

# Making maps pretty

Andrea Aime



OPENGEO

Jim Groffen



LISAsoft™



# Making maps pretty

## Introduction



# Introducing carthography

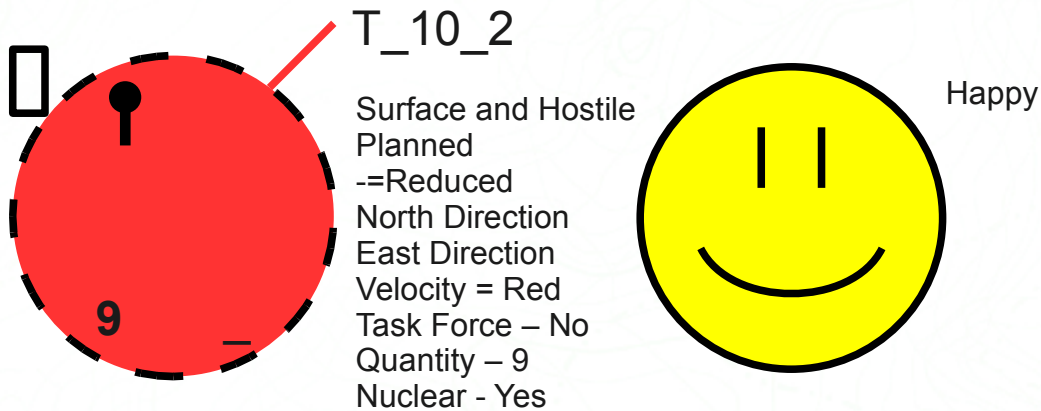
- Depiciting shape and location, conveying qualitative and quantitative information over a map
- Map: visually appealing, but still correct and useful for a specific purpose





# Target audience

- Ability to read specific symbols
- Ability do discern colors and small features on the map



The image shows two symbols side-by-side. On the left is a red circular target symbol with a dashed black border. It contains a small black rectangle at the top-left, a black dot with a vertical line extending downwards, and the number '9' at the bottom-left. A red line points from the label 'T\_10\_2' to the top-right edge of the circle. To the right of the target is a yellow smiley face with a black outline and a black smile. The word 'Happy' is written to the right of the smiley face.

T\_10\_2

Surface and Hostile  
Planned  
--Reduced  
North Direction  
East Direction  
Velocity = Red  
Task Force – No  
Quantity – 9  
Nuclear - Yes

Happy



# Target medium

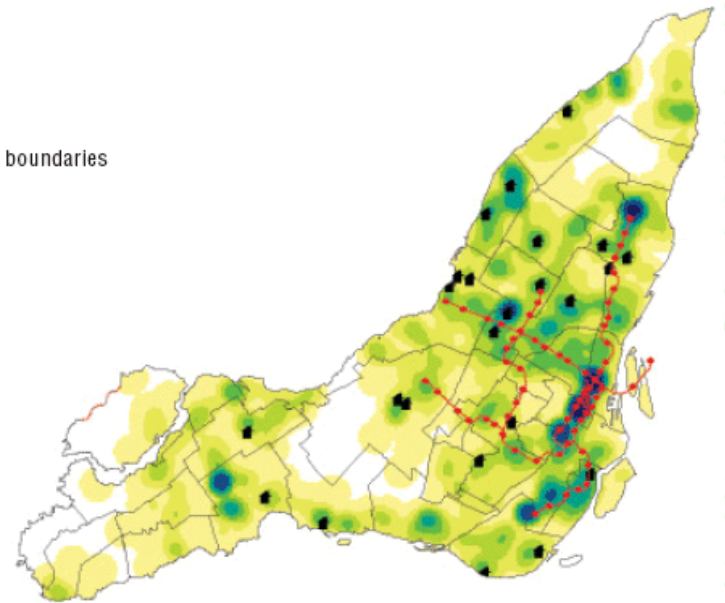
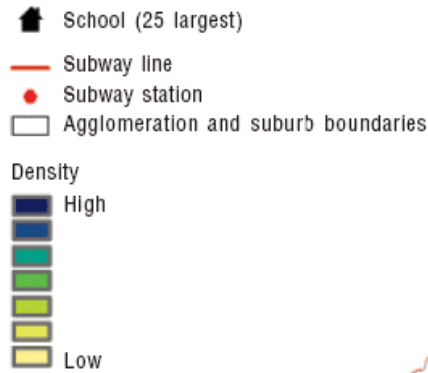
- Device type (desktop, netbook, mobile)
- Colors available?





# Purpose

- What should it display?
- How will it be used?





# Clarity – Information Density

- General Purpose or Specific Purpose
- Level of Detail
- Usability

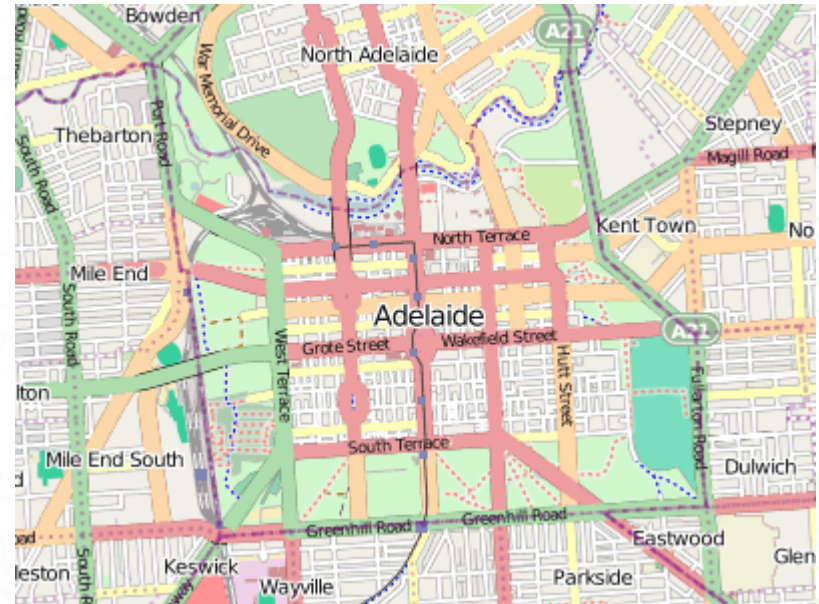


© 2008 "Club Residences at Aina Nalu"



