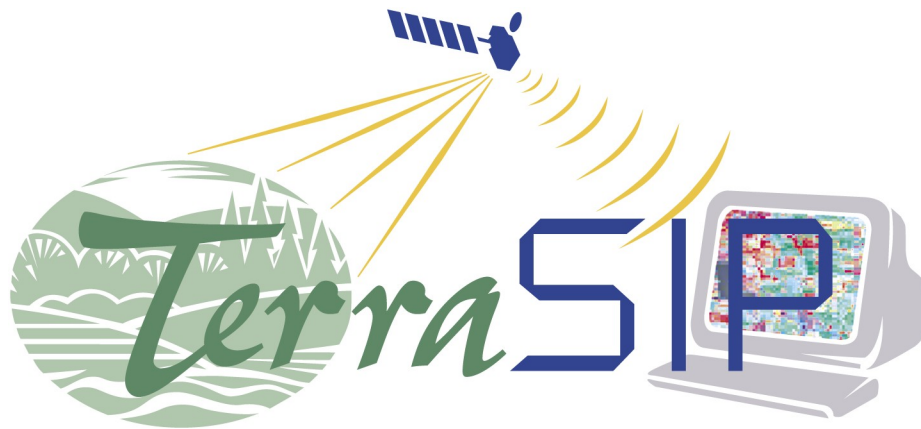


# Getting Started with The UMN MapServer

*Perry Nacionales*  
TerraSIP/MapServer Project  
University of Minnesota



# Outline

---

## MapServer Background

- Brief History
- Features/Capabilities

## Anatomy of a MapServer Application

- Creating a MapServer Application
- A Brief Look at MapScript

## Tools and Resources

- Third Party Client/Management Tools
- Resources

# Background

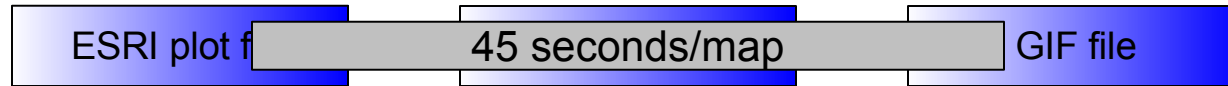
---

A Brief History of MapServer  
(or MapServer's Brief History)

# MapServer: The Past

---

- 1994 - C Program Creates ArcPlot AMLs



- 1994 - NASA-sponsored ForNet Project
- 1997 - *Shapelib* Library (LGPL or “MIT-type”)
- 1997 - NASA-sponsored TerraSIP Project
- 1998 - MapServer 3.2 Released as Open Source
- 2001 - MapServer 3.4
- 2003 - MapServer 4.0
- 2004 - MapServer 4.2

# MapServer: The Present

---

- 30 March, 2006 – MapServer 4.8.3
- 10+ Active Developers Around The World
- 2,100+ Mailing List Subscribers
- 40,000+ Global Applications Deployed
- Hosted Open Source Geospatial 2005 Conference – 3<sup>rd</sup> MapServer Users Meeting
- Open Source Geospatial (OSGeo) Foundation – Founding Member
- Supports OGC Web Services
- Supports OPeNDAP

# MapServer: The Future

---

- Continue to support OGC Standards
  - Sensor Observation Service (SOS)
- FOSS4G2006 – 4<sup>th</sup> MapServer Users Meeting
  - <http://www.foss4g2006.org/>

# Background

---

MapServer Features and Capabilities

# What isn't MapServer

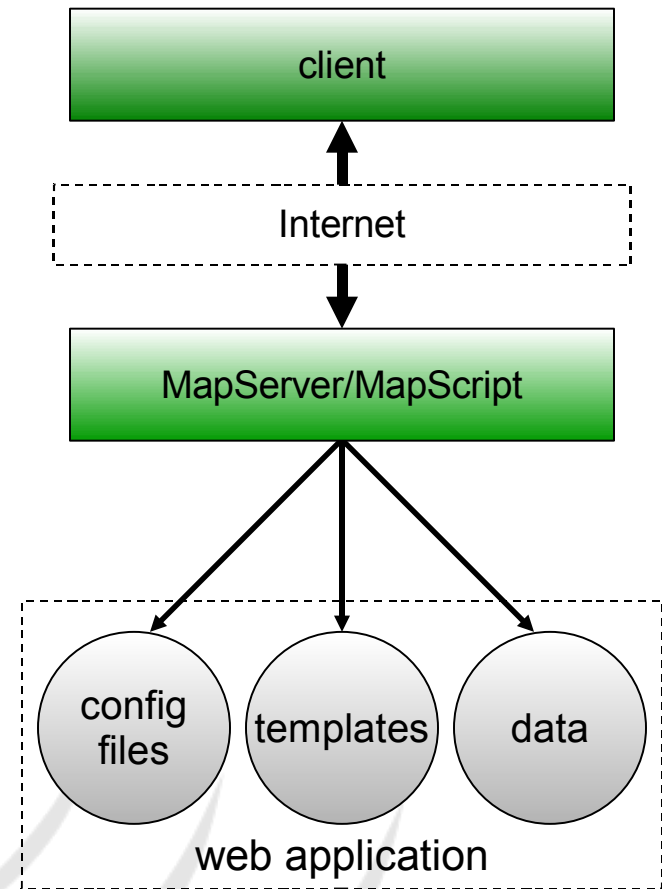
---

- A Desktop GIS!
- A Ready-To-Use Application (e.g. parcel mapping)
- Client-Based (e.g. isn't installed in your computer)
- Easy-To-Use, “Wizard” Configurable
- Expensive!



