gvSIG Desktop: written in Java, extendable through Scripting

gvSIG Desktop

Main Features

- Open Source
- Java (jts, gdal, geotools…)
- Plugable
  - Java Plugins
  - Scripting Plugins
- Actual version 2.4.1
- Road to 3.0
  - Improvements in Raster

Development

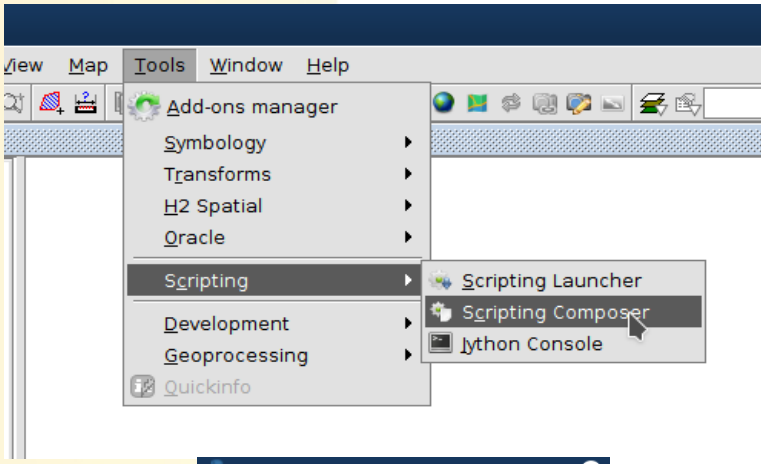
Choose wisely

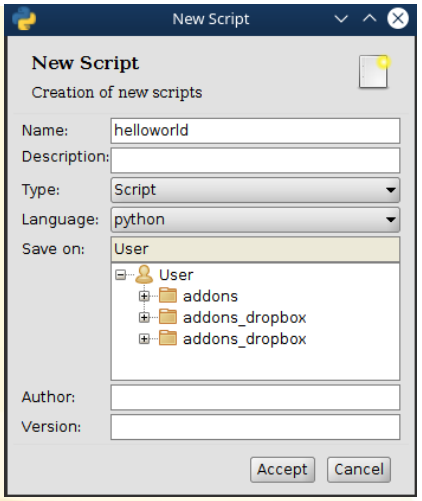
- Do you need to create a new plugin for gvSIG Desktop?
  - Java
    - Good for arquitecture (gvsig core)
    - Bad for a helloworld
  - **Scripting**
    - Easy (Less code with Jython)
    - Fast (gvsig libraries for Python, gvpy,..)
    - IDE inside gvSIG (Integrated development environment)
    - Test and run without leaving/restarting gvSIG
    - Other integrations (R)

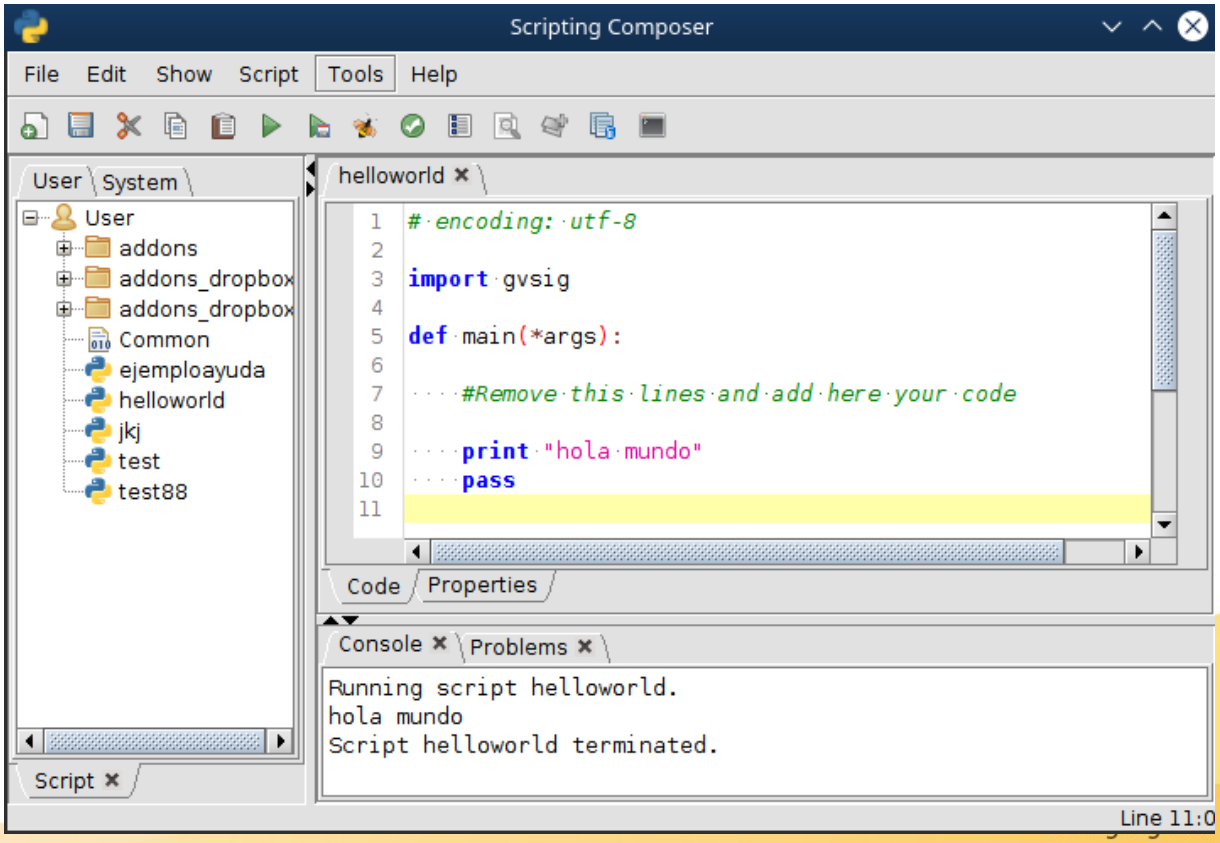
Development

Hello World

- Basic script



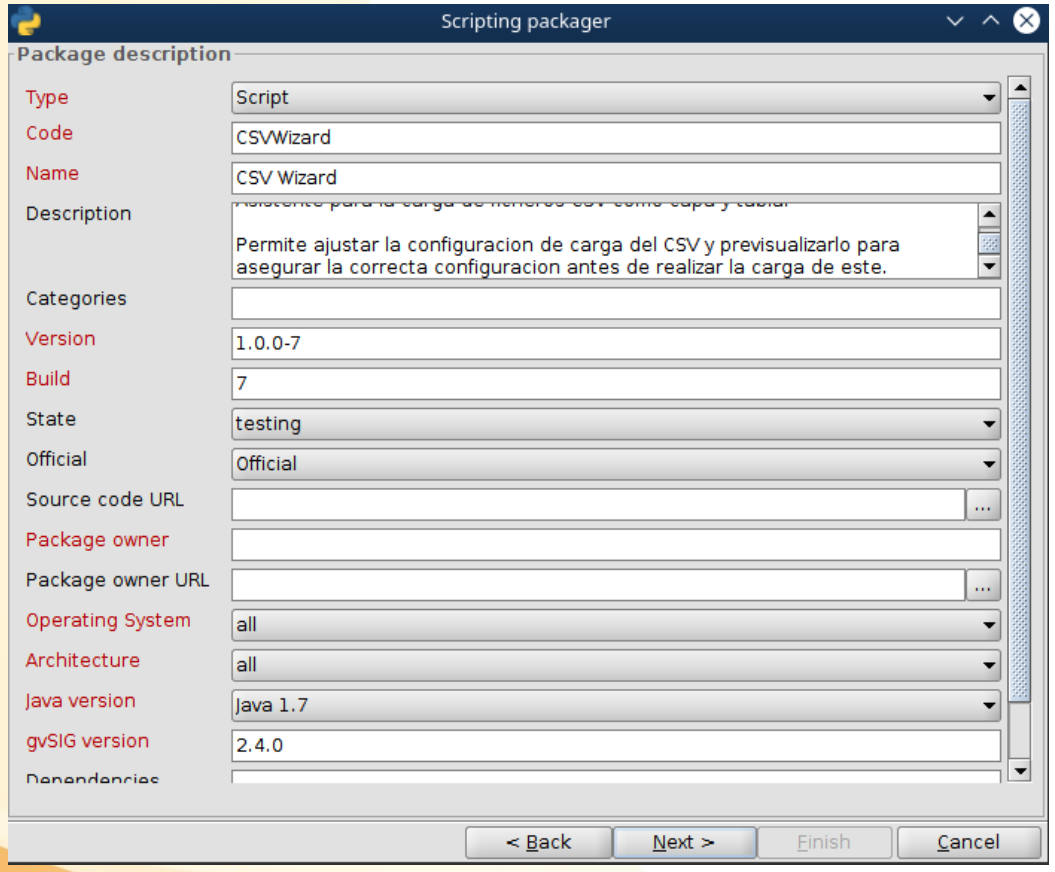


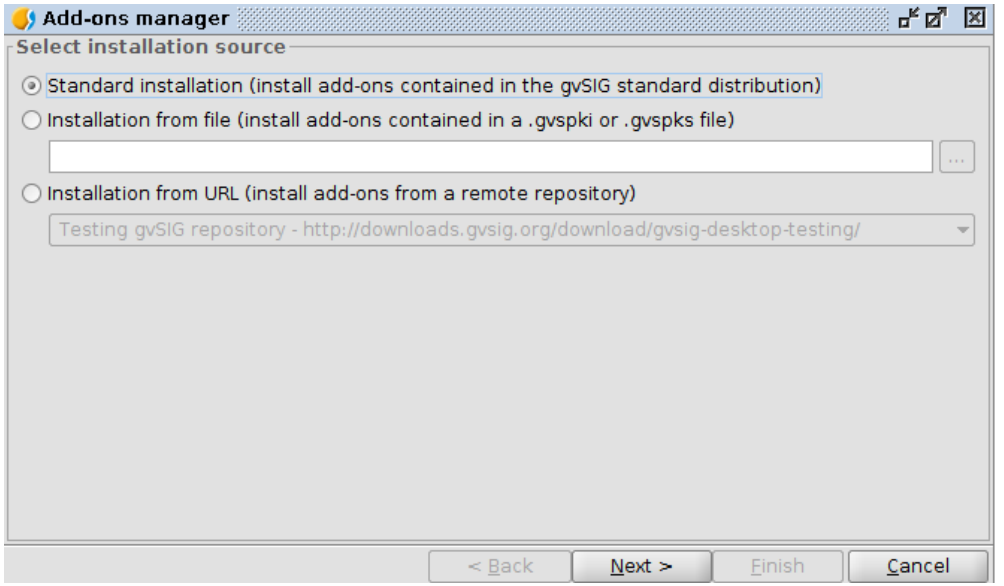


Development

Hello World

- Easy to share





Development

Easy

- **Jython**
  - Implementation of Python in Java
  - Work with Java using Python syntax
  - Available with Java & Python libraries (some limitations)
- **Easy to learn** for people without programming knowledge
  - Create a geoprocess in a few lines
- **Powerful** → Not limited for not using Java
- Easy to create buttons, toolbox, menus,.. for execution

Development

Fast

- **Less code with Jython**
  - Compared with Java
- **Libraries in Python for gvSIG**
  - Only access from Jython
  - **gvsig**
    - Utils: createShape(), currentLayer()
    - Added methods to Java classes
  - **gvpy**: execute a geoprocess with one line of code
  - **Formpanel**: easy and fast to create UI and events

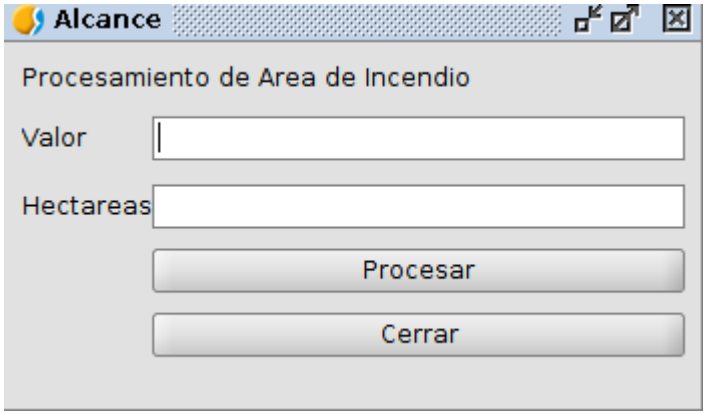
```
gvpy.runalg("randomvector",10, 0)
gvpy.runalg("randomvector",10, gvpy.TYPE_POLYGON)
```

