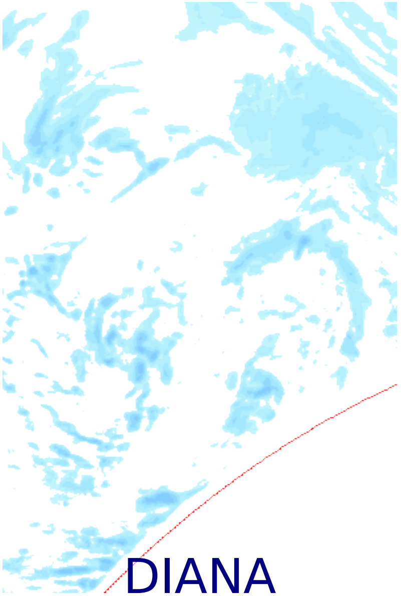
Meteorologisk
institutt
met.no

Apache tips & tricks

Using Apache to work around Mapserver limitations

Background

- We use in-house software DIANA as WMS-server for weather maps
- Mapserver is mostly used for background maps
- We wanted it to be possible to fetch a single image with background and weather data
- Configured Mapserver to function as a Cascading WMS. All requests to DIANA-WMS goes through Mapserver



Shorter URLs

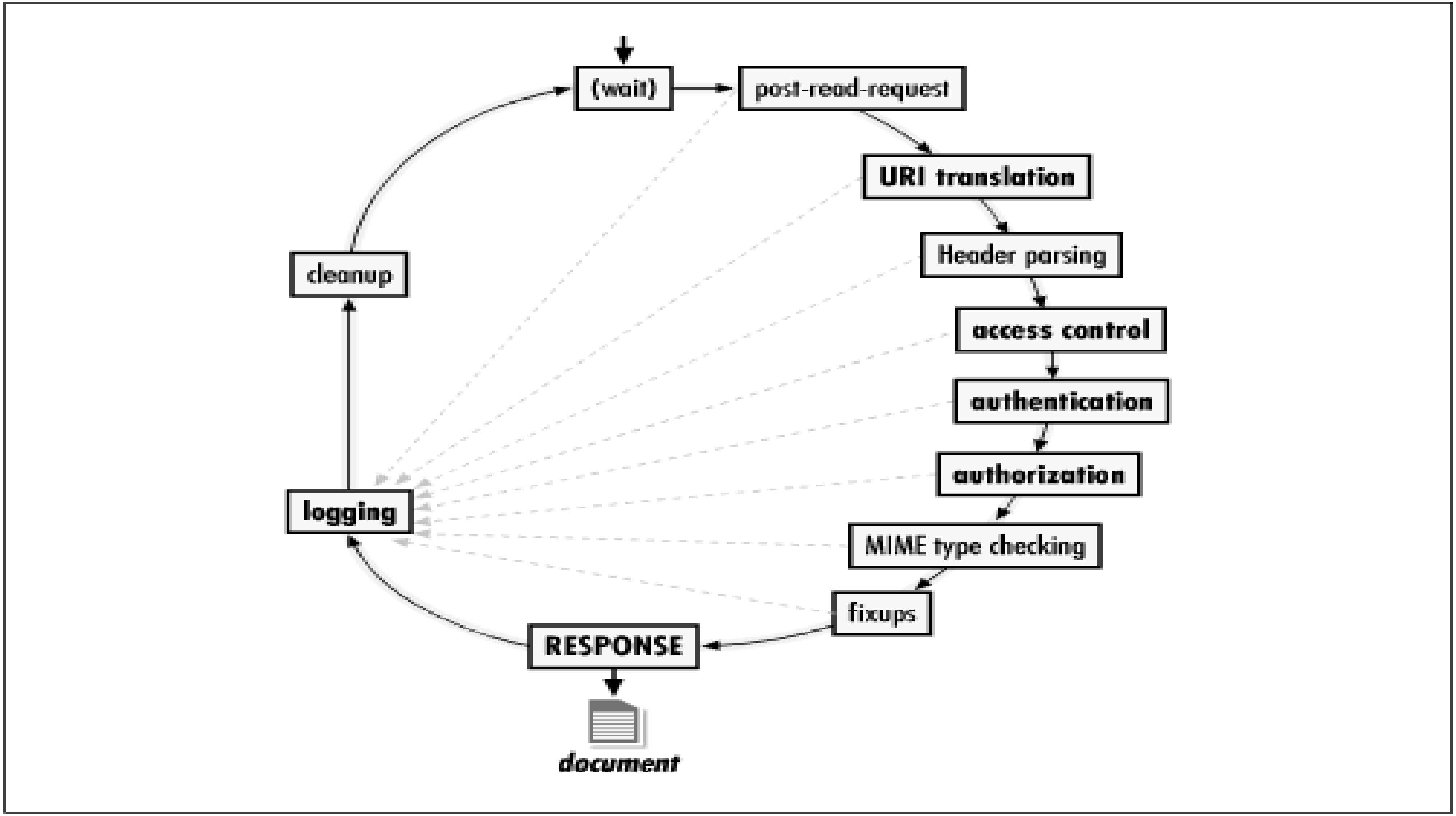
- Standard mapserver URL needs full path to location of the mapfile as a parameter

