Escola Politècnica Superior
d'Edificació de Barcelona
UNIVERSITAT POLITÈCNICA DE CATALUNYA

INGENIERÍA TÉCNICA TOPOGRÁFICA
PROJECTE FINAL DE CARRERA

APLICACIÓN WEB PARA LA EJECUCIÓN DE TAREAS DE ANÁLISIS SIG REMOTAS (WPS)

ANEJOS

Projectista/es: Álvaro Mateo Machado
Director/s: Juan Carlos González González
Convocatòria: Julio 2013

13. Anejos

13.1 MANUAL DE USUARIO

El usuario accederá vía web a la dirección donde se encuentre instalada la aplicación.

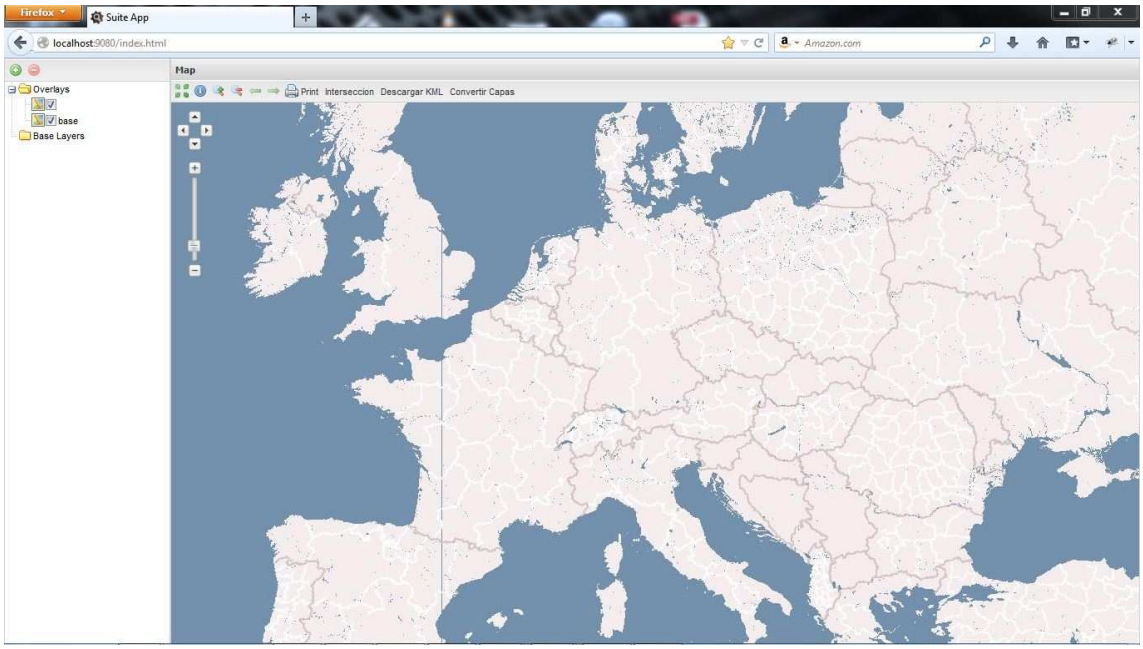


Figura 13.1.1: pantalla inicial programa

El usuario clica al icono  para cargar sus propias capas en geoserver desde el navegador donde aparecerá la ventana emergente “Available Layers”. Esta ventana tiene un botón “Upload Layers” que el usuario clicará para acceder a otra ventana emergente encargada de cargar el archivo en el servidor. El archivo cargado estará en formato comprimido .zip (contendrá como mínimo los archivos con extensión .shp, .shx y .dbf).

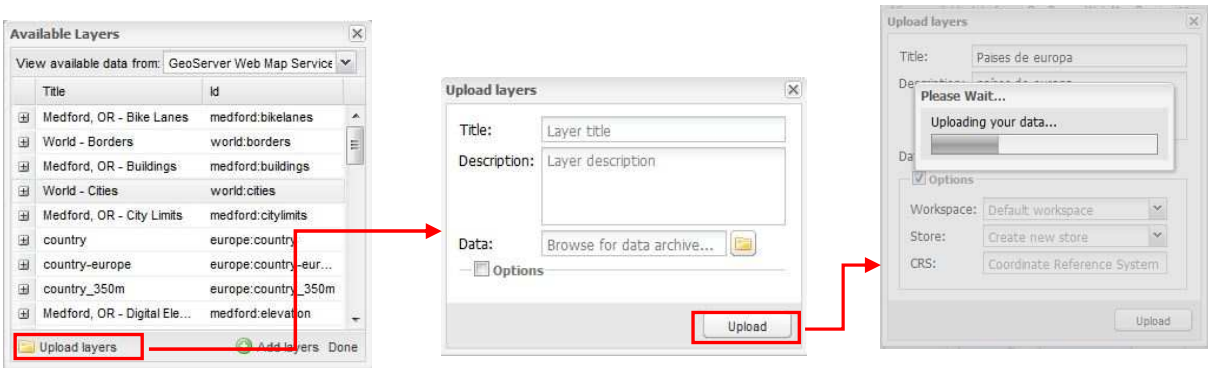


Figura 13.1.2: Secuencia pantallas cargar capas.

Añadimos las capas al visor, mediante el icono  . Se abrirá el cuadro “Available Layers” dónde encontramos las capas a añadir disponibles en el servidor. Seleccionamos la capa “country” y volvemos a clicar en el icono  de esta misma ventana emergente.

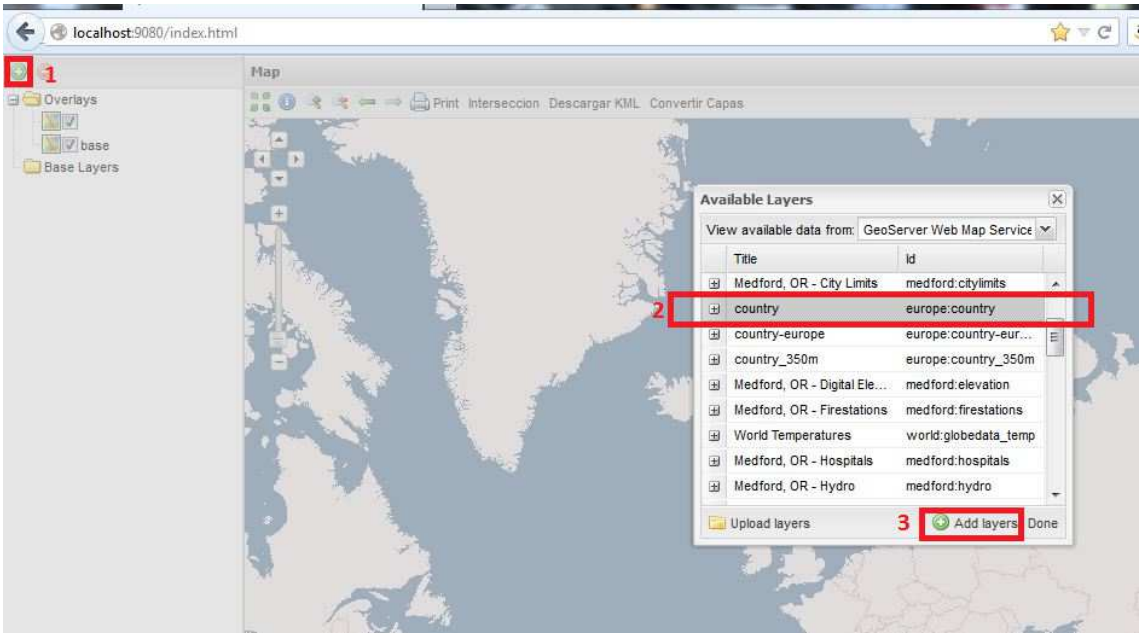


Figura 13.1.3: pantalla añadir capa 1

La capa se añadirá al árbol de capas situado a la izquierda.

Figura 13.1.4: pantalla capa 1 añadida

Igual que en el primer paso añadimos una nueva capa, en este caso se tratará de la capa “mjurban”, que también quedará añadida en el árbol de capas como muestra la siguiente figura.

