



POZnan*

geopoz*

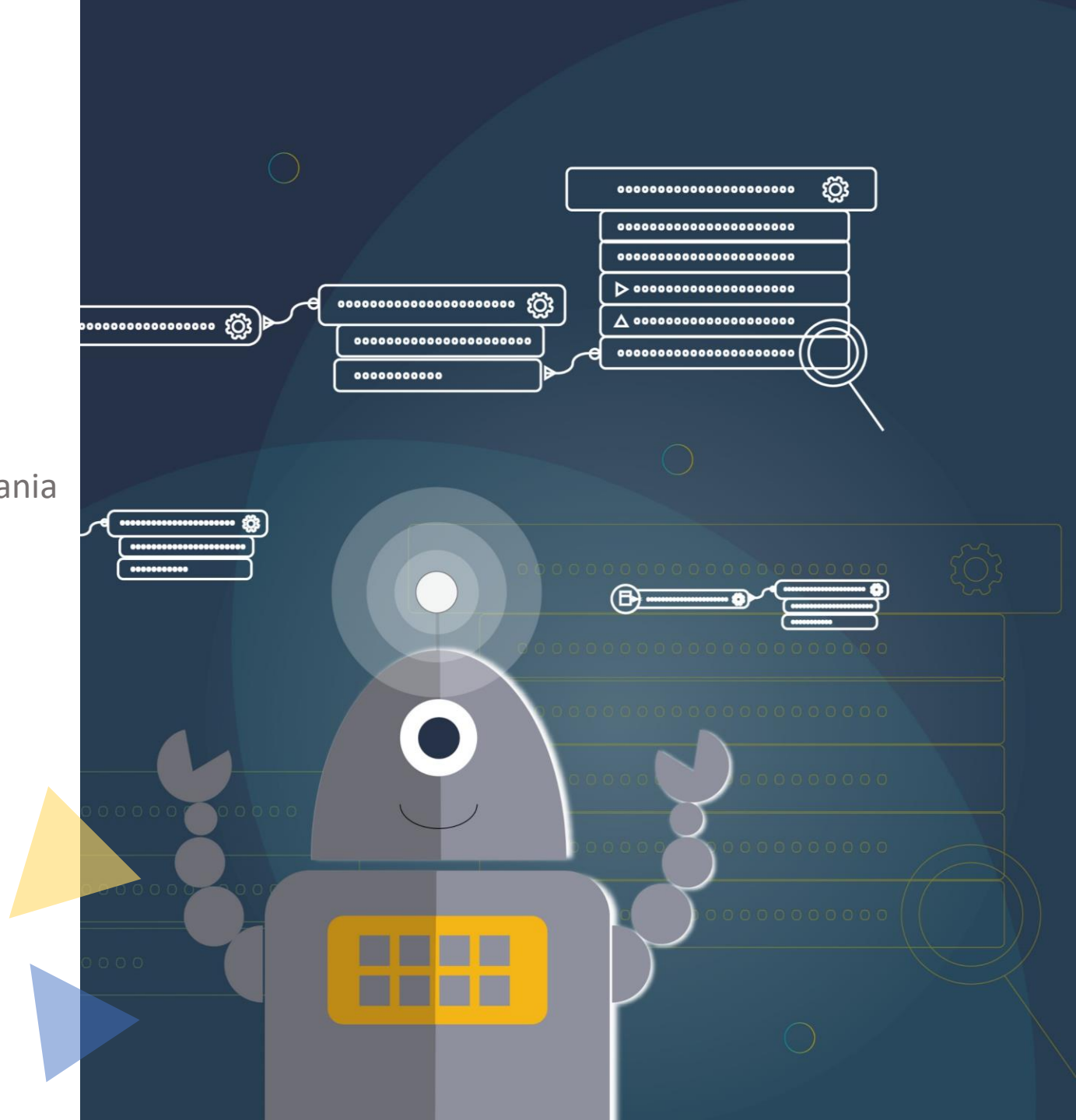


Od 2D do 3D - Rozwój SIP Miasta Poznania z wykorzystaniem FME

Krzysztof Drygalski
Marcin Lis

Plan prezentacji

- ▶ Wprowadzenie
- ▶ Aktualizacja danych LoD 1
- ▶ Generowanie zestawu danych do aktualizacji LoD 2
- ▶ Wizualizacja 3D Miejscowych Planów Zagospodarowania
- ▶ Profil terenu na żądanie w przeglądarce 2D



Wprowadzenie



Model 3D w liczbach

Realizacja projektu: 01.10.2017 – 30.06.2018

Budynki:

▶ LoD 1 – 98413



▶ LoD 2 – 92524



▶ LoD 3 – 16



▶ Budynki w trakcie budowy - 739

▶ Budynki projektowane – 1768

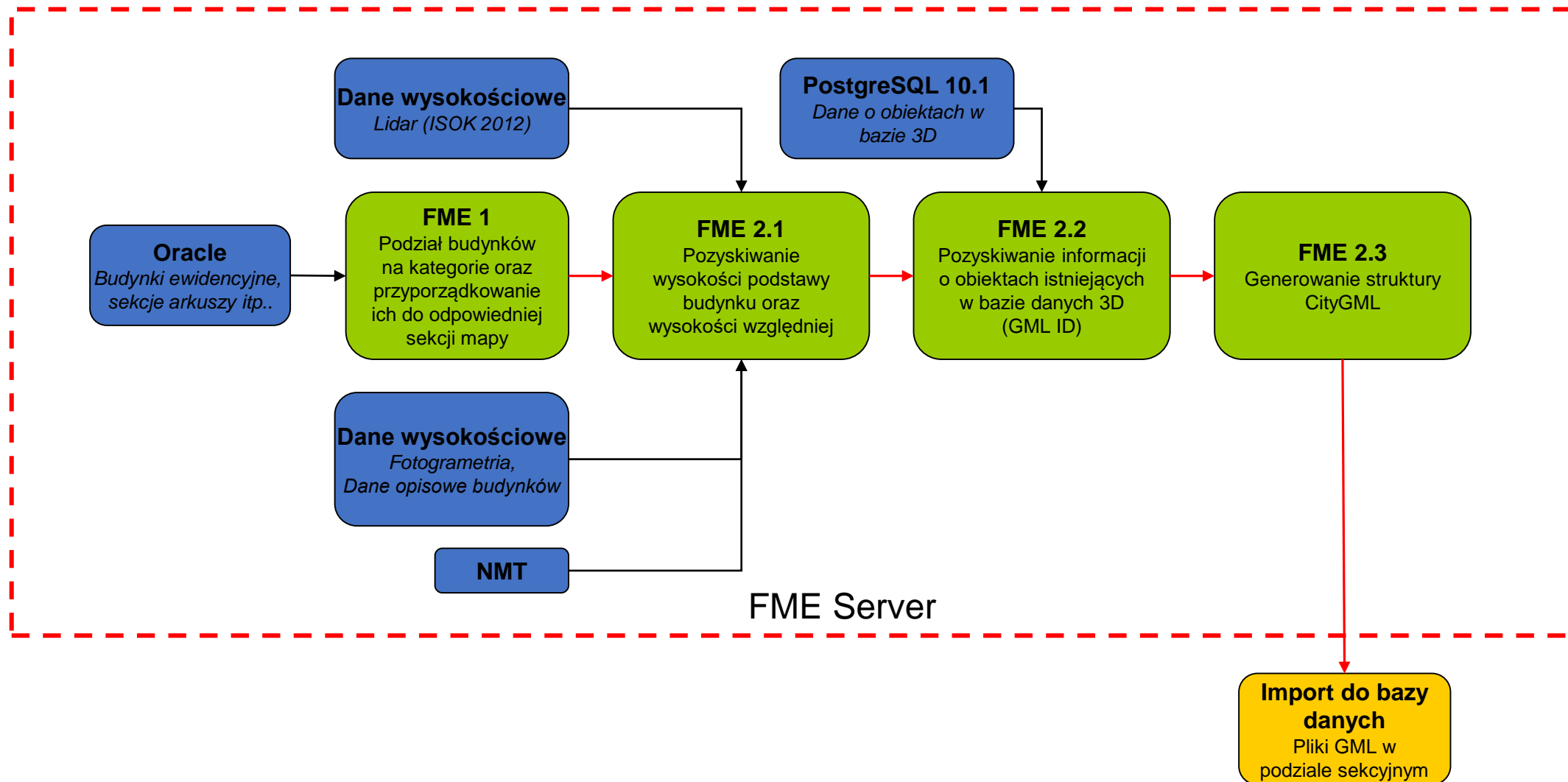
Ilość warstw aktualizowanych w cyklach tygodniowych - 13



