

Projekt JUMP a možnosti rozšiřování nástroje JUMP Workbench

Jan Růžička, Petr Fuks

Instytut geoinformatiky, VSB – TUO, 17. listopadu 15, 708 33 Ostrava – Poruba

E – mail: jan.ruzicka@vsb.cz, petr.fuks.hgf@vsb.cz

WWW: <http://gisak.vsb.cz/gportal>

Abstract

The paper briefly describes project JUMP and possibilities how to extend functionality of the JUMP Workbench GIS tool. JUMP means Java Unified Mapping Platform. We can say that the JUMP is a platform for development GIS tools in a unified form covered by JUMP Workbench. First of all the JUMP Project gives tools to develop some GIS applications, but one of the project results could be directly use as a GIS Mapping tool. JUMP Workbench GUI gives possibilities to view, edit and analyse spatial data in one GUI environment. If the user is familiar with programming in Java language, than the user can develop specific extension to the JUMP Workbench to solve different problems. JUMP Workbench is published under GPL licence.

JUMP

JUMP-Project [6] poskytuje balík rozšiřitelných API funkcí a grafické uživatelské rozhraní pro prohlížení a manipulaci s prostorovými datovými sadami. Zvolená vývojová platforma je jazyk Java (resp. Java 2 Platform), což umožňuje snadnou přenositelnost mezi různými operačními systémy (Linux, Windows, Mac OS). V současné době se skládá ze tří hlavních projektů:

1. JUMP Unified Mapping Platform [5].
2. JTS Topology Suite [4].
3. JCS Conflation Suite [3].

JUMP Unified Mapping Platform (JUMP)

První projekt se zabývá vývojem platformy JUMP resp. vývojem aplikace JUMP Workbench. Při vývoji je kladen velký důraz na rozšiřitelnost a modularitu. Rozšiřitelnost aplikace je zajištěna nejen využitím otevřených standardů, ale také vlastní koncepcí návrhu. Koncept je tvořen hlavním rámcem do něhož jsou připojeny funkční části zvané pluginy. Nad tímto rámcem je definováno grafické uživatelské rozhraní pro snadné ovládání funkcí aplikace.

Přehled charakteristik nástroje JUMP

- Interaktivní nástroje pro prohlížení, editování a zpracování geografických dat
- API funkce poskytující plný programátorský přístup ke všem funkcím pro práci se vstupem a výstupem dat, datovými sadami geoprvků, zobrazením a tvorbou prostorových dotazů.
- Vysoký stupeň modularity a rozšiřitelnosti

- Podporuje všechny významné standardy jakým je GML [9] a objektové modely vydané konsorciem OpenGIS [10].
- Otevřený kód je napsán výhradně v jazyce Java

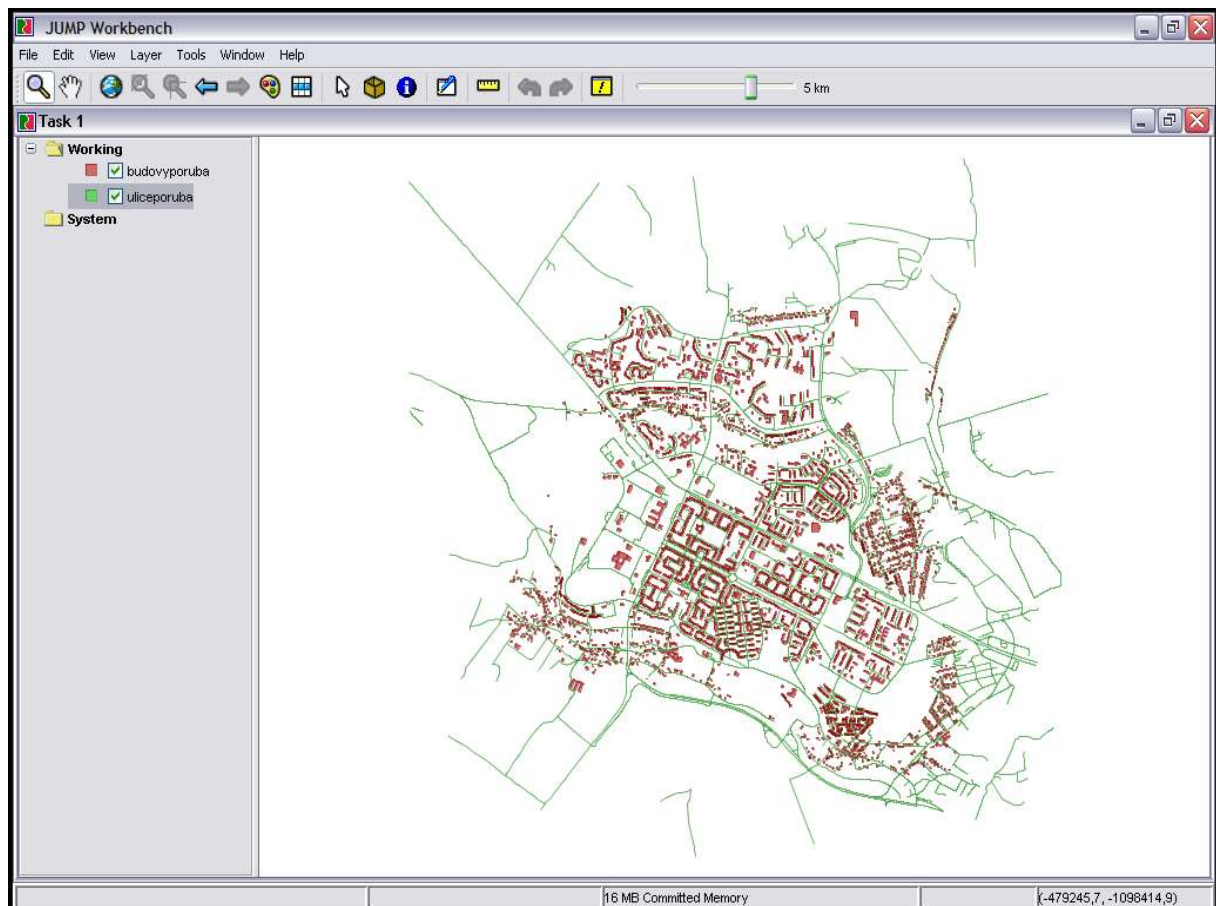
Aplikace JUMP využívá tříd z projektu JTS Topology Suite (JTS) jež poskytuje objektový model prostorových dat založený na specifikacích OpenGIS konsorcia [9], [10].