# What is MapServer?

- Web Mapping Application Development System
- Server-Based
- Open Source
- Supports OGC Web Services Specifications
- *Fast*
- Extremely Configurable
- Easy-To-Use



# MapServer Features/Capabilities

---

- Input: Supports 24+ vector formats
  - PostGIS, ORACLE Spatial, SDE geospatial databases
- Input: Supports even more raster formats
- Output: GIF, PNG (8- and 24-bits), TIFF/GeoTIFF, SVG, SWF, etc.
- Supports OGC WMS (Client/Server), WFS (Client/Server), and WCS (Server)
- Supports OPeNDAP (Client)
- MapScript: PHP, Python, Perl, Ruby, Java, C#
- Cross-platform (Linux/Unix, Windows, Mac, etc.)
- On-the-fly Reprojection
- TrueType fonts
- Advanced Labeling with Collision Mediation
- Thematic Mapping via Logical/Regular Expressions

# Creating A MapServer Application

---

The Way to Web Mapping Nirvana

# Required and Optional Software

---

- Any HTTP server that supports the CGI protocol
  - Apache, IIS, Tomcat, etc.
- MapServer CGI program and utilities
  - shp2img, shptree, sortshp, etc.
- Other geospatial programs/utilities
  - GDAL/OGR utilities (gdalinfo, ogrinfo, ogr2ogr, etc.)
  - Proj.4 utilities (proj, cs2cs)
  - GRASS, QGIS, etc.

# Application Files and Directories

---

- Application Directory: `/data/www/ms101`
  - Map and HTML template file are located here
  - Demo Application template file: `ms101.html`
  - Demo Application map file: `ms101.map`
- Web Root Directory: `/data/www/ms101/webdocs`
  - HTML files (`index.html`) and web-readable images
  - Web Root Alias (Virtual Directory): `/ms101`
  - Application URL: `http://localhost/ms101/index.html`
- Temporary Files Directory: `/data/www/tmp`
  - Files created by MapServer goes here
  - Alias: `/tmp`
- MapServer CGI Program: `/cgi-bin/mapserv(.exe)`

# Application Virtual Directories (Aliases)

---

- Apache Application Directory Alias:

```
Alias /ms101 "/data/www/ms101/webdocs"
```

```
<Directory "/data/www/ms101/webdocs">  
    AllowOverride None  
    Options Indexes FollowSymLinks  
    Order allow,deny  
    Allow from all  
</Directory>
```

- Apache Temporary Directory Alias:

```
Alias /tmp "/data/www/tmp"
```

```
<Directory "/data/www/tmp">  
    Order allow,deny  
    Allow from all  
</Directory>
```

# Example 1: The Map Object

---

- Open “mapfile.map” with a text editor

```
MAP
```

```
NAME 'appID'
```

```
EXTENT -127.460342 22.474929 -64.458549 52.448773
```

```
SIZE 600 400
```