# Labels

- Font
- Density
- Contrast



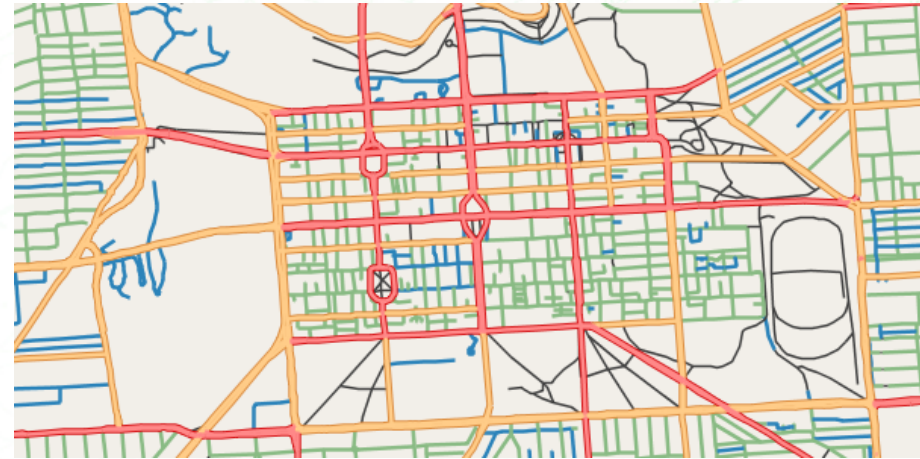




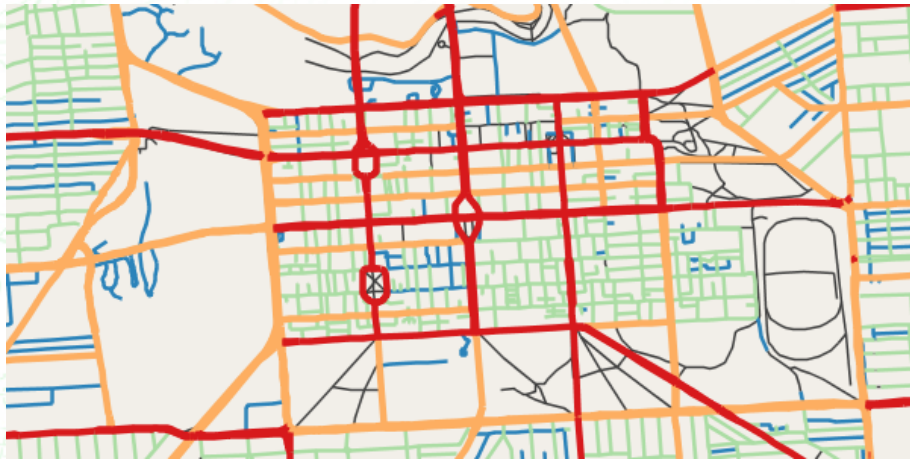
# Line styling



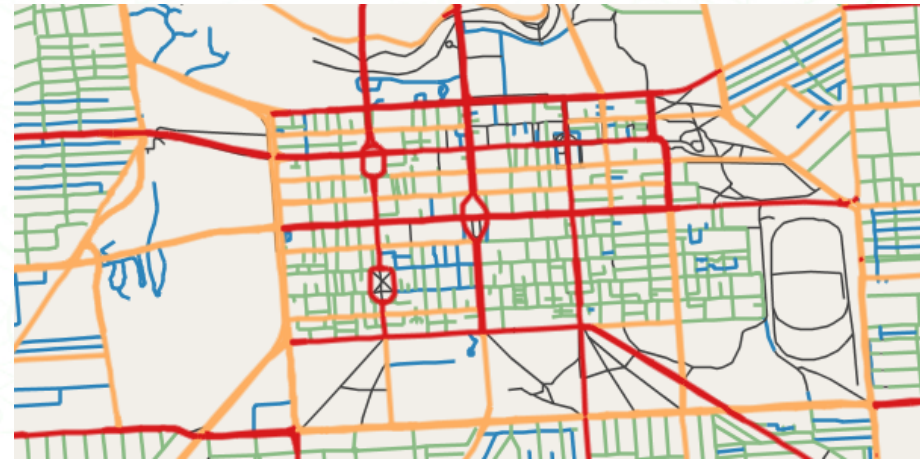
COLOUR



CASING



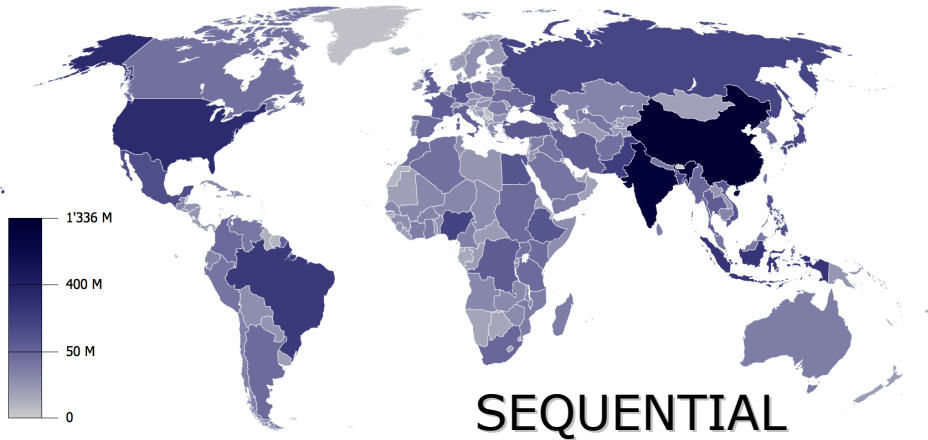
THICKNESS



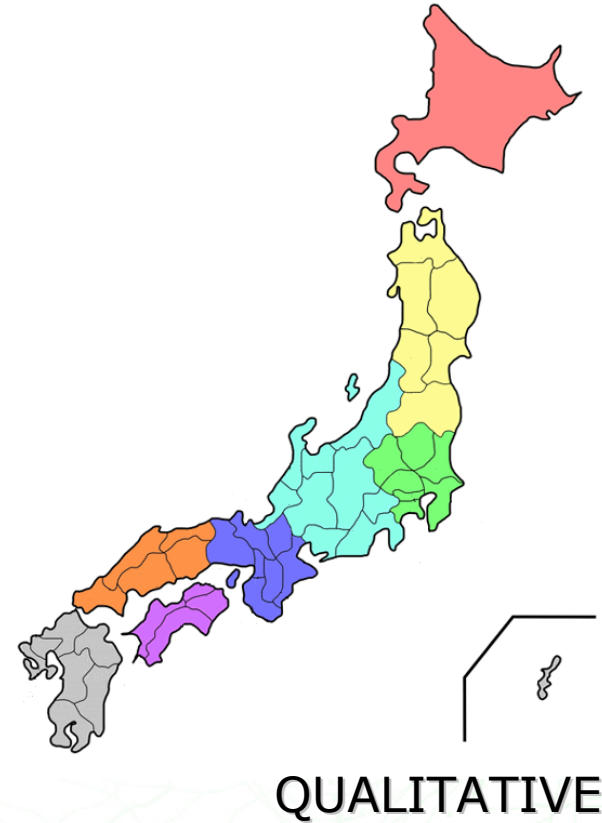
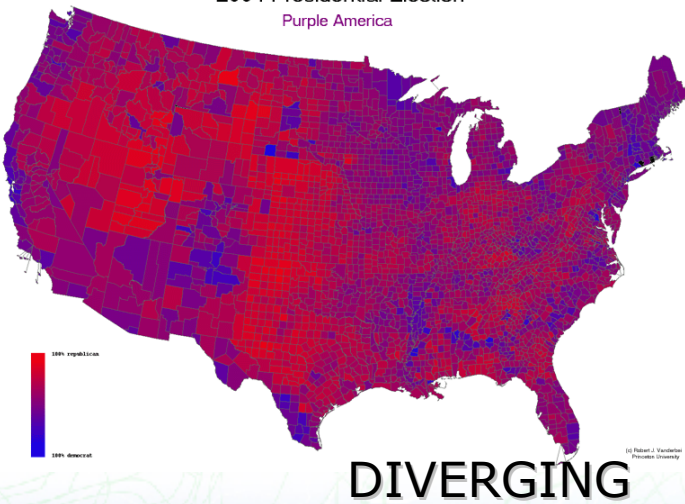
TRANSPARENT CASING



