

Building a web mapping application with GeoExt

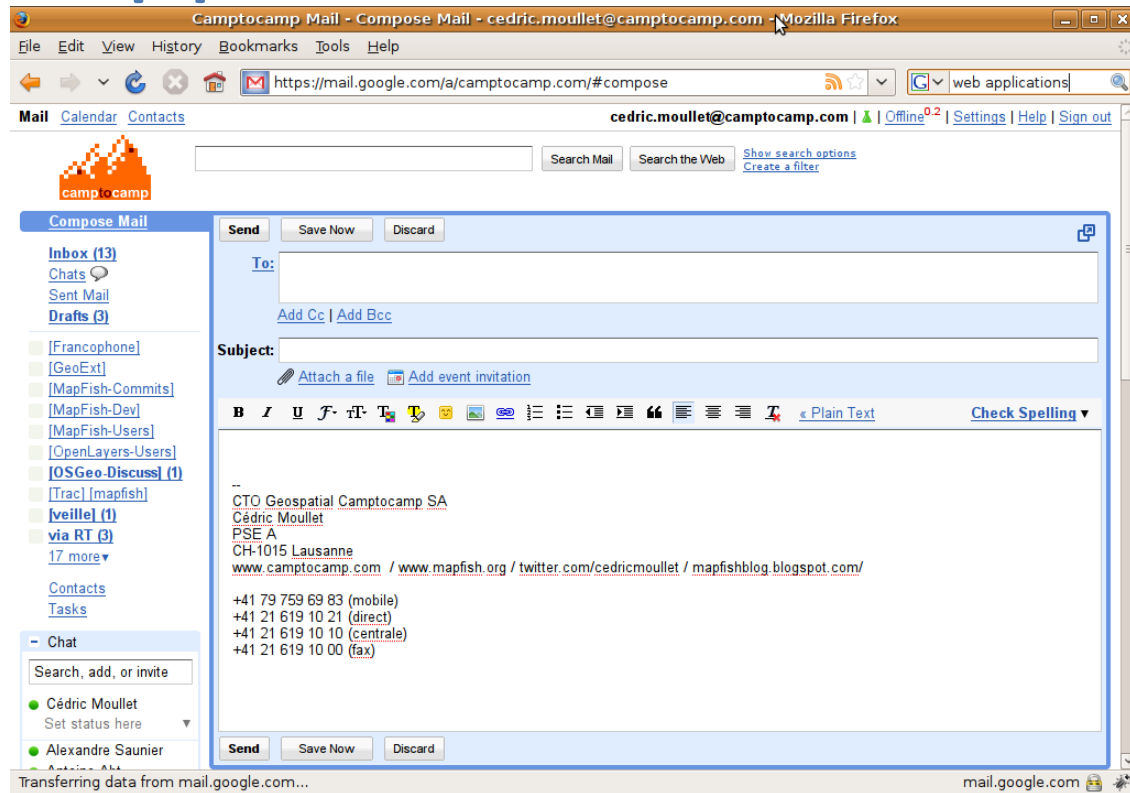
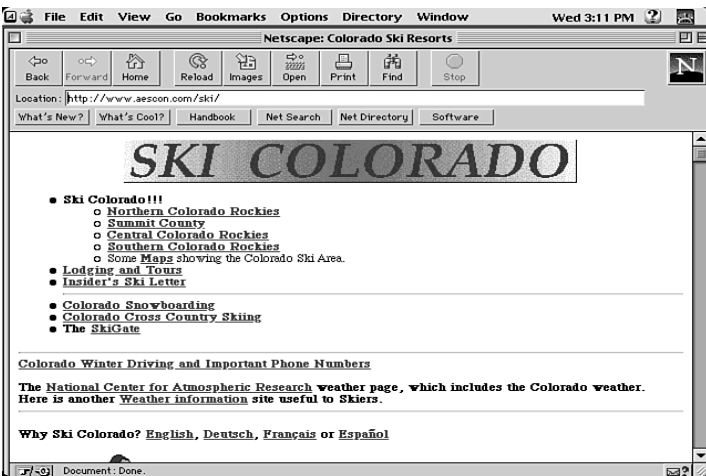
Andreas Hocevar & Cédric Moullet



Why GeoExt ?