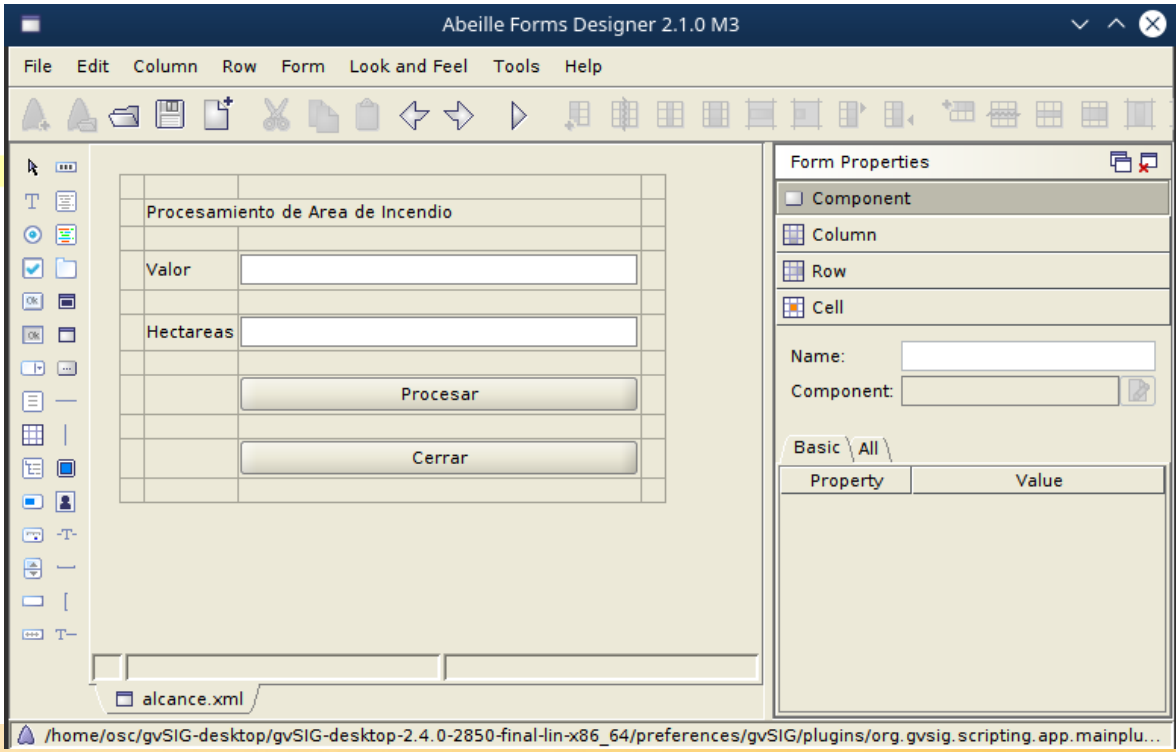
Development

Fast

- **Formpanel**

```
# encoding: utf-8

import gvsig
from gvsig import geom
from gvsig.libs.formpanel import FormPanel
import os

from process import countCells

class Alcance(FormPanel):
    def __init__(self):
        FormPanel.__init__(self,
                gvsig.getResource(__file__,"alcance.xml"))

    def btnProcess_click(self, *args):
        rasterLayer = gvsig.currentLayer()
        if rasterLayer == None:
            self.txtH.setText("Ninguna capa seleccionada")
            return
        valueLimit = float(self.txtValue.getText())
        store = rasterLayer.getDataStore()
        h = countCells(store, valueLimit)
        self.txtH.setText(str(h))
        print "Hectareas: ", h


def main(*args):
    l = Alcance()
    l.showTool("Alcance")
    pass
```