# Colour schemes

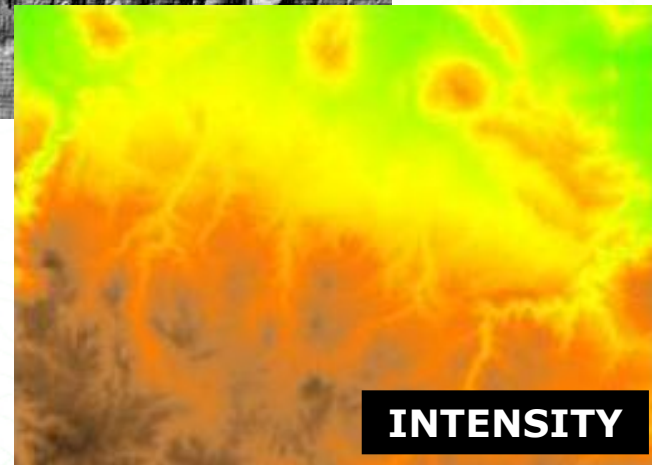
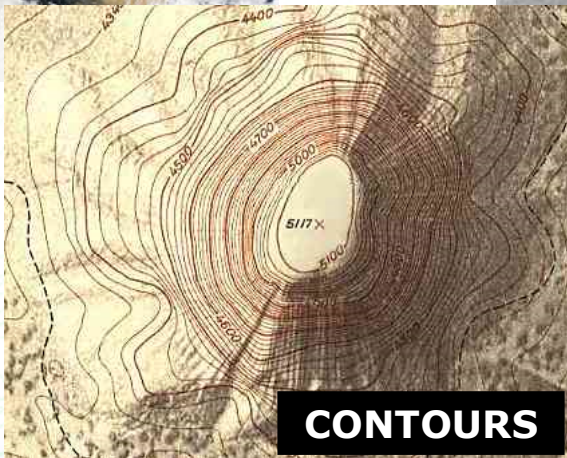
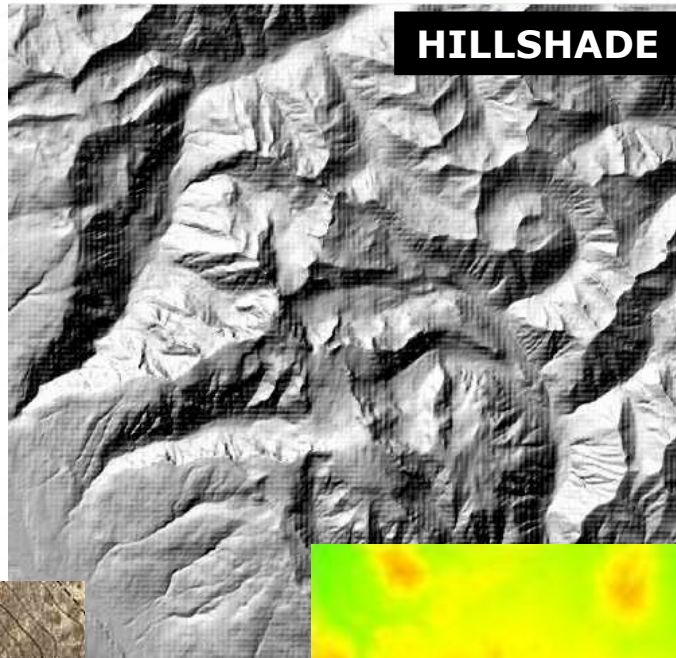
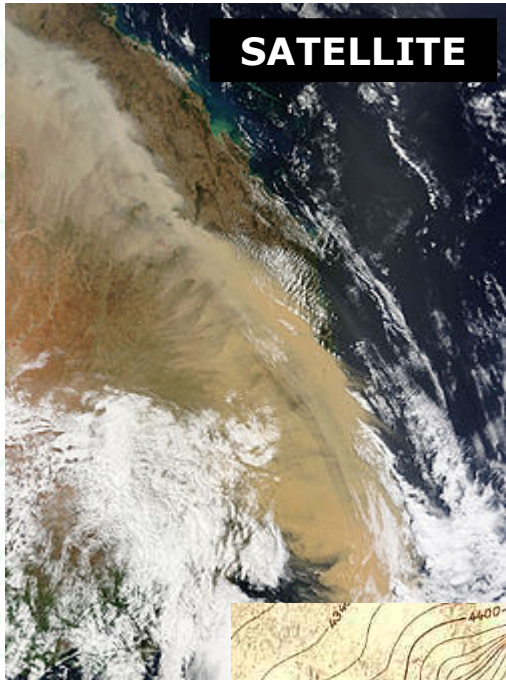


2004 Presidential Election  
Purple America



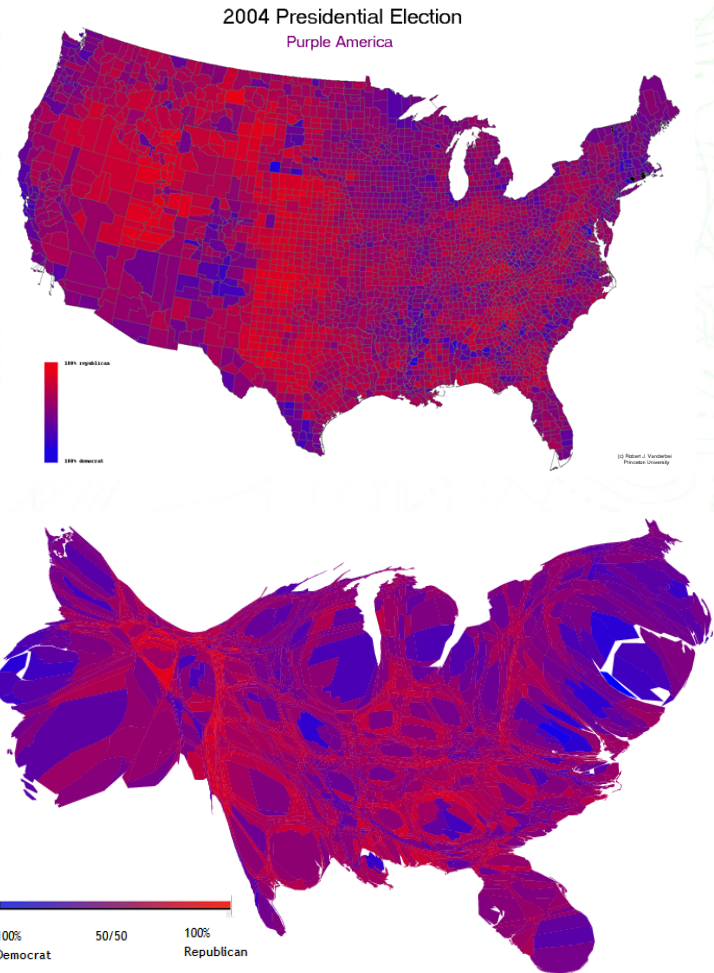
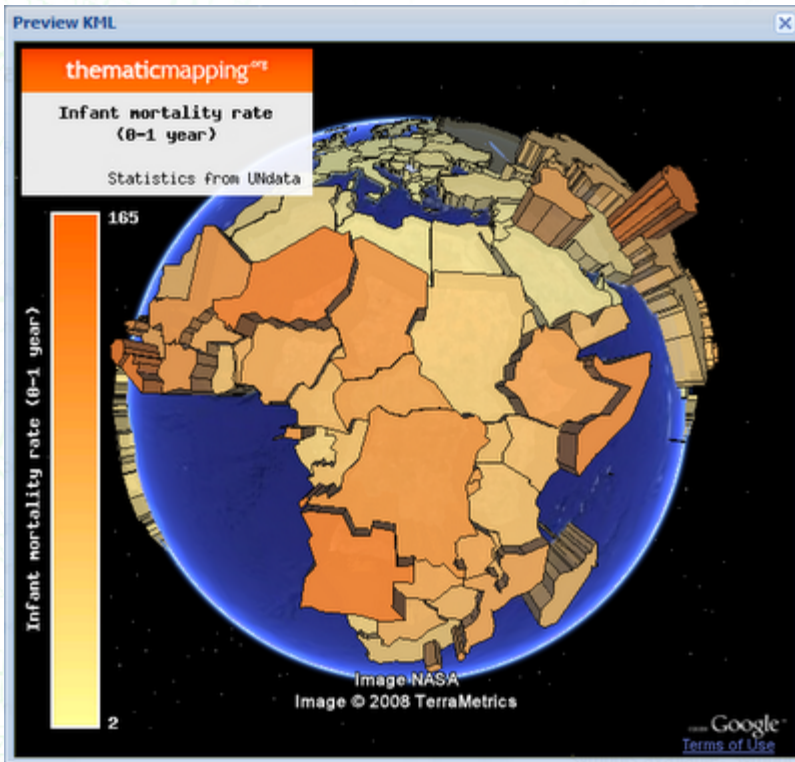


# Raster





# Fallacies





## Map elements



# Polygons

- Bounded areas
  - Physical: coastlines
  - Conceptual: administrative subdivisions
- Two stylable components
  - Outline
  - Fill
- Filling the polygon
  - To show the shape or a uniform quality
  - To show a quantity or a non uniform quality (thematic mapping)





# Points

- Entities with a position, but no dimension
  - Too small at all scales
  - Too small at small scales (cities, buildings)
- Represented
  - Small, simple symbols
  - Large, elaborate symbols
- Communicate differences
  - Using color
  - Using size
  - Using shape





# Lines

- Entities too thin to be represented as polygons
  - At all scales
  - At certain scales (rivers, roads)
- Nature:
  - Landscape features
  - Computed (isolines)
- Communicate with:
  - Color
  - Width
  - Graphics

