL utilizzate dal servizio di landing page per trovare i progetti	<>	All
QGIS_SERVER_LOG_FILE	Specifica il percorso e il nome del file. Assicurati che il server abbia i permessi corretti per scrivere sul file. Il file dovrebbe essere creato automaticamente, e può inviare alcune richieste al server. Se non c'è, controlla i permessi.	""	All

6.3 Riepilogo impostazioni

All'avvio di QGIS Server, viene visualizzato un riepilogo di tutti i parametri configurabili grazie alle variabili d'ambiente. Inoltre, viene visualizzato anche il valore attualmente utilizzato e l'origine.

Ad esempio con 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:

- 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 questo caso particolare, sappiamo che i valori **QGIS_SERVER_MAX_THREADS** e **QGIS_SERVER_PARALLEL_RENDERING** vengono letti dal file ini che si trova nella cartella **QGIS_OPTIONS_PATH** (definita tramite una variabile d'ambiente). Le voci corrispondenti nel file ini sono **/qgis/max_threads** e **/qgis/parallel_rendering** e i loro valori sono **true** e **4** thread.

6.4 Connessione a file di servizio

Per far sì che apache sia a conoscenza del file di servizio PostgreSQL (vedere la sezione pg-service-file) devi fare in modo che il tuo file *.conf assomigli a:

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

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

6.5 Aggiungere font al server Linux

Tieni presente che puoi utilizzare progetti QGIS che puntano a font che potrebbero non esistere per impostazione predefinita su altri computer. Ciò significa che se si condivide il progetto, questo potrebbe apparire diverso su altri computer (se i font non esistono sul computer di destinazione).

Per evitare che ciò accada, devi installare i font mancanti sul computer di destinazione. Sui sistemi desktop questa operazione è di solito banale (doppio clic sui font).

Per Linux, se non hai installato un ambiente desktop (o preferisci la linea di comando), devi installarlo:

- Su sistemi basati su Debian:

```
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
```

- Su sistemi basati su Fedora:

```
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
```

Server di sviluppo

Un'installazione e un deployment di produzione di QGIS Server di solito comporta la configurazione di un componente web server (ad esempio Apache o Nginx) che può inoltrare le richieste HTTP provenienti dai client all'applicazione binaria QGIS Server FastCGI.

Se vuoi testare rapidamente QGIS Server sulla tua macchina locale senza configurare e installare uno stack completo del server web, puoi usare il server QGIS Development Standalone.

Questa è un'applicazione indipendente che fornisce un server web molto semplice pronto a ospitare i file del tuo progetto.

Avvertimento: Il server di sviluppo standalone non è stato sviluppato con lo scopo di essere usato in produzione, non è stato controllato per le vulnerabilità di sicurezza o per altre condizioni di stress che normalmente si verificano su un server esposto pubblicamente.

Per avviare il server:

```
$ qgis_mapserver
```

La porta predefinita che il Development Server ascolta è 8000. Esempio in uscita:

```
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
```

Il server ha alcune opzioni che possono essere passate come argomenti della linea di comando. Puoi vederle tutte invocando il server con `-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
```

Sviluppo ed esecuzione in container

Ci sono molti modi per utilizzare applicazioni containerizzate, dai più semplici (semplici immagini Docker) ai più sofisticati (Kubernetes e così via).

Nota: Questo tipo di sviluppo ha bisogno che l'applicazione `docker` sia installata e funzionante. Guarda questo [tutorial](#).

Suggerimento: Docker esegue applicazioni preconfezionate (alias immagini) che possono essere recuperate come sorgenti (Dockerfile e risorse) da costruire o già costruite da registri (privati o pubblici).

Nota: I download dei package QGIS Debian-Ubuntu necessitano di una chiave di autenticazione gpg valida. Fare riferimento a [installation pages](#) per aggiornare il seguente file Docker.

8.1 Semplici immagini docker

Poiché l'immagine docker non esiste in un registro pubblico, dovrai crearla. Per farlo, crea una cartella `qgis-server` e all'interno di essa:

- crea un file `Dockerfile` con questo contenuto:

```
FROM debian:bullseye-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 \
```

(continues on next page)

(continua dalla pagina precedente)

```

&& 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
&& echo "deb [signed-by=/etc/apt/keyrings/qgis-archive-keyring.gpg] https://qgis.org/debian bullseye 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"]

```

- crea un file `cmd.sh` con questo contenuto:

```

#!/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

```

- costruisci l'immagine con:

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

8.1.1 Prima esecuzione

Per eseguire il server è necessario un file di progetto QGIS. Puoi utilizzare uno tuo o scegliere questo esempio <<https://gitlab.com/Oslandia/qgis/docker-qgis/-/blob/cc1798074d4a66a472721352f3984bb318777a5a/qgis-exec/data/osm.qgs>>`_.

Per farlo, crea una cartella `data` all'interno della cartella `qgis-server` e copia il tuo file al suo interno. Per rispettare le istruzioni seguenti, rinominalo in `osm.qgs`.

Nota: Potresti dover aggiungere gli URL consigliati nella scheda *QGIS Server* della *menuselection: Progetto* – *> Proprietà* se le *GetCapabilities* sono interrotte. Ad esempio, se il server è esposto sulla porta 8080, si inserirà l'URL pubblicizzato `http://localhost:8080/qgis-server/`. Ulteriori informazioni sono disponibili nella sezione *Configura il tuo progetto* e successive.

Ora, puoi avviare il server con:

```
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
```

Opzioni utilizzate:

- **-d:** esecuzione in background
- **--rm:** rimuovere il contenitore quando viene arrestato
- **--name:** nome del contenitore da creare
- **--net:** (precedentemente creata) sottorete
- **--hostname:** nome host del contenitore, per riferimenti successivi
- **-v:** cartella dati locale da installare nel contenitore
- **-p:** mappatura porte host/contenitore
- **-e:** variabile d'ambiente da utilizzare nel contenitore

Per verificare, digita `docker ps | grep qgis-server` e si dovrebbe vedere una linea con **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 Esempio utilizzabile

Poiché il server accetta solo connessioni fastcgi, hai bisogno di un server HTTP che gestisca questo protocollo. Per farlo, dobbiamo creare un semplice file di configurazione di Nginx e avviare un'immagine di Nginx.

Crea un file `nginx.conf` nella cartella corrente con questo contenuto:

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

(continues on next page)

(continua dalla pagina precedente)

```

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

```

E digita questo 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

```

Per verificare le capabilities disponibili, digita nel browser <http://localhost:8080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities>.

8.1.3 Pulizia

Per rimuovere le immagini in esecuzione, digita:

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

8.2 Stack Docker

Il metodo precedente è eseguibile tramite script, ma non è facilmente confezionabile, né standardizzabile o facilmente gestibile.

Per lavorare con un insieme di immagini docker si può usare uno stack docker gestito da un orchestratore. In uno stack, le immagini lavorano nella stessa rete privata ed è possibile avviare/arrestare l'intero stack o distribuirlo ad altri worker. Esistono molti orchestratori, ad esempio Swarm, Kubernetes e Mesos.

Di seguito presenteremo semplici configurazioni a scopo di test. Non sono adatte alla produzione.

8.2.1 Swarm/docker-compose

Docker ha ora un proprio orchestratore: Swarm (compatibile con i file docker-compose). Devi [enable it](#) (la versione per Mac funziona anche con Linux).

Descrizione stack

Ora che Swarm funziona, crea il file di servizio (vedi [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)

```

(continues on next page)

(continua dalla pagina precedente)

```

- DEBUG=1                                # display env before spawning QGIS Server