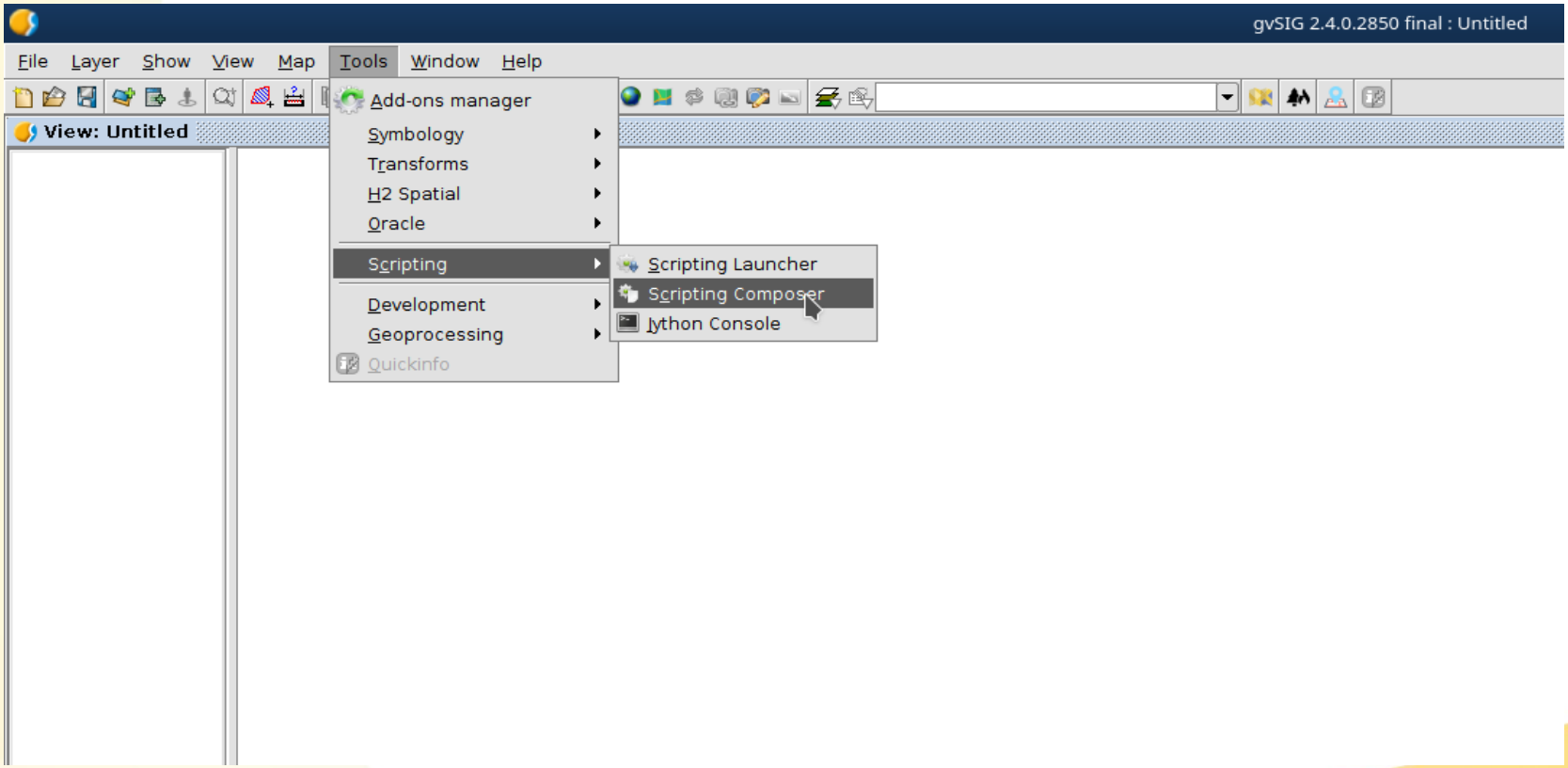




Development

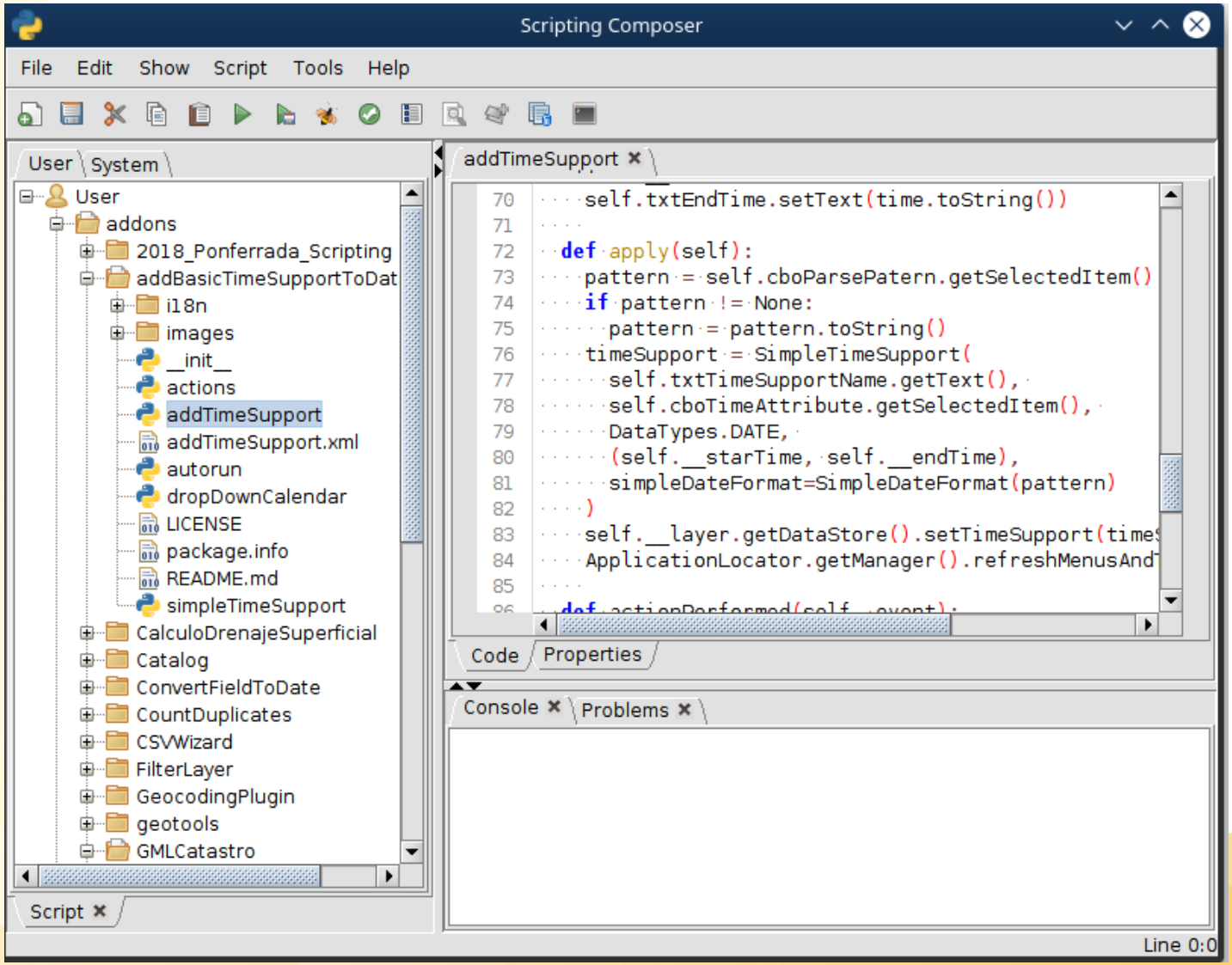
IDE inside gvSIG

- Already in your installation



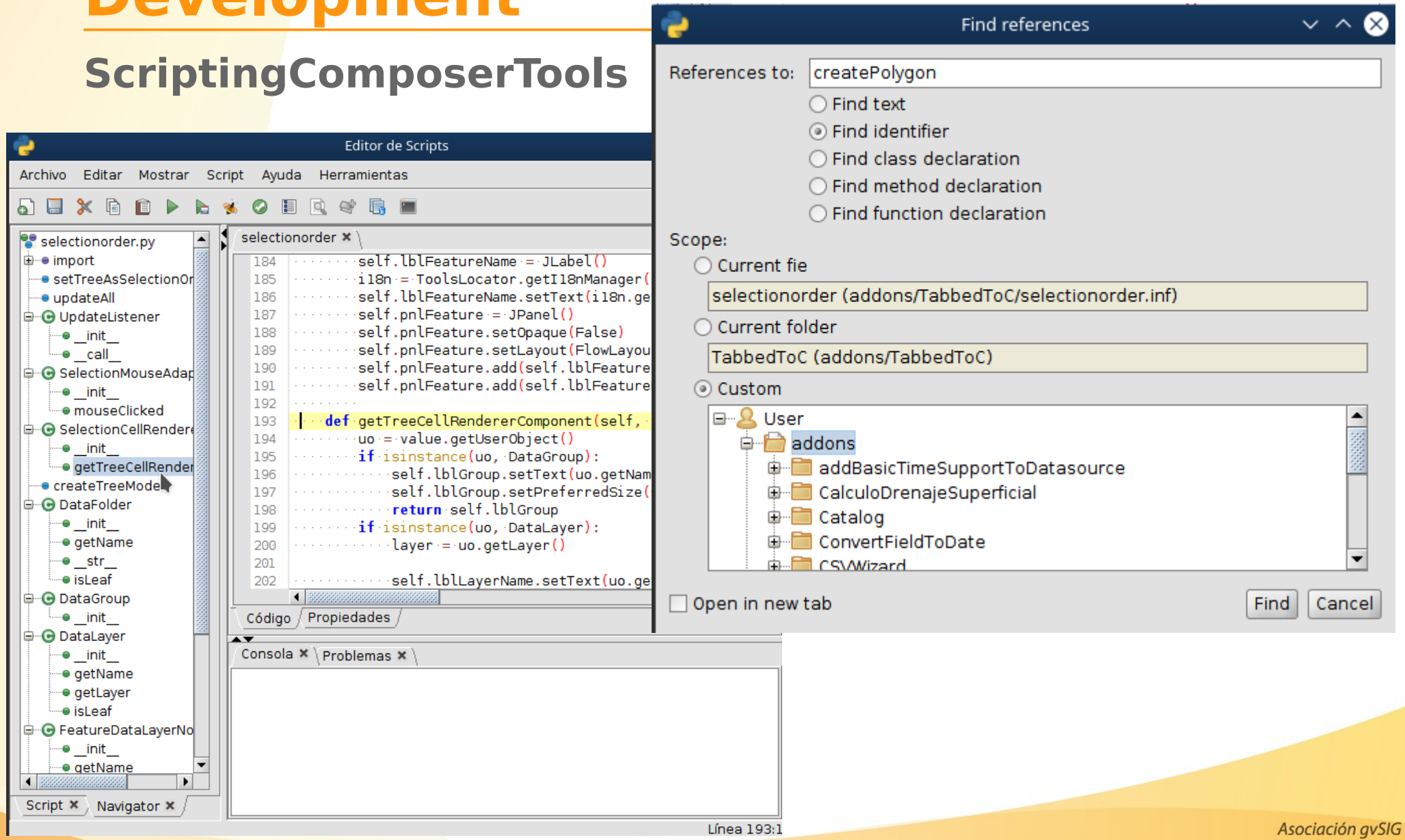
Development

IDE inside gvSIG



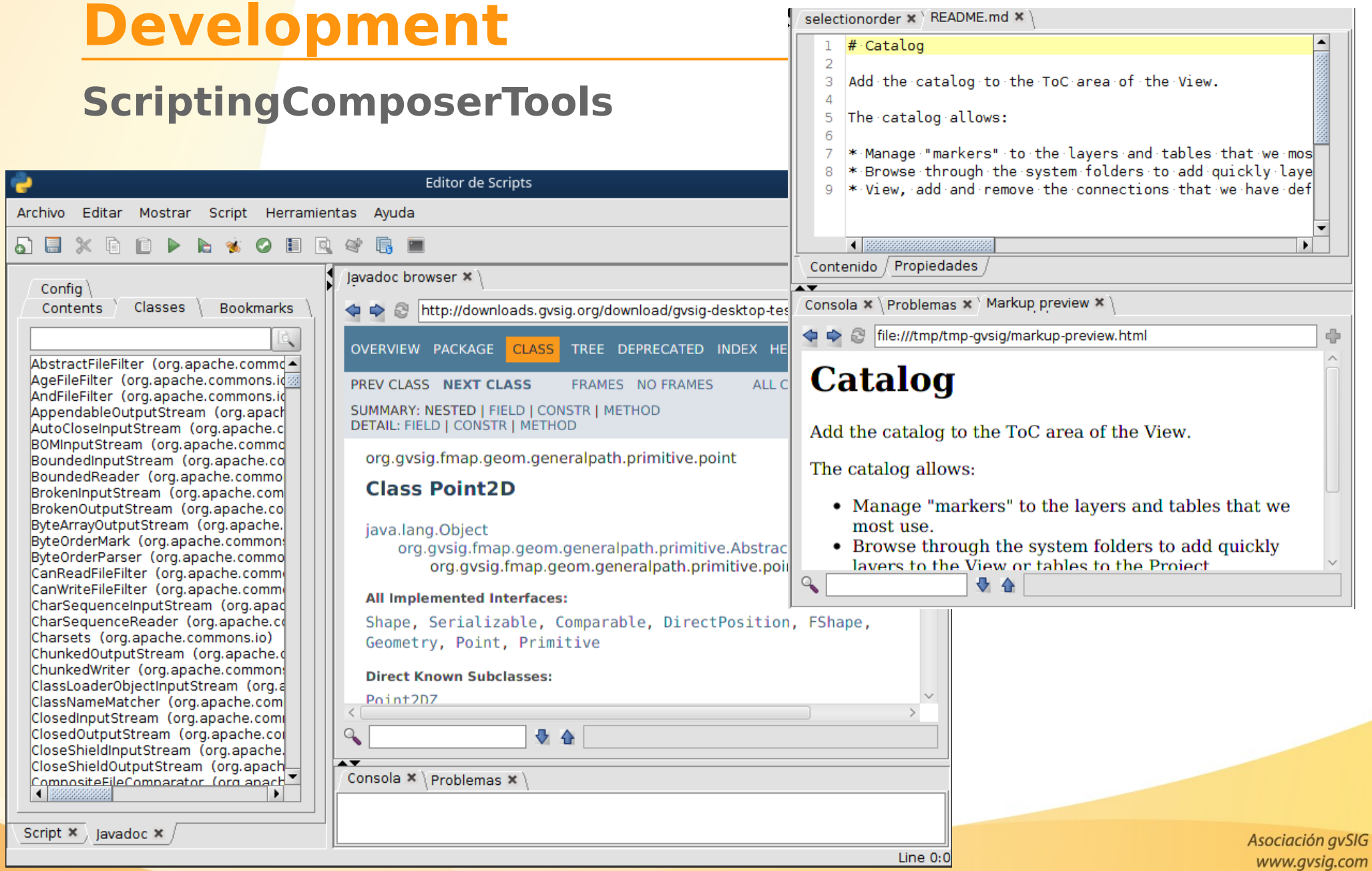
Development

ScriptingComposerTools



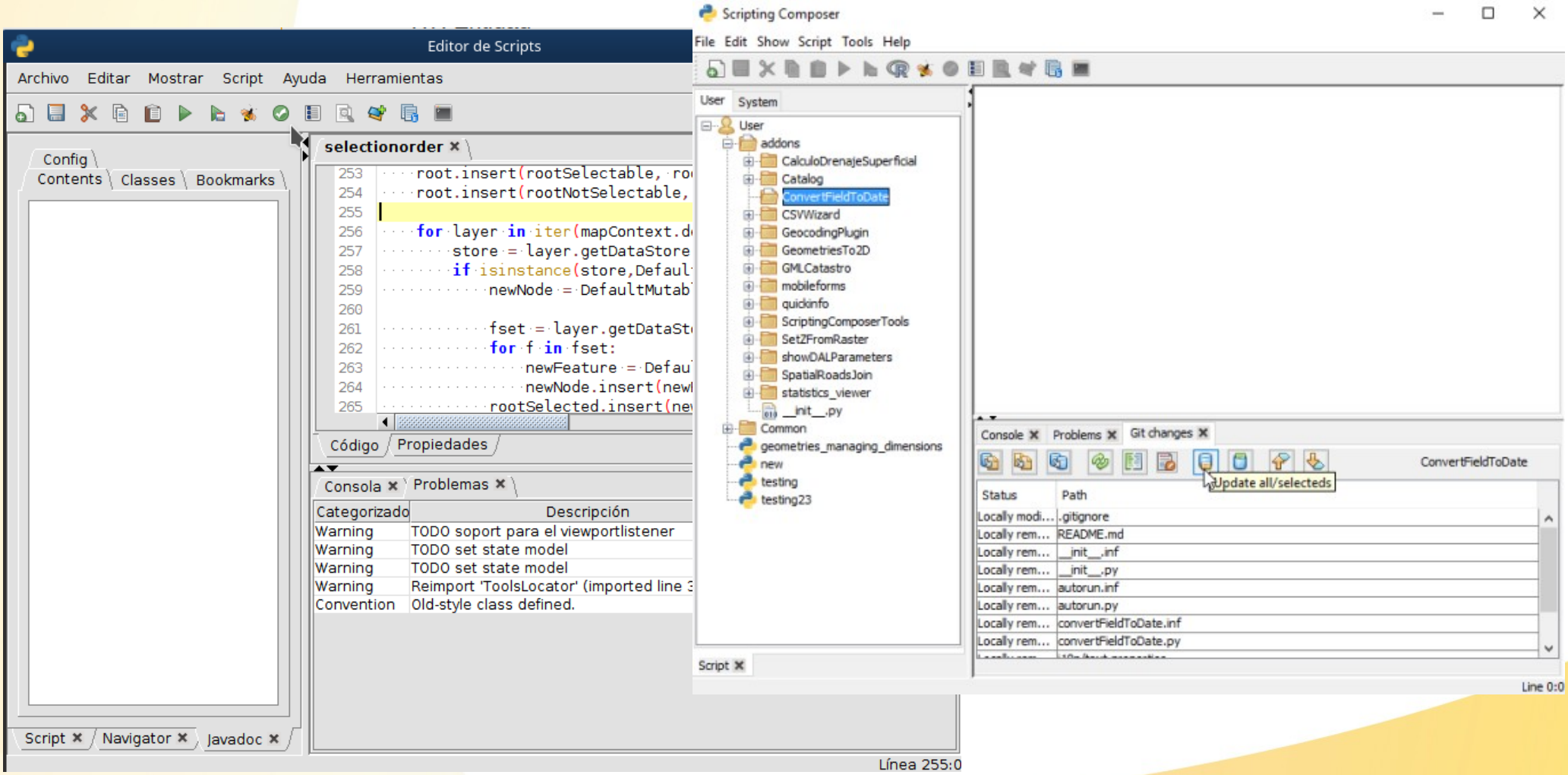
Development

ScriptingComposerTools



Development

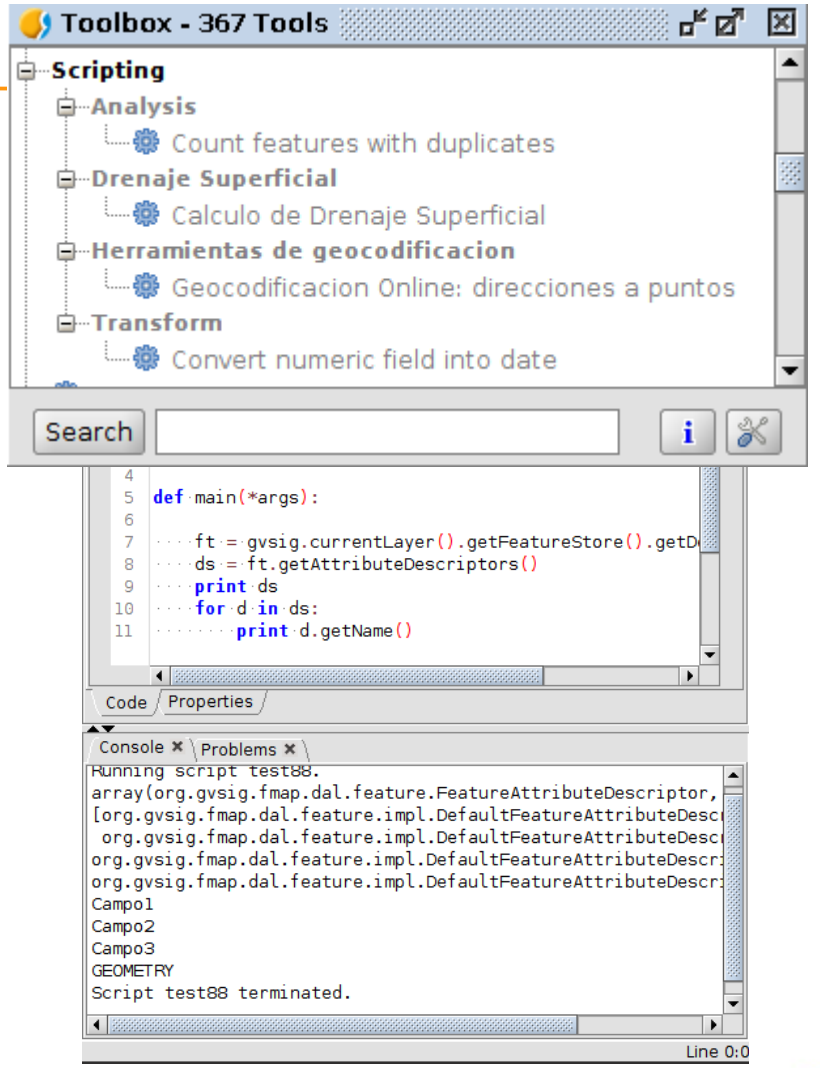
ScriptingComposerTools



Development

Test and run

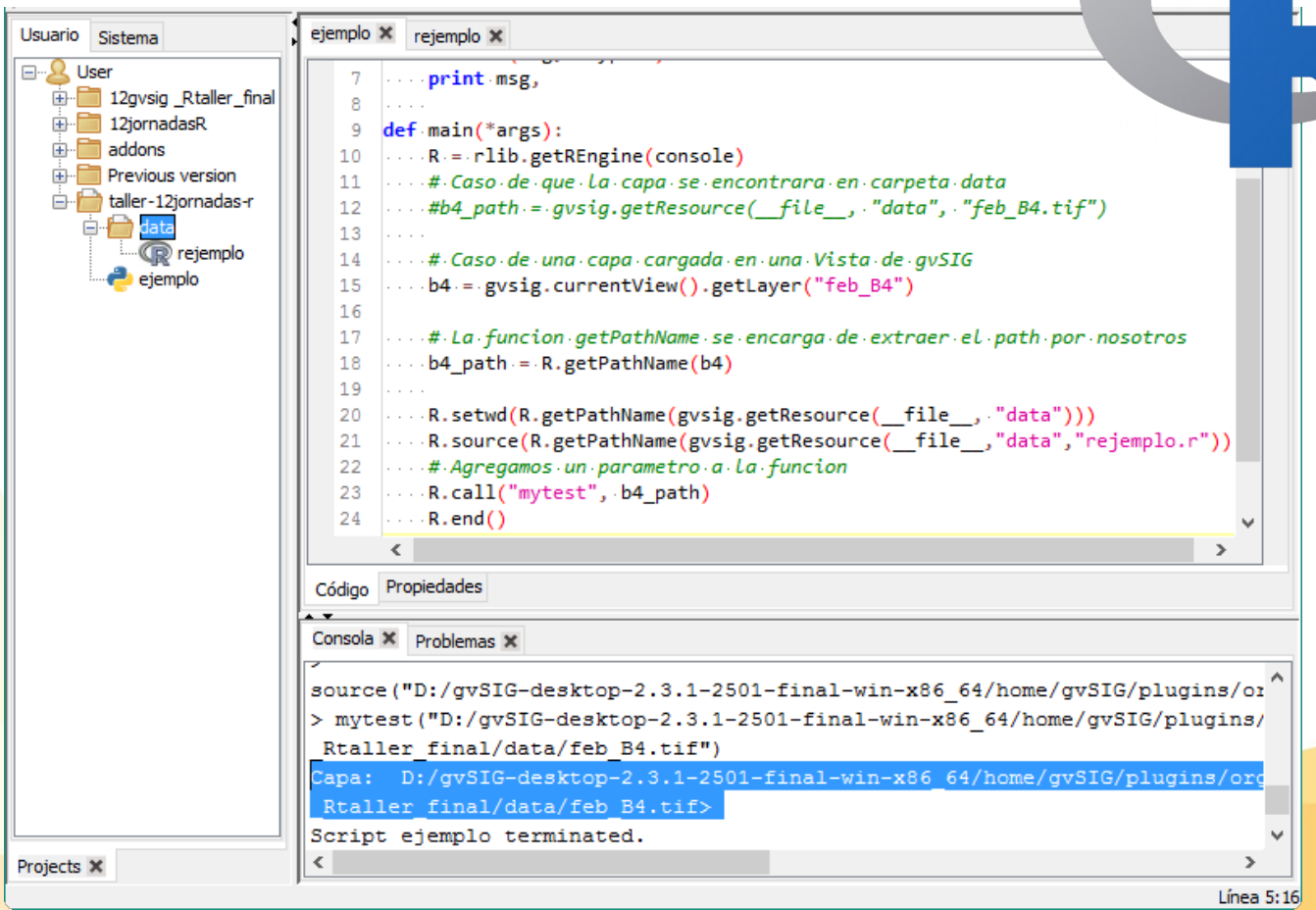
- Workflow (depending of the plugin)
  - Create
  - Execute
  - Test and check result
  - Modify
  - Execute
  - Test and check result
  - ...
- Work directly with gvSIG
  - Access to the active Layer, selected features, ...



Example

Integration with R

- Install R plugin from Addons Manager
- Execute R code from a script



Example

Geoprocess

- Add a geoprocess to the Toolbox

process

```
class Desplazamiento(ToolboxProcess):
    def defineCharacteristics(self):
        self.setName("Desplazamiento en X e Y")
        self.setGroup("Vectorial")
        params = self.getParameters()
        params.addInputVectorLayer("LAYER", "Capa de entrada", SHAPE_TYPE_POINT, True)
        params.addNumericalValue("X", "X_traslation",0, NUMERICAL_VALUE_DOUBLE)
        params.addNumericalValue("Y", "Y_traslation",0, NUMERICAL_VALUE_DOUBLE)
        self.addOutputVectorLayer("RESULT_POINT", "XYShift_point", SHAPE_TYPE_POINT)

    def processAlgorithm(self):
        features = None
        try:
            params = self.getParameters()
            layer = params.getParameterValueAsVectorLayer("LAYER")