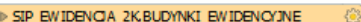
Aktualizacja LoD 1



Schemat aktualizacji bazy danych LoD 1



11/11/2019



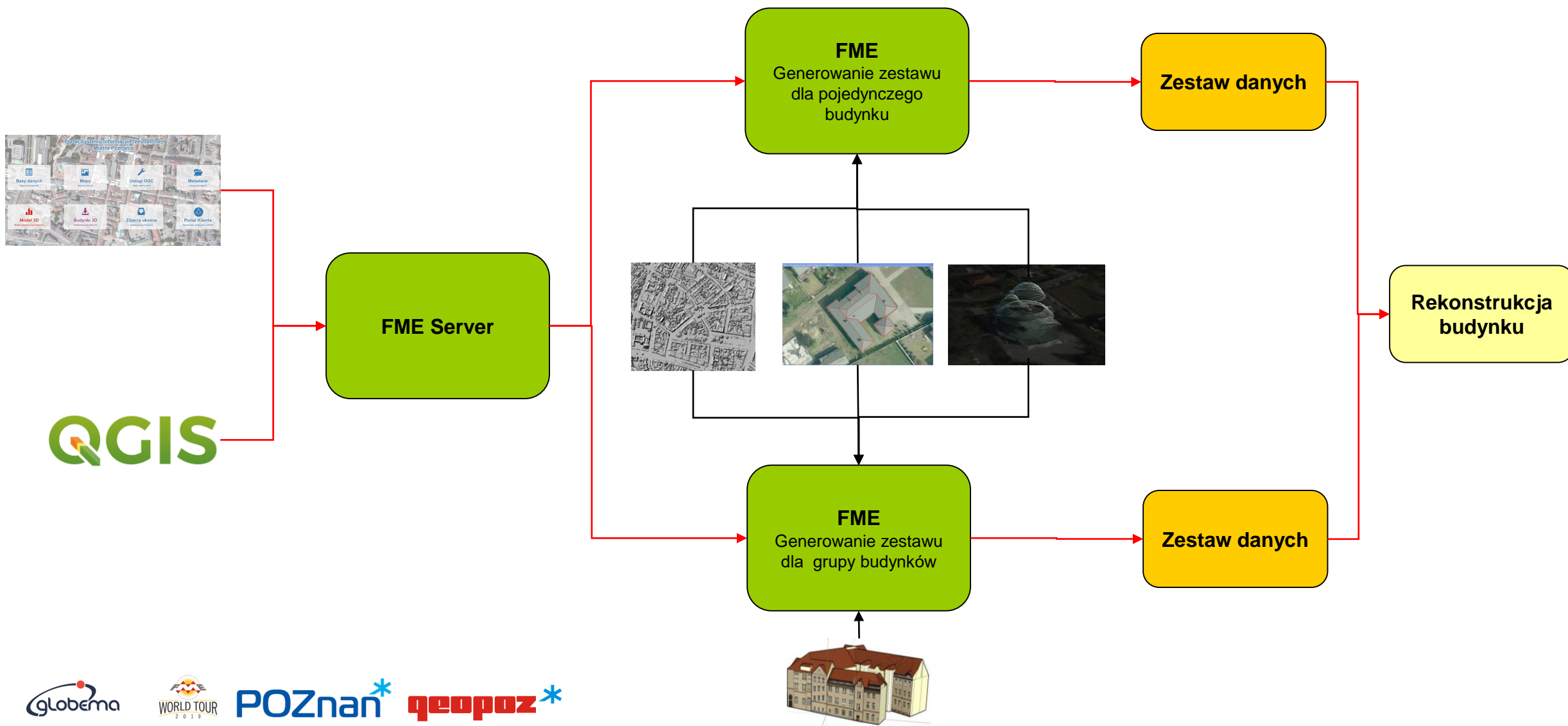
Wynik procesu



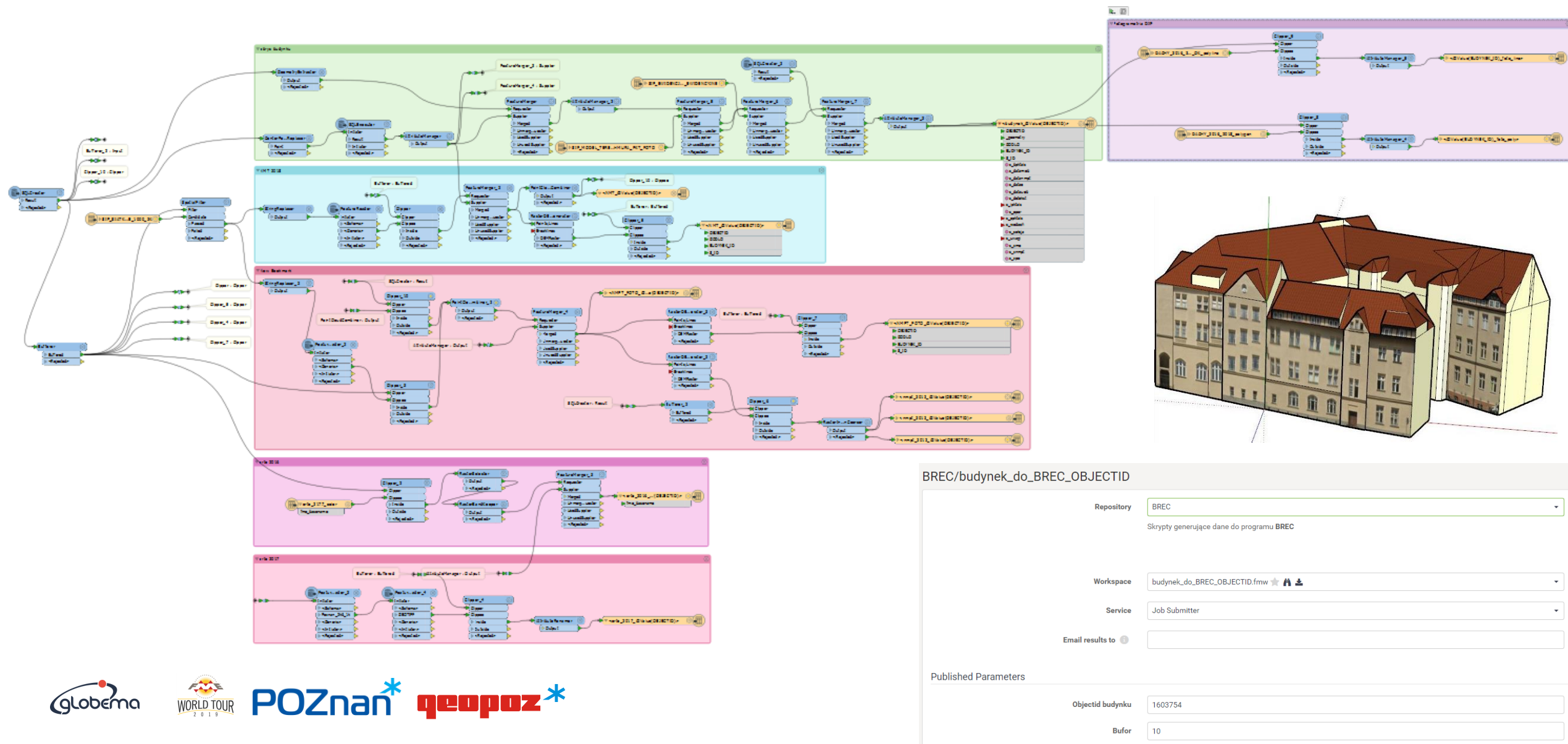
Generowanie zestawu danych do aktualizacji LoD 2