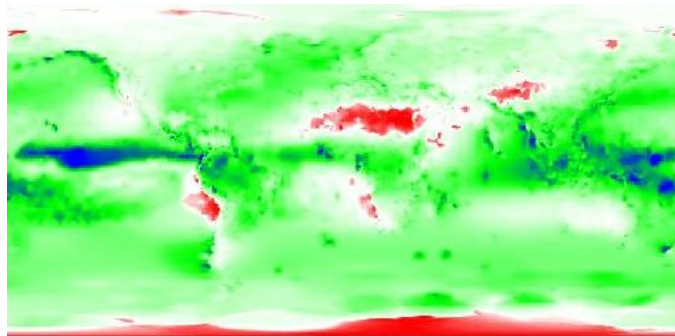
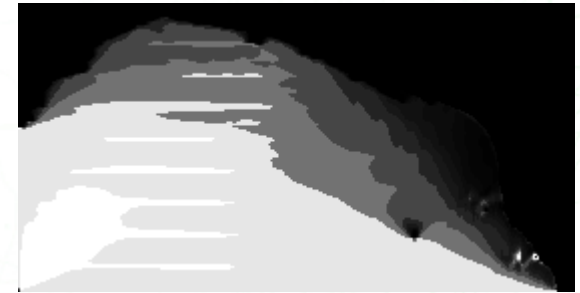
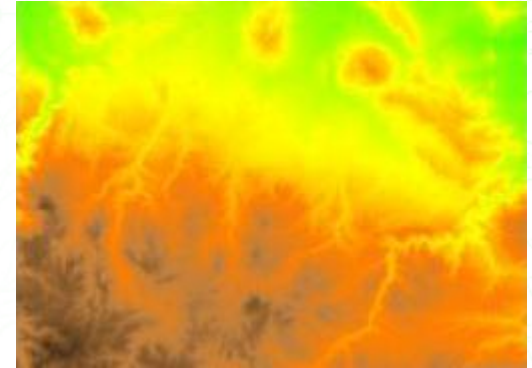






# Raster

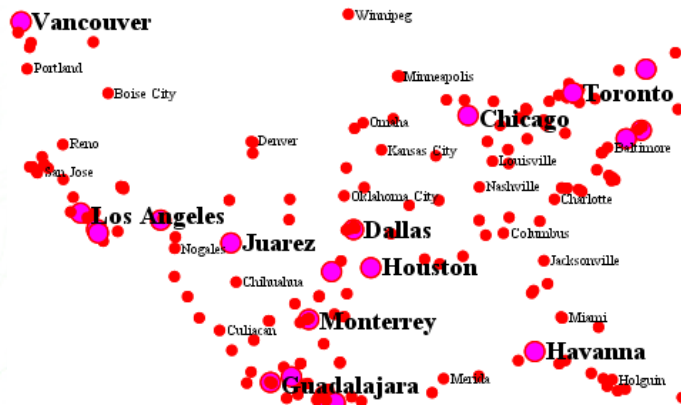
- Imagery
- Grid of physical values (temperature, elevation, ...)
- Contrast enhancement is sometimes necessary
- Color classification
  - Continuous
  - By classes





# Text

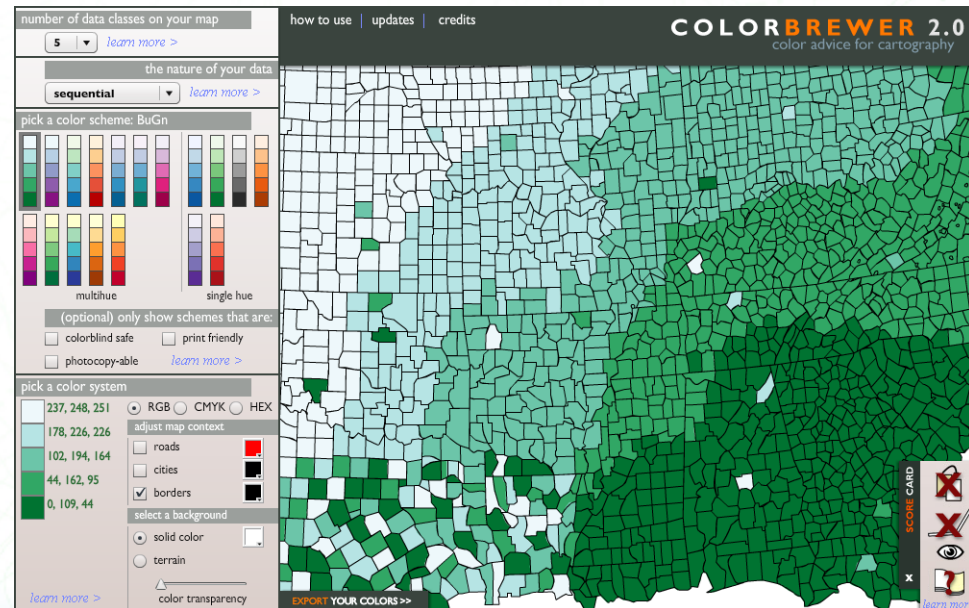
- Labels
  - Font
  - Color
  - Contrast
  - Density
  - Hierarchy





# Colour

- Choosing an appropriate color scheme
  - Culture
  - Ability to discern shades of colour
  - sing too many/too few colours
- Exercise
  - <http://colorbrewer2.org/>





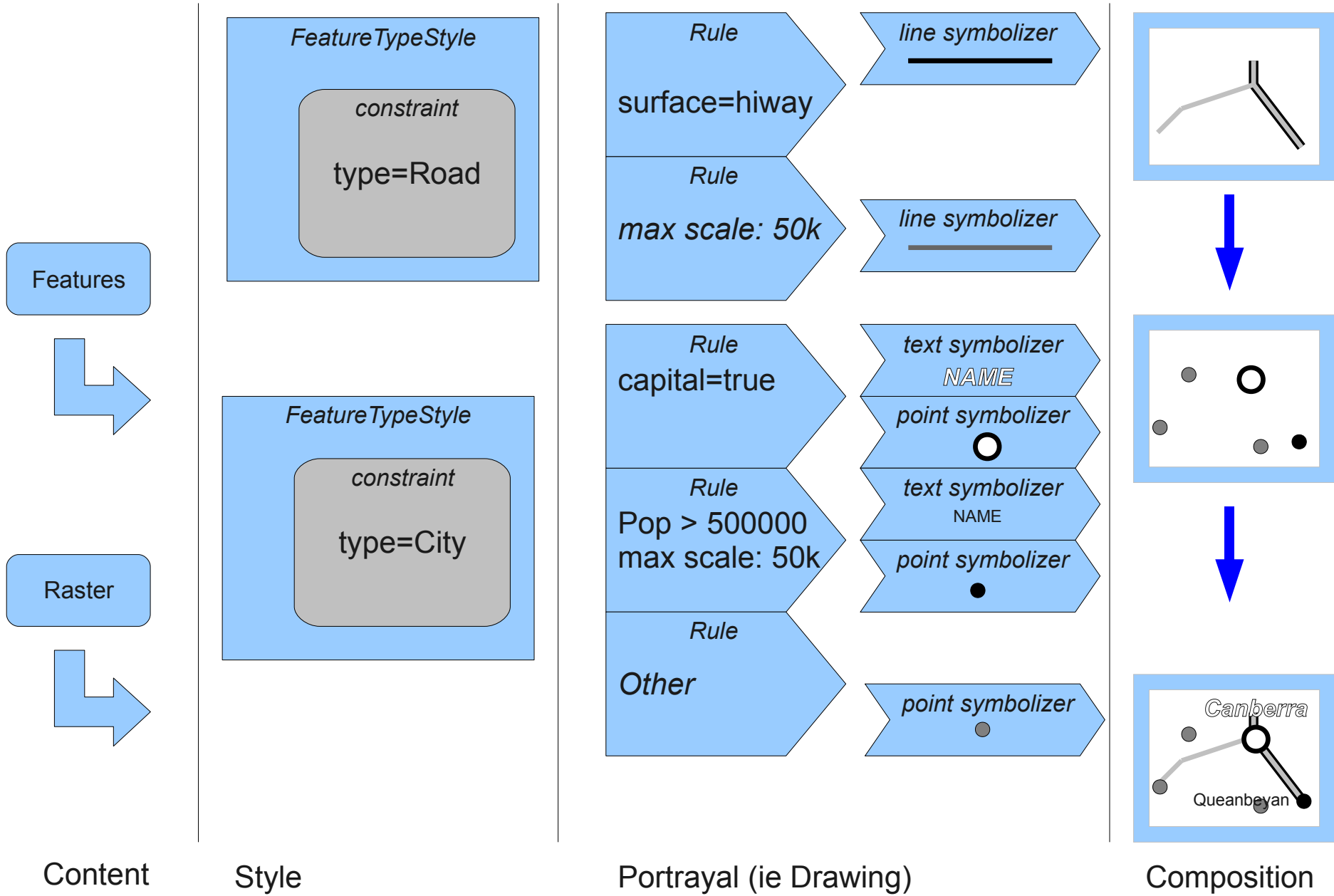
# Making maps pretty

## Using styles



# Separating styles and data

- Styled Layer Descriptor: markup specifying how to style a map
- Separate from the map
  - The same style can be reused against various data
  - The same data can be styled in different ways
- Notable examples in the IT industry
  - HTML and CSS
  - Model/View/Controller