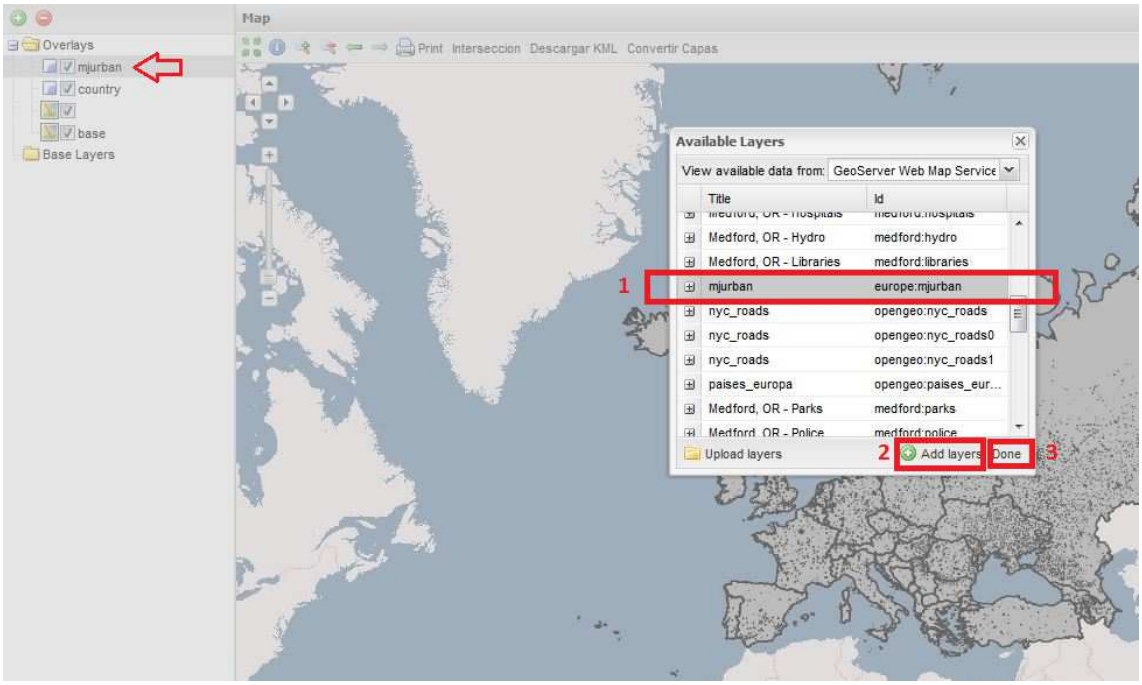
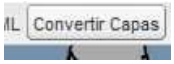


Figura 13.1.5: pantalla añadir capa 2

Una vez tenemos cargadas las capas, necesitamos convertir cada una de sus entidades en objetos seleccionables al ratón, para ello haremos clic en el botón “Convertir Capas” situado en la parte superior del visor → 

Se mostrará un prompt mostrando un mensaje donde pedirá al usuario que indique el nombre de la capa que queremos convertir en seleccionable y clicamos “ok”. Es necesario en este ejemplo indicar como primera capa la primera añadida en el árbol de capas para que no surjan problemas posteriormente en la selección.

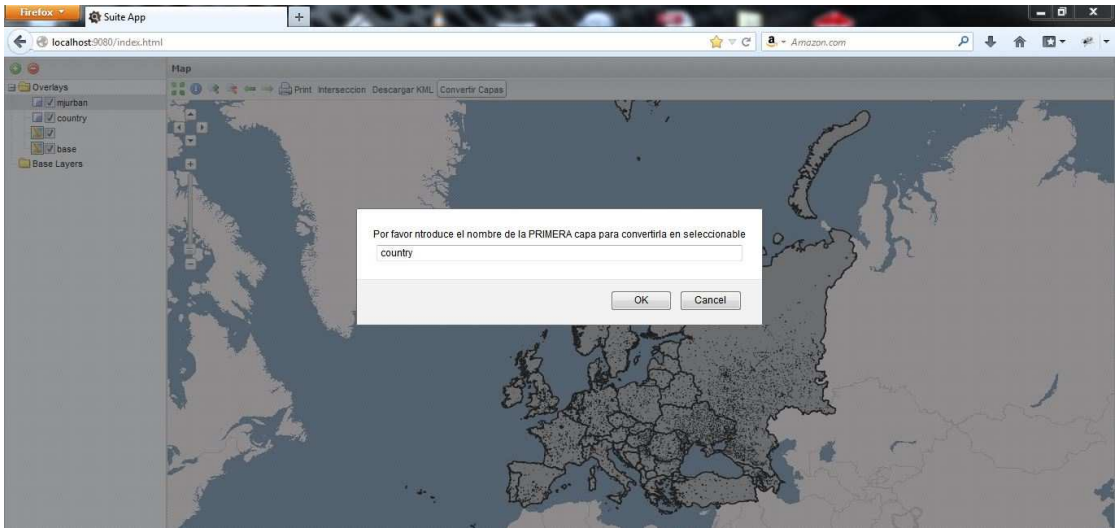


Figura 13.1.6: pantalla convertir capa 1

Hacemos lo mismo con la segunda capa.

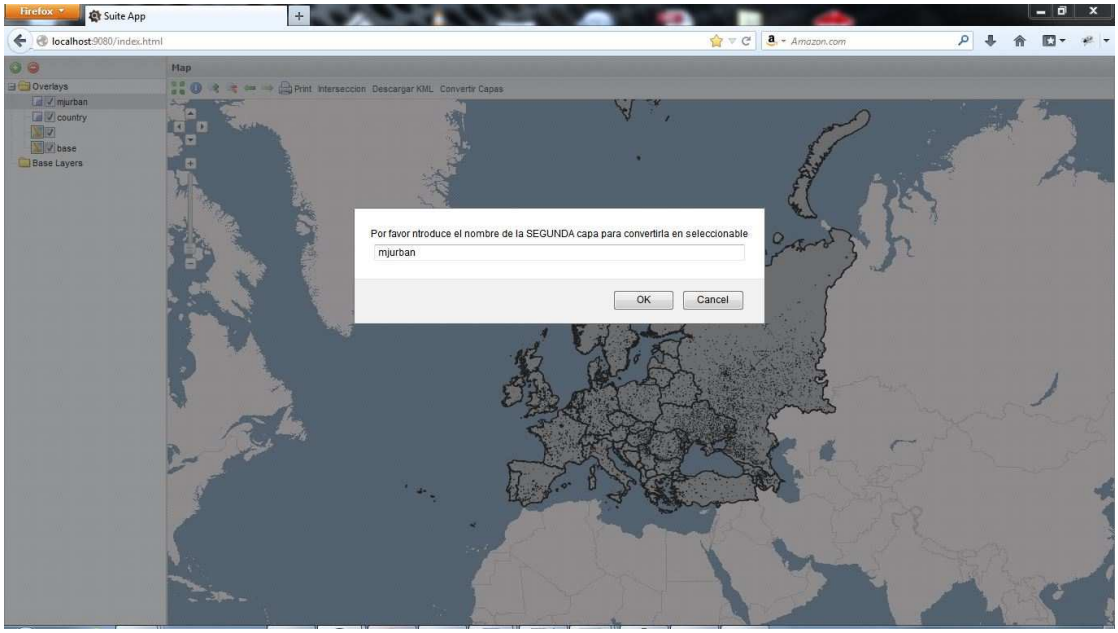


Figura 13.1.7: pantalla convertir capa 2

Esta acción genera 4 capas nuevas. Las iniciales (“country” y “mjurban”) convertidas a WMS que corresponden a la imagen que vemos, y las “Selection *nombredecapa*” que se crean como vectores con un estilo determinado que marcan el tipo de encendido de la entidad cuando ésta es seleccionada, gracias al protocolo WFS.