            x = params.getParameterValueAsDouble("X")
            y = params.getParameterValueAsDouble("Y")
            input_store = layer.getFeatureStore()
            features = input_store.getFeatureSet()
            output_store = self.buildOutPutStore(
                features.getDefaultFeatureType(),
                SHAPE_TYPE_POINT,
                "XYShift_points",
                "RESULT_POINT"
                )
            self.setRangeOfValues(0, features.getSize())
            n = 0
            for feature in features.iterator():
                if self.isCanceled():
                    print "Proceso cancelado"
                    break
                newfeature = self.createNewFeature(output_store, feature)
                geom = newfeature.getDefaultGeometry()
                geom.move(x,y)
                output_store.insert(newfeature)
                n+=1
                self.setCurValue(n)
            output_store.finishEditing()
        finally:
            DisposeUtils.disposeQuietly(features)
            print "Proceso terminado %s" % self.getCommandLineName()
            return True
```

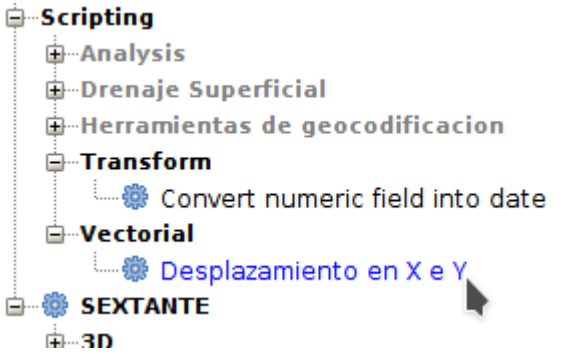
autorun

```
def selfRegister(*args):
    i18nManager = ToolsLocator.getI18nManager()
    i18nManager.addResourceFamily("text",File(gvsig.getResource(__file__,"i18n")))

    process = Desplazamiento()
    process.selfregister("INEGI")
    process.updateToolbox()
```

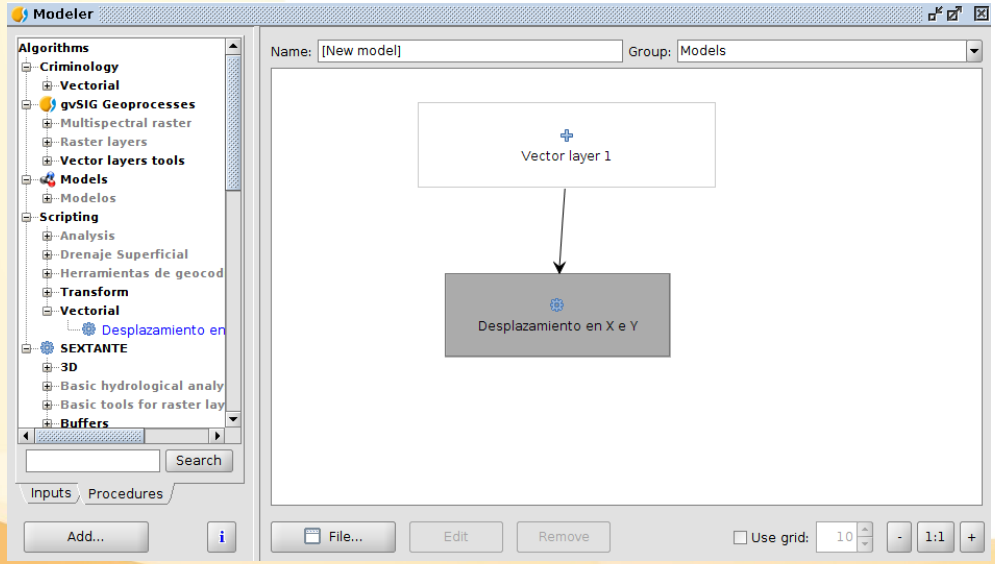
Example

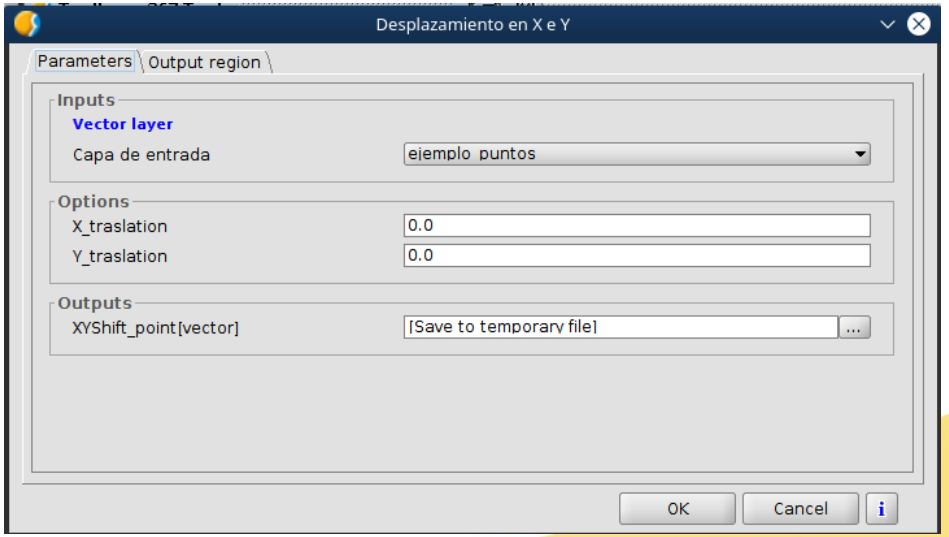
Geoprocess

- Use it from the model builder or gvpy




```
gvpy.runalg("randomvector",10, 0)
gvpy.runalg("randomvector",10, gvpy.TYPE_POLYGON)
```