# Root SLD elements

- StyledLayerDescriptor
  - NamedLayer (UserLayer)
    - UserStyle (NamedStyle)
- We concentrate on the UserStyle contents since that is where a Style is defined
- Unless you're making dynamic styling requests (that few WMS support) you can safely ignore the other elements and treat them as boilerplate

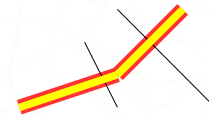
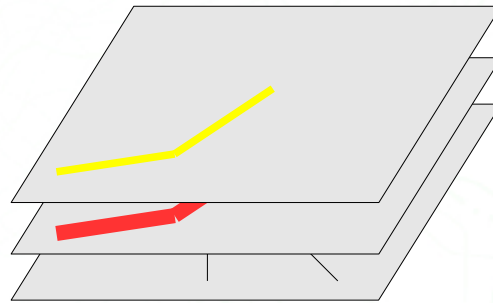
```
<StyledLayerDescriptor ... >
  <NamedLayer>
    <Name>name for style layer</Name>
    <UserStyle>
      <Title>Title for style layer</Title>
      <FeatureTypeStyle>
        ...
      </FeatureTypeStyle>
    </UserStyle>
  </NamedLayer>
</StyledLayerDescriptor>
```



# FeatureTypeStyle

- Contains rules driving the map renderer
- Can be used to ensure inner layering (ensure certain features will be drawn before others)\*
- Can mention the type of feature to be styled (safer to just skip it)

```
<FeatureTypeStyle>
  <Rule>
    ...
  </Rule>
  <Rule>
    ...
  </Rule>
</FeatureTypeStyle>
<FeatureTypeStyle>
  <Rule>
    ...
  </Rule>
  <Rule>
    ...
  </Rule>
</FeatureTypeStyle>
```



\*: the SLD specification does not tell exactly how rendering order should be applied





# Rule

- Contains the symbolizers (what actually draws features on the map)
- Controls them:
  - Scale controls: display features only at certain scales
  - Filtering: display only features matching certain conditions

```
<Rule>
  <MinScaleDenominator>400000</MinScaleDenominator>
  <MaxScaleDenominator>1000000</MaxScaleDenominator>
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>MTFCC</ogc:PropertyName>
      <ogc:Literal>S1200</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
  <LineSymbolizer>
    ...
  </LineSymbolizer>
</Rule>
```



# Filtering

- Spatial or attribute filters
- Logical combinations
- ElseFilter

```
<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>oneway</ogc:PropertyName>
      <ogc:Literal>0</ogc:Literal>
    </ogc:PropertyIsEqualTo>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>maxspeed</ogc:PropertyName>
      <ogc:Literal>60</ogc:Literal>
    </ogc:PropertyIsGreaterThan>
    <ogc:Not>
      <ogc:Disjoint>
        <ogc:PropertyName>Geometry</ogc:PropertyName>
        <gml:Envelope srsName="urn:x-ogc:def:crs:EPSG:6.3:4326">
          <gml:lowerCorner>138.0 -35.5</gml:lowerCorner>
          <gml:upperCorner>139.0 -34.5</gml:upperCorner>
        </gml:Envelope>
      </ogc:Disjoint>
    </ogc:Not>
  </ogc:And>
</ogc:Filter>
```

- PropertyIsEqualTo
- PropertyIsNotEqualTo
- PropertyIsLessThan
- PropertyIsLessThanOrEqualTo
- PropertyIsGreaterThan
- PropertyIsGreaterThanOrEqualTo
- PropertyIsBetween
- PropertyIsLike
- PropertyIsNull

- BBOX
- Intersects
- Contains
- Crosses
- Overlaps
- Touches
- Equals
- Within
- Beyond
- Dwithin
- And
- Or
- Not



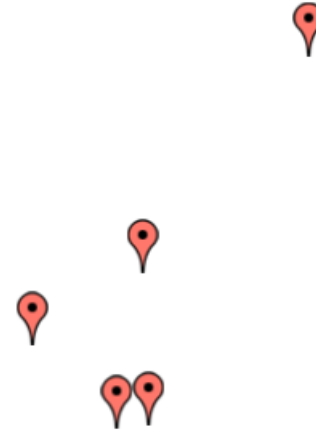
# Point symbolizer

- Applies to all geometry types (centroids used)
- Can use simple marks or external images

```
<PointSymbolizer>
  <Graphic>
    <Mark>
      <WellKnownName>square</WellKnownName>
      <Fill>
        <CssParameter name="fill">
          #FF0000
        </CssParameter>
      </Fill>
    </Mark>
    <Size>6</Size>
  </Graphic>
</PointSymbolizer>
```



```
<PointSymbolizer>
  <Graphic>
    <ExternalGraphic>
      <!-- avoid hot linking, this is just a sample -->
      <OnlineResource xlink:type="simple"
        xlink:href="http://www.google.com/mapfiles/marker.png" />
      <Format>image/png</Format>
    </ExternalGraphic>
  </Graphic>
</PointSymbolizer>
```





# Line symbolizer

- Controls how a line is displayed
- Can use continuous lines, dash arrays, but also graphical symbols

```
<LineSymbolizer>
  <Stroke>
    <CssParameter name="stroke">#0000FF</CssParameter>
    <CssParameter name="stroke-width">6</CssParameter>
    <CssParameter name="stroke-linejoin">round</CssParameter>
    <CssParameter name="stroke-linecap">round</CssParameter>
  </Stroke>
</LineSymbolizer>
```



```
<LineSymbolizer>
  <Stroke>
    <CssParameter name="stroke">
      #000000
    </CssParameter>
    <CssParameter name="stroke-dasharray">
      10 5 1 5
    </CssParameter>
  </Stroke>
</LineSymbolizer>
```





# Polygon symbolizer

- Has both a stroke and a fill
- Can use solid color, partial transparency, or graphic fills

```
<PolygonSymbolizer>
  <Fill>
    <CssParameter name="fill">
      #AAAAAA
    </CssParameter>
  </Fill>
  <Stroke>
    <CssParameter name="stroke">
      #000000
    </CssParameter>
    <CssParameter name="stroke-width">1</CssParameter>
  </Stroke>
</PolygonSymbolizer>
```



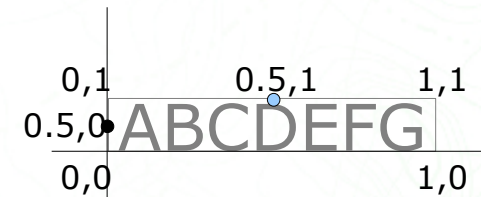
```
<PolygonSymbolizer>
  <Fill>
    <GraphicFill>
      <Graphic>
        <ExternalGraphic>
          <OnlineResource xlink:type="simple"
            xlink:href="grass_fill.png" />
          <Format>image/png</Format>
        </ExternalGraphic>
      </Graphic>
    </GraphicFill>
  </Fill>
  <Stroke />
</PolygonSymbolizer>
```





