```
# encoding: utf-8

import gvsig
from gvsig.libs.formpanel import FormPanel, getResource
from gvsig import openStore

from java.awt import BorderLayout

from org.gvsig.tools.evaluator import AbstractEvaluator

from org.gvsig.fmap.dal import DALLocator
from javax.swing.table import DefaultTableModel

def getBlockByPoint(blocksStore, point):
  attrGeomName = blocksStore.getDefaultFeatureType()
    .getDefaultGeometryAttributeName()
  features = blocksStore.features(IntersectEvaluator(attrGeomName,point))
  if features.isEmpty():
    return None
  firstFeature = iter(features).next()
  block = Block(
    firstFeature.getDefaultGeometry(),
    firstFeature.get("HOJA"),
    firstFeature.get("FECHAALTA")
  )
  return block

class IntersectEvaluator(AbstractEvaluator):
  def __init__(self, attrGeomName, point):
    self.__attrGeomName = attrGeomName
    self.__point = point
    self.__sqlwhere = " ST_intersects(GeomFromText('%s','%s'),%s) " % (
      self.__point.convertToWKT(),
      "EPSG:29030",
      self.__attrGeomName
    )

  def getName(self):
    return "intersectByPoint"

  def evaluate(self, data):
    geom = data.getDataValue(self.__attrGeomName)
    if geom == None:
      return False
    return geom.intersects(self.__point)

  def getCQL(self):
    return self.__sqlwhere


class Block(object):
  def __init__(self, geom, hoja, fechaAlta):
    self.__geom = geom
    self.__hoja = hoja
    self.__fechaAlta = fechaAlta

  def __repr__(self):
    return "Block(%s, %r, %r)" % (
        self.__geom.convertToWKT(),
        self.__hoja,
        self.__fechaAlta
    )
  __str__ = __repr__

  def getHoja(self):
    return self.__hoja

  def getFechaAlta(self):
    return self.__fechaAlta
  def getGeom(self):
    return self.__geom

  def getProperties(self):
    properiesStore = openStore(
      "Shape",
      shpFile=getResource(__file__,"data","properties.shp"),
      CRS="EPSG:23030"
    )
    attrGeomName = properiesStore.getDefaultFeatureType()
      .getDefaultGeometryAttributeName()
    query = properiesStore.createFeatureQuery()
    query.setFilter(IntersectEvaluator(attrGeomName, self.__geom))
    return properiesStore.getFeatures(query,100)

class PropertiesTableModel(DefaultTableModel):
  def __init__(self,properties):
    self.__properties = properties
    self.addColumn("Hoja")
    self.addColumn("Parcela")
    self.addColumn("Num.patio")

  def getRowCount(self):
    return self.__properties.size()

  def getValueAt(self, row, column):
    prop = self.__properties.get(row)
    if column == 0:
      return prop.get("HOJA")
    if column == 1:
      return prop.get("PARCELA")
    return "%s %s" % (
      prop.get("NUMERO"),
      prop.get("NUMERODUP")
    )

class BlockPanel(FormPanel):
  def __init__(self):
    FormPanel.__init__(self,getResource(__file__,"blockpanel.xml"))
    self.setPreferredSize(300,300)

  def setBlock(self, block):
    self.txtHoja.setText(str(block.getHoja()))
    self.txtFechaDeAlta.setText(str(block.getFechaAlta()))
    self.txtArea.setText(str(block.getGeom().area()))
    self.txtPosicion.setText(block.getGeom().centroid().convertToWKT())
    self.tableProperties.setModel(
      PropertiesTableModel(block.getProperties())
    )

def main(*args):
  from gvsig.geom import createPoint2D

  blocksStore = openStore(
    "Shape",
    shpFile=getResource(__file__,"data","blocks.shp"),
    CRS="EPSG:23030"
  )
  block = getBlockByPoint(blocksStore, createPoint2D(725822,4374000))
  #print block
  if block != None:
    panel = BlockPanel()
    panel.setBlock(block)
    panel.showTool("Informacion de la manzana")
```