Workbench

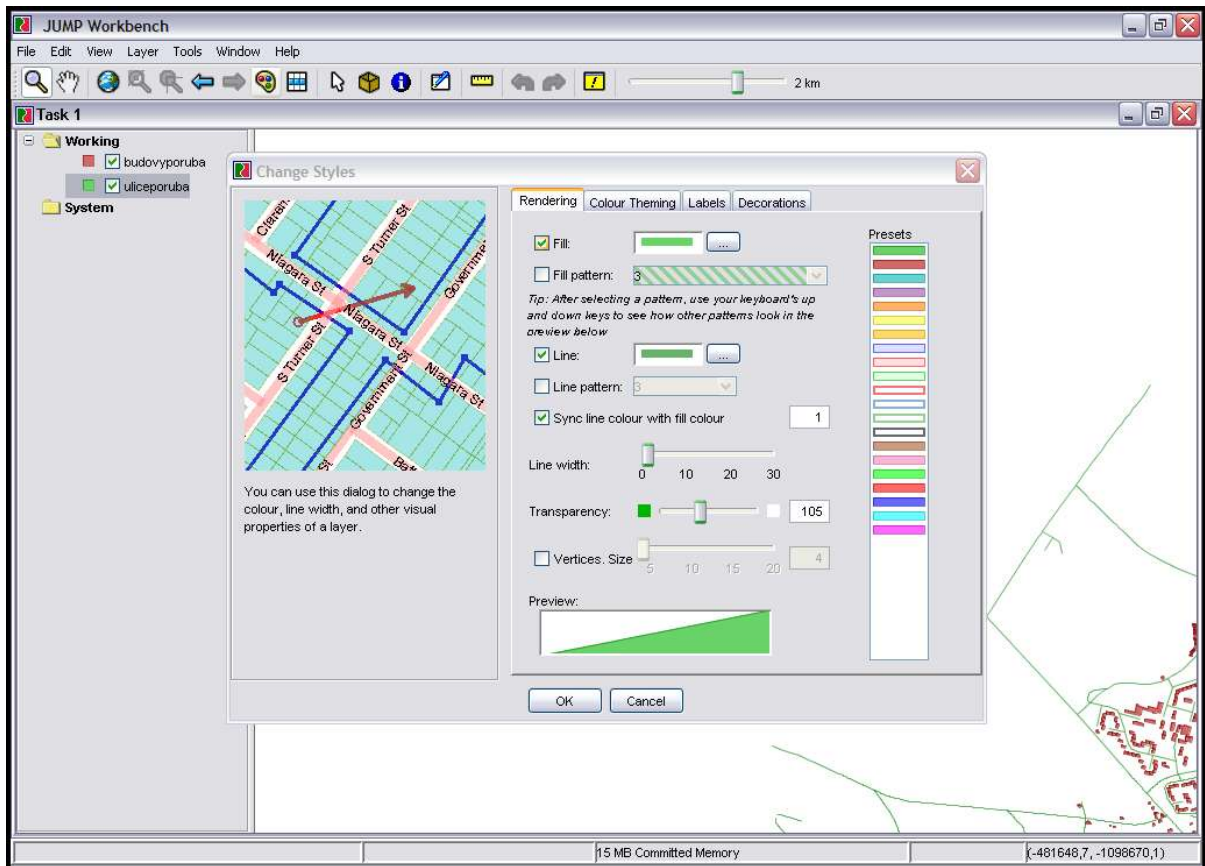
Workbench (pracovní rámeček) představuje snadno rozšiřitelné, interaktivní grafické uživatelské rozhraní sloužící pro vizualizaci a manipulaci s prostorovými daty a jejich atributy. Je tvořeno grafickým uživatelským rozhraním složeným z pracovních oken. V každém takovém pracovním okně lze zobrazit datové vrstvy. Data lze nahrávat z mnoha různých datových formátů včetně GML a ESRI shapefile. U geoprvků lze editovat jak jejich prostorovou tak i atributovou složku, kopírovat je mezi vrstvami i pracovními okny, atd.