# Text symbolizer

```
<TextSymbolizer>
  <Label>
    <ogc:PropertyName>NAME</ogc:PropertyName>
  </Label>
  <Font>
    <CssParameter name="font-family">Arial</CssParameter>
    <CssParameter name="font-weight">Bold</CssParameter>
    <CssParameter name="font-size">14</CssParameter>
  </Font>
  <LabelPlacement>
    <PointPlacement>
      <AnchorPoint>
        <AnchorPointX>0.5</AnchorPointX>
        <AnchorPointY>0.5</AnchorPointY>
      </AnchorPoint>
      <Displacement>
        <DisplacementX>0</DisplacementX>
        <DisplacementY>-15</DisplacementY>
      </Displacement>
    </PointPlacement>
  </LabelPlacement>
  <Halo>
    <Radius>
      <ogc:Literal>2</ogc:Literal>
    </Radius>
    <Fill>
      <CssParameter name="fill">#FFFFFF</CssParameter>
    </Fill>
  </Halo>
  <Fill>
    <CssParameter name="fill">#000000</CssParameter>
  </Fill>
</TextSymbolizer>
```



AnchorPoint legend

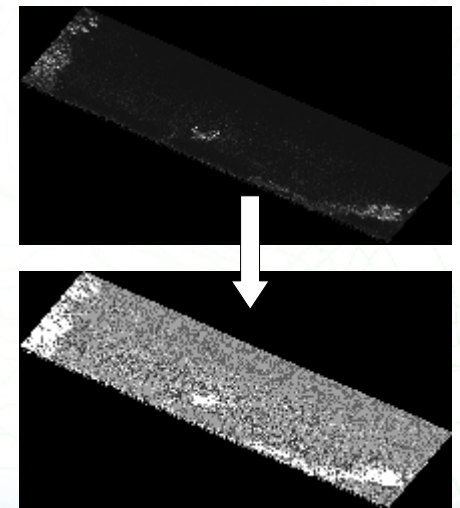
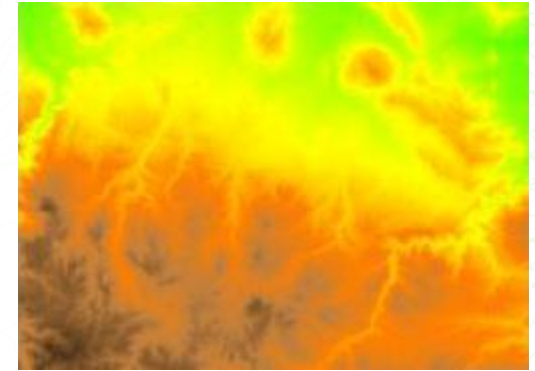


# Raster symbolizer

- Can apply normalizations, extract bands, apply colormaps

```
<RasterSymbolizer>
  <Opacity>1.0</Opacity>
  <ColorMap>
    <ColorMapEntry color="#AAFFAA" quantity="0" label="values" />
    <ColorMapEntry color="#00FF00" quantity="1000" label="values"/>
    <ColorMapEntry color="#FFFF00" quantity="1200" label="values" />
    <ColorMapEntry color="#FF7F00" quantity="1400" label="values" />
    <ColorMapEntry color="#BF7F3F" quantity="1600" label="values" />
    <ColorMapEntry color="#000000" quantity="2000" label="values" />
  </ColorMap>
</RasterSymbolizer>
```

```
<RasterSymbolizer>
  <ChannelSelection>
    <GrayChannel>
      <SourceChannelName>1</SourceChannelName>
      <ContrastEnhancement>
        <Histogram/>
      </ContrastEnhancement>
    </GrayChannel>
  </ChannelSelection>
  <ContrastEnhancement>
    <GammaValue>1</GammaValue>
  </ContrastEnhancement>
</RasterSymbolizer>
```





## Advanced styling





# Thematic mapping with a twist

- Let's assume you have some polygon data and that you want to make a simple thematic map, qualitative approach: green areas, water and urban
- The original data has a classification field, CFCC, that is much more detailed
- However:
  - All green areas are like D84, D85, D86, ...
  - All water areas start with H
  - All other areas can be considered urban
- We could pre-process the data and reclassify it, but if we cannot or just don't want to...



# Thematic map with a twist

```
<FeatureTypeStyle>
  <!-- park and green spaces -->
  <Rule>
    <ogc:Filter>
      <!-- D84, D85, D86, ... -->
      <ogc:PropertyIsLike wildCard="%" singleChar="_" escape="\ ">
        <ogc:PropertyName>CFCC</ogc:PropertyName>
        <ogc:Literal>D8%</ogc:Literal>
      </ogc:PropertyIsLike>
    </ogc:Filter>
    <PolygonSymbolizer>
      <Fill>
        <CssParameter name="fill">#B4DFB4</CssParameter>
      </Fill>
      <Stroke>
        <CssParameter name="stroke">#88B588</CssParameter>
      </Stroke>
    </PolygonSymbolizer>
  </Rule>
  <!-- water -->
  <Rule>
    <ogc:Filter>
      <ogc:PropertyIsLike wildCard="%" singleChar="_" escape="\ ">
        <ogc:PropertyName>CFCC</ogc:PropertyName>
        <ogc:Literal>H%</ogc:Literal>
      </ogc:PropertyIsLike>
    </ogc:Filter>
    <PolygonSymbolizer>
      <Fill>
        <CssParameter name="fill">#8AA9D1</CssParameter>
      </Fill>
      <Stroke>
        <CssParameter name="stroke">#436C91</CssParameter>
      </Stroke>
    </PolygonSymbolizer>
  </Rule>
```

```
<!-- urban -->
<Rule>
  <ElseFilter />
  <PolygonSymbolizer>
    <Fill>
      <CssParameter name="fill">
        #A5A5A5
      </CssParameter>
    </Fill>
    <Stroke>
      <CssParameter name="stroke">
        #6E6E6E
      </CssParameter>
    </Stroke>
  </PolygonSymbolizer>
</Rule>
</FeatureTypeStyle>
```





# Stacking multiple symbolizers - points

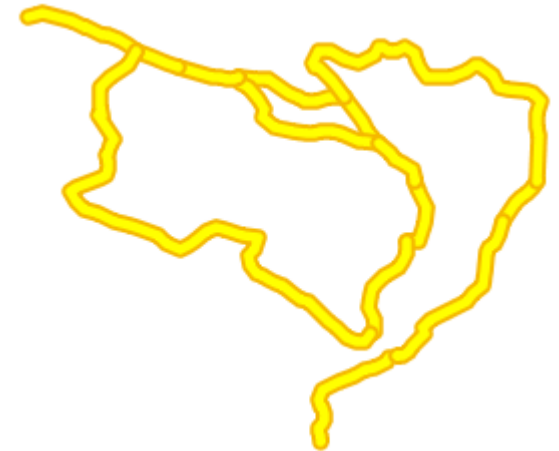
```
<FeatureTypeStyle>
  <Rule>
    <PointSymbolizer>
      <Graphic>
        <Mark>
          <WellKnownName>circle</WellKnownName>
          <Fill>
            <CssParameter name="fill">#FF0000</CssParameter>
          </Fill>
        </Mark>
        <Size>11</Size>
      </Graphic>
    </PointSymbolizer>
    <PointSymbolizer>
      <Graphic>
        <Mark>
          <WellKnownName>circle</WellKnownName>
          <Fill>
            <CssParameter name="fill">#EDE513</CssParameter>
          </Fill>
        </Mark>
        <Size>7</Size>
      </Graphic>
    </PointSymbolizer>
  </Rule>
</FeatureTypeStyle>
```