http:// wms.met.no/cgi-bin/mapserver? map=/var/www/weather/mapfiles/atmosphere.map& ...

- Long and impractical
- Would prefer something shorter, like this:

http:// wms.met.no/weather/atmosphere.map? ...

Apache request cycle



mod_rewrite

```
RewriteRule
  /([^/]*)/(.*\.map$)
  /cgi-bin/mapserv?map=/var/www/$1/mapfiles/$2
  [PT,QSA]
```

mod_rewrite

```
RewriteRule
  /([^/]*)/(.*\.map$)
  /cgi-bin/mapserv?map=/var/www/$1/mapfiles/$2
  [PT,QSA]
```

Incoming URL:

*http:// wms.met.no/**weather**/atmosphere.map? ...*

mod_rewrite

```
RewriteRule
  /([^/]*)/(.*\.map$)
  /cgi-bin/mapserv?map=/var/www/$1/mapfiles/$2
  [PT,QSA]
```

Incoming URL:

*http:// wms.met.no/weather/**atmosphere.map**? ...*

mod_rewrite

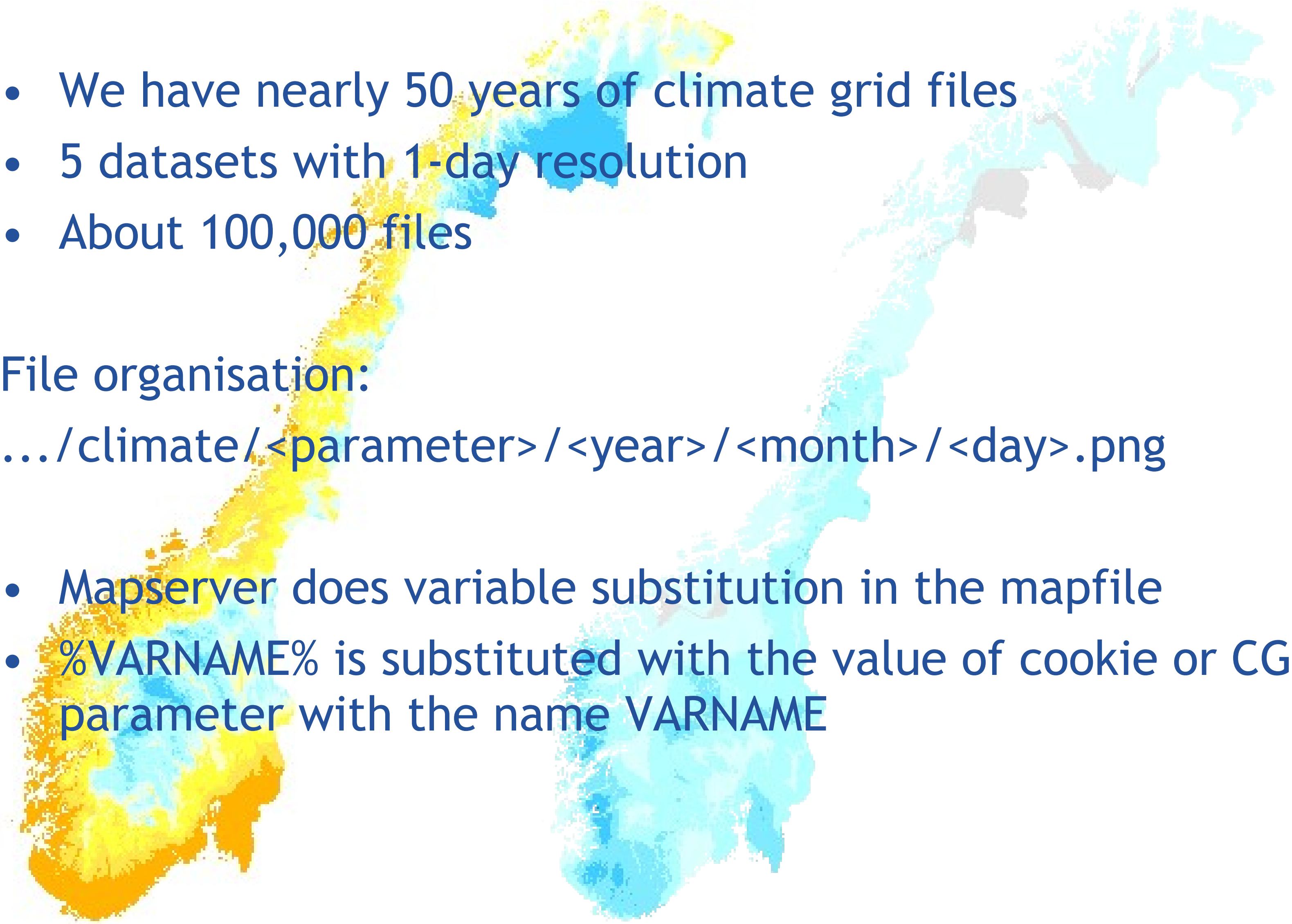
```
RewriteRule
  /([^/]*)/(.*\.map$)
  /cgi-bin/mapserv?map=/var/www/$1/mapfiles/$2
  [PT,QSA]
```