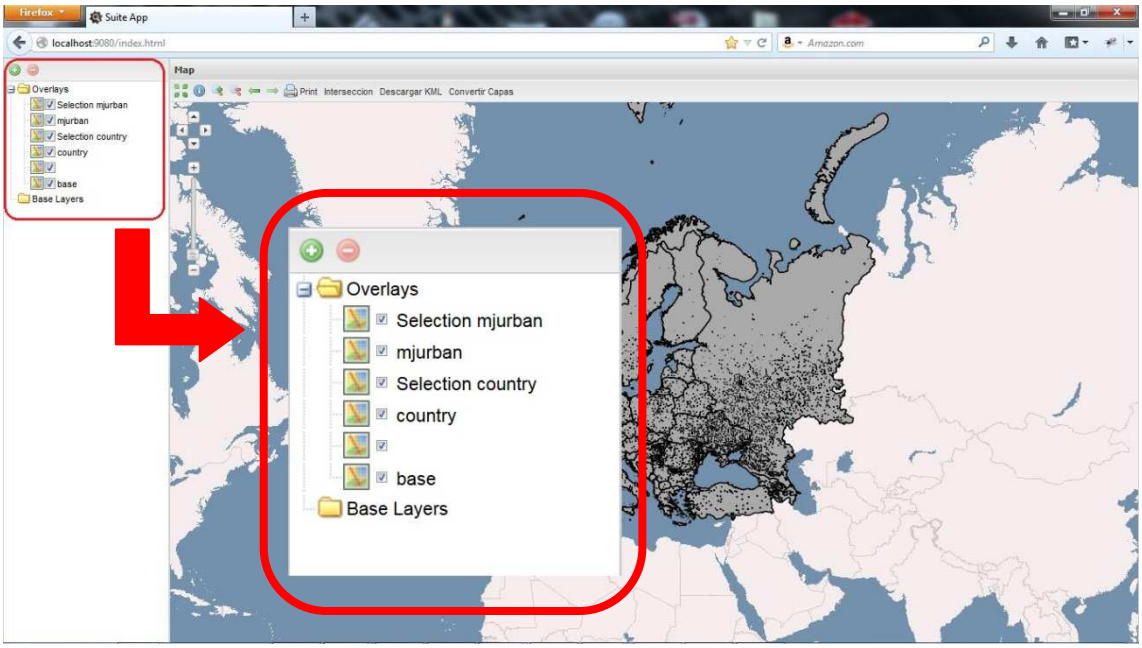


Figura 13.1.8: pantalla capas seleccionables añadidas

Una vez tenemos las capas listas para seleccionar se seleccionan las entidades a las que queremos hacer pasar por el proceso de intersección mediante clics de ratón (existe la multiselección manteniendo la tecla “mayúsculas” pulsada).

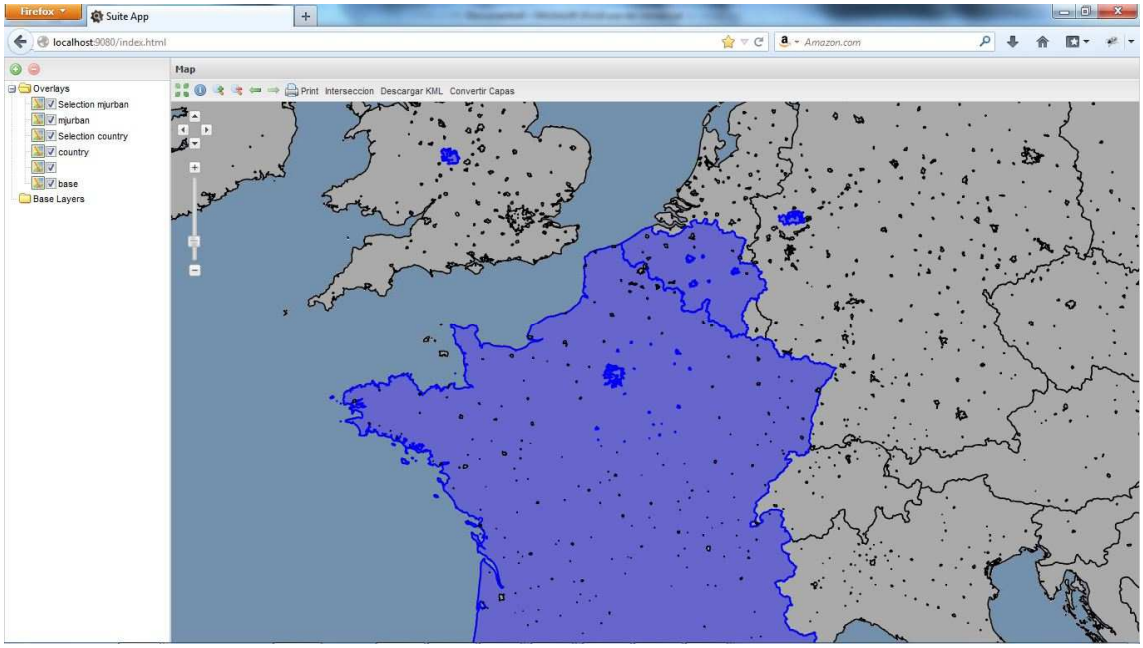
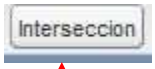


Figura 13.1.9: pantalla seleccionar entidades

Clicamos al botón intersección 

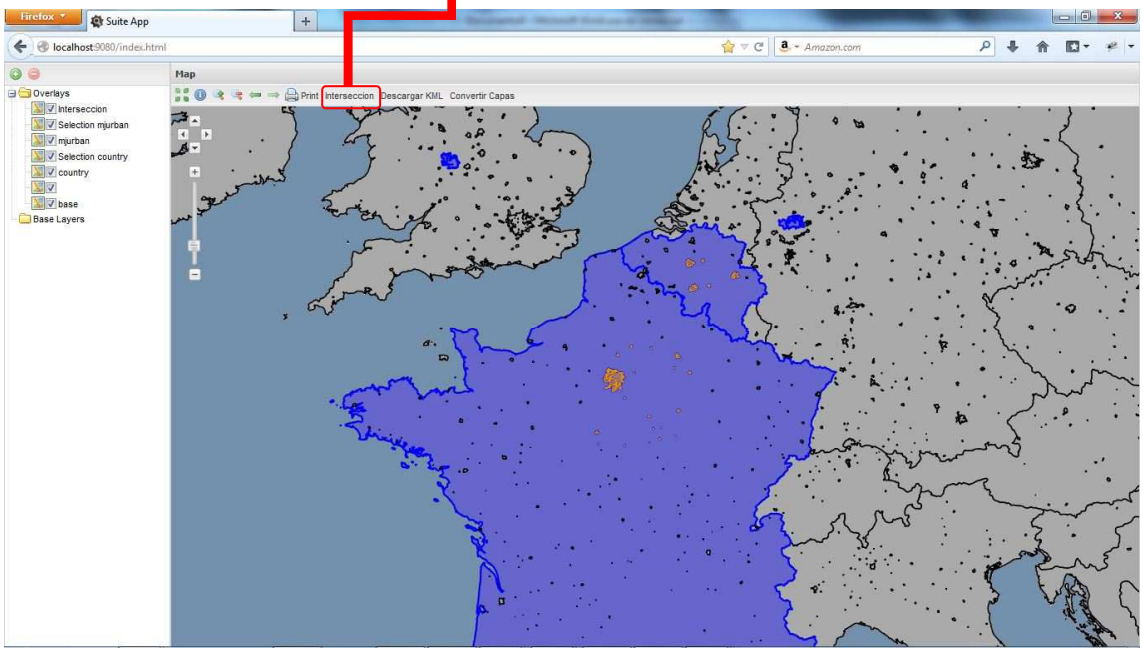


Figura 13.1.10: pantalla resultado entidades intersecadas

Haciendo “zoom” comprobamos las entidades resultantes de la intersección.

