**GIS and the Spatial Web**

Tutorial 20 – GeoServer 1

Please make every effort to complete this tutorial before next week’s lecture.

**Practical exercise**

1. If you wish install a *localhost* GeoServer on a laptop or home PC consult the slides on Bb. Make sure you have Java Version 8 on your machine. Either head to the website at [www.geoserver.org](http://www.geoserver.org/) to download the latest software, or use the Install link on Bb (I recommend the latter – it is not the latest version, but is much easier to install). Installation is quite self-explanatory and there are many on-line sites to guide you if needed. Once installed, you can log in via the web admin page using **admin / geoserver**

Then create a new Workspace for yourself to use (see the Bb slides for details)

1. To work on our Geoserver, start a browser and enter the URL

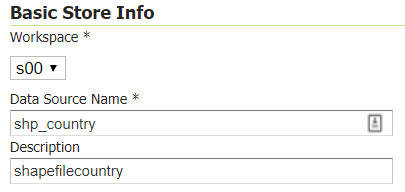
<http://ces-gis.usw.southwales.ac.uk:8080/geoserver/web/>

You need to use the account name and password send to you (e.g. **s00 / gVbeT**).  
Once logged in you have access to your workspace only (called s00, or whatever)

1. Follow the Bb slides to create/set up a ***Store*** in which to place a single shapefile. Select the Shapefile to be used by this store (see Bb slides for details). Use the data pre-loaded for you in the data directory [data\_dir/](http://ces-gis.usw.southwales.ac.uk:8080/geoserver/web/wicket/bookmarkable/org.geoserver.web.data.store.DataAccessEditPage?3-1.ILinkListener-dataStoreForm-parametersPanel-url-dialog-content-breadcrumbs-path-0-pathItemLink&storeName=shp_country&wsName=s00) source\_data/

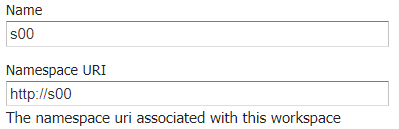
I suggest you start with the Shapefile holding the Outline for Wales (outine.shp).

**Set the Workspace to your workspace, and give the data source a name and description…**

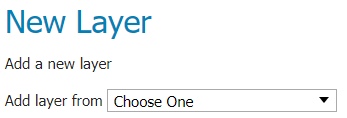


Then Save

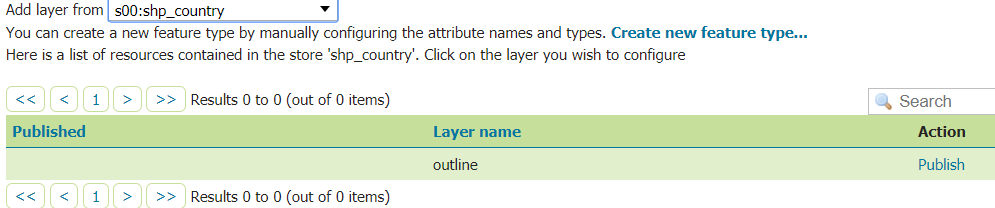
1. Next, we set up a new Layer using this new Store. First click on Workspaces in the left-hand panel, select your worlkspace, and go to the **Namespace URI** textbox.



You must edit the entry here in some way – I suggest you add the “/” character to the end if not already there, or delete the “/” character from the end if it is. Now click Save. This is just a fix for a long-running bug: re-registering the web address internally seems to solve the problem.  
Now click on Layers in the left-hand panel to create and publish a new WMS Map Layer based on the new Shapefile Store. Add a new layer, and select the store that you just created.



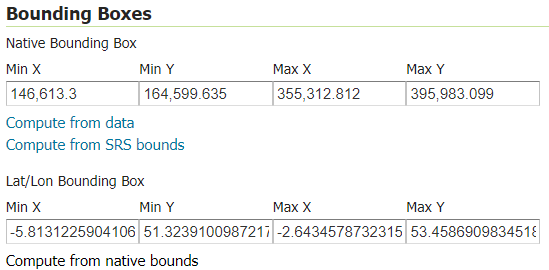
This will result in something like…



Click Publish to configure the Layer’s details (see Bb slides for more information). Set its map extents by clicking on **“Compute** **from data”** item under Bounding Boxes, and then **“Compute from native bounds”** below this.

\*\*If clicking on **“Compute from data”** fails (does not complete the boxes), you have either not performed the Workspace URI rename fix above, or must do so again \*\*

Using **“Compute from data”** should fill with the Layer’s map limits for you, e.g:



Scroll down and click Save.

1. You have now set up a WMS map Layer. Test its availability in all of the following ways:  
     
   (a) Use **Layers Preview** in GeoServer

(b) Using the URL generated above, edit it to generate a direct URL call from the browser. You would not normally do this, but it proves the point that a WMS request simply results in the supply of an image file posted back to the client making the call.

Go to the end of the URL and edit

*&format=application/openlayers*

to read  
 &format=image/png

(c) Create an html script to use the OpenLayers or Leaflet APIs to request the supply of the WMS later: this is the normal way in which you will consume these services…

Leaflet API (details of workspace and layer name need to be edited to suite) code…

*L.tileLayer.wms('http://81.87.34.57:8080/geoserver/s00/wms',{*

*opacity: 1.0,*

*layers: 's00:country',*

*format: 'image/png',*

*transparent: true,*

*attribution: 'from CES-GIS GeoServer'*

*}).addTo(map);*

OpenLayers API (details of workspace and layer name need to be edited to suite) code…

*new ol.layer.Image({*

*source: new ol.source.ImageWMS({*

*url:'http://ces-gis.usw.southwales.ac.uk:8080/geoserver/s00/wms',*

*params: {'LAYERS': 's00:country'},*

*ratio: 1,*

*serverType: 'geoserver'})*

*}) ],*

(d) Use QGIS to issue the WMS request and display the map - this proves that it is not only a web browser that can use a WMS feed. Also, by using QGIS we can go full-circle and display both the original shapefile (vector data) and corresponding raster image delivered via WMS, and see both simultaneously on the same map. This is a little odd, but conceptually interesting. The shapefiles are available on Bb for you to try this.

1. Repeat the process to set up a Layer using the **dentists** and **aonb** shapefiles. Note that you will probably have to return to your Workspace each time and edit its URI (e.g. add or remove the trailing “/”) to get the Bounding Boxes to configure properly.

1. Modify your OpenLayers/Leaflet script to requests two WMS layers (e.g. outine and dentists) as separate map calls, and display appropriately (see my map examples @ <https://ces-web2.southwales.ac.uk/staff/mlangfor/index.html>). Then try to ahieve the same outcome by publishing a **Group Layer** in GeoServer instead (as in my examples).