Incoming URL:

http:// wms.met.no/weather/atmosphere.map? ...

Translated URL:

http:// wms.met.no/cgi-bin/mapserv?
*map=/var/www/**weather**/mapfiles/**atmosphere.map**& ...*



- We have nearly 50 years of climate grid files
- 5 datasets with 1-day resolution
- About 100,000 files

File organisation:

.../climate/<parameter>/<year>/<month>/<day>.png

- Mapserver does variable substitution in the mapfile
- %VARNAME% is substituted with the value of cookie or CGI-parameter with the name VARNAME

- We have nearly 50 years of climate grid files
- 5 datasets with 1-day resolution
- About 100,000 files

File organisation:

.../climate/<parameter>/<year>/<month>/<day>.png

- Mapserver does variable substitution in the mapfile
- %VARNAME% is substituted with the value of cookie or CGI-parameter with the name VARNAME

We wrote a URI Translation handler that splits the TIME parameter and creates new parameters for each of the components

TIME=2009-11-24T00:00:00

is split into:

YEAR=2009
MONTH=11
DAY=24

HOUR=00
MINUTES=00
SECONDS=00

Mapserver configuration

```
LAYER
  NAME tam
  DATA .../climate/tam/%YEAR%/%MONTH%/%DAY%.png
  ...
END
LAYER
  NAME rr
  DATA .../climate/rr/%YEAR%/%MONTH%/%DAY%.png
  ...
END
```

Mapserver configuration

With TIME=2009-11-24T00:00:00

```
LAYER
  NAME tam
  DATA .../climate/tam/2009/11/24.png
  ...
END
LAYER
  NAME rr
  DATA .../climate/rr/2009/11/24.png
  ...
END
```

Passthrough of Dimensions

- Mapserver does not pass through TIME, ELEVATION or any other dimensional parameters to remote WMS servers when acting as a cascading server
- It's possible to add TIME=%TIME%&ELEVATION=%ELEVATION% to connection string, to work around this
- %VARNAME% is case sensitive, WMS-parameter names are not

Passthrough of Dimensions

Solution:

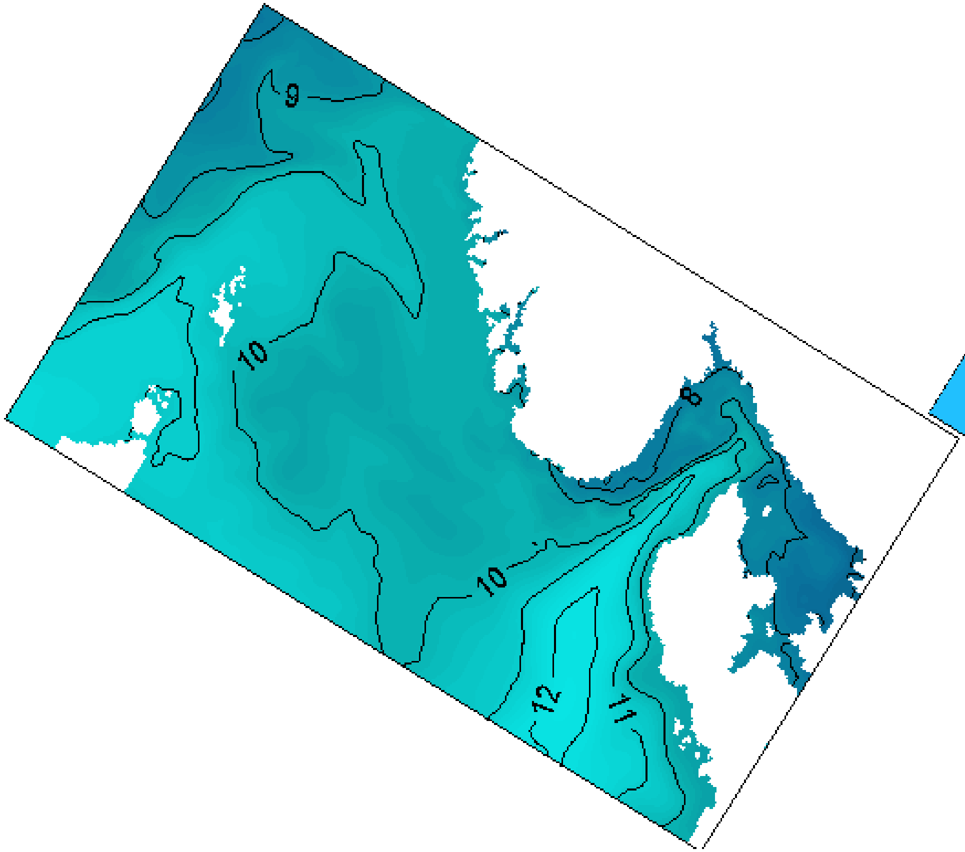
- Wrote a URI Translation handler that packs all dimension parameters (TIME, ELEVATION, DIM_*) into a string
- Adds parameter PASSTHROUGH=<dimension string> to the query string

Mapserver config:

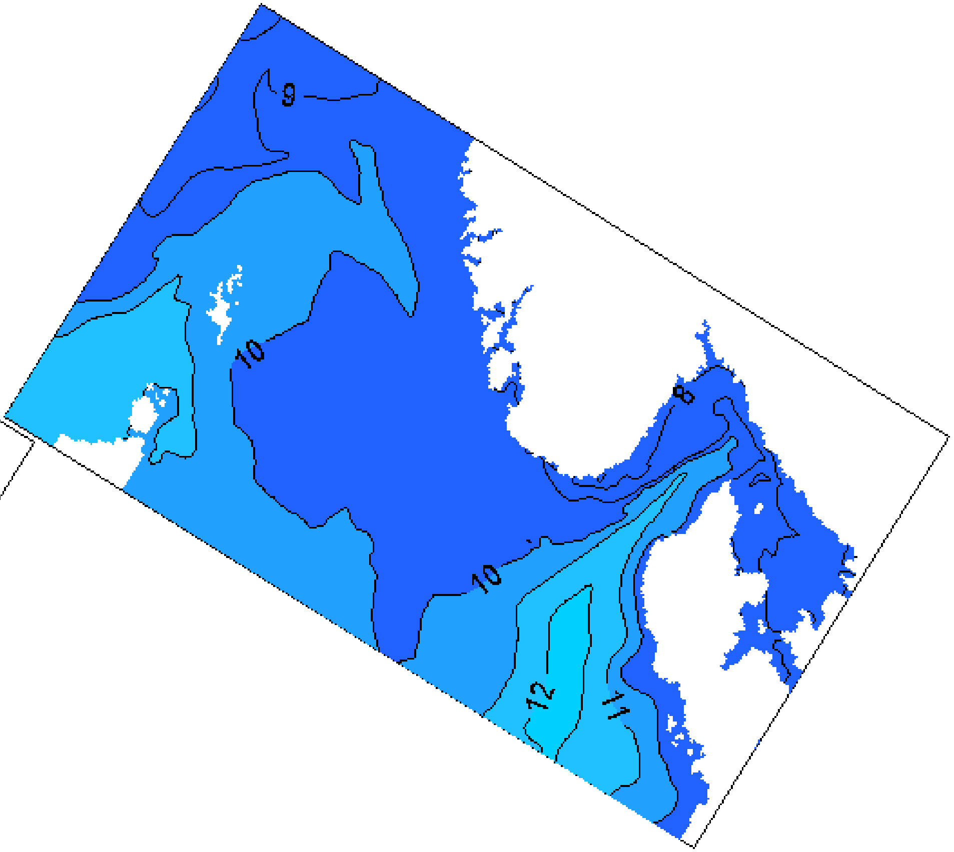
```
CONNECTION http:// localhost/diana?%PASSTHROUGH%& ...
```

