

Web Services Geográficos



Taller de Sistemas de Información
Geográficos Empresariales



OGC Web Services (OWS)

- ❑ OWS es el nombre genérico con el que se agrupa todos los estándares de Web Services de OGC (WMS, WFS, WCS, CSW, etc).
- ❑ Definidos en base a estándares de Internet: HTTP, URL, tipos MIME y XML
- ❑ Deben soportar obligatoriamente el método GET. El método POST es obligatorio para WFS pero opcional para WMS.
- ❑ Se define la operación GetCapabilities para todos los protocolos.



OGC Web Services (OWS)

❑ Operación GetCapabilities

- Permite que el cliente conozca las capacidades del servidor
- Devuelve la metadata del servicio (ej. datos que posee, formatos que soporta, valores admitidos de los parámetros, etc.)
- Permite operación de “binding”
- No soporta WSDL.
- Parámetros de la Solicitud:
 - Service: el tipo de servicio que se desea (ej. WMS). Un mismo servidor puede soportar varios servicios.
 - Request: el nombre de la operación (GetCapabilities, en este caso)
 - Version: el número de versión del protocolo que se está consultando. Las versiones tienen el formato x.y.z.



OGC Web Services (OWS)

- **AcceptVersions:** versiones del protocolo que soporta el cliente, en orden de preferencia. El cliente y el servidor negocian la versión del protocolo. El servidor elige una versión dentro del AcceptVersions y se la envía al cliente. Si no soporta ninguna de las versiones, envía una excepción.
- **Sections:** secciones de la metadata del servicio que se deben incluir en la respuesta (ServiceIdentification, ServiceProvider, OperationsMetadata, Contents, All).
- **UpdateSequence:** permite que el cliente averigüe si su caché es consistente con el servidor.
- **AcceptFormats:** formatos de respuesta (tipos MIME) que soporta el cliente, en orden de preferencia. Por omisión es “text/XML”.



OGC Web Services (OWS)

- Ejemplo de GetCapabilities en KVP:

`http://hostname:port/path?SERVICE=WCS&REQUEST=GetCapabilities&ACCEPTVERSIONS=1.0.0,0.8.3&SECTIONS=Contents&UPDATESEQUENCE=XYZ123&ACCEPTFORMATS=text/xml`

- Ejemplo de GetCapabilities en XML:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<GetCapabilities xmlns="http://www.opengis.net/ows/1.1" xmlns:ows="http://www.opengis.net/ows/1.1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/ows/1.1 fragmentGetCapabilitiesRequest.xsd" service="WCS"
  updateSequence="XYZ123">
```

```
<AcceptVersions><Version>1.0.0</Version><Version>0.8.3</Version></AcceptVersions>
```