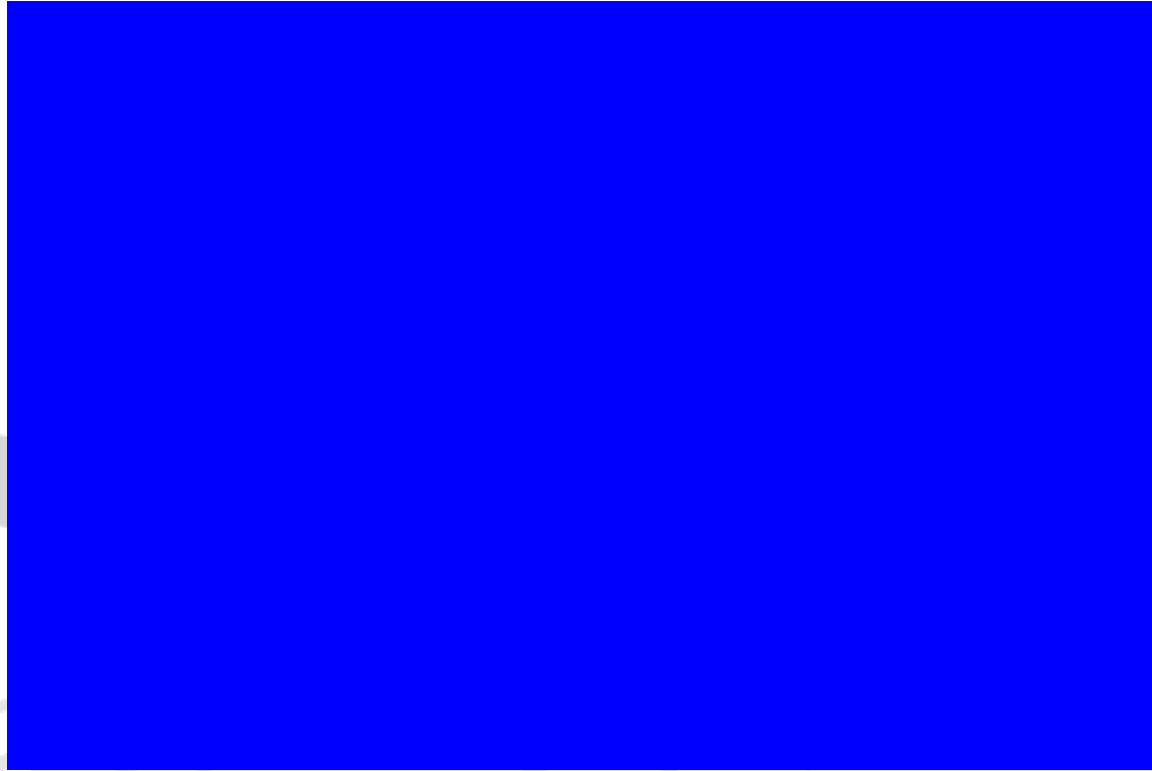
```
IMAGECOLOR 0 0 255
```

```
END
```

- Keyword MAP denotes start of map object
- NAME is the application ID—used as prefix to images created by application
- EXTENT is the output map extent
- SIZE is the width and height of output image in pixels
- IMAGECOLOR defines the background map color
- END denotes the end of the map object
- To View map, open URL in browser:  
<http://localhost/cgi-bin/mapserv?map=/data/www/ms101/mapfile.map&mode=map>
- **Map File Reference** for more information

# Example 1: The MapServer Request

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/mapfile.map&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/mapfile.map&mode=map)



# Example 2: Adding a Polygon Layer

---

- Open “layer\_polygon.map”

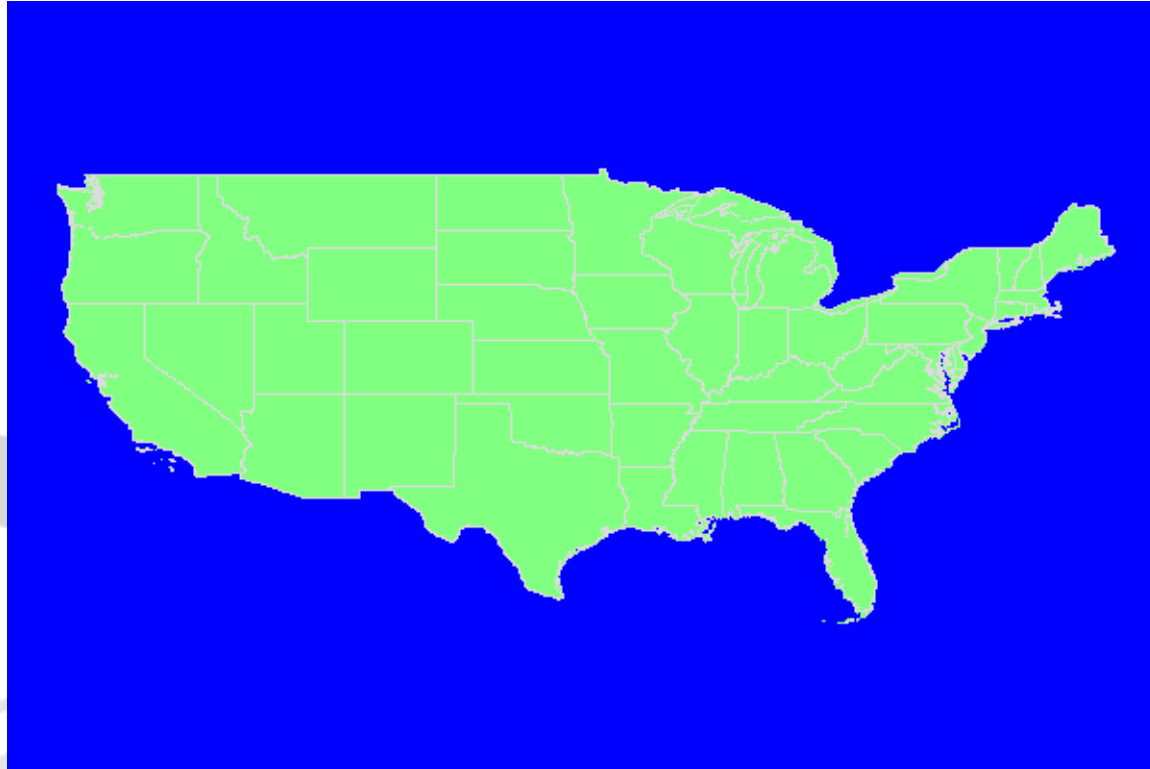
```
...
  LAYER
    NAME 'states'
    DATA 'data/states_conus_geo.shp'
    TYPE POLYGON
    STATUS OFF

    CLASS
      NAME 'States'
      STYLE
        COLOR 128 255 128
        #OUTLINECOLOR 220 220 220
      END
    END
  END
...
```

- View the Map
- [Vector Data Access](#) for more information

# Example 2 Output

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_polygon.map&layer=states&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_polygon.map&layer=states&mode=map)