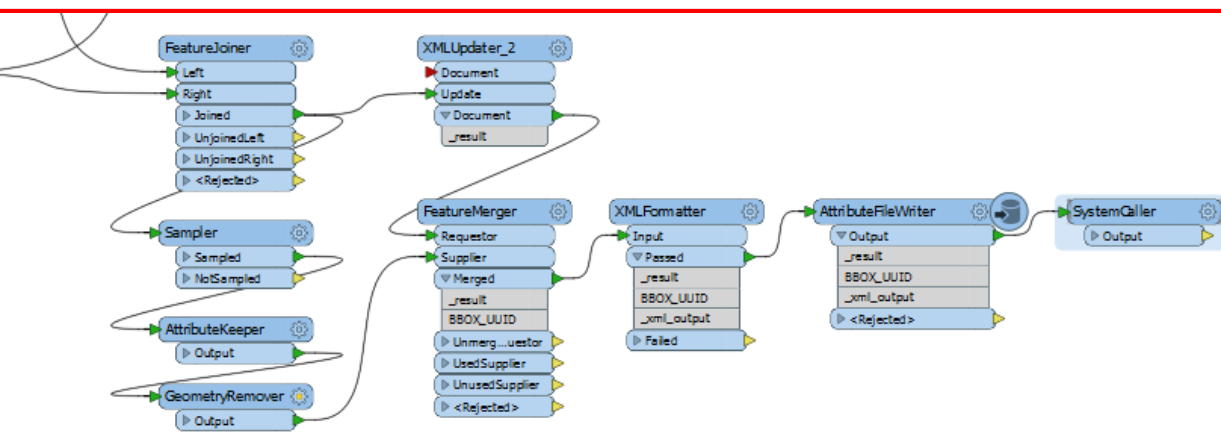
Schemat generowania zestawu danych do aktualizacji Lod 2



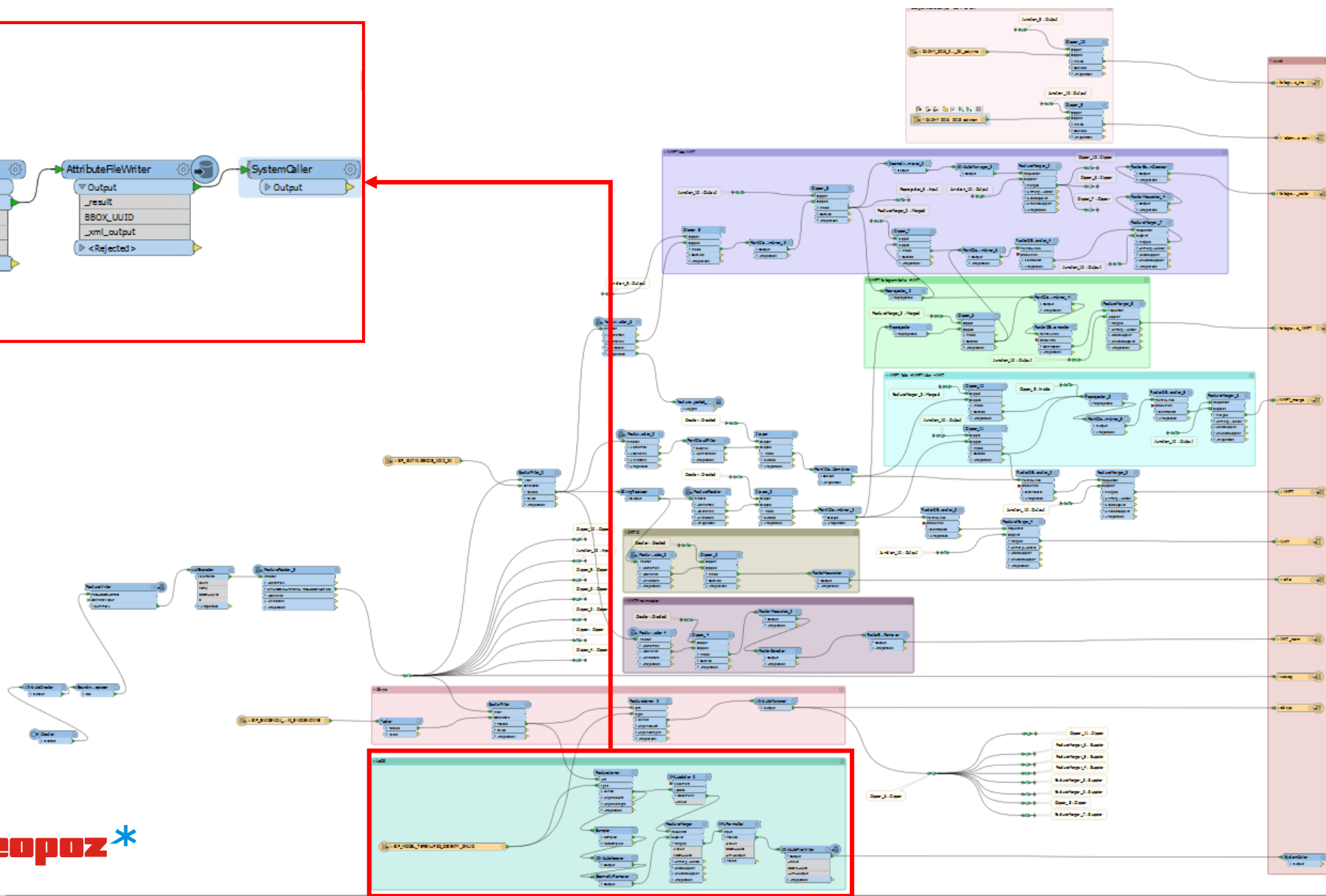
Aktualizacja LoD 2 – generowanie pojedynczego obiektu