```
<Sections><Section>Contents</Section></Sections>
```

```
<AcceptFormats><OutputFormat>text/xml</OutputFormat></AcceptFormats>
```

```
</GetCapabilities>
```



OGC Web Services (OWS)

- Parámetros de la Respuesta:
 - Version: el número de versión del protocolo en que se envía la respuesta.
 - UpdateSequence: informa al cliente si hubo actualizaciones.
- Secciones de la Respuesta:
 - ServiceIdentification: metadata del servidor (tipo de servicio, version del servicio, título del servidor, descripción del servidor, palabras clave, costos, etc).
 - ServiceProvider: metadata del operador del servidor (nombre, sitio Web, información de contacto, etc.)
 - OperationsMetadata: metadata de cada operación definida para este servicio y soportada por el servidor, incluyendo las URLs de las mismas.



OGC Web Services (OWS)

- ❑ Contents: metadata de los juegos de datos (datasets) existentes y opcionalmente los bounding boxes de los mismos. El servidor no tiene obligación de incluir información de todos sus juegos de datos, por lo que podría devolver una referencia a un servicio de catálogos (un CSW). Ej. un WCS podría agregar miles de nuevos rasters por día, por lo que sería inapropiado incluirlos todos en la respuesta.



Web Map Service (WMS)

- ❑ Define un protocolo para obtener mapas dinámicos a partir de información geográfica distribuida.
- ❑ El mapa que se obtiene es una archivo de imagen (PNG, JPEG, etc.)



Web Map Service (WMS)

- ❑ Operación GetCapabilities: Dentro de los elementos específicos de WMS en la respuesta de esta operación se incluyen los siguientes parámetros:
 - LayerLimit: Cantidad máxima de capas que pueden pedirse.
 - MaxWidth, MaxHeight: Ancho y Altura máximo (en pixels) del mapa que puede pedirse.
 - Layer: Elemento con metadata sobre cada capa geográfica. Los layers se anidan entre sí para heredar propiedades (Ej. Todos los layers que usan la misma proyección pueden anidarse dentro de un mismo layer padre que define la misma).
- Parámetros:
 - **Title:** Título de la capa. No se utiliza con GetMap.
 - **Name:** Nombre de la capa. Se utiliza con GetMap. Un capa con *title* pero sin *name* solo sirve como agrupación formal de capas.



Web Map Service (WMS)

- **Style:** Estilo visual soportado por la capa.
- **EX_GeographicBoundingBox:** Rectángulo delimitador en latitud/longitud. Si la capa no se encuentra en coordenadas geográficas, los valores pueden ser aproximados.
- **CRS:** Coordinate Reference System, Sistema de Referencia de la Capa. Anteriormente llamado SRS. Puede haber varios por capa.
- **BoundingBox:** Rectángulo delimitador en determinado CRS. Puede haber varios por capa. Cada capa debe tener al menos uno en su CRS nativo (en el que están almacenados los datos).
- **MetadataURL:** URL para obtener metadatos sobre la capa.
- **Queryable:** Indica si se soporta la operación GetFeatureInfo para esta capa.
- **Cascaded:** Indica si la capa se obtuvo de otro servidor.



Web Map Service (WMS)

❑ Operación GetMap

- Es la operación básica que solicita un mapa a partir de ciertas capas.
- Parámetros:
 - **Layers:** Lista de nombres de capas que se deben superponer en el mapa.
 - **Styles:** Lista de estilos que se aplicaran a las capas.
 - **BBox:** Rectángulo de extensión total de mapa (extent) en coordenadas del CRS.
 - **CRS:** El CRS en el que se especifica el Bbox.
 - **Format:** Formato (tipo MIME) en el que se generará el mapa.