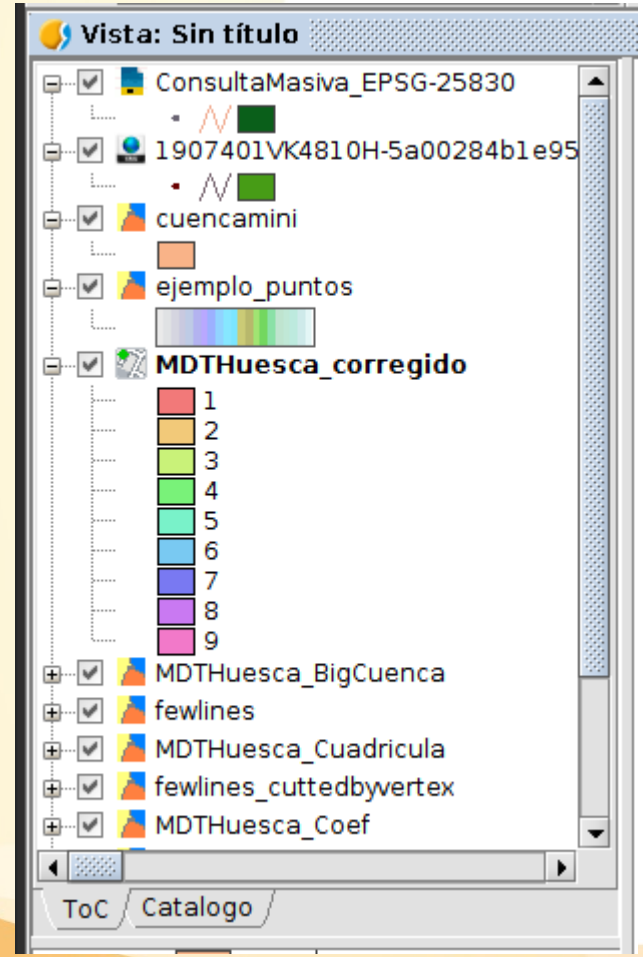


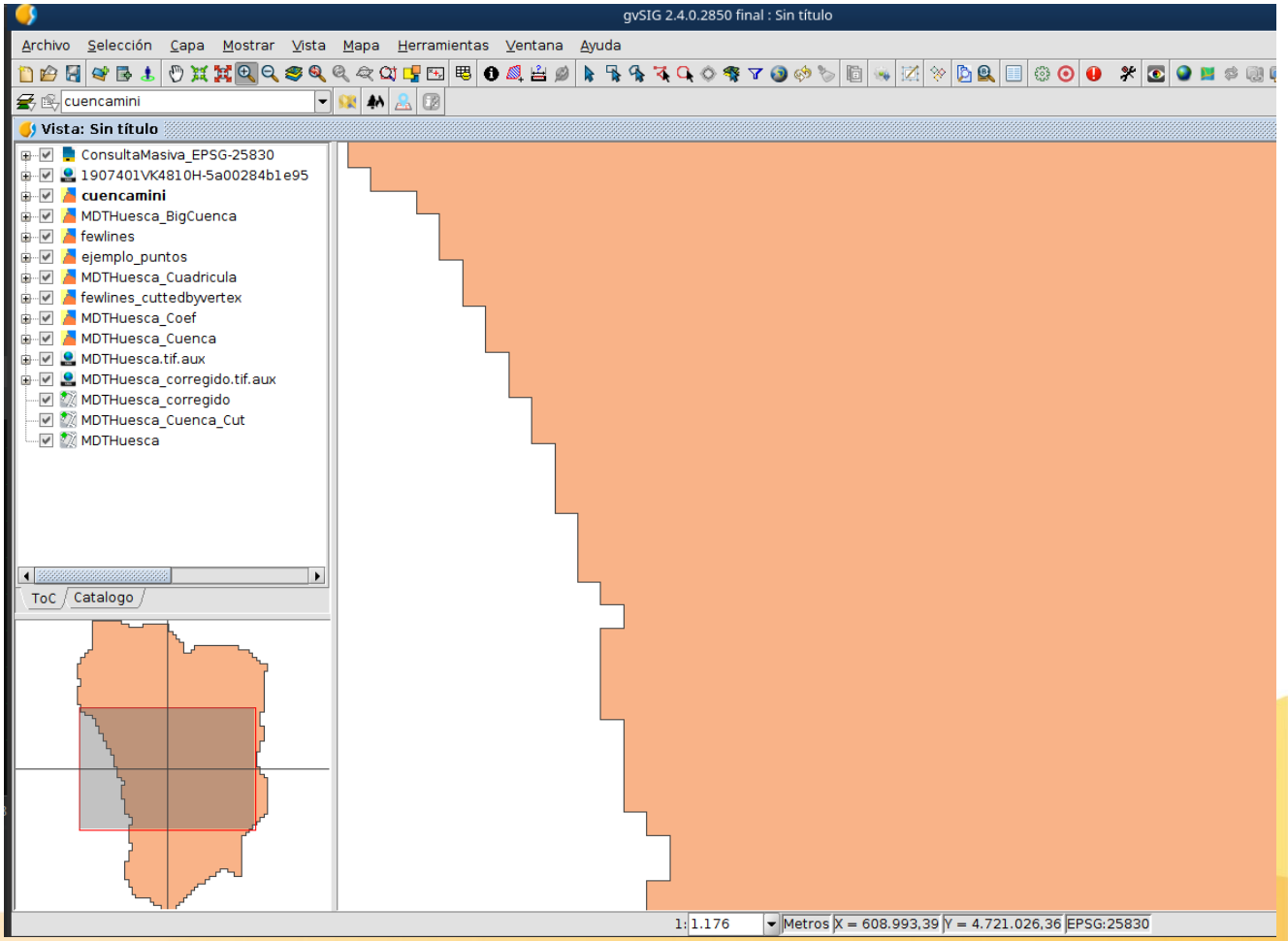


Example

Table of Contents (ToC)