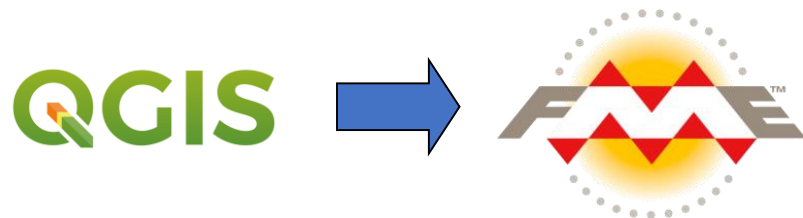
Aktualizacja LoD 2 – generowanie obiektów w zasięgu



```
</resources>
</import>
<export>
  <query version="2.0.0" targetSRS="UUID_b392162e-ee3f-4802-b6b8-0db93d99807e" mode="
  useCallNameFilter="false" useBboxFilter="false" useTiling="false">
    <typeNames>
      <typeName>Building</typeName>
    </typeNames>
    <filter bboxMode="overlap">
      <gmlIds>
        <id>BID_179777_517f6aff-fa57-4885-91dc-bcce9a1e9879</id>
        <id>BID_123847_32e70aa4-9505-4931-9540-6730d939d5d0</id>
      </gmlIds>
      <gmlName wildcard="" singleCharacter="" escapeCharacter="">gmlName</gmlName>
      <bbox>
        <valueReference>gml:boundedBy</valueReference>
        <envelope srsRef="UUID_b392162e-ee3f-4802-b6b8-0db93d99807e">
          <lowerCorner>6428282.458470761 5807221.290224273</lowerCorner>
          <upperCorner>6429769.738525888 5808541.739342429</upperCorner>
        </envelope>
      </bbox>
    </filter>
    <count>/count</count>
    <lod mode="maximum">
      <lod>2</lod>
    </lod>
    <tilingOptions>
      <rows>1</rows>
      <columns>1</columns>
      <tilePath>tile</tilePath>
      <tilePathSuffix>row_column</tilePathSuffix>
      <tileNameSuffix>none</tileNameSuffix>
      <includeTileAsGenericAttribute>false</includeTileAsGenericAttribute>
      <genericAttributeValue>min_xmin_xmax_yMax</genericAttributeValue>
    </tilingOptions>
  </query>
  <path>
    <mode>lastUsed</mode>
    <standardPath></standardPath>
    <lastUsedPath>D:\Projekty\Roboczy\lastUsedPath</lastUsedPath>
  </path>
```



Integracja FME Server z innymi aplikacjami GIS



Brec Data

Zakres

Bbox (x_min y_min x_max ymax):

FME

Wybierz budynek

OBJECTID:

FME

```
xmin,ymin,xmax,ymax=box.toRectF().getCoords()

wspl=str(xmin)+" "+str(ymin)+" "+str(xmax)+" "+str(ymax)
self.dockwidget.coords.setText(wspl)

#print(xmin,ymin,xmax,ymax)
def start(self):
    self.dockwidget.coords.clear()
    global layer
    layer=iface.addVectorLayer("Polygon?crs=epsg:2177","zasieg","memory")
    #pr=layer.dataProvider()

    layer.startEditing()
    iface.actionAddFeature().trigger()

    layer.featureAdded.connect(self.update)
    layer.geometryChanged.connect(self.update)

def FME_zakres(self):
    SERVER_URL =
    REPOSITORY = "BREC"
    WORKSPACE = "bbox_to_BREC.fmw"
    TOKEN =

    # Set up the published parameters as object
    params={ "publishedParameters":[
        {"name": "COORDS",
         "value": '{0}'.format(wspl)}
        ]
    }
    url = '{0}/fmerest/v2/transformations/commands/submit/{1}/{2}'.format(SERVER_URL, REPOSITORY, WORKSPACE)

    # Request constructor expects bytes, so we need to encode the string
    body = json.dumps(params).encode('utf-8')

    headers = {
        'Content-Type' : 'application/json',
        'Accept' : 'application/json',
        'Authorization' : 'fmetoken token={0}'.format(TOKEN)
    }

    # This will use POST, since we are including data
    req = urllib.request.Request(url, body, headers)

    r = urllib.request.urlopen(req)

    resp = r.read()
    resp = resp.decode('utf-8')
    resp = json.loads(resp)

    if r.status == 202:
        self.dockwidget.fme_1.setText('Job ID - {0}'.format(resp['id'])+' - przyjęto do realizacji')
        layer.commitChanges()
        QgsProject.instance().removeMapLayer(layer.id())
        self.dockwidget.coords.setText("")
```