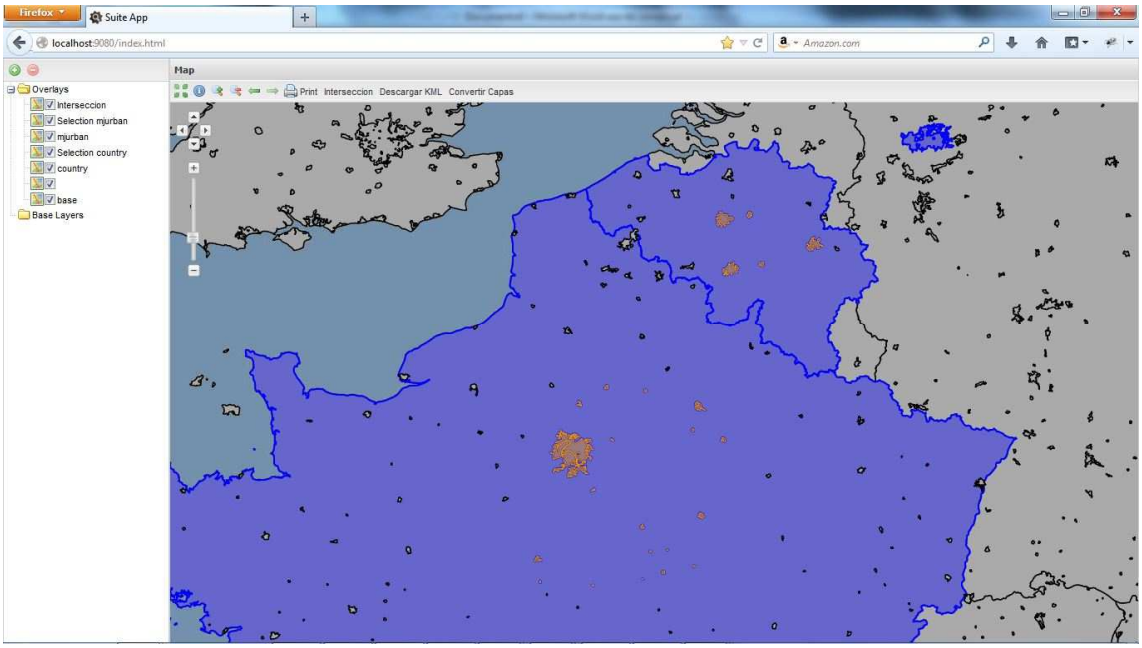


Figura 13.1.11: pantalla resultado entidades intersecadas “zoom”

Las entidades marcadas en amarillo son las comunes entre ambas capas, es decir, son las entidades resultantes en el proceso de intersección.

Estas entidades se añadirán en un nuevo vector o capa resultante llamado “Interseccion”.

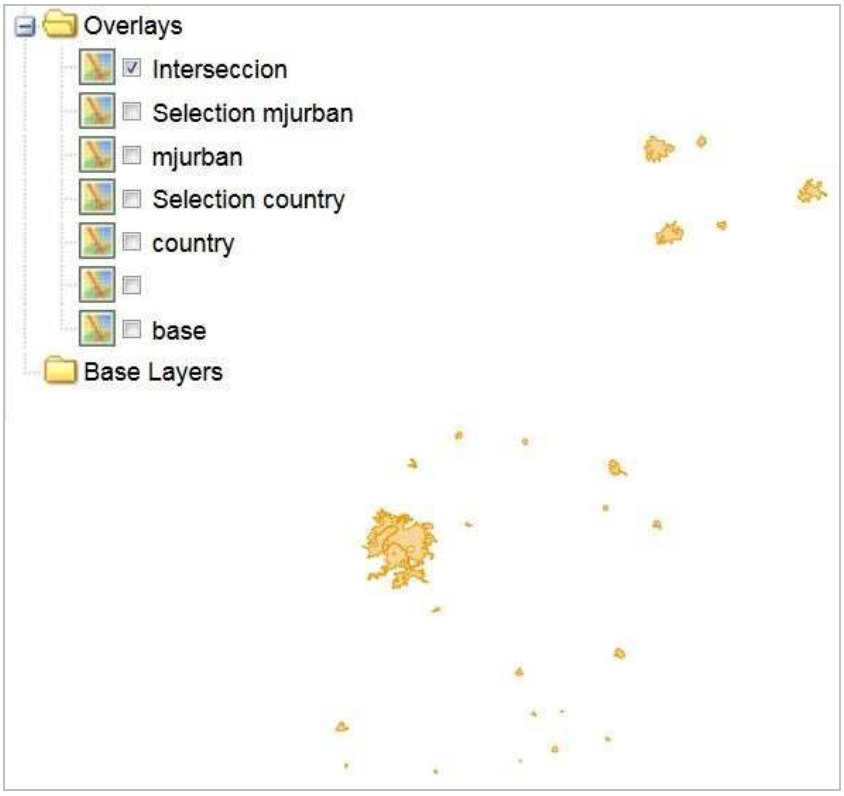


Figura 13.1.12: pantalla capa resultado “Intersección”

Para descargar esta capa en formato KML el usuario debe clicar en el botón "DescargarKML" → Descargar KML El usuario podrá abrirlo o salvarlo en memoria.

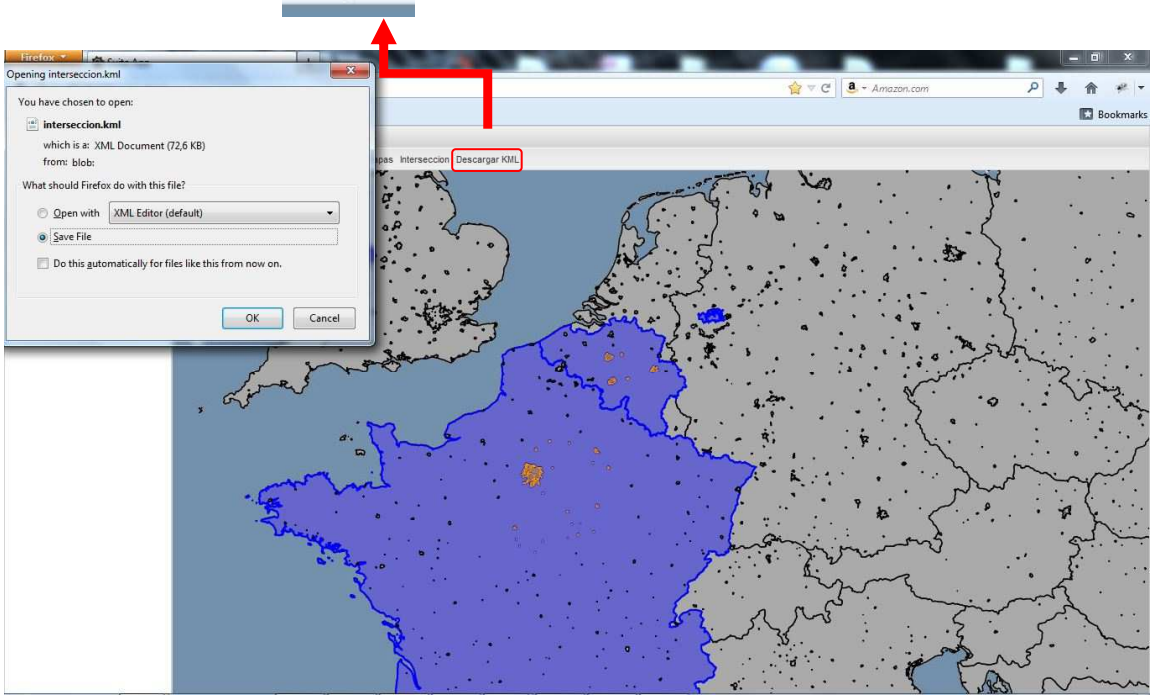


Figura 13.1.13: pantalla capa resultado "Intersección"