# Example 3: Adding a Line Layer

---

- Open “layer\_line.map”

```
...
  LAYER
    NAME 'states'
    DATA 'data/states_conus_geo.shp'
    TYPE LINE
    STATUS OFF

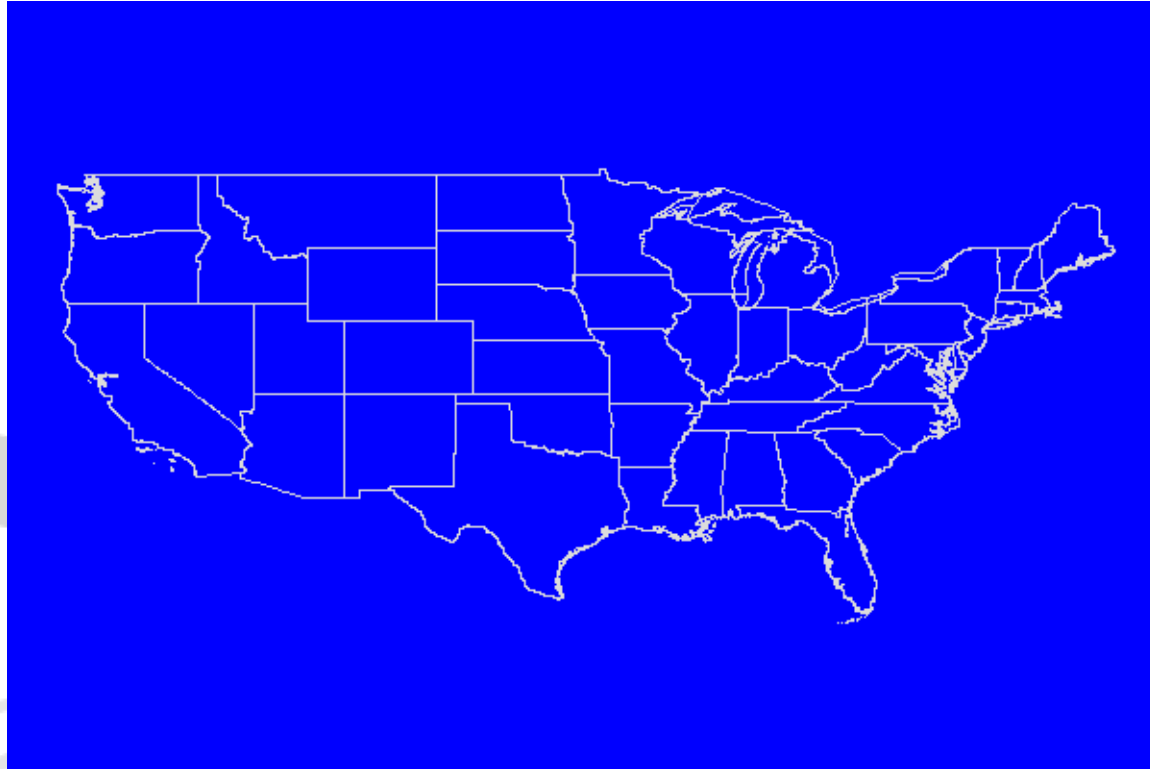
    CLASS
      NAME 'States'
      STYLE
        COLOR 220 220 220
      END
    END
  END
...

```

- View the Map

# Example 3 Output

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_line.map&layer=states&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_line.map&layer=states&mode=map)

# Example 4: Adding an Annotation (Label) Layer

## ■ Open “layer\_line.map”

```
...
FONTSET
'/data/www/ms101/fonts/fonts.list'

LAYER
  NAME 'states'
  DATA
'data/states_conus_geo.shp'
  TYPE ANNOTATION
  STATUS OFF

LABELITEM 'STATE'
CLASS
  NAME 'States'
  STYLE
    COLOR -1 -1 -1
  END
...
```

```
...
LABEL
  COLOR 32 32 32
  OUTLINECOLOR 244 244 244
  SIZE 8
  TYPE TRUETYPE
  FONT 'vera_sans-bold'
  ANTIALIAS TRUE
  MINDISTANCE 150
END
END
END
...
```

## ■ View the Map

# Example 4 Output



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_label.map&layer=states&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_label.map&layer=states&mode=map)

# Example 5: Adding a Point Layer

## ■ Open “layer\_point.map”

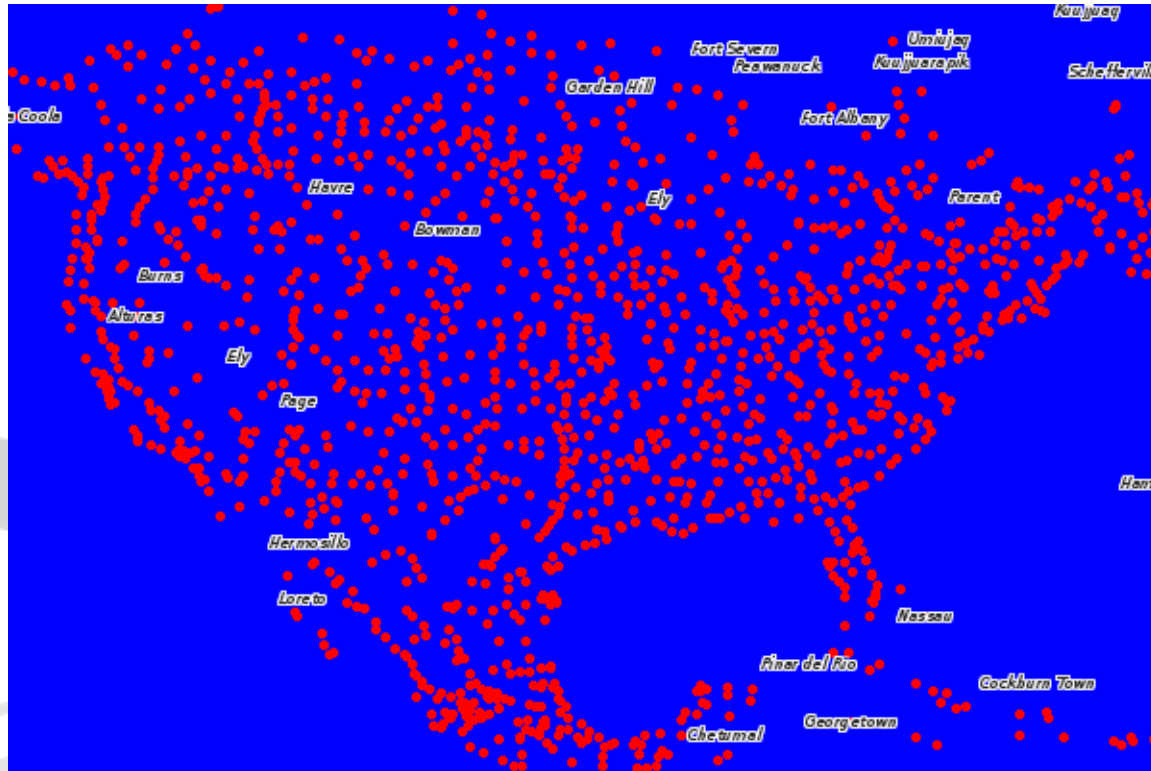
```
...  
FONTSET 'fonts/fonts.list'  
SYMBOLSET 'symbols/symbols.txt'
```

```
LAYER  
  NAME 'cities'  
  DATA 'data/pop_pnt.shp'  
  TYPE POINT  
  STATUS OFF  
  LABELITEM 'NAME'  
  CLASS  
    NAME 'States'  
    STYLE  
      COLOR 255 0 0  
      SYMBOL 'circle'  
      SIZE 5  
    END  
  END
```

```
...  
  LABEL  
    COLOR 32 32 32  
    OUTLINECOLOR 244 244 244  
    SIZE 6  
    TYPE TRUETYPE  
    FONT 'vera_sans-italic'  
    ANTIALIAS TRUE  
  END  
END  
...
```