Wizualizacja 3D danych systemu informacji przestrzennej oraz zewnętrznych



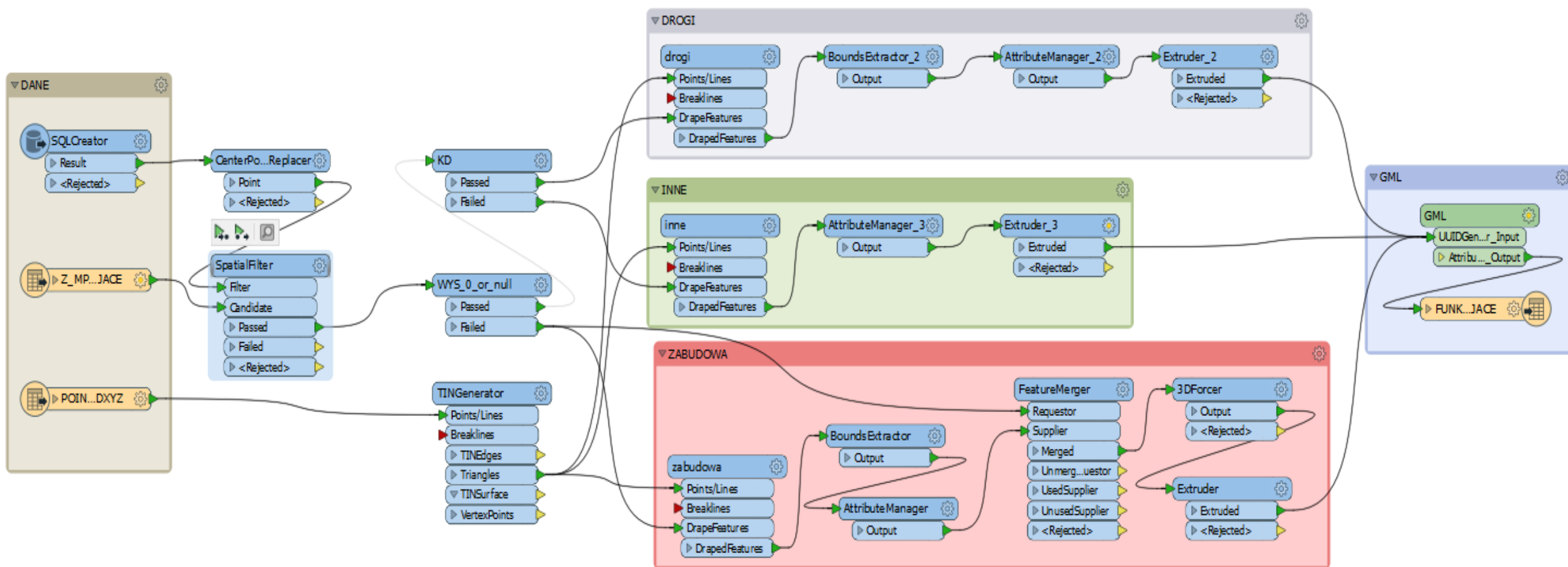
Wizualizacja Miejscowych Planów Zagospodarowania Przestrzennego