From web pages to web applications



Rich UI

Toolbars

Window

Menu

AJAX

Tree

Combo

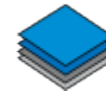
Tab

Grid

Layout

Panel

ExtJS + OpenLayers = GeoExt



... not a revolutionary idea: MapFish, OpenGeo, MapGears, ExtMap, others...



PSC

- Bart van den Eijnden
- Tim Schaub
- Eric Lemoine
- Andreas Hocevar
- Cédric Moullet

Committers

- Andreas Hocevar
- Bart van den Eijnden
- Eric Lemoine
- Frédéric Junod
- Patrick Valsecchi
- Pierre Giraud
- Tim Coulter
- Tim Schaub
- * Sebastian Benthall
- * Bruno Binet

Timeline

July 2009

October 2009



GeoExt 0.5
ExtJS 2.2.1
OL 2.8

GeoExt 0.6
ExtJS 2.x, 3.0
OL 2.8+

Licenses and Copyrights

- BSD License
- Copyrights to OSGEO
- Ext JS Dual Licensing (GPL3 or commercial)

Latest News ?

- Release 0.6
 - More WMS and WFS readers
 - Synchronized grid & layer selection of features
 - More Layer Tree options (incl. drag-n-drop)
 - LegendPanel
 - Sliders
- Roadmap in discussion
- www.geoext.org

The Concept

- Ext JS is great for rich web applications, but lacks mapping
- OpenLayers is great for maps, but lacks a rich UI
- Put the OpenLayers map in an Ext panel
- Wrap OpenLayers objects in Ext data components



JavaScript Toolkit for Rich Web Mapping Applications

GeoExt brings together the geospatial know how of OpenLayers with the user interface savvy of [Ext JS](#) to help you build powerful web mapping applications with the web with JavaScript.

[Documentation](#) | [Examples](#) | [Download](#) | [Development](#)

Using GeoExt

See GeoExt in action.

```
new Ext.Window({
  title: "GeoExt in Action",
  height: 280, width: 450, layout: "fit",
  items: [{
    xtype: "gx_mappanel",
    layers: [new OpenLayers.Layer.WMS(
      "Global Imagery", "http://demo.opengeo.org/geoserver/wms",
      {layers: 'bluemarble'}
    )],
    zoom: 1
  }]
}).show();
```

Run it!

Download

Current release: 0.5



12 lines of code

Create and show a window with a GeoExt.MapPanel



About and Around The MapPanel

- Content is an OpenLayers.Map
- Access layers through a LayerStore
- Add button controls through Actions
- Add anchored Popup windows
- Panel populates a ScaleStore
- Panel's items interact with the map (e.g. ZoomSlider)



24 lines of code

Add 3 controls to the bottom bar using GeoExt.Action

7+6 lines of code

Pimp the ZoomBar with a GeoExt.ZoomSlider

7 lines of code

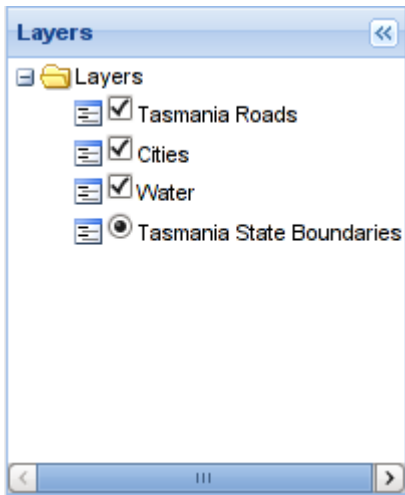
Create and show an anchored GeoExt.Popup window

27 lines of code

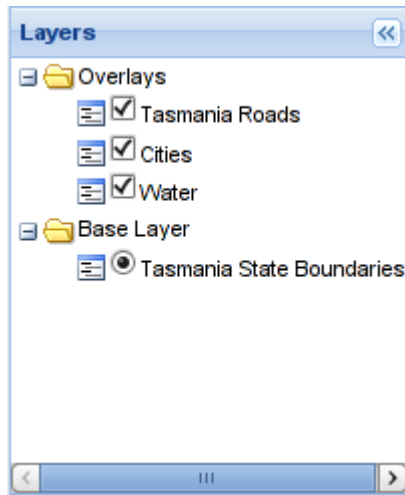
Show/set scale in a drop-down using GeoExt.ScaleStore