## ■ View the Map

# Example 5 Output



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_point.map&layer=cities&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_point.map&layer=cities&mode=map)



# Example 6: On-the-fly Reprojection

---

- Open “projection.map”

MAP

```
...  
EXTENT -2501519 -2113908 2966025 1038597 #laea  
UNITS meters
```

```
PROJECTION  
  "init=epsg:2163"  
END
```

LAYER

```
...  
PROJECTION  
  "init=epsg:4326"  
END
```

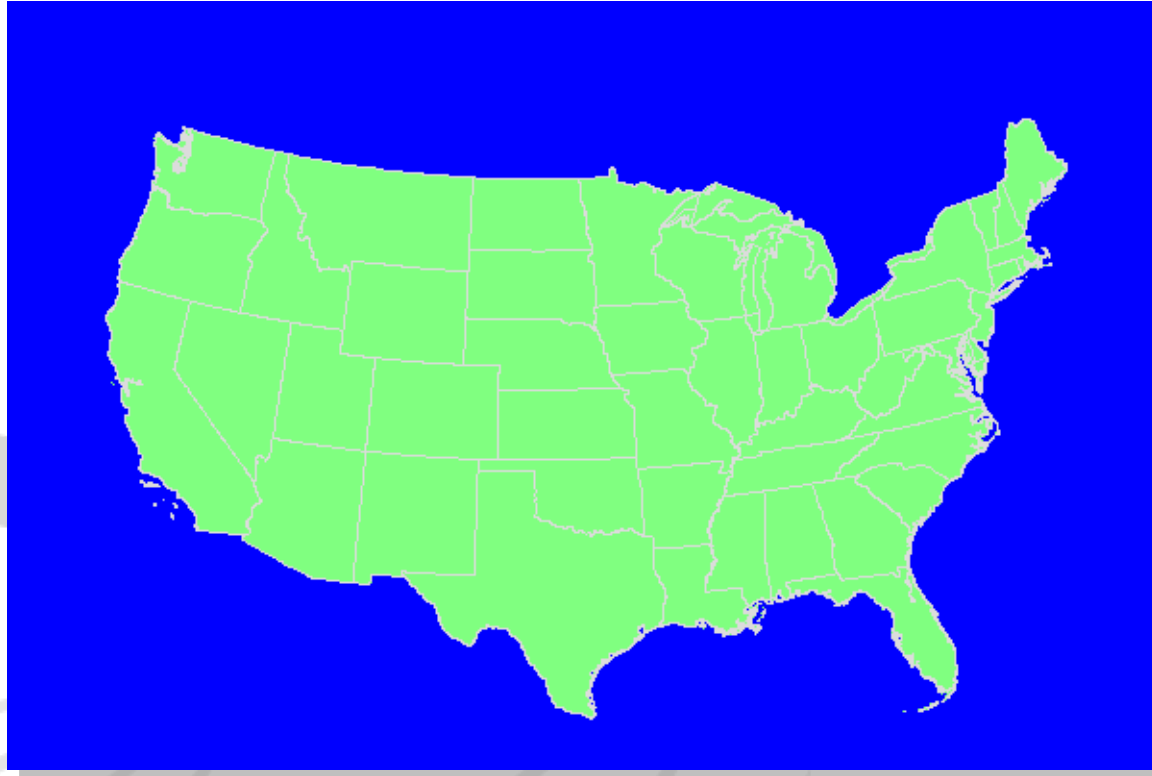
END

END

- View The Map

# Example 6 Output

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/projection.map&layer=states&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/projection.map&layer=states&mode=map)