 - **Width, Height:** Ancho y altura en pixels. Si la relación de aspecto no coincide con el BBox, la imagen será deformada.



Web Map Service (WMS)

- **Transparent:** Solicita fondo del mapa transparente.
- **Bgcolor:** Solicita fondo del mapa en un determinado color.
- **Time y Elevation:** En capas que poseen estas dimensiones, permite obtener la capa para un determinado valor de la misma. (WMS-T)



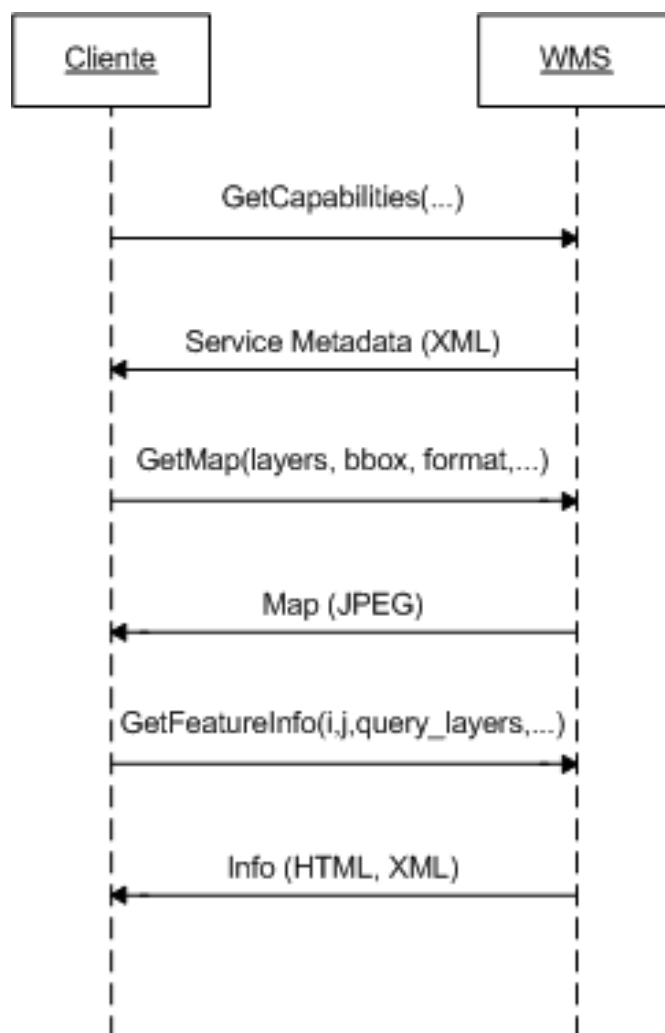
Web Map Service (WMS)

❑ Operación GetFeatureInfo

- Es la operación de consulta de datos de una capa.
- Típicamente se invoca cuando el usuario hace clic sobre el mapa para obtener datos de una o varias entidades (función “identify”).
- Parámetros:
 - **I, J**: Coordenadas de pantalla del punto de interés.
 - **Query_layers**: Nombre de las capas que se quiere identificar.
 - **Info_format**: Formato en el que se desea la respuesta.
 - Además es necesario incluir parámetros de GetMap (BBOX, CRS, WIDTH, HEIGHT), ya que WMS no mantiene estado.

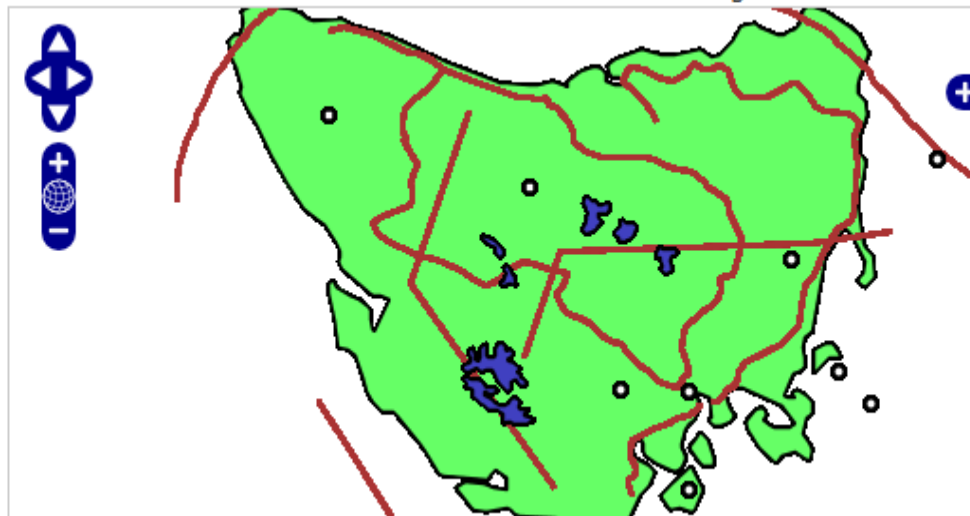


Web Map Service (WMS)



Web Map Service (WMS)

Demonstrates the WMSGetFeatureInfo control for fetching information about a position from WMS (via GetFeatureInfo request).



Tasmania

Click on the map to get feature info.

ATTRIBUTE	TASMANIA_WATER_BODIES.4
AREA	1066494066
PERIMETER	1071999090
WATER_TYPE	Lake
CNTRY_NAME	Australia
CONTINENT	Australia

Web Feature Service (WFS)

- ❑ Define un protocolo para consultar y modificar información geográfica en GML.
- ❑ Operaciones propias: DescribeFeatureType, GetFeature, GetGmlObject, Transaction, LockFeature.
- ❑ Se clasifican los WFS de acuerdo a las operaciones que soportan:
 - WFS Básico: DescribeFeatureType y GetFeature
 - WFS con Xlink: Básico + GetGmlObject
 - WFS Transaccional: Básico + Transaction (+LockFeature)
 - WFS Transaccional con Xlink



Web Feature Service (WFS)

❑ Operación DescribeFeatureType

- Solicita un esquema de descripción de un tipo de entidad geográfica en un determinado lenguaje, por ejemplo GML.
- Parámetros de la Solicitud:
 - TypeName: Nombre del tipo de entidad.
 - OutputFormat: Formato del esquema. Por defecto es GML.
- Respuesta:
 - Esquema de descripción el formato solicitado.



Web Feature Service (WFS)

❑ Operación GetFeature

- Es la operación de consulta (analogía con SELECT)
- Obtiene un conjunto de entidades geográficas.
- Parámetros de la Solicitud:
 - Query
 - **TypeName:** Lista con los nombres de los tipos de entidades que se consultan. Es similar al FROM del un SELECT.
Ej. typeName="ns1:Ciudad=A,ns1:Ciudad=B,ns2:CiudadCostera"
 - **PropertyName:** Propiedad de un tipo de entidad que debe incluirse en la respuesta. Las propiedades obligatorias de un determinado type siempre se incluyen en la respuesta para que el documento se valide (no es necesario especificarlas acá). Es similar a los atributos de un SELECT.
 - **Filter:** Permite imponer restricciones sobre las entidades que se incluyen en la respuesta, por ejemplo, condiciones geográficas (WITHIN, OVERLAP, etc.) Es similar al WHERE.



Web Feature Service (WFS)

- ❑ **OutputFormat:** Formato del esquema de descripción de la respuesta. Los tipos válidos se obtienen mediante GetCapabilities.
- ❑ **ResultType:** Indica si se deben devolver las entidades (Results) o la cantidad de las mismas (Hits). Último caso, similar al COUNT.
- ❑ **MaxFeatures:** Limita el número de entidades devueltas. Similar al LIMIT.



Web Feature Service (WFS)