Using Named Styles with mapserver

LAYER=sea.temp
STYLE=256colour



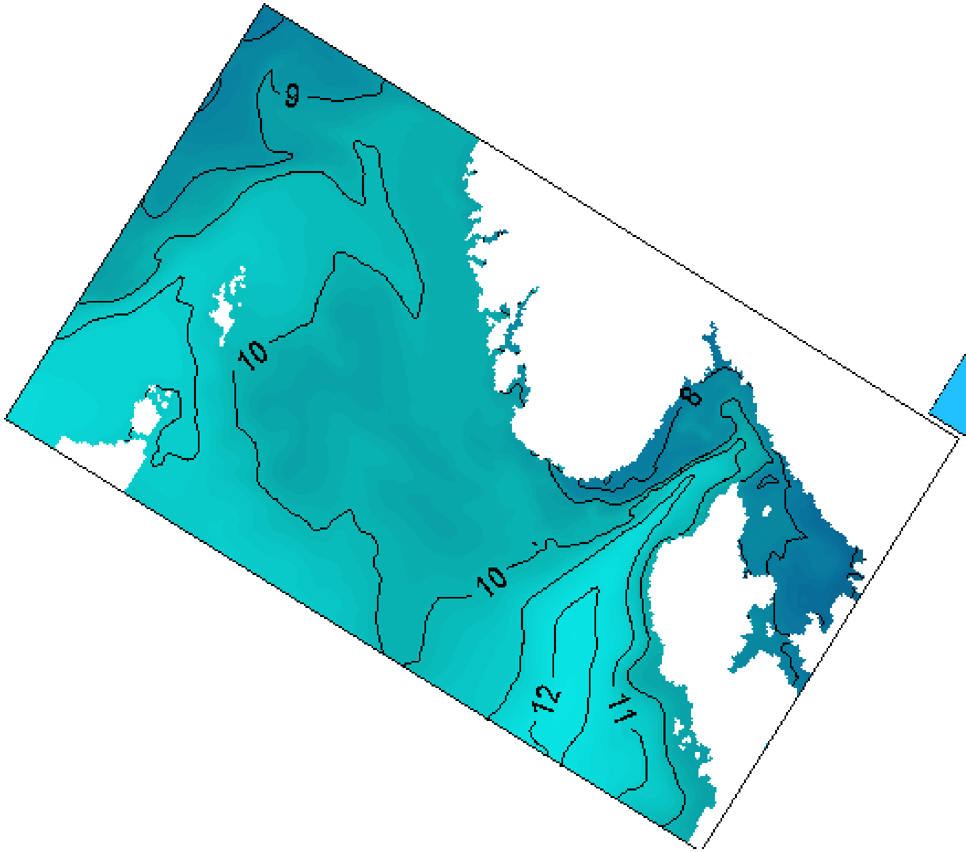
LAYER=sea.temp
STYLE=default



First attempt

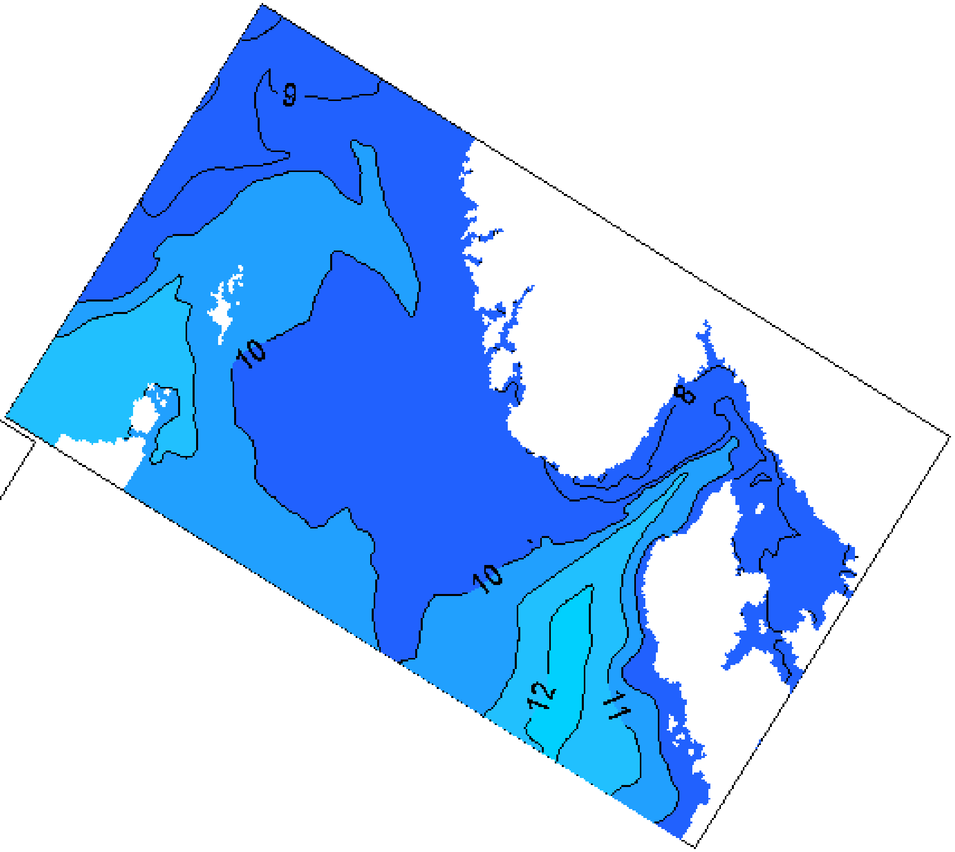
LAYER=sea.temp_STYLE_256colour

STYLE=default



LAYER=sea.temp

STYLE=default



Using Named Styles with mapserver

- Works
- Requires special handling by client, or the styles will appear as two layers
- Poor support for named styles in clients, so we'd have to modify the client anyway

- Non-standard solution means we can't commit solution back to community
- Better to fix it server-side, so the client will think it's proper named styles

Using Named Styles with mapserver

Solution:

- Keep separate layers and naming convention in Mapserver
- Rewrite input from clients
- Filter GetCapabilities output from Mapserver

GetMap request translation

From client:

- LAYERS=background,sea.temp,coastlines
- STYLES=default,256colours,default

Translated:

- LAYERS=background,**sea.temp_STYLE_256colours**,coastlines
- STYLES=default,**default**,default

Mapserver will only see the translated LAYERS and STYLES parameters, and does not know what the client actually requested.

GetCapabilities filtering

- Mapserver's GetCapabilities response will treat each style as a separate layer
- Need to filter output from mapserver
  - Apache 1.3: RewriteRule + cgi-script
  - Apache 2.0: OutputFilter
- Locate all <Layer>-nodes where name contains _ STYLE _
  - Delete the <Layer> node
  - Add <STYLE>-node to the real layer

GetCapabilities filtering

```
<Layer>
   <Name>sea.temp</Name>
   <Title>Temperature</Title>
</Layer>
<Layer>
   <Name>sea.temp_STYLE_256colours</Name>
   <Title>Temperature (256colours)</Title>
</Layer>
```

Mapserver returns the two styles as two different layers

GetCapabilities filtering

```
<Layer>
    <Name>sea.temp</Name>
    <Title>Temperature</Title>
    <Style>
      <Name>default</Name>
      <Title>default</Title>
    </Style>
    <Style>
      <Name>256colours</Name>
      <Title>256colours</Title>
    </Style>
</Layer>
```

But the client will see it as one layer with two named styles

Thank You!