Tree View For Layers

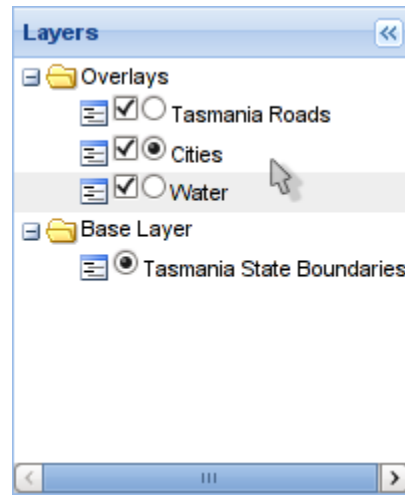
- Synchronized with map
- Control visibility and order of layers (drag & drop)
- Group layers with LayerContainers
- LayerParamNode for sub-layers (e.g. &LAYERS=sub1,sub2,sub3)
- Flexible configuration with Nodes and Loaders, mixable with other content



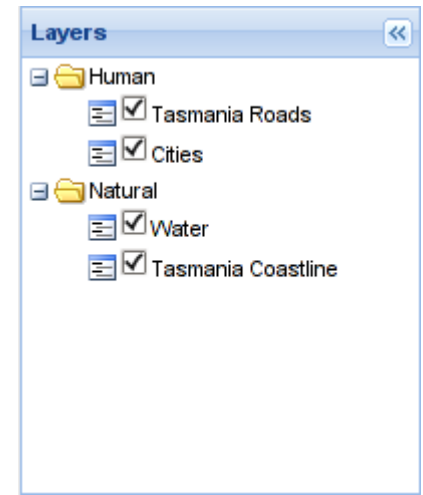
```
[[
  "nodeType": „gx_layercontainer“
]]
```



```
[[
  "nodeType": „gx_overlaylayercontainer“
},{
  "nodeType": „gx_baselayercontainer“
}]
```



```
[[
  "nodeType": „gx_overlaylayercontainer“,
  "loader": {
    "baseAttrs": { "radioGroup": "active" }
  }
},{
  "nodeType": „gx_baselayercontainer“
}]
```



```
[[
  "nodeType": "async",
  "text": "Human",
  "children": [{
    "nodeType": "gx_layer",
    "layer": "Tasmania Roads"
  }, {
    "nodeType": "gx_layer",
    "layer": "Cities"
  }
]}, {
  "nodeType": "async",
  "text": "Natural",
  "children": [{
    "nodeType": "gx_layer",
    "layer": "Water"
  }, {
    "nodeType": "gx_layer",
    "layer": "Tasmania State Boundaries",
    "text": "Tasmania Coastline"
  }
]}
]]
```

With radio buttons

GeoExt.tree.OverlayLayerContainer
GeoExt.tree.BaseLayerContainer

GeoExt.tree.LayerContainer

Custom groups
Custom node text
Map with „allOverlays: true“

Data Access Components

- Treat OpenLayers Formats and objects like Ext JS remote data, using Readers and Stores
- The ProtocolProxy wraps OpenLayers Protocols as input for Readers
- 2-way synchronization between Stores and OpenLayers objects

4 lines of code

Load remote WMS Capabilities into a
`GeoExt.data.WMSCapabilitiesStore`

WMS Capabilities	
Name	Title
tiger:poly_landmark	Manhattan (NY) landmarks
tiger:poi	Manhattan (NY) points of interest
tiger:tiger_roads	Manhattan (NY) roads
topp:tasmania_citie	Tasmania cities
topp:tasmania_roac	Tasmania roads
topp:tasmania_stat	Tasmania state boundaries
topp:tasmania_wat	Tasmania water bodies
topp:states	USA Population
tiger:giant_polygon	World rectangle
og:archsites	archsites_Type
og:buasites	buasites_Type

Add Layer

Map Layers

- Manhattan (NY) points of interest
- Manhattan (NY) landmarks
- Manhattan (NY) roads