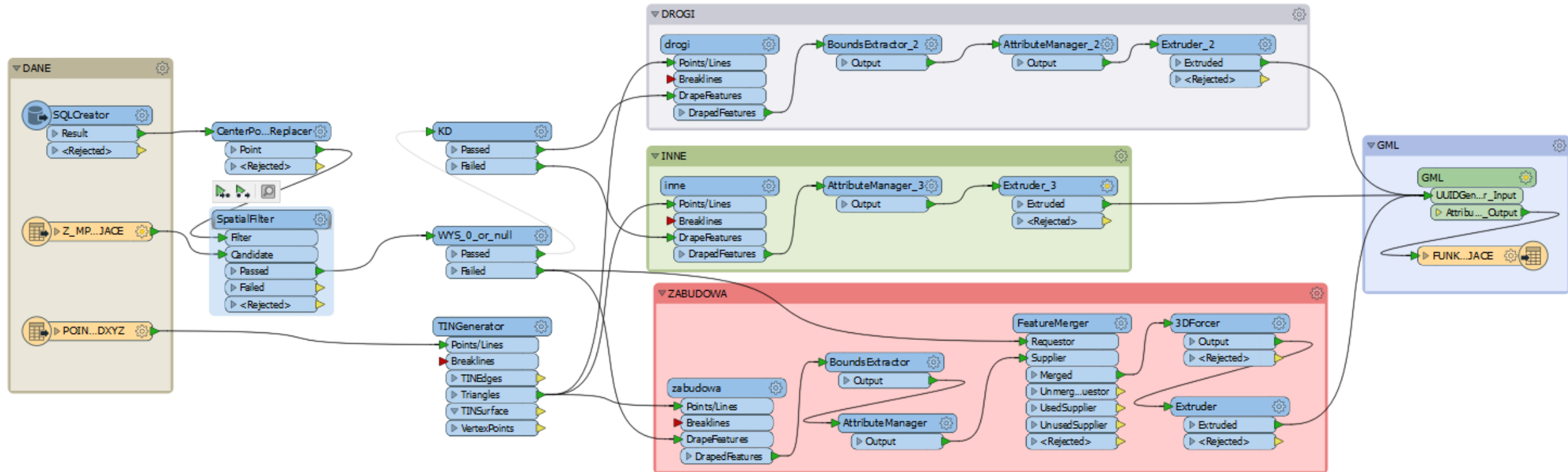


Wizualizacja Miejscowych Planów Zagospodarowania Przestrzennego



Wizualizacja Miejscowych Planów Zagospodarowania Przestrzennego





Profil terenu na żądanie w przeglądarce 2D

01



Profil terenu na żądanie w przeglądarce 2D

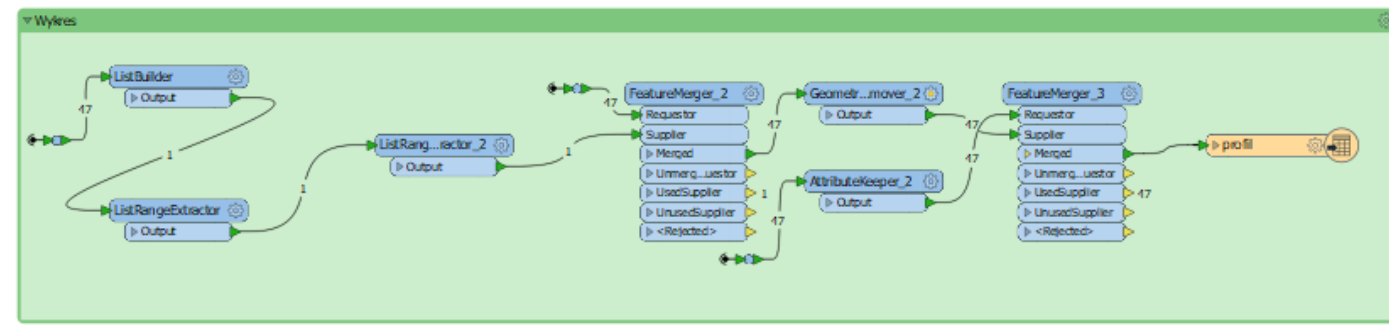
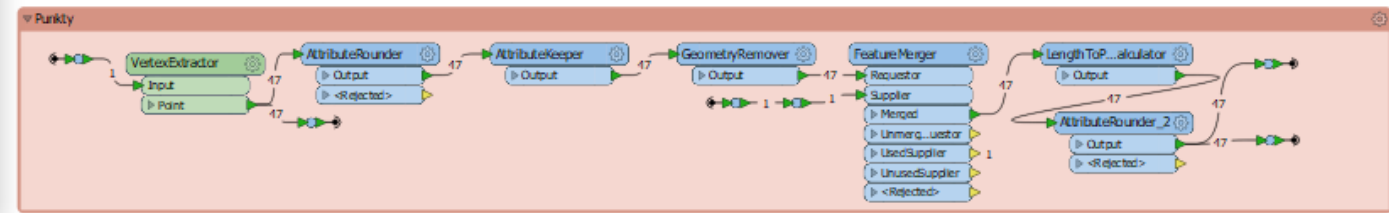
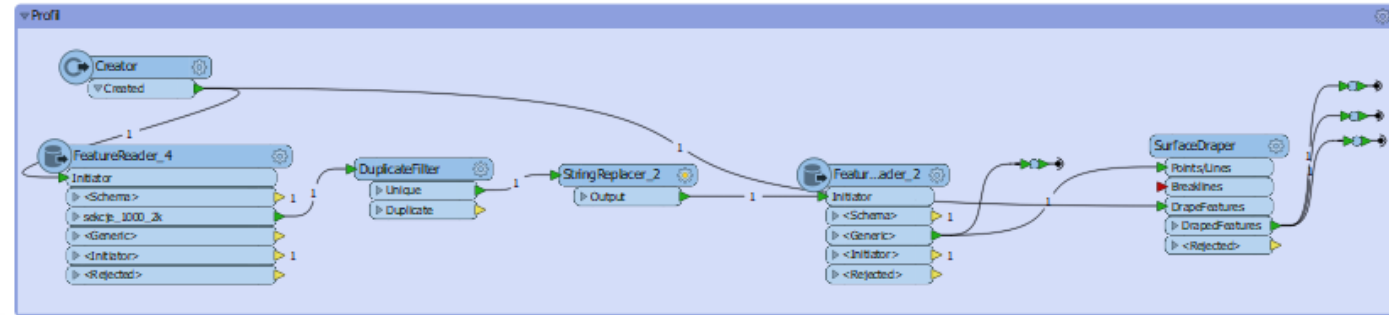
02