13.2 CÓDIGO FUENTE

Código fuente “ConvertirSeleccionable.js”

```
var map = this.target.mapPanel.map;
        map.events.on({
            addlayer: this.raiseLayer,
            scope: this
        });

        // Add action buttons when the viewer is ready
        target.on('ready', function() {


            var actionDefaults = {
                map: target.mapPanel.map,
                enableToggle: true,
                toggleGroup: this.ptype,
                allowDepress: true
            };

            this.addActions([

                new GeoExt.Action({
                    text: 'Convertir Capas',
                    handler:function(evt) {


                        var me = this;
                        var capa1,capa2,layer1, layer2, nombre1, nombre2, fin;
                        fin = 0;
                        nombre1 = prompt("Por favor ntroduce el nombre de la PRIMERA capa para
convertirla en seleccionable","");
                        if (nombre1 != null){
                            nombre2 = prompt("Por favor ntroduce el nombre de la SEGUNDA capa
para convertirla en seleccionable","");
                            if(nombre2 != null){
                                capa1= map.getLayersByName (nombre1);
                                capa2 = map.getLayersByName(nombre2);
                                alert(capa1.geometryType);
                            }
                            else { fin =2}
                        }
                        else{fin = 1 }

                        if (fin == 0){
                        if (capa1.length != 0  && capa2.length != 0){
                            document.capauno="Selection "+nombre1;
                            document.capados="Selection "+nombre2;

                            layer1= new OpenLayers.Layer.WMS(
                                nombre1,
                                "http://localhost:9080/geoserver/wms",
                                {layers: 'europe:'+nombre1, transparent: true,
isBaseLayer:false }
                            );

                            select = new OpenLayers.Layer.Vector("Selection "+nombre1,
{styleMap:
                                new
OpenLayers.Style(OpenLayers.Feature.Vector.style["select"])
                            });

                            map.addLayers([layer1, select]);


                            control = new OpenLayers.Control.GetFeature({
                                protocol: OpenLayers.Protocol.WFS.fromWMSLayer(layer1),
                                box: true,
                                hover: false,
                                //toggle:false,
                                //multiple: false,
                                clickout: true,
                                multipleKey: "shiftKey",
                                toggleKey: "ctrlKey"
```

```
                            });

                            control.events.register("featureselected", this, function(e) {

                                if(!map.getLayersByName('Selection
'+nombre2)[0].features.length) {
                                    //alert(map.getLayersByName("Selection Nucleos
Urbanos")[0].features.length);
                                    select.addFeatures([e.feature]);

//alert(map.getLayersByName("Selection")[0].features[0].id);
                                }

                            });

                            control.events.register("featureunselected", this, function(e) {
                                //alert("pais") ;
                                select.removeFeatures([e.feature]);
                            });

                            map.addControl(control);
                            control.activate();

////////////////////////////////////////////////////////////////////////////////
                            layer2= new OpenLayers.Layer.WMS(
                                nombre2,
                                "http://localhost:9080/geoserver/wms",
                                {layers: 'europe:'+nombre2, transparent: true,
isBaseLayer:false }
                            );

                            select2 = new OpenLayers.Layer.Vector("Selection "+nombre2,
{styleMap:
                                new
OpenLayers.Style(OpenLayers.Feature.Vector.style["select"])
                            });

                            map.addLayers([layer2, select2]);


                            control2 = new OpenLayers.Control.GetFeature({
                                protocol: OpenLayers.Protocol.WFS.fromWMSLayer(layer2),
                                box: true,
                                hover: false,
                                //toggle:false,
                                //multiple: false,
                                clickout: true,
                                multipleKey: "shiftKey",
                                toggleKey: "ctrlKey"

                            });

                            control2.events.register("featureselected", this, function(e) {


                                //alert(map.getLayersByName("Selection Nucleos
Urbanos")[0].features.length);
                                select2.addFeatures([e.feature]);
                                //alert(map.getLayersByName("Selection")[0].features[0].id);


                            });

                            control2.events.register("featureunselected", this, function(e) {
                                //alert("pais") ;
                                select2.removeFeatures([e.feature]);
                            });

                            map.addControl(control2);
                            control2.activate();
                            map.removeLayer(capa1[0]);
                            map.removeLayer(capa2[0]);


                        }
                        else {alert('ERROR: nombre de la capa incorrecto');}
```

```
                    }
                }

            },actionDefaults)
        ]);
    }, this);
},
```

Código fuente "SeleccionaCapas.js"

```
ptype: "myapp_selecciona",

    addActions: function() {


        var map = this.target.mapPanel.map;
            //{allOverlays: false};
            map.events.on({
              addlayer: this.raiseLayer,
              scope: this
            });

           //blayer = new OpenLayers.Layer.OSM("Mapnik") ; //opcion 1
           var blayer = new OpenLayers.Layer.WMS(
                "base",
                "http://vmap0.tiles.osgeo.org/wms/vmap0",
            {layers: 'basic', isBaseLayer:true});
            ////////////////////////////////

           map.addLayers([blayer]);



  },
```

Código fuente "WPSDemoSinCambios.js"

```
var WPSDemo = Ext.extend(gxp.plugins.Tool, {

    ptype: 'app_wpsdemo',

    /** inicio plugin */
    init: function(target) {

        WPSDemo.superclass.init.apply(this, arguments);
        var map = this.target.mapPanel.map;



        // Add action buttons when the viewer is ready
        target.on('ready', function() {

            // Get a reference to the vector layer from app.js
           this.layer = target.getLayerRecordFromMap({
                name: "sketch",
                source: 'ol'
            }).getLayer();
            // Some defaults
            var actionDefaults = {
                map: target.mapPanel.map,
                enableToggle: true,
                toggleGroup: this.ptype,
                allowDepress: true
            };

            this.addActions([

                new GeoExt.Action({
                    text: 'Interseccion',
                    handler:function(evt) {
                        this.wpsClient = new OpenLayers.WPSClient({
                            servers: {
                                local: '/geoserver/wps'
                                //opengeo: 'http://demo.opengeo.org/geoserver/wps'
                            }
                        });
```

```
                        var me = this;
                        var capa1, capa2;
                        capa1 = document.capauno;
                        capa2 = document.capados;
                        var numNucleos = map.getLayersByName(capa2)[0].features.length;
                        var numPais = map.getLayersByName(capa1)[0].features.length ;



                        if ((numNucleos > 0) && (numPais > 0)){
                            var resultado = new OpenLayers.Layer.Vector("Interseccion");
                            var features1 =
map.getLayersByName(capa2)[0].features;//OpenLayer.Feature.Vector
                            var features2 =
map.getLayersByName(capa1)[0].features;//OpenLayer.Feature.Vector

                            var intersection = wpsClient.getProcess('local',
'JTS:intersection');
                            var intersects = wpsClient.getProcess('local', 'JTS:intersects');
                            var s,f1,f2,outputs;

                            var maxpais = features2.length;
                            var maxciudad = features1.length;

                            for (i = 0 ; i < maxpais; ++i){
                                for (j = 0; j < maxciudad; ++j){

                                    f1 = features1[j];
                                    f2 = features2[i];
                                    var salida,salida2,outputs;
                                    me = this;
                                    intersection.execute({
                                        inputs:{
                                            a:f1,
                                            b:f2
                                        },
                                        success: function(salida2) {
                                            if(salida2 != null && salida2.result != null ){
                                                resultado.addFeatures(salida2.result[0]);
                                            }
                                        }
                                    });

                                }
                            }

                            map.addLayer(resultado);
                        }
                        else{ alert("no hay ninguna feature seleccionada!");}
                    }

                },actionDefaults)
            ]);
        }, this);
    },
```