- Basic from Java

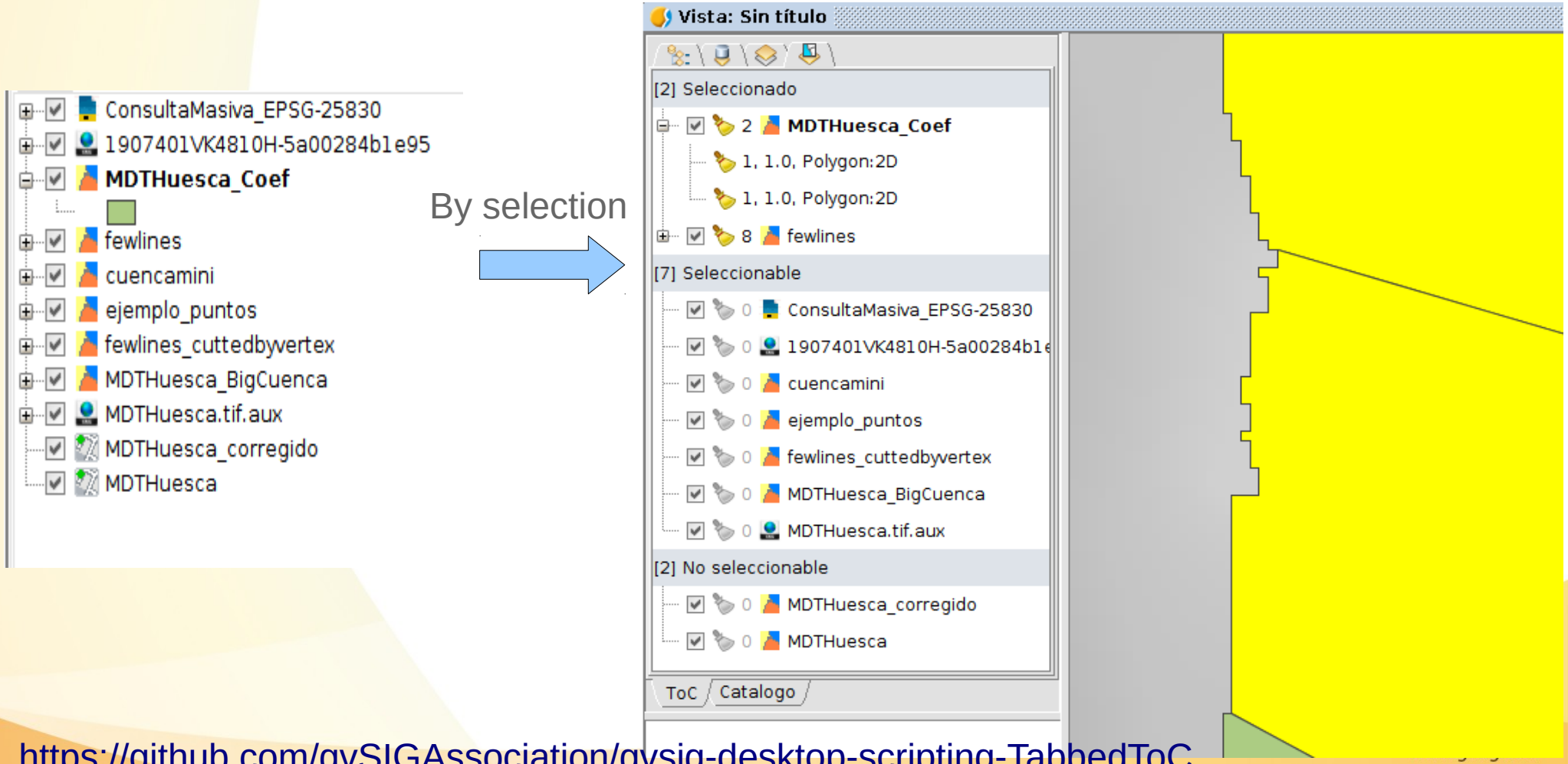




Example

Table of Contents (ToC) → TabbedToC

- Advanced plugin created from Scripting



https://github.com/gvSIGAssociation/gvsig-desktop-scripting-TabbedToC

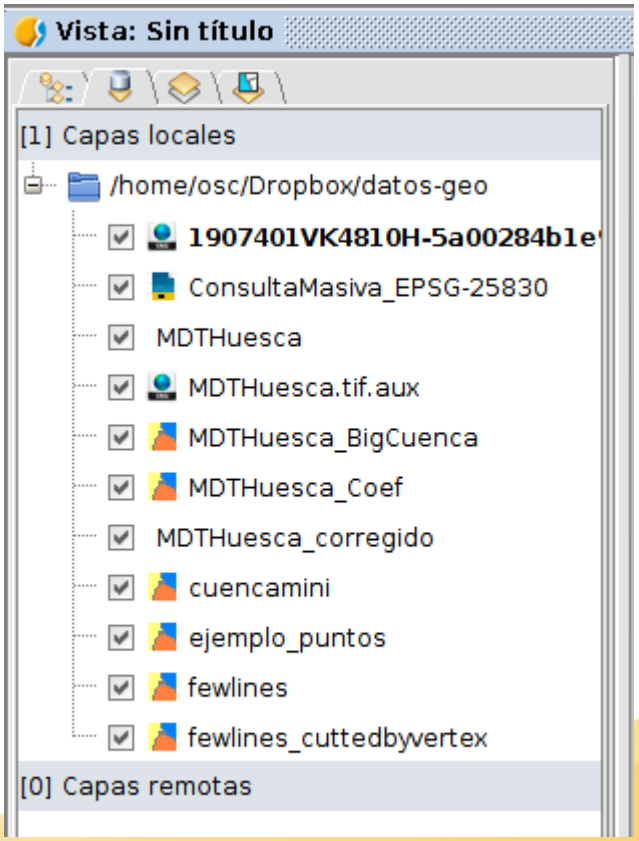
Example

Table of Contents (ToC) → TabbedToC

- Advanced plugin created from Scripting

By visibility

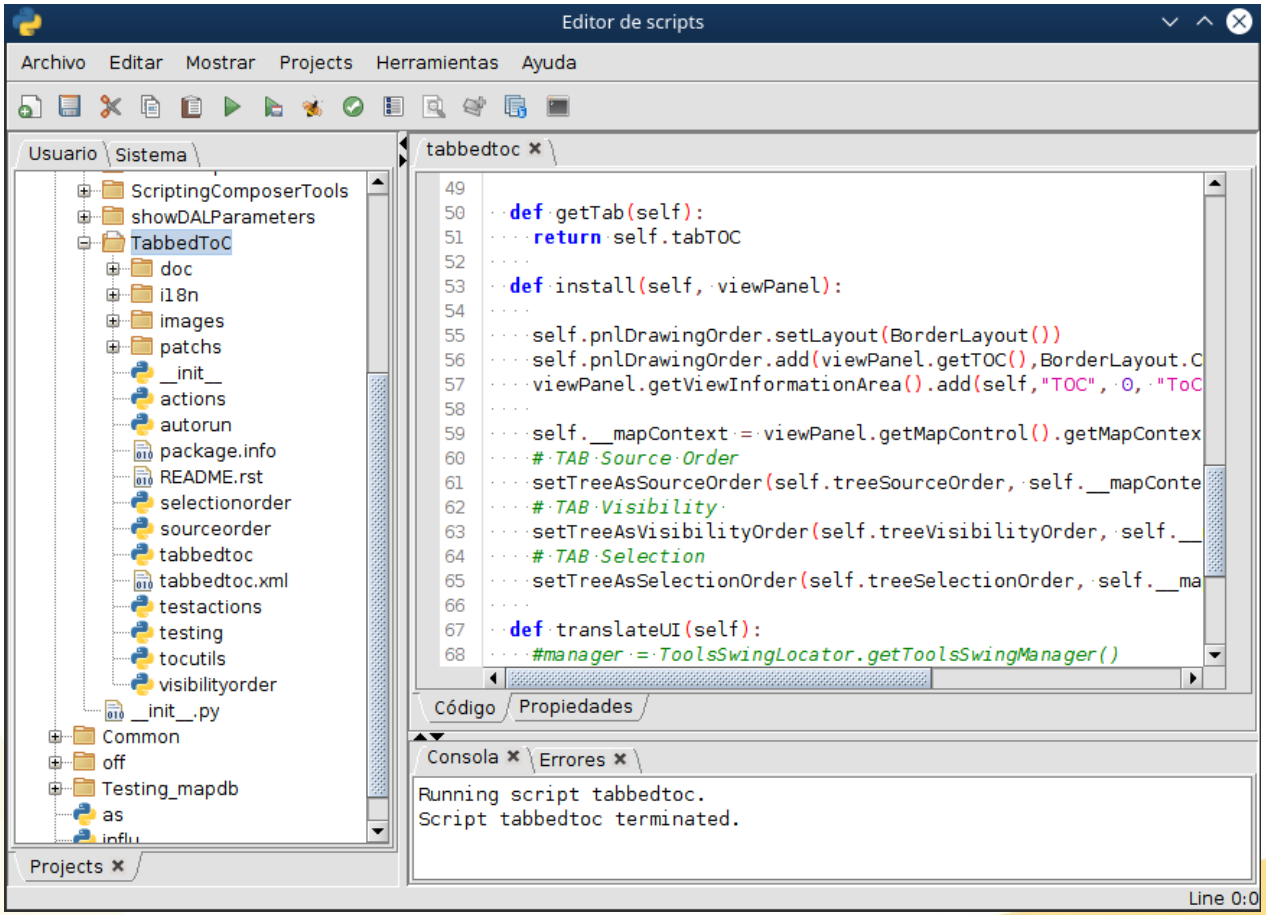
By source



Example

Table of Contents (ToC) → TabbedToC

- Advanced plugin created from Scripting

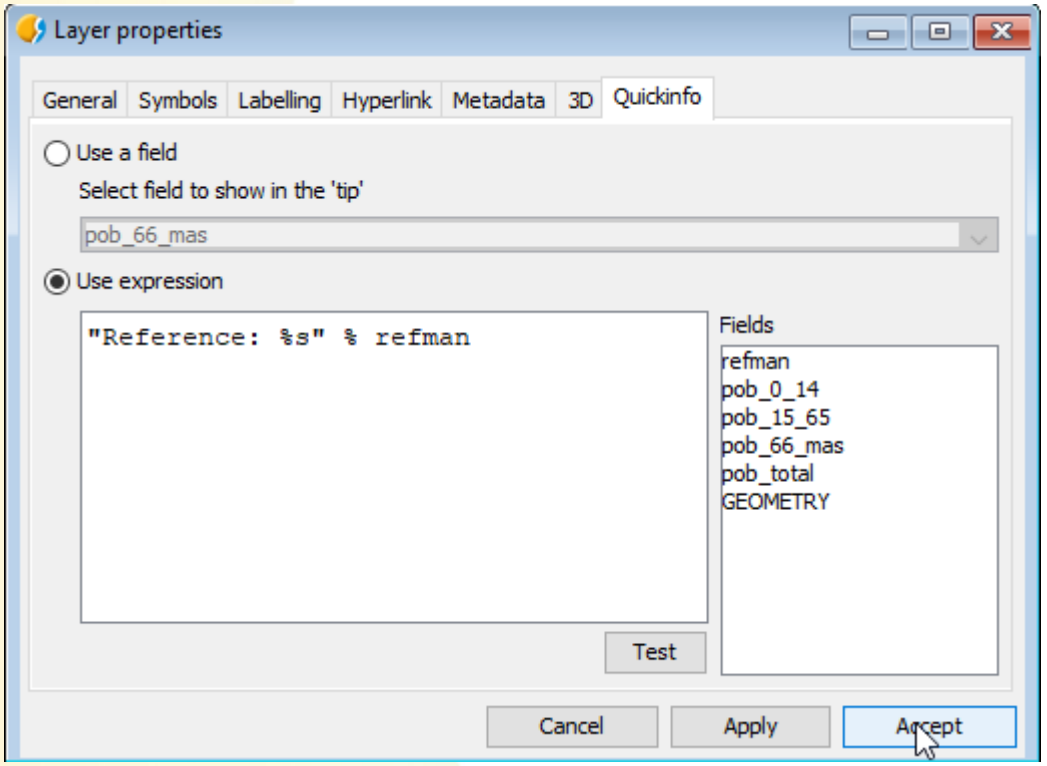


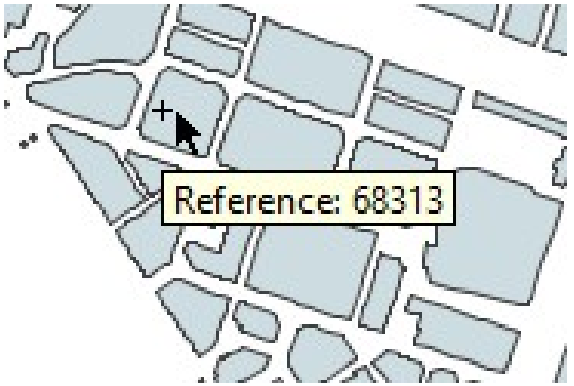
https://github.com/gvSIGAssociation/gvsig-desktop-scripting-TabbedToC

Example

Quick Info

- New type of data visualization



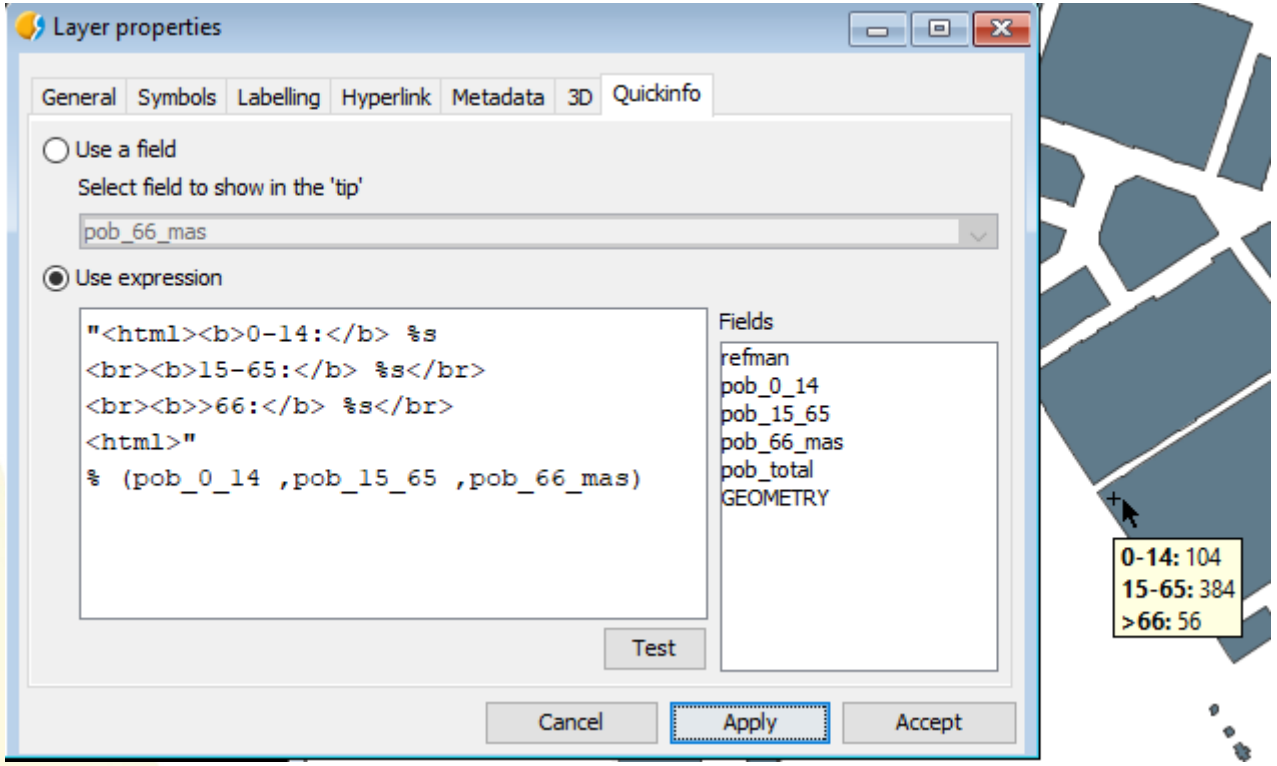


https://blog.gvsig.org/2017/12/15/towards-gvsig-2-4-quick-info-tool/
https://github.com/gvSIGAssociation/gvsig-desktop-scripting-quickinfo

Example

Quick Info

- New type of data visualization

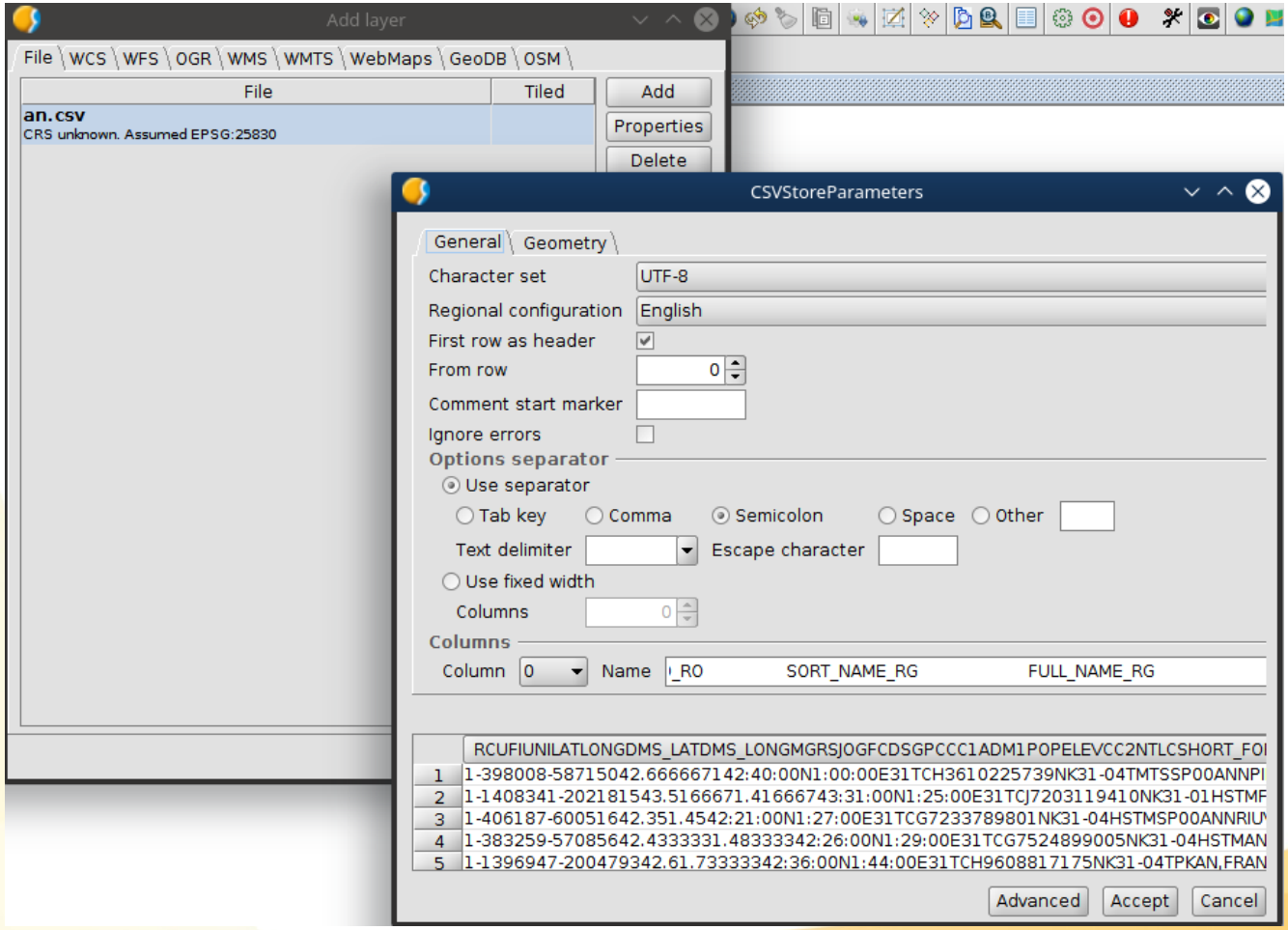


https://blog.gvsig.org/2017/12/15/towards-gvsig-2-4-quick-info-tool/
https://github.com/gvSIGAssociation/gvsig-desktop-scripting-quickinfo