# Stacking multiple symbolizers - lines

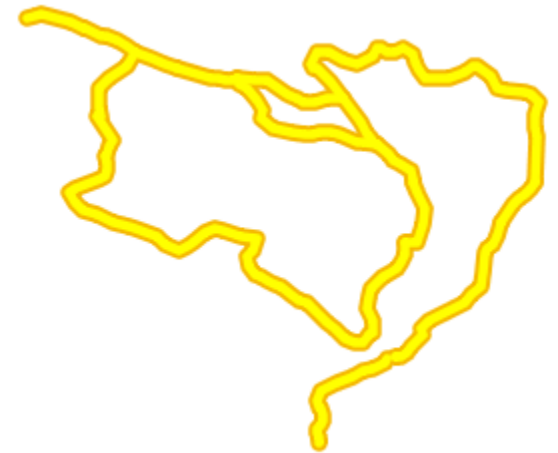
```
<FeatureTypeStyle>
  <Rule>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#F5B800</CssParameter>
        <CssParameter name="stroke-width">8</CssParameter>
        <CssParameter name="stroke-linejoin">round</CssParameter>
        <CssParameter name="stroke-linecap">round</CssParameter>
      </Stroke>
    </LineSymbolizer>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#FFFF00</CssParameter>
        <CssParameter name="stroke-width">4</CssParameter>
        <CssParameter name="stroke-linejoin">round</CssParameter>
        <CssParameter name="stroke-linecap">round</CssParameter>
      </Stroke>
    </LineSymbolizer>
  </Rule>
</FeatureTypeStyle>
```





# Stacking multiple symbolizers – lines 2

```
<FeatureTypeStyle>
  <Rule>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#F5B800</CssParameter>
        <CssParameter name="stroke-width">8</CssParameter>
        <CssParameter name="stroke-linejoin">round</CssParameter>
        <CssParameter name="stroke-linecap">round</CssParameter>
      </Stroke>
    </LineSymbolizer>
  </Rule>
</FeatureTypeStyle>
<FeatureTypeStyle>
  <Rule>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#FFFF00</CssParameter>
        <CssParameter name="stroke-width">4</CssParameter>
        <CssParameter name="stroke-linejoin">round</CssParameter>
        <CssParameter name="stroke-linecap">round</CssParameter>
      </Stroke>
    </LineSymbolizer>
  </Rule>
</FeatureTypeStyle>
```





# Scale dependencies – show labels

```
<FeatureTypeStyle>
  <Rule>
    <!-- no scale dependencies: DISPLAYS ALWAYS -->
    <PointSymbolizer>
      <Graphic>
        <Mark>
          <WellKnownName>circle</WellKnownName>
          <Fill>
            <CssParameter name="fill">
              #FF0000
            </CssParameter>
          </Fill>
        </Mark>
        <Size>11</Size>
      </Graphic>
    </PointSymbolizer>
    <PointSymbolizer>
      <Graphic>
        <Mark>
          <WellKnownName>circle</WellKnownName>
          <Fill>
            <CssParameter name="fill">
              #EDE513
            </CssParameter>
          </Fill>
        </Mark>
        <Size>7</Size>
      </Graphic>
    </PointSymbolizer>
  </Rule>
```

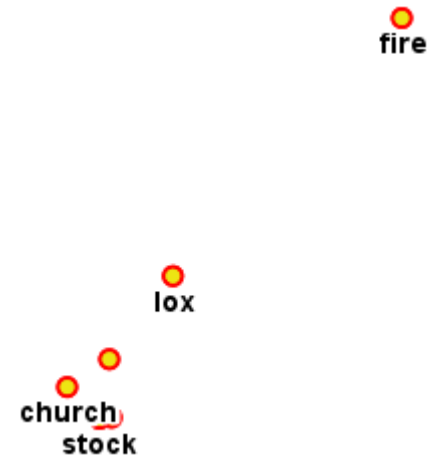
```
<Rule>
  <MaxScaleDenominator>32000</MaxScaleDenominator>
  <TextSymbolizer>
    <Label><ogc:PropertyName>NAME</ogc:PropertyName></Label>
    <Font>
      <CssParameter name="font-family">Arial</CssParameter>
      <CssParameter name="font-weight">Bold</CssParameter>
      <CssParameter name="font-size">14</CssParameter>
    </Font>
    <LabelPlacement>
      <PointPlacement>
        <AnchorPoint>
          <AnchorPointX>0.5</AnchorPointX>
          <AnchorPointY>0.5</AnchorPointY>
        </AnchorPoint>
        <Displacement>
          <DisplacementX>0</DisplacementX>
          <DisplacementY>-15</DisplacementY>
        </Displacement>
      </PointPlacement>
    </LabelPlacement>
    <Halo>
      <Radius><ogc:Literal>2</ogc:Literal></Radius>
      <Fill>
        <CssParameter name="fill">#FFFFFF</CssParameter>
      </Fill>
    </Halo>
    <Fill>
      <CssParameter name="fill">#000000</CssParameter>
    </Fill>
  </TextSymbolizer>
</Rule>
</FeatureTypeStyle>
```



# Scale dependencies – show labels



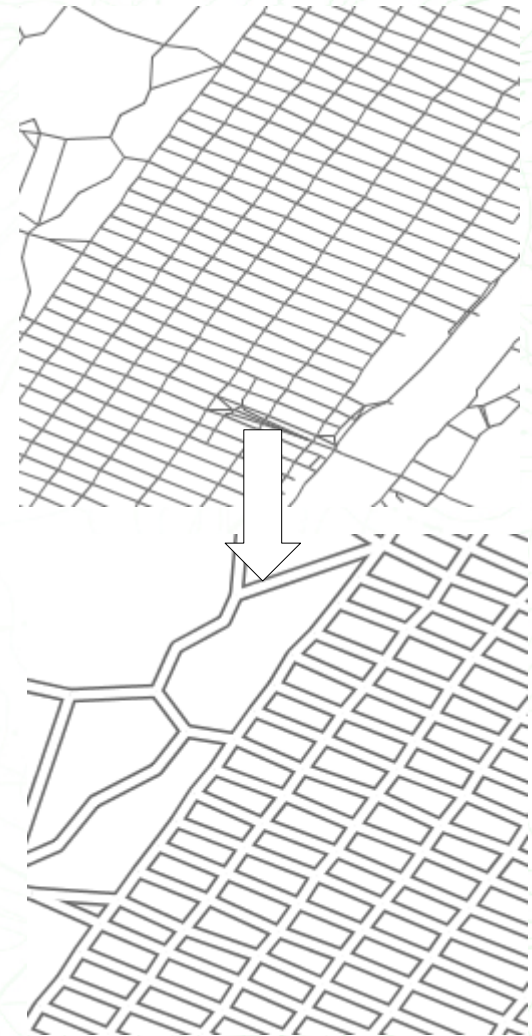
Zoom in





# Scale dependencies – alternate styling

```
<FeatureTypeStyle>
  <Rule> <!-- thin line only at lower scales -->
    <MinScaleDenominator>32000</MinScaleDenominator>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#666666</CssParameter>
        <CssParameter name="stroke-width">0.5</CssParameter>
      </Stroke>
    </LineSymbolizer>
  </Rule>
  <Rule> <!-- thick line drawn first-->
    <MaxScaleDenominator>32000</MaxScaleDenominator>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#666666</CssParameter>
        <CssParameter name="stroke-width">7</CssParameter>
      </Stroke>
    </LineSymbolizer>
  </Rule>
</FeatureTypeStyle>
<FeatureTypeStyle>
  <FeatureTypeName>Feature</FeatureTypeName>
  <Rule> <!-- thin line drawn second -->
    <MaxScaleDenominator>32000</MaxScaleDenominator>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#FFFFFF</CssParameter>
        <CssParameter name="stroke-width">4</CssParameter>
      </Stroke>
    </LineSymbolizer>
  </Rule>
</FeatureTypeStyle>
```







# Simple labelling

```
<TextSymbolizer>
  <Label>
    <ogc:PropertyName>NAME</ogc:PropertyName>
  </Label>
  <Font>
    <CssParameter name="font-family">Times New Roman</CssParameter>
    <CssParameter name="font-style">Normal</CssParameter>
    <CssParameter name="font-size">14</CssParameter>
    <CssParameter name="font-weight">bold</CssParameter>
  </Font>
  <LabelPlacement>
    <LinePlacement>
    </LinePlacement>
  </LabelPlacement>
  <Halo>
    <Radius>
      <ogc:Literal>2</ogc:Literal>
    </Radius>
    <Fill>
      <CssParameter name="fill">
        #FFFFFF
      </CssParameter>
    </Fill>
  </Halo>
</TextSymbolizer>
```