Código fuente DescargarKML.js

```
(function(view) {
"use strict";
var
      Uint8Array = view.Uint8Array
    , HTMLCanvasElement = view.HTMLCanvasElement
    , is_base64_regex = /\s*;\s*base64\s*(?:;|$)/i
    , base64_ranks
    , decode_base64 = function(base64) {
        var
              len = base64.length
            , buffer = new Uint8Array(len / 4 * 3 | 0)
            , i = 0
            , outptr = 0
            , last = [0, 0]
            , state = 0
```

```
            , save = 0
            , rank
            , code
            , undef
        ;
        while (len--) {
            code = base64.charCodeAt(i++);
            rank = base64_ranks[code-43];
            if (rank !== 255 && rank !== undef) {
                last[1] = last[0];
                last[0] = code;
                save = (save << 6) | rank;
                state++;
                if (state === 4) {
                    buffer[outptr++] = save >>> 16;
                    if (last[1] !== 61 /* padding character */) {
                        buffer[outptr++] = save >>> 8;
                    }
                    if (last[0] !== 61 /* padding character */) {
                        buffer[outptr++] = save;
                    }
                    state = 0;
                }
            }
        }
        // 2/3 chance there's going to be some null bytes at the end, but
that
        // doesn't really matter with most image formats.
        // If it somehow matters for you, truncate the buffer up outptr.
        return buffer;
    }
;
if (Uint8Array) {
    base64_ranks = new Uint8Array([
          62, -1, -1, -1, 63, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, -1
        , -1, -1,  0, -1, -1, -1,  0,  1,  2,  3,  4,  5,  6,  7,  8,  9
        , 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
        , -1, -1, -1, -1, -1, -1, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35
        , 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51
    ]);
}
if (HTMLCanvasElement && !HTMLCanvasElement.prototype.toBlob) {
    HTMLCanvasElement.prototype.toBlob = function(callback, type /*,
...args*/) {
          if (!type) {
            type = "image/png";
        } if (this.mozGetAsFile) {
            callback(this.mozGetAsFile("canvas", type));
            return;
        }
        var
              args = Array.prototype.slice.call(arguments, 1)
            , dataURI = this.toDataURL.apply(this, args)
            , header_end = dataURI.indexOf(",")
            , data = dataURI.substring(header_end + 1)
            , is_base64 = is_base64_regex.test(dataURI.substring(0,
header_end))
            , blob
        ;
        if (Blob.fake) {
```

```
            // no reason to decode a data: URI that's just going to become
a data URI again
            blob = new Blob
            if (is_base64) {
                blob.encoding = "base64";
            } else {
                blob.encoding = "URI";
            }
            blob.data = data;
            blob.size = data.length;
        } else if (Uint8Array) {
            if (is_base64) {
                blob = new Blob([decode_base64(data)], {type: type});
            } else {
                blob = new Blob([decodeURIComponent(data)], {type: type});
            }
        }
        callback(blob);
    };
}
}(self));
```

```
/* FileSaver.js
 * A saveAs() FileSaver implementation.
 * 2013-01-23
 *
 * By Eli Grey, http://eligrey.com
 * License: X11/MIT
 *   See LICENSE.md
 */

/*global self */
/*jslint bitwise: true, regexp: true, confusion: true, es5: true, vars:
true, white: true,
  plusplus: true */

/*! @source
http://purl.eligrey.com/github/FileSaver.js/blob/master/FileSaver.js */

var saveAs = saveAs
  || (navigator.msSaveBlob && navigator.msSaveBlob.bind(navigator))
  || (function(view) {
    "use strict";
    var
          doc = view.document
          // only get URL when necessary in case BlobBuilder.js hasn't
overridden it yet
        , get_URL = function() {
            return view.URL || view.webkitURL || view;
        }
        , URL = view.URL || view.webkitURL || view
        , save_link = doc.createElementNS("http://www.w3.org/1999/xhtml",
"a")
        , can_use_save_link = "download" in save_link
        , click = function(node) {
            var event = doc.createEvent("MouseEvents");
```

```
            event.initMouseEvent(
                "click", true, false, view, 0, 0, 0, 0, 0
                , false, false, false, false, 0, null
            );
            node.dispatchEvent(event);
        }
        , webkit_req_fs = view.webkitRequestFileSystem
        , req_fs = view.requestFileSystem || webkit_req_fs ||
view.mozRequestFileSystem
        , throw_outside = function (ex) {
            (view.setImmediate || view.setTimeout)(function() {
                throw ex;
            }, 0);
        }
        , force_saveable_type = "application/octet-stream"
        , fs_min_size = 0
        , deletion_queue = []
        , process_deletion_queue = function() {
            var i = deletion_queue.length;
            while (i--) {
                var file = deletion_queue[i];
                if (typeof file === "string") { // file is an object URL
                    URL.revokeObjectURL(file);
                } else { // file is a File
                    file.remove();
                }
            }
            deletion_queue.length = 0; // clear queue
        }
        , dispatch = function(filesaver, event_types, event) {
            event_types = [].concat(event_types);
            var i = event_types.length;
            while (i--) {
                var listener = filesaver["on" + event_types[i]];
                if (typeof listener === "function") {
                    try {
                        listener.call(filesaver, event || filesaver);
                    } catch (ex) {
                        throw_outside(ex);
                    }
                }
            }
        }
        , FileSaver = function(blob, name) {
            // First try a.download, then web filesystem, then object URLs
            var
                  filesaver = this
                , type = blob.type
                , blob_changed = false
                , object_url
                , target_view
                , get_object_url = function() {
                    var object_url = get_URL().createObjectURL(blob);
                    deletion_queue.push(object_url);
                    return object_url;
                }
                , dispatch_all = function() {
                    dispatch(filesaver, "writestart progress write
writeend".split(" "));
                }
                // on any filesys errors revert to saving with object URLs
```

```
                , fs_error = function() {
                    // don't create more object URLs than needed
                    if (blob_changed || !object_url) {
                        object_url = get_object_url(blob);
                    }
                    if (target_view) {
                        target_view.location.href = object_url;
                    }
                    filesaver.readyState = filesaver.DONE;
                    dispatch_all();
                }
                , abortable = function(func) {
                    return function() {
                        if (filesaver.readyState !== filesaver.DONE) {
                            return func.apply(this, arguments);
                        }
                    };
                }
                , create_if_not_found = {create: true, exclusive: false}
                , slice
            ;
            filesaver.readyState = filesaver.INIT;
            if (!name) {
                name = "download";
            }
            if (can_use_save_link) {
                object_url = get_object_url(blob);
                save_link.href = object_url;
                save_link.download = name;
                click(save_link);
                filesaver.readyState = filesaver.DONE;
                dispatch_all();
                return;
            }
            // Object and web filesystem URLs have a problem saving in
Google Chrome when
            // viewed in a tab, so I force save with application/octet-
stream
            // http://code.google.com/p/chromium/issues/detail?id=91158
            if (view.chrome && type && type !== force_saveable_type) {
                slice = blob.slice || blob.webkitSlice;
                blob = slice.call(blob, 0, blob.size, force_saveable_type);
                blob_changed = true;
            }
            // Since I can't be sure that the guessed media type will
trigger a download
            // in WebKit, I append .download to the filename.
            // https://bugs.webkit.org/show_bug.cgi?id=65440
            if (webkit_req_fs && name !== "download") {
                name += ".download";
            }
            if (type === force_saveable_type || webkit_req_fs) {
                target_view = view;
            } else {
                target_view = view.open();
            }
            if (!req_fs) {
                fs_error();
                return;
            }
            fs_min_size += blob.size;
```

```
            req_fs(view.TEMPORARY, fs_min_size, abortable(function(fs) {
                fs.root.getDirectory("saved", create_if_not_found,
abortable(function(dir) {
                    var save = function() {
                        dir.getFile(name, create_if_not_found,
abortable(function(file) {
                            file.createWriter(abortable(function(writer) {
                                writer.onwriteend = function(event) {
                                    target_view.location.href =
file.toURL();
                                    deletion_queue.push(file);
                                    filesaver.readyState = filesaver.DONE;
                                    dispatch(filesaver, "writeend", event);
                                };
                                writer.onerror = function() {
                                    var error = writer.error;
                                    if (error.code !== error.ABORT_ERR) {
                                        fs_error();
                                    }
                                };
                                "writestart progress write abort".split("
").forEach(function(event) {
                                    writer["on" + event] = filesaver["on" +
event];
                                });
                                writer.write(blob);
                                filesaver.abort = function() {
                                    writer.abort();
                                    filesaver.readyState = filesaver.DONE;
                                };
                                filesaver.readyState = filesaver.WRITING;
                            }), fs_error);
                        }), fs_error);
                    };
                    dir.getFile(name, {create: false},
abortable(function(file) {
                        // delete file if it already exists
                        file.remove();
                        save();
                    }), abortable(function(ex) {
                        if (ex.code === ex.NOT_FOUND_ERR) {
                            save();
                        } else {
                            fs_error();
                        }
                    }));
                }), fs_error);
            }), fs_error);
        }
        , FS_proto = FileSaver.prototype
        , saveAs = function(blob, name) {
            return new FileSaver(blob, name);
        }
    ;
    FS_proto.abort = function() {
        var filesaver = this;
        filesaver.readyState = filesaver.DONE;
        dispatch(filesaver, "abort");
    };
    FS_proto.readyState = FS_proto.INIT = 0;
    FS_proto.WRITING = 1;
```

```
    FS_proto.DONE = 2;

    FS_proto.error =
    FS_proto.onwritestart =
    FS_proto.onprogress =
    FS_proto.onwrite =
    FS_proto.onabort =
    FS_proto.onerror =
    FS_proto.onwriteend =
        null;

    view.addEventListener("unload", process_deletion_queue, false);
    return saveAs;
}(self));
```

```
var DescargarKML = Ext.extend(gxp.plugins.Tool, {

    ptype: 'app_descargarkml',

    /** inicio plugin */
    init: function(target) {

        DescargarKML.superclass.init.apply(this, arguments);
        var map = this.target.mapPanel.map;


        // Add action buttons when the viewer is ready
        target.on('ready', function() {

            // Get a reference to the vector layer from app.js
           this.layer = target.getLayerRecordFromMap({
                name: "sketch",
                source: 'ol'
            }).getLayer();
            // Some defaults
            var actionDefaults = {
                map: target.mapPanel.map,
                enableToggle: true,
                toggleGroup: this.ptype,
                allowDepress: true
            };

            this.addActions([

                new GeoExt.Action({
                    text: 'Descargar KML',
                    handler:function(evt) {

                        //si existe la capa intersecci□uiere decir que hay
entidades resultantes, sino no se podr↘escargar
                        var me = this;
                        existe = map.getLayersByName ("Interseccion");
                        if (existe.length != 0){

                            var NumInterseccion =
map.getLayersByName("Interseccion")[0].features.length;
```

```
                        if ((NumInterseccion > 0) ){

                            var format = new OpenLayers.Format.KML({
                                //'maxDepth':10,
                                'extractStyles':true,
                                'internalProjection':
map.baseLayer.projection,
                                'externalProjection': new
OpenLayers.Projection("EPSG:4326")
                            });
                            var s =
format.write(map.getLayersByName("Interseccion")[0].features);
                            alert(s);
                            /* FileSaver.js demo script
                             * 2012-01-23
                             *
                             * By Eli Grey, http://eligrey.com
                             * License: X11/MIT
                             *   See LICENSE.md
                             */

                            /*! @source
http://purl.eligrey.com/github/FileSaver.js/blob/master/demo/demo.js */

                            (function(view) {
                            "use strict";
                            // The canvas drawing portion of the demo
is based off the demo at
                            //
http://www.williammalone.com/articles/create-html5-canvas-javascript-
drawing-app/
                            var
                                  document = view.document
                                , $ = function(id) {
                                    return document.getElementById(id);
                                }
                                , session = view.sessionStorage
                                // only get URL when necessary in case
Blob.js hasn't defined it yet
                                , get_blob = function() {
                                    return view.Blob;
                                }
                            ;
                            var BB = get_blob();
                            saveAs(
                                  new BB(
                                      [s]
                                    , {type: "text/xml;
subtype=gml/3.1.1; charset=" + document.characterSet}
                                )
                                , "interseccion.kml"
                            );
                            }(self));
                        }

                        else{ alert("Error en la descarga: no existen
entidades en la capa Interseccion");}
                    }
                    else {alert('no existe la capa Interseccion');}
                }
```

```
                },actionDefaults)
            ]);
        }, this);
    },


});

//Ext.preg(myapp.plugins.DescargarKML.prototype.ptype,
myapp.plugins.DescargarKML);
Ext.preg(DescargarKML.prototype.ptype, DescargarKML);
```