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

```

Per distribuire (o aggiornare) lo stack, digita:

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

Controlla lo stato di distribuzione dello stack fino a ottenere **1/1** nella colonna **replicas**:

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

Qualcosa come:

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

Per verificare le capabilities del WMS, digita in un browser web <http://localhost:8080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities>.

Pulizia

Per pulire, digita:

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

8.2.2 Kubernetes

Installazione

Se hai un'installazione di **Docker Desktop**, l'uso di Kubernetes (alias k8s) è piuttosto semplice: [enable k8s](#).

In caso contrario, segui [minikube tutorial](#) o [microk8s for Ubuntu](#).

Poiché l'installazione di Kubernetes può essere molto complessa, ci concentreremo solo sugli aspetti utilizzati in questa demo. Per ulteriori informazioni, consulta [official documentation](#).

microk8s

microk8s necessita di ulteriori passaggi: devi abilitare il registro e etichettare l'immagine qgis-server per far sì che Kubernetes trovi le immagini create.

Per prima cosa, abilita il registro:

```
microk8s enable dashboard dns registry
```

Quindi, etichetta e inoltra l'immagine al registro appena creato:

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

Infine, aggiungi o completa il `/etc/docker/daemon.json` per avere il registro **127.0.0.1:32000** elencato nel campo **insecure-registries**:

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

Creare manifesti

Kubernetes descrive gli oggetti da distribuire in manifesti yaml. Ne esistono di diversi tipi, ma noi useremo solo i `deploy` (gestire i pod, cioè le immagini docker) e i servizi per esporre i `deploy` a scopi interni o esterni.

Manifesti di distribuzione

Crea un file `deployments.yaml` con questo contenuto:

```
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: IfNotPresent
          env:
            - name: LANG
              value: en_EN.UTF-8
            - name: QGIS_PROJECT_FILE
              value: /data/osm.qgs
            - name: QGIS_SERVER_LOG_LEVEL
              value: "0"
            - name: DEBUG
              value: "1"
      ports:
```

(continues on next page)

(continua dalla pagina precedente)

```

      - 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/default.conf
      volumes:
        - name: nginx-conf
          hostPath:
            path: REPLACE_WITH_FULL_PATH/nginx.conf

```

Manifesti di servizio

Crea un file `services.yaml` con questo contenuto:

```

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

```

(continues on next page)

(continua dalla pagina precedente)

```

namespace: default
spec:
  type: NodePort
  selector:
    myLabel: qgis-nginx
  ports:
    - port: 80
      targetPort: 80
      nodePort: 30080

```

Distribuzione manifesti

Per distribuire le immagini e i servizi in Kubernetes, si può usare la dashboard (fai clic su + in alto a destra) o la linea di comando.

Nota: Quando utilizzi la linea di comando con `microk8s` dovrai anteporre a ogni comando il prefisso `microk8s.`

Per distribuire o aggiornare i manifesti:

```
kubectl apply -k ./
```

Per verificare cosa è attualmente distribuito:

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

Dovresti ottenere qualcosa di simile a:

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	↪
↪ 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

Per leggere i log di nginx/qgis, digita:

```
kubectl logs -f POD_NAME
```

Per verificare le capabilities del WMS, digita in un browser web <http://localhost:30080/qgis-server/?SERVICE=WMS&VERSION=1.3.0&REQUEST=GetCapabilities>.

Pulizia

Per ripulire, digita:

```
kubectl delete -n default service/qgis-server service/qgis-nginx deployment/qgis-  
↪nginx deployment/qgis-server
```

8.3 Distribuzione in cloud

Gestire il tuo cluster di server per gestire la distribuzione di applicazioni containerizzate è un lavoro complesso. Devi gestire diversi problemi, come l'hardware, la larghezza di banda e la sicurezza a diversi livelli.

Le soluzioni di distribuzione in cloud possono essere una buona alternativa quando non ci si vuole concentrare sulla gestione dell'infrastruttura.

Una distribuzione nel cloud può utilizzare meccanismi proprietari, ma anche compatibili con le fasi spiegate in precedenza (*docker images* e *stack management*).

8.3.1 Caso d'uso AWS

Con Amazon AWS, grazie alle funzionalità di [ECS \(Elastic Container Service\)](#), puoi utilizzare docker-compose o wrapper compatibili con Kubernetes per gestire il tuo stack. Dovrai creare un [image registry](#) per rendere accessibili le tue immagini personalizzate.

Per utilizzare le funzionalità simili a docker-compose, devi installare il client **ecs-cli** e disporre di [permessi/ruoli adeguati](#). Quindi, con l'aiuto dei comandi [ecs-cli compose](#), puoi riutilizzare la [stack description](#).

Per utilizzare Kubernetes, puoi utilizzare la console web di AWS o lo strumento a linea di comando [eksctl](#) e disporre dei permessi/ruoli appropriati. Quindi, con un ambiente kubectl ben configurato, puoi riutilizzare il [Kubernetes manifests](#).

Domande frequenti

- *Quali sono le differenze tra QGIS Desktop e QGIS Server?

QGIS Desktop ha un'interfaccia utente grafica e consente di creare e modificare mappe. QGIS Server è un'applicazione server che fornisce i file di progetto QGIS alle applicazioni dell'utente finale tramite servizi web OGC come [WMS](#), [WFS](#), ecc..

- Cos'è OGC?

L' [OGC \(Open Geospatial Consortium\)](#) è un'organizzazione internazionale senza scopo di lucro impegnata a creare standard aperti di qualità per la comunità geospaziale globale.

- *Nomi di altri server di web mapping?

ArcGIS server, Geoserver, Mapserver, Mapnik etc.

- Come confrontare il server QGIS con altri server di web mapping? (2021/01/01)

Caratteristiche	QGIS Server	GeoServer	ArcGIS Server
Dal	2006	2001	1999
Licenza	GPL	GPL	commerciale
Supporto commerciale	Più società	Più società	ESRI e la sua rete di venditori
Tecnologia	C++/python	Java	C++
Cache tassello	Sì	sì (via GeoWebCache)	Sì
3D	No	No	Sì
Interrogazione	Filtri FES (2.0) e OGC (1.0)	Filtri CQL e OGC	Filtri OGC
Generazione di report	Sì	Sì	Sì
Amministrazione Server	sì tramite parti terze (LizMap, QWC2, ecc.)	web + API REST	web + API REST
Progetto GIS edizione Layer/simbologia	completo tramite GUI dedicato	semplice tramite interfaccia web	completo tramite GUI dedicato

- Quali sono le versioni delle specifiche OGC implementate in QGIS server rispetto ad altri server di web mapping? (2021/01/01)

OGC standard	QGIS Server	GeoServer	ArcGIS Server
WMS (Web Map Service)	1.3.0 - 1.1.1	1.3.0 - 1.1.1	1.3.0 - 1.1.1
WFS (Web Feature Service)	1.1.0 - 1.0.0	2.0.0 - 1.0.0	2.0.0 - 1.0.0
OAPIF (aka WFS3)	1.0.0	no	no
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)	no	1.0.0	1.0.0
CSW (Catalogue Service for the Web)	no	2.0.2	no
SLD (Styled Layer Descriptor)	Si	Si	Si

- **Che cos'è una cache tassello?*

Le mappe sono spesso statiche. Poiché la maggior parte dei client di mappatura esegue la visualizzazione dei dati WMS (Web Map Service) a ogni interrogazione, ciò può comportare un'elaborazione non necessaria e un aumento dei tempi di attesa.

La cache tassello ottimizza questa esperienza salvando (memorizzando nella cache) le immagini delle mappe, o tassello, man mano che vengono richieste, agendo di fatto come un proxy tra il client (come OpenLayers o Google Maps) e il server (qualsiasi server conforme a WMS). Quando vengono richieste nuove mappe e tasselli, il server QGIS intercetta queste chiamate e restituisce i tasselli pre-renderizzati, se memorizzati, oppure chiama il motore QGIS per visualizzare nuovi tasselli, se necessario. In questo modo, una volta memorizzati i tasselli, la velocità di visualizzazione delle mappe aumenta di molte volte, migliorando notevolmente l'esperienza dell'utente.

- *Che cos'è PostgreSQL?*

PostgreSQL è un potente database relazionale a oggetti open source che QGIS supporta.

- *Che cos'è PostGIS?*

PostGIS è un'estensione del database spaziale per il database relazionale a oggetti PostgreSQL. Aggiunge il supporto per gli oggetti geografici, consentendo l'esecuzione di query di geolocalizzazione in SQL.

- Continua...