1 line of code

`GeoExt.data.LayerStore` shared
between map and tree node

11 lines of code

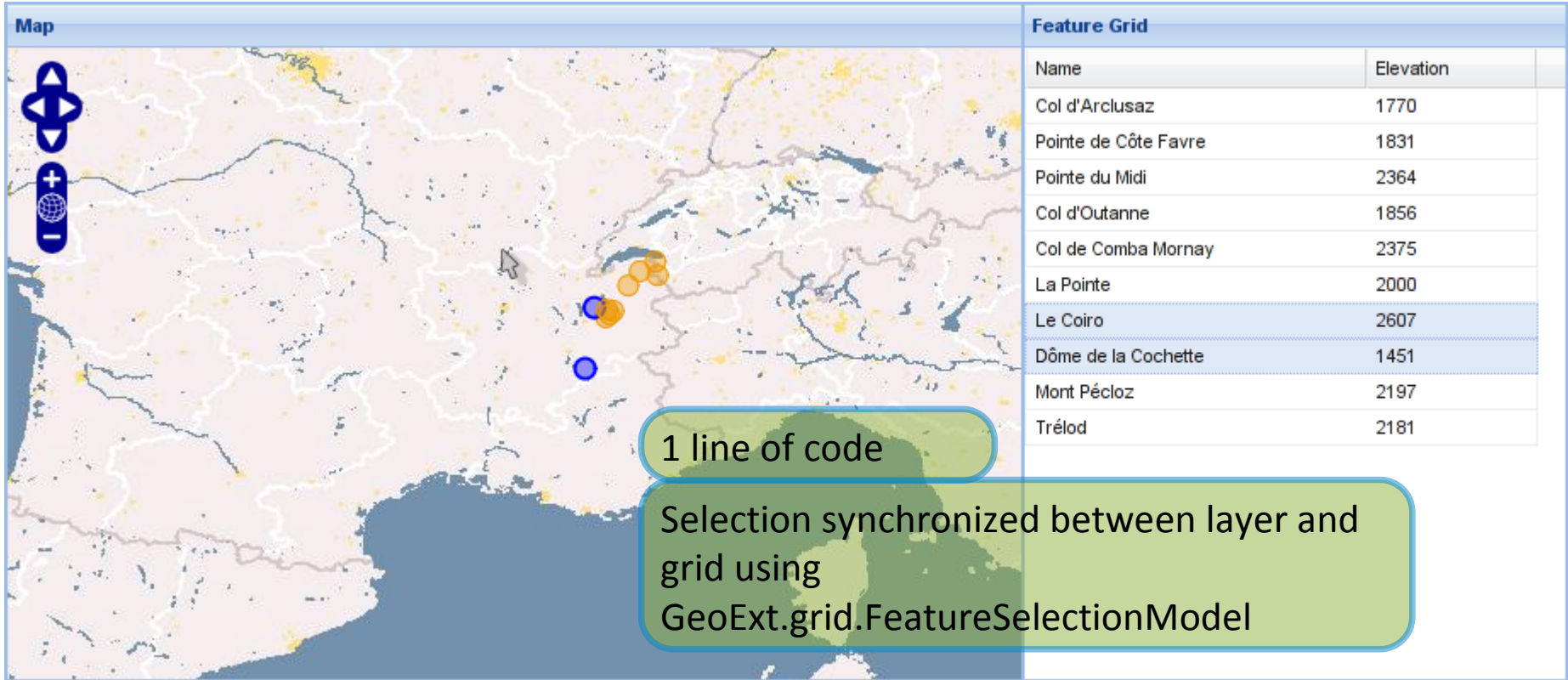
Copy layer to the map's
`GeoExt.data.LayerStore`

5 lines of code

Get remote features (GeoJSON) using `GeoExt.data.ProtocolProxy`

8 lines of code

`GeoExt.data.FeatureStore` shared between vector layer and grid



Map

Feature Grid

Name	Elevation
Col d'Arclusaz	1770
Pointe de Côte Favre	1831
Pointe du Midi	2364
Col d'Outanne	1856
Col de Comba Mornay	2375
La Pointe	2000
Le Coiro	2607
Dôme de la Cochette	1451
Mont Pécloz	2197
Trélod	2181

1 line of code

Selection synchronized between layer and grid using `GeoExt.grid.FeatureSelectionModel`