Example

CSV Wizard

- Advanced tool for import CSV

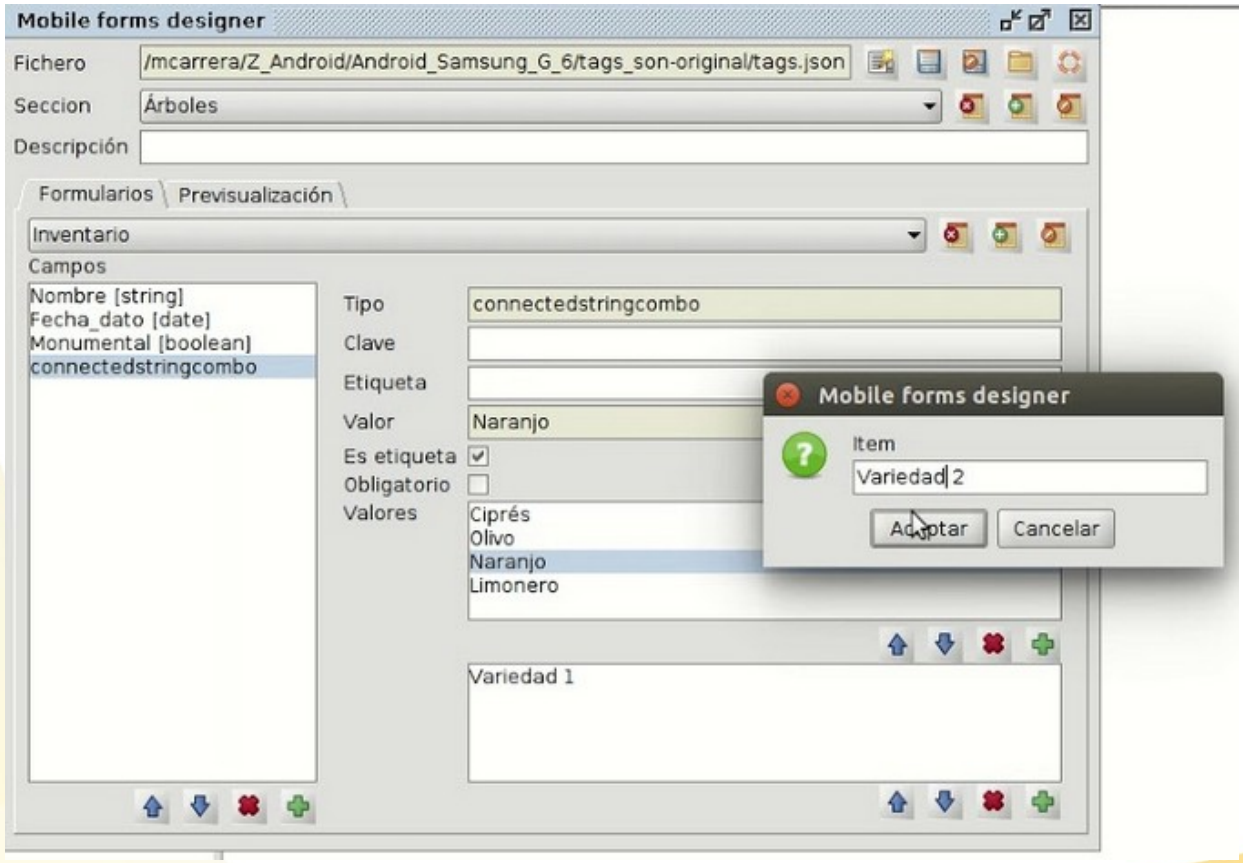


https://github.com/gvSIGAssociation/gvsig-desktop-scripting-CSVWizard

Example

Mobile forms

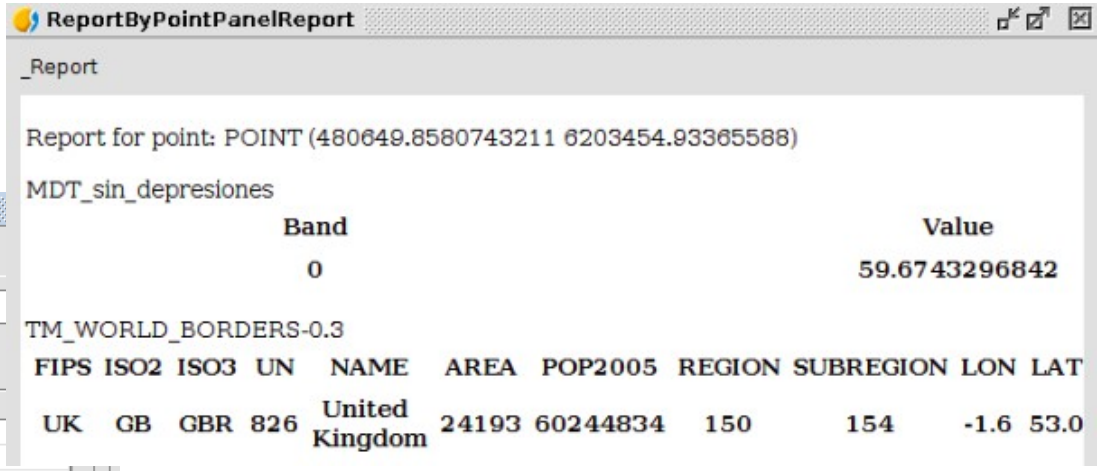
- Create mobile forms

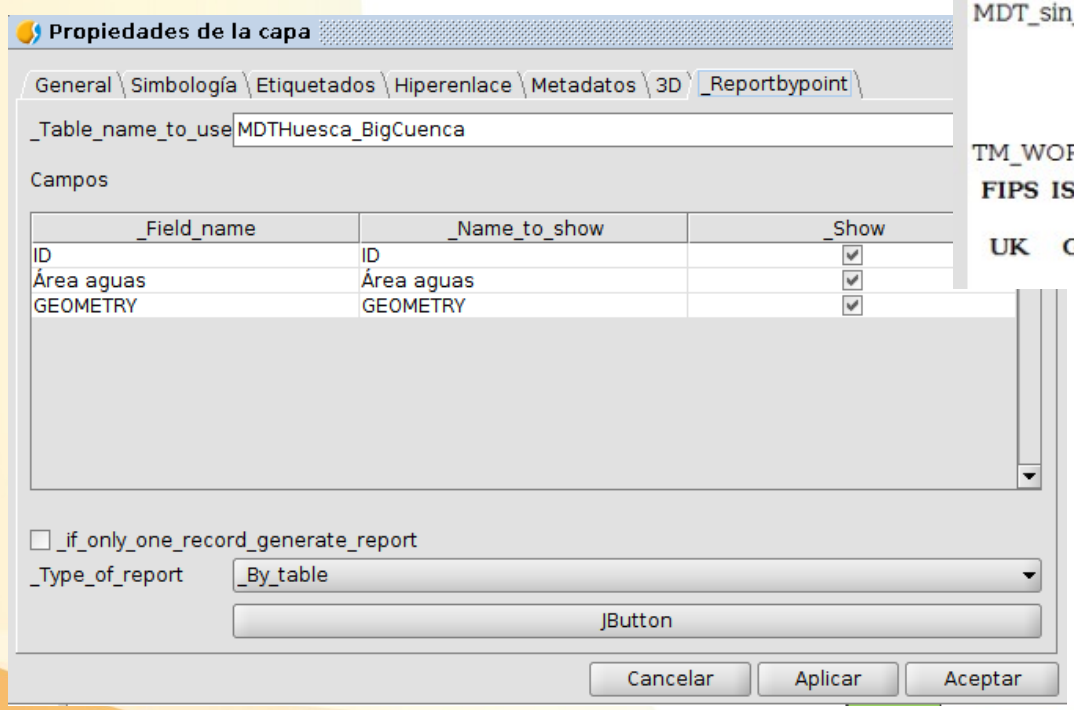


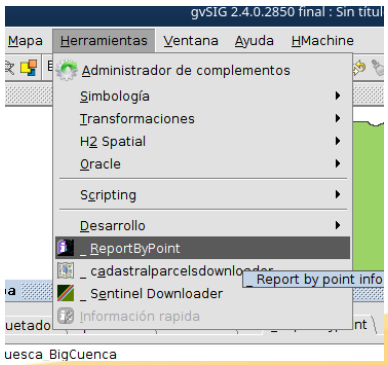
Example

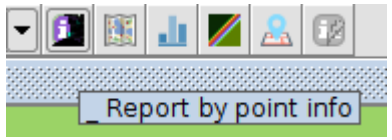
Report by point

- Generate a point with the info in all the layers in a view





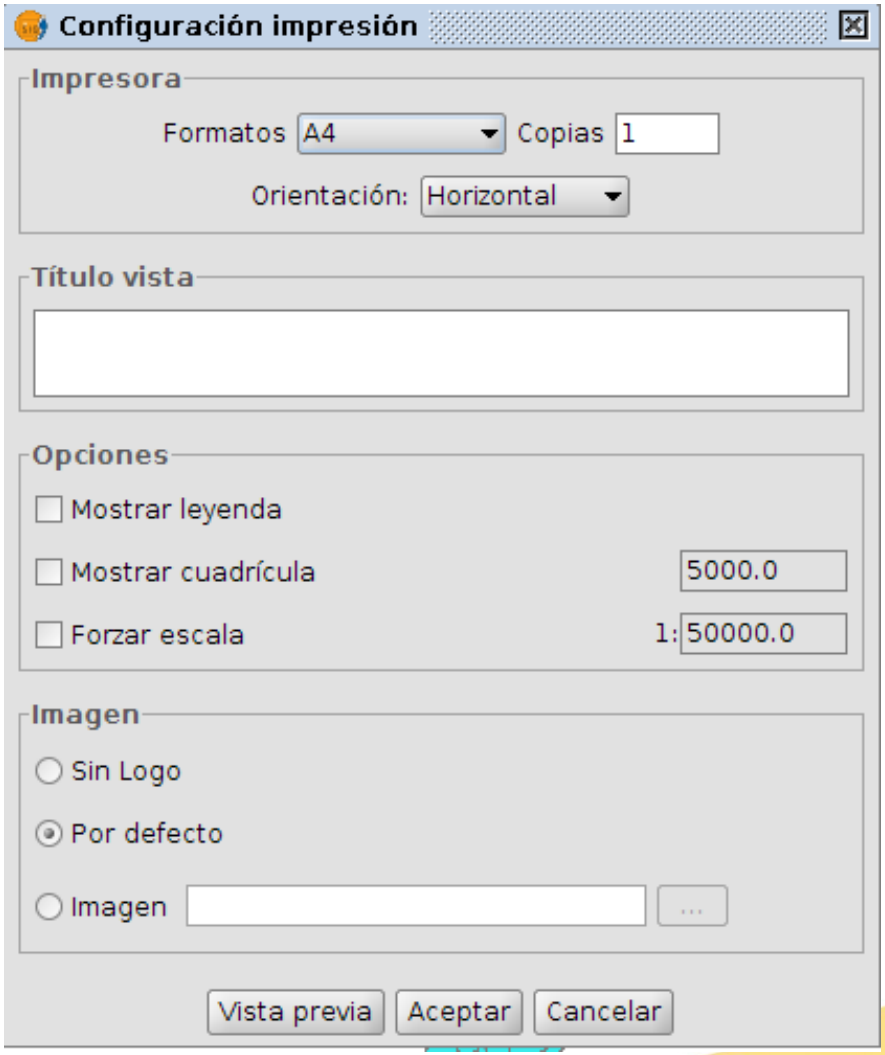




Example

Quick Export Map

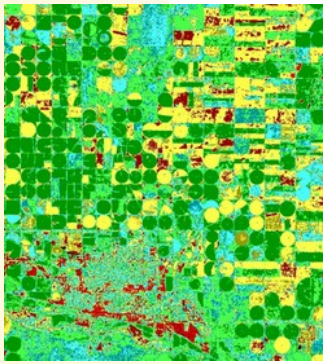
- Create quick map

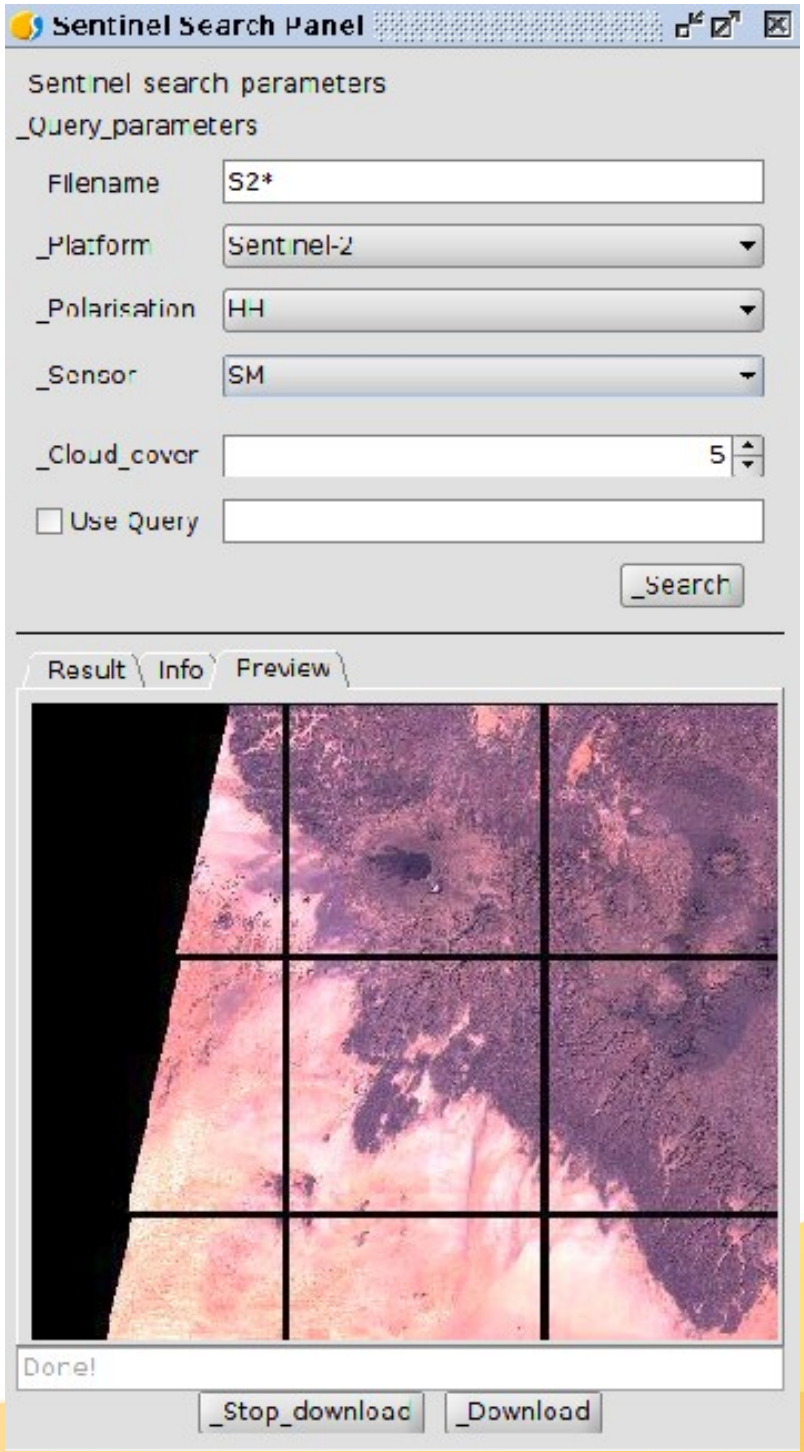


Example

Sentinel Downloader