13.3 MENSAJES ENVIADOS

MENSAJE 1
hola a tots,

sóc nou en aquesta llista, estic fent el projecte final de carrera amb l'aplicació OpenGeo Suite per windows i tinc un dubte a l`hora d'importarels meus shapes amb el postgis importer. En teoria s'hauríen de crear a la meva base de dades dues taules *geometry_columns*, y *spatial_ref_sys, només em crea la taula spatial_ref_sys, es normal?, donarà problemes en un futur?es pot solucionar?

Gràcies
Alvaro.

Hola Álvaro, benvingut a la llista.

La taula spatial_ref_sys de PostGIS no és creada per cap shp, sino que és la taula que desa les diferents projeccions que pots utilitzar (4326, 23030 etc). La taula geometry_columns és un tesaure de les columnes geomètriques de tota la BD. Si, quan importes un .shp se t'hauria de crear una nova taula amb el nom del shape. Si no et funciona bé, poden ser coses molt diferents depenent de la versió i de com ho hagis instalat, per exemple que no tens la BDD amb suport geoespacial (http://postgis.refractions.net/documentation/manual-1.5/ch02.html#id2661925) o problemes de permisos (mira l'usuari amb el que importes té prous permisos). Intenta seguir les instruccions que hagis fet servir al peu de la lletra (desinstala/reinstala si has fet quelcom extrany), mira els logs (quin missatge d'error et dóna?) i googleja quan tinguis algun missatge d'error extrany. OpenGeo Suite sobre Windows hauria de ser molt fàcil.

Salut i sort,
Martí

Si, Álvaro, és normal.
A versions més antigues, es creava la taula geometry_columns per desar quines eren els columnes de les taules de la bbdd que tenien dades georeferenciades. A partir de les últimes versions, això no és així.
Quan una taula té una columna de tipus geometry, el sistema ja ho sap i no li cal desar-ho en una taula especial. És normal i tot et funcionarà bé.
A més, és molt millor, perquè abans s'havia de vigilar sempre que la taula geometry_columns estigués actualitzada.

Roger

MENSAJE 2

geometry_columns table

hi
I'm doing my final year project with the application OpenGeo Suite for windows and i have a doubt in importing shapes with PostGIS importer. In fact it should be created two database tables, which are geometry_columns, and spatial_ref_sys, but the system only just created the table spatial_ref_sys, is it normal?, shall it cause problems in the future? Can i solve it?

Thanks
Alvaro.

Jill Clark
If you are using PostGIS 2.0, geometry_columns is now a view and not a table.

Justin Deoliveira (Employee)
What version of the OpenGEo suite are you running and on what Windows version? When you say "Postgis importer" do you mean following the "Import data" link from the dashboard? Or one of the import utilities that comes directly with postgis?

Álvaro Mateo
Hello Justin, and thanks for reply me.
The version of my OpenGeo is 2.5 and I use it on winXP because I had many problems with Win Vista. When I said "postgis importer" I meant the "Import data" link from the dashboard, sorry for not being more specific in my explanations. Anywhere Jill commented me if i am using postgis2.0
geometry_columns is now a view not a table and i found it.

Thanks you very much.
Alvaro

MENSAJE 3

Álvaro Mateo

how i can add my own maps in the suite?

hi,
i am using app sdk suite to work with my own maps, but i would need to know
how i can add my own maps in the suite? For example, in the suite when you
click the addLayer button (+) a window with the title "availables layers" appears.
Now i get OSM base layer, but how i can do to see the layers that i have stored in geoserver (like when i use geoexplorer in opengeo suite).
I'm modifying the app.js and i guess that the question is in this part of the code, and to contact with the layers that are in geoserver.

```
// layer sources
sources: {
local: {
ptype: "gxp_wmscsource",
url: "/geoserver/wms",
```

version: "1.1.1"
},
osm: {
ptype: "gxp_osmsource"
}
},

I'm sure it can be done, but i don't know how, could you help me please?
Thanks!