Translation Parameters

User Parameters

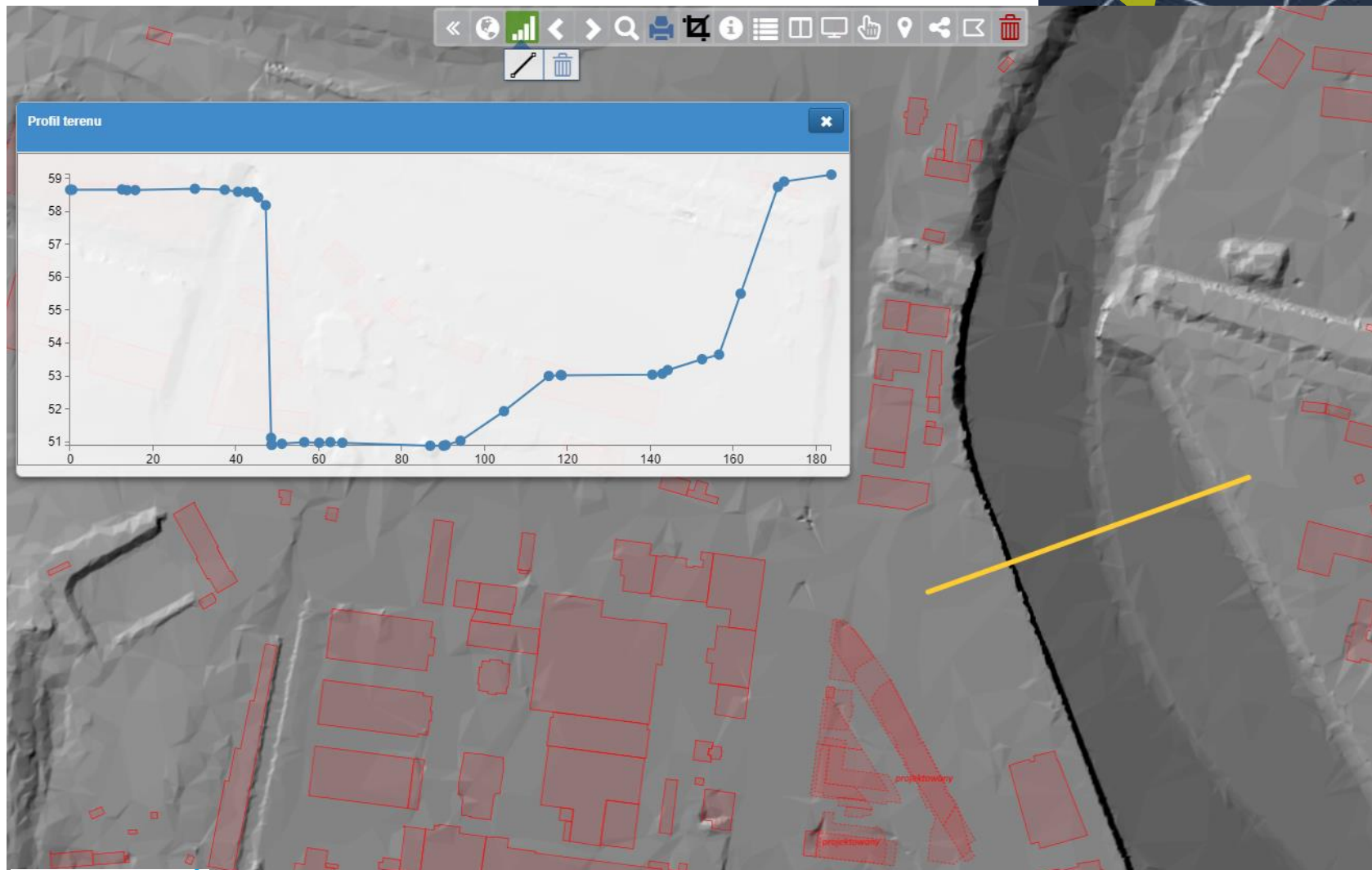
COORD_LIST: 6144142 5806848.448550672

OK Cancel



Profil terenu na żądanie w przeglądarce 2D

03



sip.poznan.pl

Portal Systemu Informacji Przestrzennej Miasta Poznania



Bazy danych

Raporty tematyczne



Mapy

Warstwy danych



Usługi OGC

WMS, WMTS, WFS



Metadane

Usługa katalogowa



Model 3D

Model trójwymiarowy Poznania



Budynki 3D

Pobieranie budynków 3D



Zdjęcia ukośne

Ortofotomapa Poznania



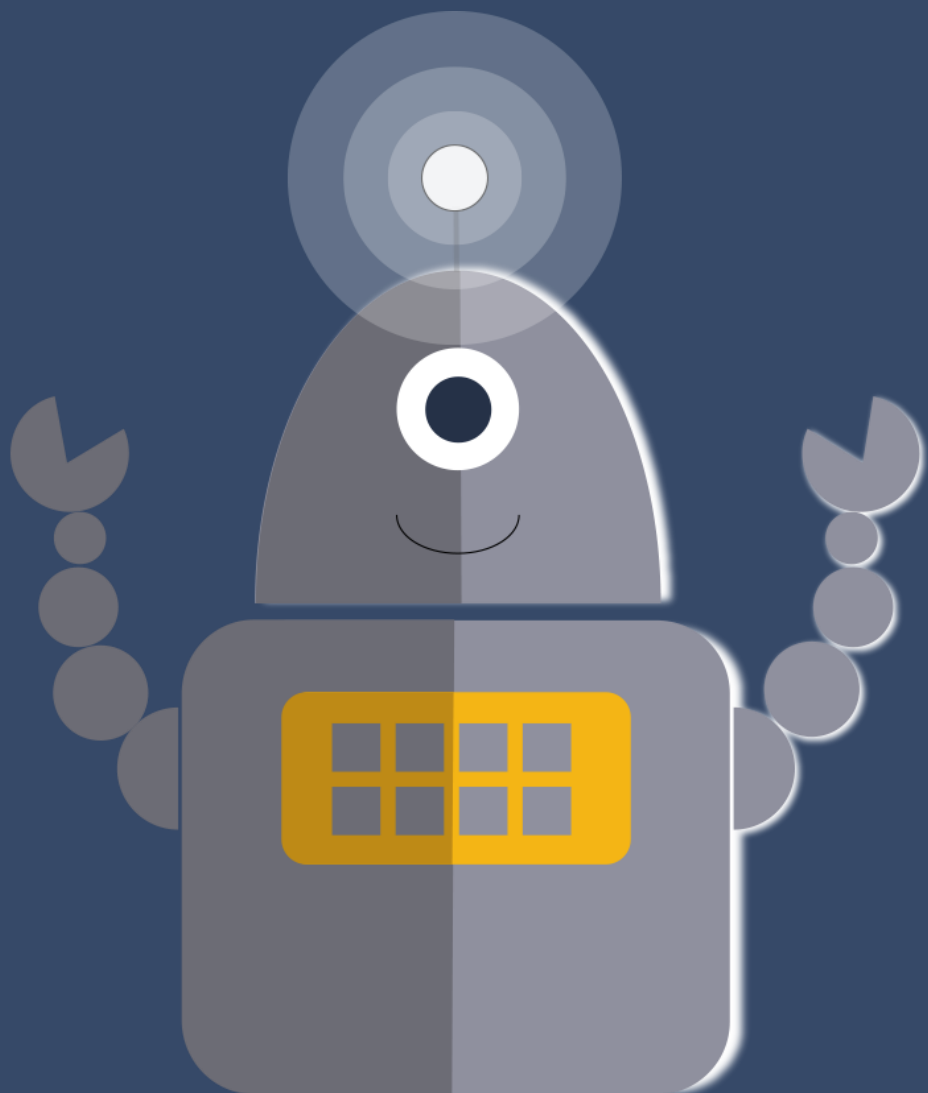
Portal Klienta

Zamawianie materiałów z PZGIK

[Poprzednia wersja portalu](#)

[Kontakt](#)

[Informacje o portalu](#)



Dziękujemy za uwagę

Krzysztof Drygalski

Marcin Lis