- Search and download Sentinel products

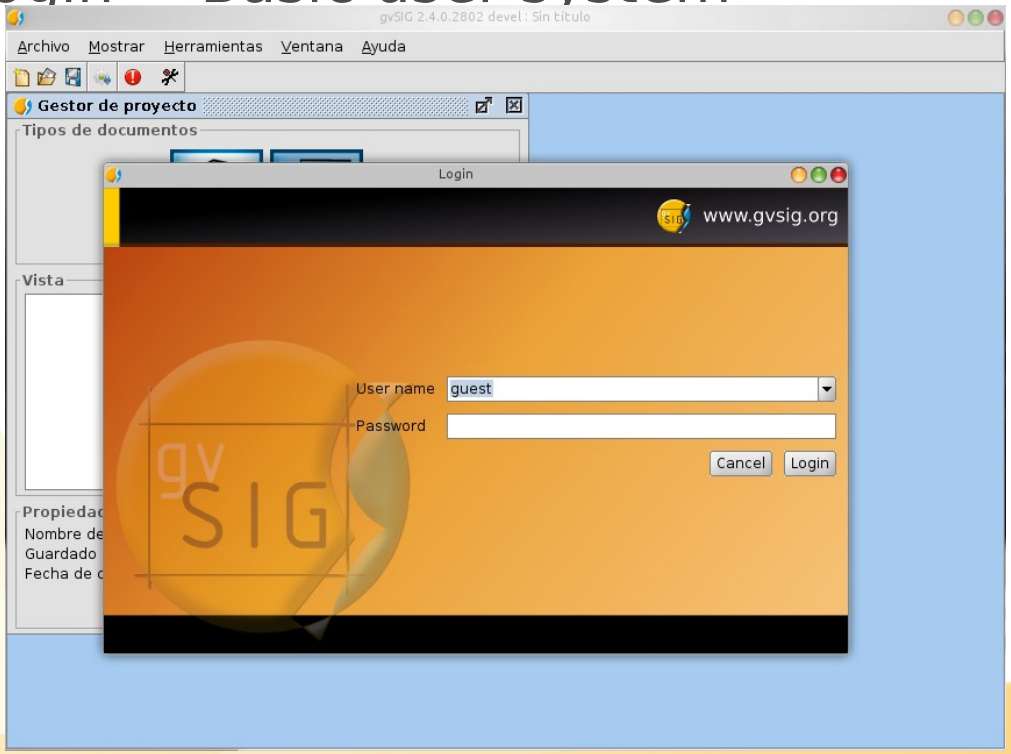




Example

Other type of plugins

- GMLCatastro → create new export format file
- CadastralDownloader → download municipalities
- Geocoding Online → using python libraries
- SimpleLogin → Basic user system



+Info

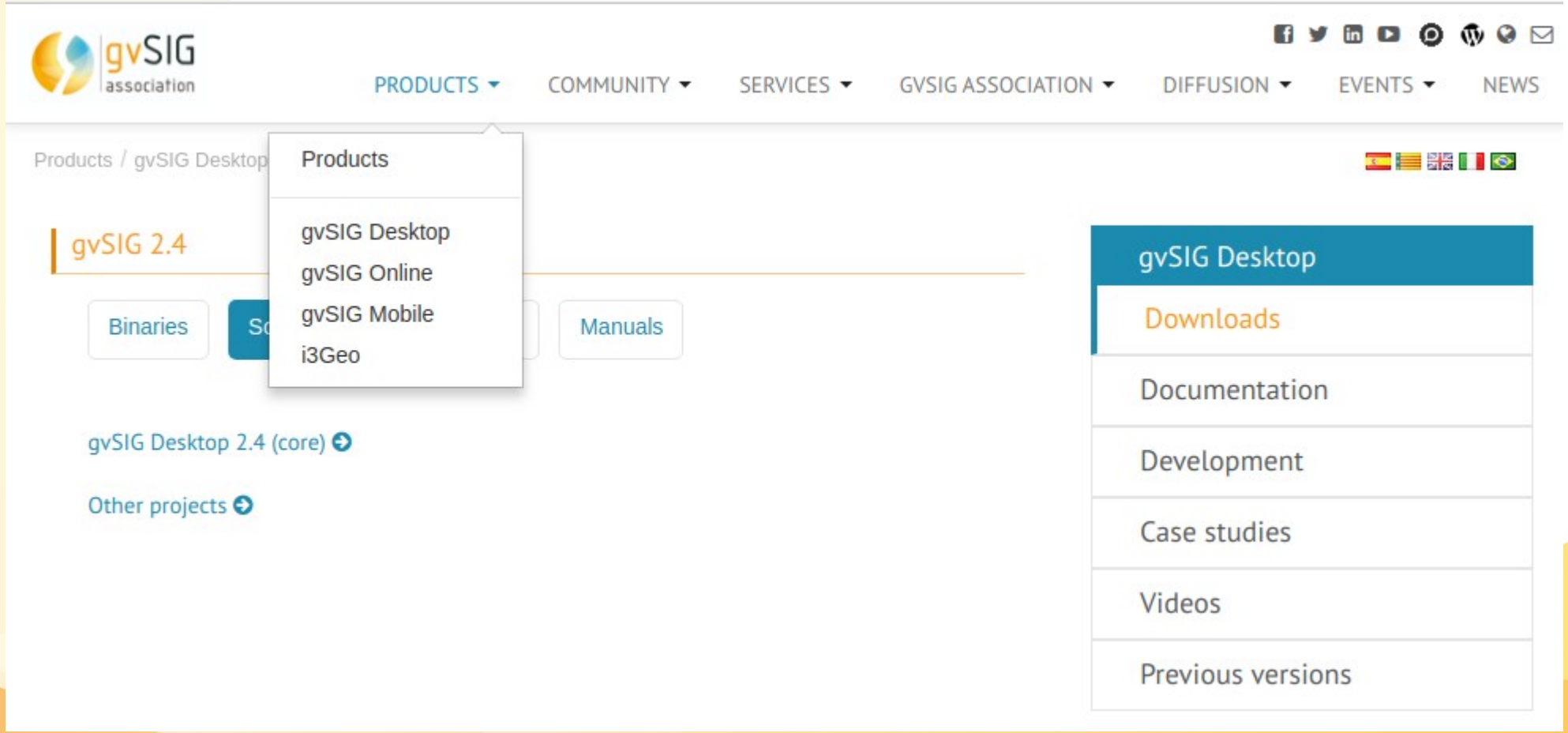
Official docs

- gvsig.com
- blog.gvsig.com
  - Workshops
- Mailing list
- Scripting guide 2.4
- Youtube
  - Gvsig + scripting

+Info

Code

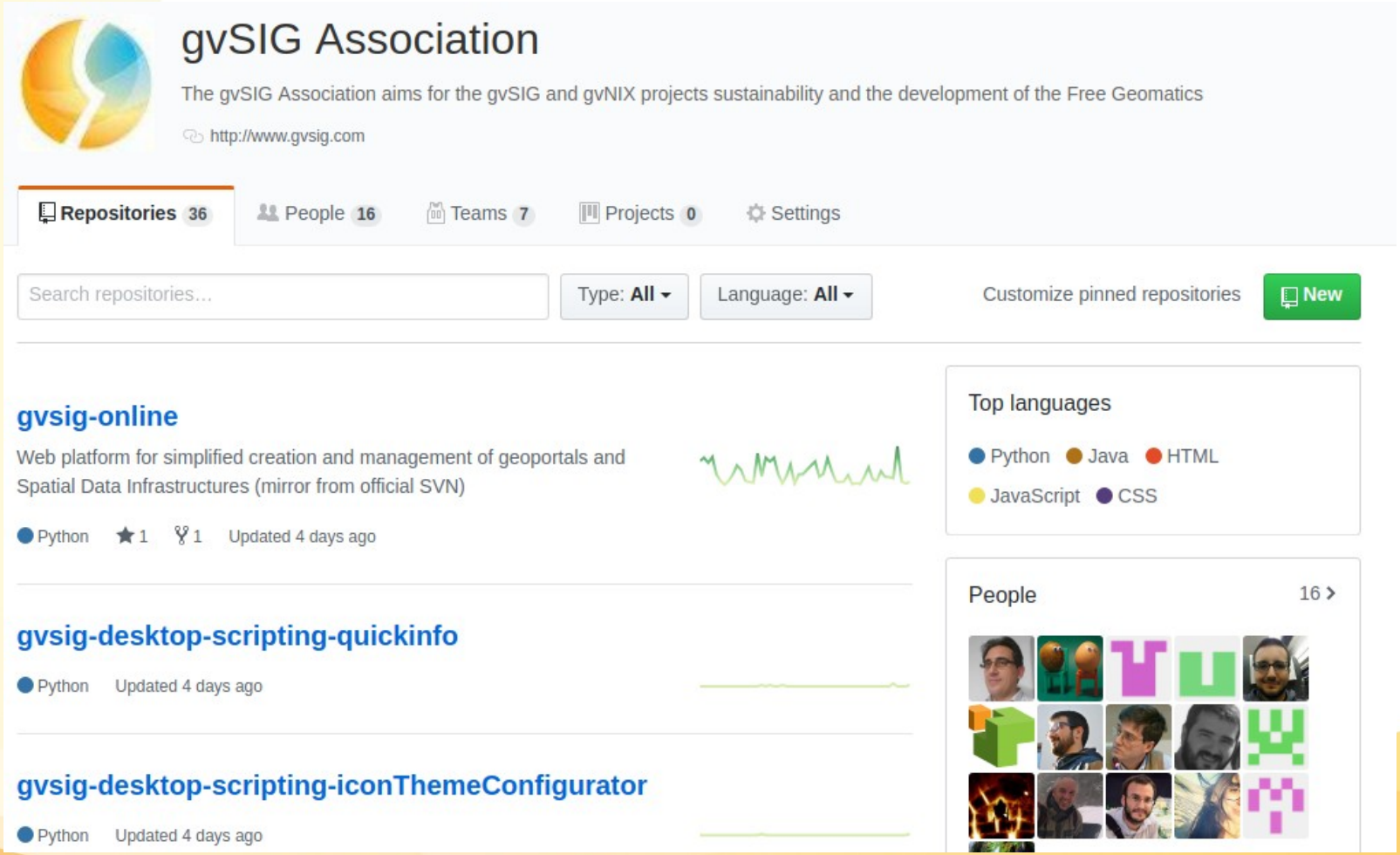
- Source code



+Info

Code

- Github https://github.com/gvSIGAssociation



¡Thanks!

Óscar Martínez

@masquesig / masquesig.com

omartinez@gvsig.com