```
<?xml version="1.0" ?>  
<GetFeature  
  version="1.1.0"  
  service="WFS"  
  handle="Example Query"  
  xmlns="http://www.opengis.net/wfs"  
  xmlns:ogc="http://www.opengis.net/ogc"  
  xmlns:gml="http://www.opengis.net/gml"  
  xmlns:myns="http://www.someserver.com/myns"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:schemaLocation="http://www.opengis.net/wfs ../wfs/1.1.0/WFS.xsd">
```



Web Feature Service (WFS)

```
<Query typeName="myns:Calle">  
  <wfs:PropertyName>myns:geom</wfs:PropertyName>  
  <wfs:PropertyName>myns:nombre</wfs:PropertyName>  
  <ogc:Filter>  
    <ogc:Within>  
      <ogc:PropertyName>myns:path</ogc:PropertyName>  
      <gml:Envelope srsName="EPSG:63266405">  
        <gml:lowerCorner>50 40</gml:lowerCorner>  
        <gml:upperCorner>100 60</gml:upperCorner>  
      </gml:Envelope>  
    </ogc:Within>  
  </ogc:Filter>  
</Query>  
</GetFeature>
```



Web Feature Service (WFS)

❑ Operación GetGmlObject:

- Permite obtener cualquier elemento GML a través de su identificador único (gml:id), que posee todo elemento GML.
- Parámetros de la solicitud:
 - GmlObjectId: ID del objeto
 - OutputFormat: Formato de la respuesta.
 - TraverseXlinkDepth: profundidad de búsqueda de elementos referenciados mediante Xlinks. Si es 1, devuelve solo el elemento identificado, si es 2, realiza una llamada extra (con profundidad 1) por cada elemento referenciado (recursividad). Si es * la profundidad es ilimitada.
 - TraverseXlinkExpiry: tiempo que debe esperar el servidor para obtener respuesta de los otros servidores en consultas recursivas.



Web Feature Service (WFS)

- ❑ Respuesta:
 - ❑ Elemento XML solicitado.
 - ❑ No es un documento XML como en el caso de GetFeature.

- ❑ En el siguiente ejemplo se muestran los resultados con profundidad 1 y luego con profundidad * (se resuelven xlink con profundidad 2 y 3 en este ejemplo).



Web Feature Service (WFS)

```
<?xml version="1.0" encoding="UTF-8"?>
<wfs:GetGmlObject
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs ../wfs/1.1.0/WFS.xsd"
service="WFS"
version="1.1.0"
outputFormat="text/xml; subtype=gml/3.1.1"
traverseXlinkDepth="1"
traverseXlinkExpiry="1">
<ogc:GmlObjectId gml:id="c1"/>
</wfs:GetGmlObject>
```



Web Feature Service (WFS)