# Example 7: Defining Classes

---

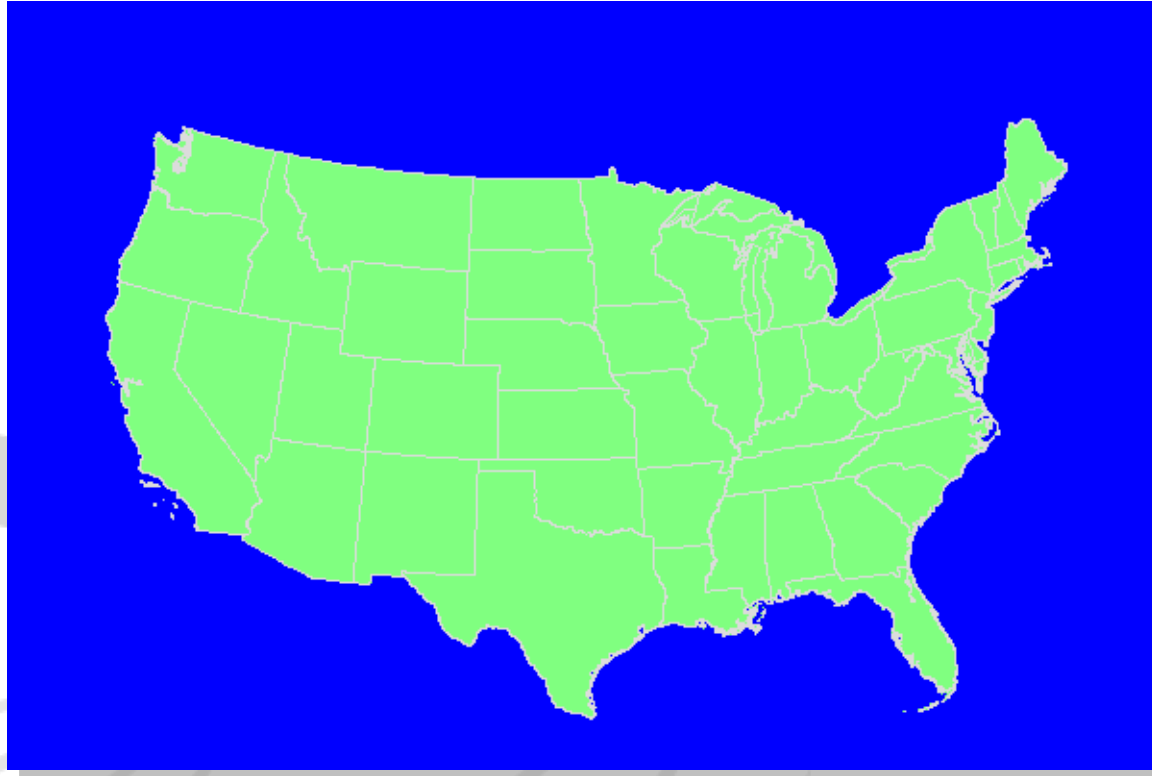
- Open “layer\_class.map”

```
...
CLASSITEM 'CAT'
CLASS
  NAME 'States'
  EXPRESSION 'Land'
  STYLE
    COLOR 128 255 128
  END
END
CLASS
  EXPRESSION 'Water'
  STYLE
    COLOR 128 128 255
  END
END
...
```

- View The Map

# Example 7 Output

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_class.map&layer=states&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_class.map&layer=states&mode=map)

# Example 8: Styles and Antialiasing

---

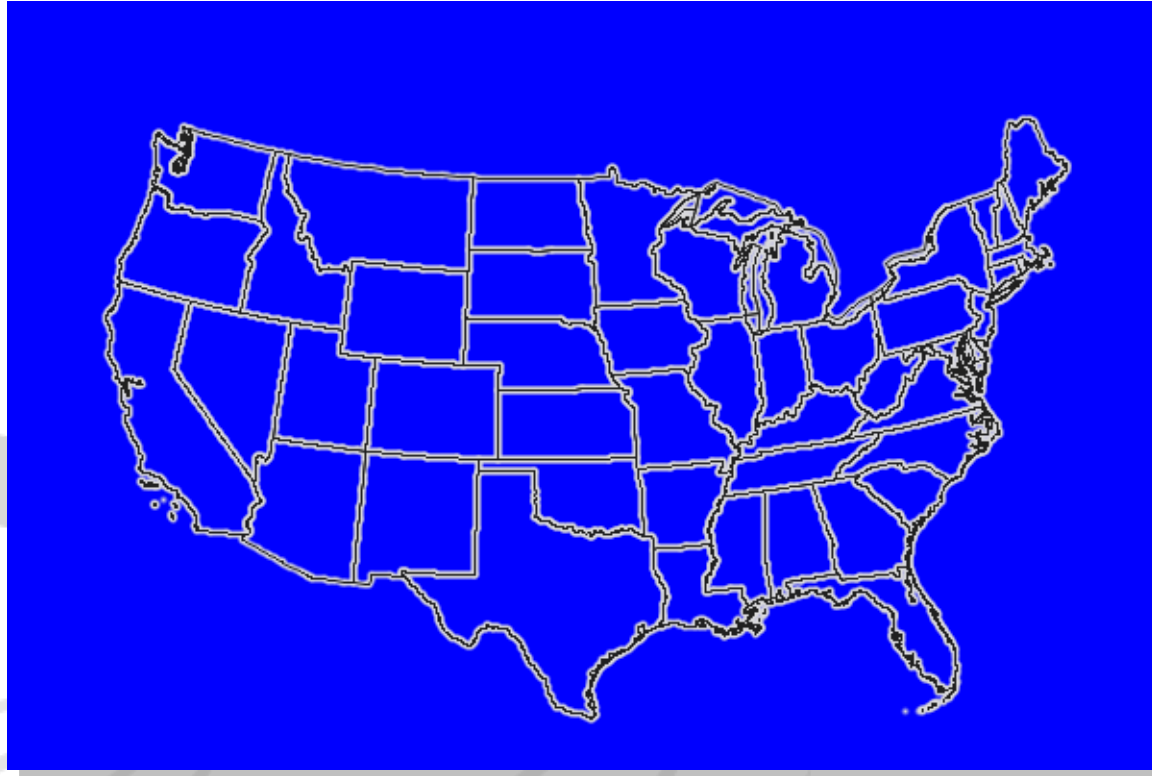
- Open “layer\_antialias.map”

```
...  
IMAGETYPE png24  
LAYER  
  
...  
TRANSPARENCY ALPHA  
CLASS  
  
...  
STYLE  
  COLOR 220 220 220  
  WIDTH 3  
  ANTIALIAS TRUE  
END  
STYLE  
  COLOR 32 32 32  
  ANTIALIAS TRUE  
END  
END ...
```