Bart van den Eijnden (Employee)
Are you proxying your geoserver with the SDK?

bart-van-den-eijndens-macbook-pro:mobile bartvde$ suite-sdk debug --help
Usage: suite-sdk debug [<options>] <app-path>
Debug an existing application. The <app-path> argument must be the path to an existing application.

List of options:
-l | --local-port port Port for the local debug server. Default is 9080.

-g | --geoserver url URL for a remote GeoServer to proxy. The debug server will make the remote GeoServer available from the '/geoserver' path within the application.

Álvaro Mateo
thanks Bart!

MENSAJE 4
example

hello I'm practicing with "http://workshops.opengeo.org/openlaye..." example to check if I can use it, but I have a problem. When I run the code, an error appears in the firebug of firefox. I have copied the exact same code in the example so I suppose that the problem is not the code. The error displayed says: "array length" in OpenLayers.js, and the layer "buildings" is never introduced. I have the latest version of OpenLayers already installed. Anyone knows why is this happening?

thanks 

Bart van den Eijnden (Employee)
What version of the OpenGeo Suite / GeoServer are you using?

It works for me with OpenGeo Suite 3.0 and the latest OpenLayers.

Álvaro Mateo
Thanks Bart, I solved the problem using my own shp. I do not know why this happened. thanks anyway.
Álvaro Mateo
i 'm using OpenGeo suite 2.5, sorry for late response.

MENSAJE 5

Álvaro Mateo
selected features in app suite

Hi!

I'm trying to get features can be select from my own shp in my app suite. I've practised with the example in your web and it works ok, i've gotten to select several features at the same time and in different layers.

i have mi own shapes load in geoserver and when i add my layers to the suite they appear properly, but where should i add the code to get my new layer can be selected? In the app.js? I mean, is it posible to select entities from own shp when thay are already charged in the app suite environment of opengeo sdk?

Thanks in advance,

Álvaro

Andreas Hocevar (Geospatial Solutions Engineer)
Can you please try to refine your question? I'm afraid I do not understand what you are asking. Maybe add links to the examples you are talking about, and point out if you want to select layers or features and what you want to do with the selected layers/features.

Álvaro Mateo
What I would like to do a selection of different features on my own loaded layers from geoserver, to make an intersection between them and get the map.
I 've gotten to select different features in different layers mixing these examples:http://openlayers.org/dev/examples/se... and http://workshops.opengeo.org/openlaye .... html. but outside the sdk environment ("GeoExplorer").
My question is this: can i do this in the sdk app environment as if it was an application programming in app.js? or it should have done it differently?

Andreas Hocevar (Geospatial Solutions Engineer)
You can do all this in the SDK, preferably by creating your own plugins. We have atutorial section that explains how to create custom plugins and add them to your application.

Álvaro Mateo
thanks Andreas

MENSAJE 6

Álvaro Mateo
slow process

hi experts,

I'm working with openlayers.

When I use Intersects method (geometry class) between two features, the process takes a long time, around 1 minute. And that's only between two features...when I try to do it with more features, it takes ages.

isIntersected = features2[i].geometry.intersects(features1[j].geometry);

I think that there is a problem with the speed of the geoserver , and I would need to optimize it.

I have tried to generalize the layers, but did not work.

Does anybody knows what's happening?...and how to ompitimize it?

I have OpenGeo Suite 2.5 and I work with suite SDK

thanks in advance.

Justin Deoliveira (Employee)
This isn't doing the intersection in geoserver, it is doing it in the browser. My guess for the slow down would be large geometries with lots of vertices but you mentioned that simplifying didn't change anything.

An alternative would be to do the intersection in GeoServer (via wms or wfs request) first before you return features to the client.

Álvaro Mateo
thanks for your reply Justin.

The problem now is, when i call JTS:intersects method in WPSClient.execute, the output seems not to be compatible. This type error is appeared.

TypeError: output.complexOutput is undefined

WPSProcess.js

```
setResponseForm: function(options) {
options = options || {};
var output = this.description.processOutputs[options.outputIndex || 0];

this.description.responseForm = {
```

```
rawDataOutput: {
identifier: output.identifier,
mimeType: this.findMimeType(output.complexOutput.supported.formats,
options.supportedFormats)
}
};
},
```

I think the problem is the output format because when the method tries to call complexOutput property,it should return a Boolean data and does not.(There is no problem when i use "JTS:intersection" process).
Does this mean I have to implement another method "setResponseForm" that uses this type of output? If I have to, can you help me about how can i do it?

thanks in advance.

MENSAJE 7
Álvaro Mateo
asynchronous processes

Hi,
I think I have a problem with the synchronization between processes server customer. I want to synchronize them, in a sequential way. How can I do it?
This is the code:

```
var resultado = new OpenLayers.Layer.Vector("Interseccion");
var intersection = wpsClient.getProcess('local', 'JTS:intersection');
var intersects = wpsClient.getProcess('local', 'JTS:intersects');
var features1 = map.getLayersByName("Selection Nucleos Urbanos")[0].features;
var features2 = map.getLayersByName("Selection")[0].features;
var salida,salida2,outputs;
for (i = 0 ; i < maxpais; ++i){
for (j = 0; j < maxciudad; ++j){

f1 = features1[j];
f2 = features2[i];

intersects.execute2({
inputs:{
a:f1,
b:f2
},
success: function(outputs) {
salida = outputs.result;
}
});
if(salida){
intersection.execute({
inputs:{
a:f1,
b:f2
},
```

```
success: function(salida2) { resultado.addFeatures(salida2.result[0]);
}
});
}
}
map.addLayer(resultado);
```

The problem is that in the first loop "salida" is undefined, because whereas jts:intersects is executing itself, if, is doing the comparison. Besides, whereas this process is running at first time, the variables j & i have been increasing, and they haven't to be increasing.
This problem is because openlayers.request or xmlHttpRequest. Could you help me, please? Thank you

MENSAJE 8

1 LayerUploadPanel not works

- how can i add a layer upload panel tool in my suite sdk? The main problem is when i click "upload layers" button. Appear errors in the firebug. I don't know the reason why those errors appear TypeError:b[e] is not a constructor... TypeError:this.dataStore is undefined:this.dataStore.emptyText = this.dataStoreEmptyText; (in LayerUploadPanel.js line 327). I'm not sure if i haven't added the necessary code to app.js. i want to upload mi own shapes from layeruploadpanel, but i don't how to do it. Anyone could help me please? – user17918

- The LayerUploadPanel has a dependency on FileUploadField.js. To satisfy this dependency, copyFileUploadField.js to your app directory, and add @require FileUploadField.js at the top of your app.js file. – ahocevar

- thanks for answering so quickly, Andreas. That was exactly the problem, but it appeared another one. I can't upload the map with the files .shp .shx .prj .dbf; then appears that message: (!) "source cities.zip is INCOMPLETE". I have tried several times with different files, but it always appears the same message. I can't solve it on my own. Thank you for helping. Álvaro. – user17918

- You can go to the GeoServer Importer UI to see why the source is incomplete. I seem to remember that this can happen when GeoServer does not recognize a PRJ file. – ahocevar

- It is possible, but I have tried upload other maps as nyc_roads.zip and this is uploaded perfectly into my suite and into suite.opengeo.org/geoexplorer/composer but, for example medford_citylimits is uploaded into the composer, but not into my suite. Always appears the next failure "cities.zip is INCOMPLETE", but if I try upload the zip without the file .prj, appears "cities.shp is INCOMPLETE". – user17918

- On the Geoserver Importer UI appear these failures: org.apache.wicket.WicketRuntimeException: Exception in rendering component: [MarkupContainer [Component id = kml.regionateAttribute]] at org.apache.wicket.Component.renderComponent(Component.java:2725) at org.apache.wicket.MarkupContainer.onRender(MarkupContainer.java:1538)..... I thought that it would be a failure, because of the kind of projection EPGS:4326, but it doen't meanto be that. Could be by anything written in the .prj, and the way geoserver has of managing that? – user17918