```
<Ciudad gml:id="c1">  
<gml:id="c1">  
<gml:name>CiudadGotica</gml:name>  
<gml:directedNode orientation="+" xlink:href="#n1"/>  
</Ciudad>
```



Web Feature Service (WFS)

```
<Ciudad gml:id="c1">
  <gml:name>CiudadGotica</gml:name>
  <gml:directedNode orientation="+">
    <!-- xlink:href="#n1" -->
    <gml:Node gml:id="n1">
      <gml:pointProperty>
        <!-- xlink:href="http://www.ciudadgotica.gov/gps.gml#townHall" -->
        <gml:Point gml:id="townHall" srsName="...">
          <gml:pos>147 234</gml:pos>
        </gml:Point>
      </gml:pointProperty>
    </gml:Node>
  </gml:directedNode>
</Ciudad>
```



Web Feature Service (WFS)

❑ Operación Transaction

- Define una serie de operaciones Insert, Delete y Update.
- El servidor debe hacer un Commit al finalizar, o un Rollback si ocurrió algún error.
- Insert
 - Inserta elementos correspondientes al esquema utilizado, con la geometría expresada en GML.
 - El atributo idgen determina como se generan los gml:id para las nuevas entidades (GenerateNew, UseExisting, ReplaceDuplicate).
- Update
 - Modifica propiedades de elementos existentes.



Web Feature Service (WFS)

- Parámetros:
 - **Property:** permite indicar el nombre de la propiedad y el valor de reemplazo.
 - **TypeName:** permite indicar el tipo de entidad.
 - **Filter:** permite restringir las entidades afectadas por la modificación. Se pueden utilizar operadores geográficos o elementos GmlObjectId.
- Delete
 - Elimina elementos existentes.
 - Parámetros:
 - **TypeName:** permite indicar el tipo de entidad.
 - **Filter:** permite restringir las entidades afectadas por la modificación. Se pueden utilizar operadores geográficos o elementos GmlObjectId.
- Respuesta:
 - Como mínimo, devuelve el número de entidades creadas, modificadas y eliminadas.



Web Feature Service (WFS)

❑ Operación LockFeature

- Permite bloquear una entidad para asegurar la seriabilidad de las transacciones.



Comparación de WS (Para completar)

	W3C	REST	OWS
Operaciones			Fijas (con operaciones opcionales)
Definición de Interfaz	WSDL		
Metadata del servicio			GetCapabilities
Transporte		HTTP	
Seguridad, Transaccionalidad, Orquestación, etc.	WS-*		
Datos binarios	Dentro del XML		



Implementaciones de OWS

- ❑ Libres:
 - MapServer
 - GeoServer
- ❑ Comerciales:
 - ArcGIS Server



Caso de Estudio: MapServer

- ❑ Servidor de mapas libre creado por la Universidad de Minnesota.
- ❑ Escrito en language C
- ❑ Modos de funcionamiento:
 - CGI: Se coloca un ejecutable sobre un servidor Web, usualmente, Apache.
 - API: La API MapScript permite acceder a todas las funcionalidades de MapServer desde otra aplicación. La versión original (y la más utilizada) es para el language PHP, pero existe para varios lenguajes más.



Caso de Estudio: MapServer

❑ Protocolos:

- Nativo
- Compatibilidad con WMS, WFS Básico (sólo lectura), WCS

❑ Configuración de MapServer

- Toda la información sobre las capas se almacena en un archivo de texto plano denominado *mapfile* (extensión .map).
- Cada capa indica su fuente de datos (shapefile, bd geográfica, raster, etc), su proyección, sus clases (clasificación de sus entidades en base a condiciones sobre los atributos), su representación visual (colores, formas, etc), su metadata.
- Para que una capa pueda ser accedida mediante un OWS, se utilizan items especiales en la metadata, dependientes del protocolo.



Referencias

- ❑ WMS

<http://www.opengeospatial.org/standards/wms>

- ❑ WFS

<http://www.opengeospatial.org/standards/wfs>

- ❑ WCS

<http://www.opengeospatial.org/standards/wcs>

- ❑ OWS

<http://www.opengeospatial.org/standards/common>

- ❑ MapServer

<http://mapserver.org/>