- View The Map

# Example 8 Output

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_antialias.map&layer=states&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_antialias.map&layer=states&mode=map)

# Example 9: Adding Raster Data

---

- Open “layer\_raster.map”

```
MAP
...
LAYER
  NAME 'relief'
  DATA 'data/shdr1fi0201.tif'
  TYPE RASTER
  STATUS OFF

  PROJECTION
    "init=epsg:2163"
  END
END
END
```

- View The Map
- Raster Data Access for more information

# Example 9 Output

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_raster.map&layer=relief&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_raster.map&layer=relief&mode=map)



# Example 10: Adding OGC WMS Layer

## ■ Open “layer\_wms.map”

```
MAP
...
WEB
  IMAGEPATH '/ms4w/tmp/ms_tmp/'
  IMAGEURL  '/ms_tmp/'
END

PROJECTION
  "init=epsg:2163"
END

LAYER # MODIS WMS map from JPL
  NAME      modis
  TYPE      RASTER
  OFFSITE   0 0 0
  STATUS    OFF
  CONNECTIONTYPE WMS
  CONNECTION
    "http://wms.jpl.nasa.gov/wms.cgi?"
```

```
METADATA
  "wms_srs" "EPSG:4326"
  "wms_name" "BMNG"
  "wms_server_version" "1.1.1"
  "wms_format" "image/png"
  "wms_style" "Jul"
END

PROJECTION
  "init=epsg:4326"
END
END # Modis WMS image ends here
END
```

## ■ View the Map

# Example 10 Output

---



- <http://localhost/cgi-bin/mapserv>
- [?map=/data/www/ms101/layer\\_wms.map&layer=modis&mode=map](http://localhost/cgi-bin/mapserv/?map=/data/www/ms101/layer_wms.map&layer=modis&mode=map)

# Example 11: Adding a Legend

- Open “legend.map”

```
MAP
```

```
...  
LEGEND  
  STATUS ON  
  LABEL  
    COLOR 0 0 0  
    SIZE SMALL  
  END  
END  
...  
END
```

- View The Map

- Output:

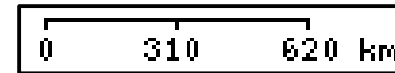


# Example 12: Adding a Scalebar

## ■ Open “scalebar.map”

```
...  
SCALEBAR  
  LABEL  
    COLOR 0 0 0  
    SIZE TINY  
  END  
  STYLE 1  
  SIZE 100 2  
  UNITS KILOMETERS  
  INTERVALS 2  
  COLOR 0 0 0  
  BACKGROUND_COLOR 128 128 128  
  STATUS ON  
  TRANSPARENT ON  
END  
...
```

## ■ Output:



## ■ View The Map

# Example 13: Adding a Reference

- Open “reference.map”

```
...  
REFERENCE  
  IMAGE 'webdocs/usref.png'  
  SIZE 150 100  
  EXTENT -2491519 -2350170  
          2976025 1294859  
  COLOR -1 -1 -1  
  OUTLINECOLOR 255 128 0  
  STATUS ON  
END  
...
```

- View The Map

- Output:



# Example 14: HTML Template

---

- Open “template.html”

...

```
<table border="1">
  <tr>
    <td valign="top">
      <p><br /></p>
    </td>
    <td valign="top">
      <p><b>Legend:</b><br /></p>
      <p><b>Reference:</b><br /></p>
    </td>
  </tr>
</table>
```

...