# Advanced labelling with vendor options

```
<TextSymbolizer>
  <Label>
    <ogc:PropertyName>NAME</ogc:PropertyName>
  </Label>
  <Font>
    <CssParameter name="font-family">Times New Roman</CssParameter>
    <CssParameter name="font-style">Normal</CssParameter>
    <CssParameter name="font-size">14</CssParameter>
    <CssParameter name="font-weight">bold</CssParameter>
  </Font>
  <LabelPlacement>
    <LinePlacement>
    </LinePlacement>
  </LabelPlacement>
  <Halo>
    <Radius>
      <ogc:Literal>2</ogc:Literal>
    </Radius>
    <Fill>
      <CssParameter name="fill">#FFFFFF</CssParameter>
    </Fill>
  </Halo>
  <VendorOption name="followLine">True
</VendorOption>
  <VendorOption name="group">True
</VendorOption>
  <VendorOption name="repeat">200
</VendorOption>
  <VendorOption name="maxDisplacement">50
</VendorOption>
</TextSymbolizer>
```





# Auto-wrapping labels

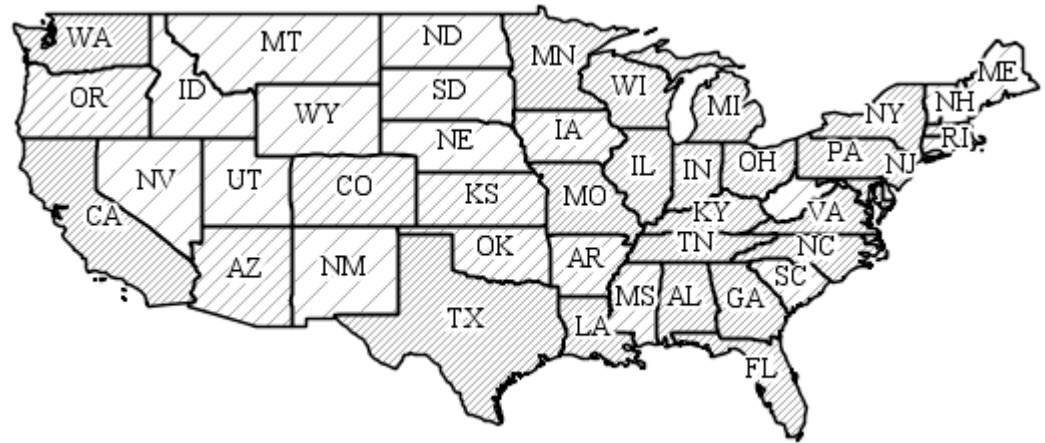
```
<FeatureTypeStyle>
  <Rule>
    <TextSymbolizer>
      <Label>
        <ogc:PropertyName>LANAME</ogc:PropertyName>
      </Label>
      <Font>
        <CssParameter name="font-family">Times New Roman
        </CssParameter>
        <CssParameter name="font-style">Normal</CssParameter>
        <CssParameter name="font-size">14</CssParameter>
        <CssParameter name="font-weight">bold</CssParameter>
      </Font>
      <LabelPlacement>
        <PointPlacement>
          <AnchorPoint>
            <AnchorPointX>0.5</AnchorPointX>
            <AnchorPointY>0.5</AnchorPointY>
          </AnchorPoint>
        </PointPlacement>
      </LabelPlacement>
      <Halo>
        <Radius>
          <ogc:Literal>2</ogc:Literal>
        </Radius>
        <Fill>
          <CssParameter name="fill">#FDE5A5</CssParameter>
          <CssParameter name="fill-opacity">0.75</CssParameter>
        </Fill>
      </Halo>
      <Fill>
        <CssParameter name="fill">#000000</CssParameter>
      </Fill>
      <VendorOption name="group">>true</VendorOption>
      <VendorOption name="autoWrap">100</VendorOption>
    </TextSymbolizer>
  </Rule>
</FeatureTypeStyle>
```





# Thematic mapping with hatch density

```
<FeatureTypeStyle>
  <Rule>
    <Title>&lt; 2M</Title>
    <ogc:Filter>
      <ogc:PropertyIsLessThan>
        <ogc:PropertyName>PERSONS</ogc:PropertyName>
        <ogc:Literal>2000000</ogc:Literal>
      </ogc:PropertyIsLessThan>
    </ogc:Filter>
    <PolygonSymbolizer>
      <Fill>
        <GraphicFill>
          <Graphic>
            <Mark>
              <WellKnownName>shape://slash</WellKnownName>
              <Stroke>
                <CssParameter name="stroke">0xAAAAAA</CssParameter>
              </Stroke>
            </Mark>
            <Size>16</Size>
          </Graphic>
        </GraphicFill>
      </Fill>
    </PolygonSymbolizer>
  </Rule>
  ...
  ...
</FeatureTypeStyle>
```



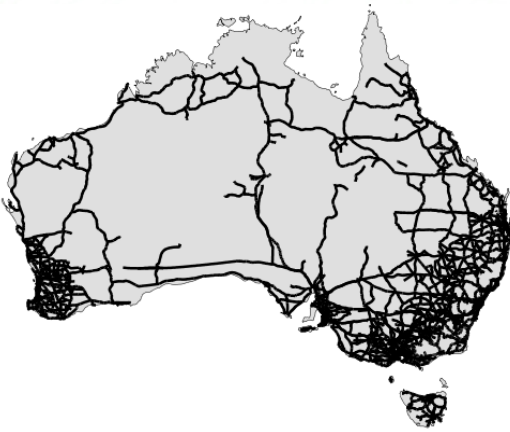


## Performance



# Performance considerations

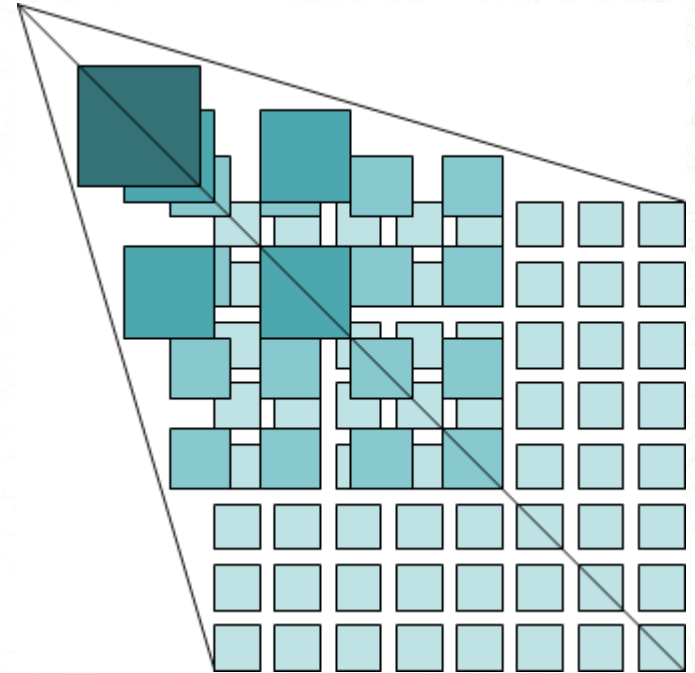
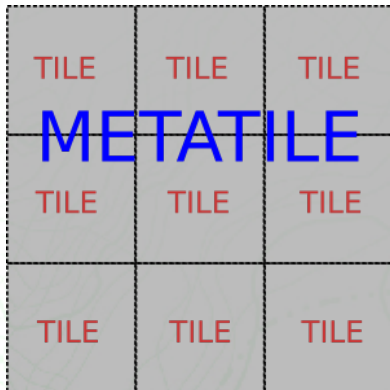
- Draw less
  - Style filters
  - Scale filters
  - Swap layers
- Simplify geometry
  - Don't draw detail that can't be discerned





# Performance considerations

- Avoid expensive styling options
  - Transparency
  - Labels are expensive
  - Halos are very expensive
- Are you Tile Caching?
  - Labels and metatiling





# Thanks!

- FOSS4G Survey – please provide feedback
- Link: <http://tinyurl.com/foss4g-survey>
- Tutorial Code: TUT-08-96
- Materials:
- [http://dl.getdropbox.com/u/2361103/FOSS4G\\_SLD.tar.gz](http://dl.getdropbox.com/u/2361103/FOSS4G_SLD.tar.gz)