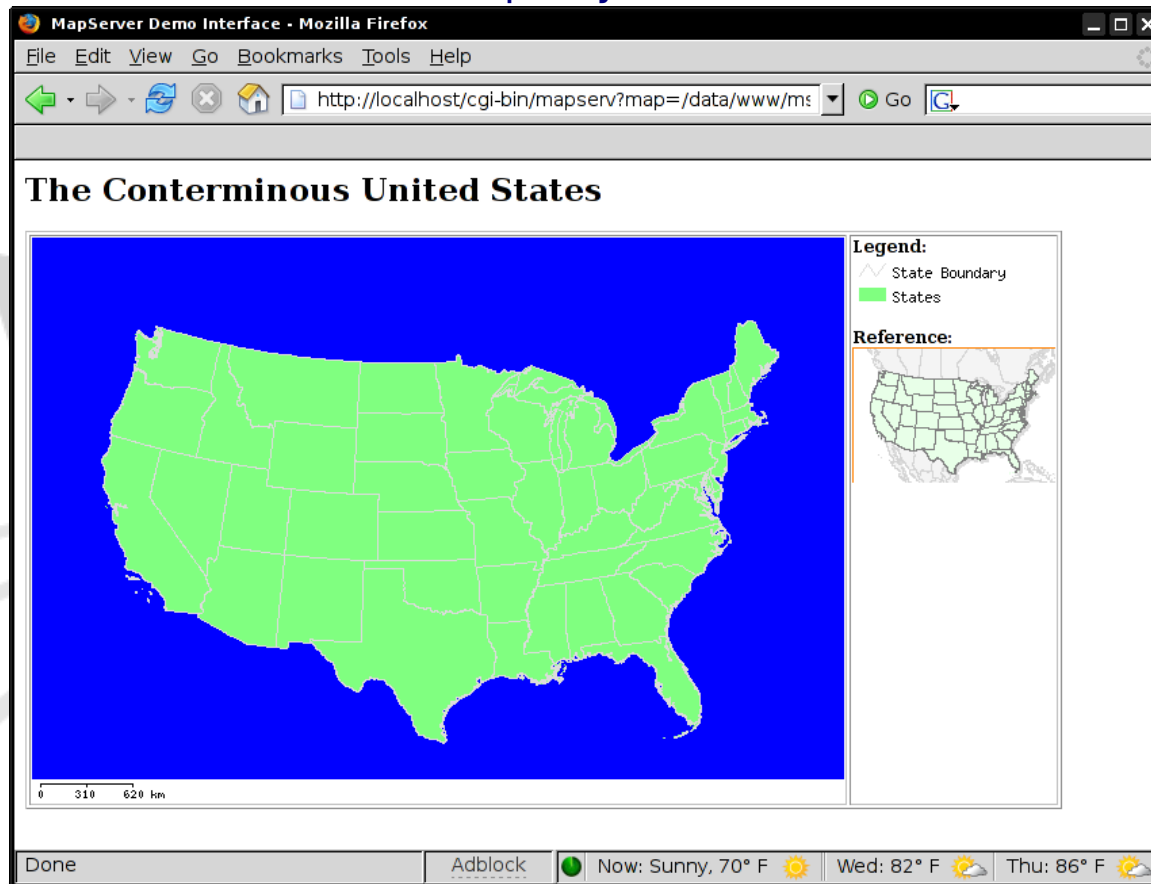
- Contains MapServer Template tags.
- Template tags are enclosed in “[ ]”.
- Processed by MapServer CGI Program.
- [HTML Template Reference](#) for more info.

# Example 14: HTML Template Output

- MapServer Request via “browse” mode:

`http://localhost/cgi-bin/mapserv`

`?map=/data/www/ms101/reference.map&layer=states&mode=browse`



# Example 15: Demo Application

---

## ■ Open “webdocs/index.html”

```
<form method="get" action="/cgi-bin/mapserv">
...
<input type="hidden" name="layer" value="states" />
<input type="text" size="20" name="program" value="/cgi-bin/mapserv" />
<input type="text" size="30" name="map"
value="/data/www/ms101/ms101.map" />
...
</form>
```

## ■ Open “ms101.html”

```
...
<form method="get" action="[program]">
<input type="hidden" name="map" value="[map]" />
<input type="hidden" name="imgext" value="[mapext]" />
<input type="hidden" name="imgxy" value="[center]" />
<input type="hidden" name="zoomsize" value="2" />
<input type="hidden" name="program" value="[program]" />
...
```



# Example 15: Demo Application

---

- Can be initialized via hyperlink or via a web form
  - Web forms can be used to change some settings
- Open “<http://localhost/ms101/index.html>” and click on the demo application hyperlink or the “Proceed to the MapServer Demo” button.
- See the [CGI MapServer Reference](#) for more information

# MapScript

---

Exposing the MapServer API to  
Scripting Languages

# MapScript Overview

---

- Extends MapServer capabilities
- Seamless Integration with Other Applications
- Uses SWIG (except PHP)
- Support for Several Languages:
  - PHP
  - Python
  - Perl
  - Ruby
  - Java
  - C#
  - TCL/Tk
- See the [MapScript API Reference](#) for more information
- Demo: [Tcl/Tk MapScript](#)
- Demo: [PHP/MapScript](#)

# Third Party Tools and Support

---

Client and Management Tools for  
MapServer Applications

# Third Party Tools/Support

---

- Open Source Client Tools

- [MapLab](#)
- [Chameleon](#)
- [ka-Map! – Demo](#)
- [CartoWeb – Demo](#)
- [MapServer Workbench](#)
- [MapBender](#)
- [MapBuilder](#)

- Proprietary Client Tools

- [Neapoljs](#)

# Support and Other Resources

---

- Support Available via Mailing Lists
  - [MapServer-Users](#)
  - [MapServer-Dev](#)
- Additional Support via Consulting/Services Companies
- MapServer Website: <http://mapserver.gis.umn.edu>
- MapTools: <http://www.maptools.org>
- GDAL/OGR: <http://www.gdal.org>
- PROJ.4: <http://proj.maptools.org>
- OSGeo: <https://www.osgeo.org>,
- OSGeo Wiki: <http://wiki.osgeo.org>

---

# Questions?