Možnosti vizualizace

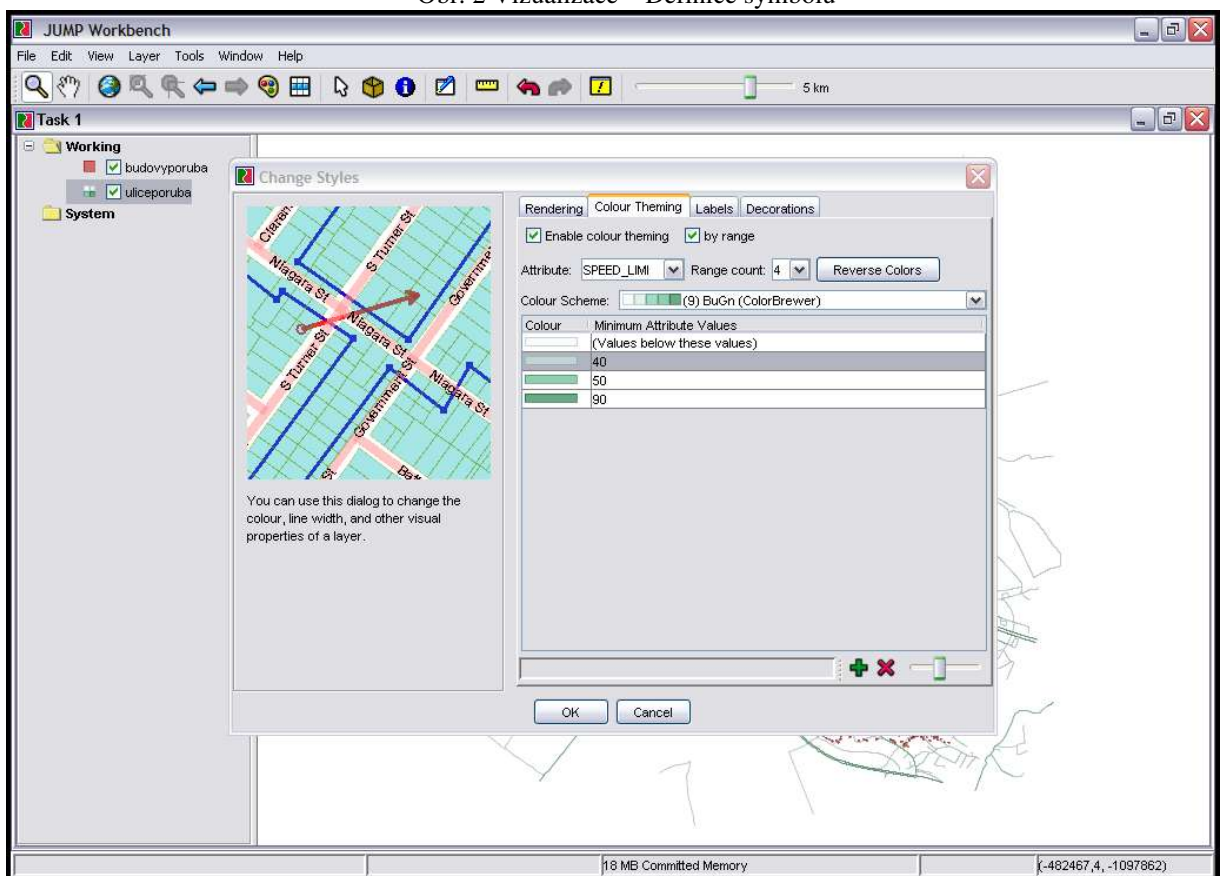
Aplikace JUMP dovoluje mnohonásobné zobrazení vrstev. Pro každou vrstvu lze nastavit styly geoprvků zahrnující barevné zvýraznění, šířku linií, průhlednost a styl linie. Klasifikaci geoprvků lze nastavit podle diskretních hodnot nebo jejich rozsahu. Ke každému geoprvcu lze přidat popisek. U popisku lze nastavit barvu, směr natočení a velikost v závislosti na hodnotě atributu.



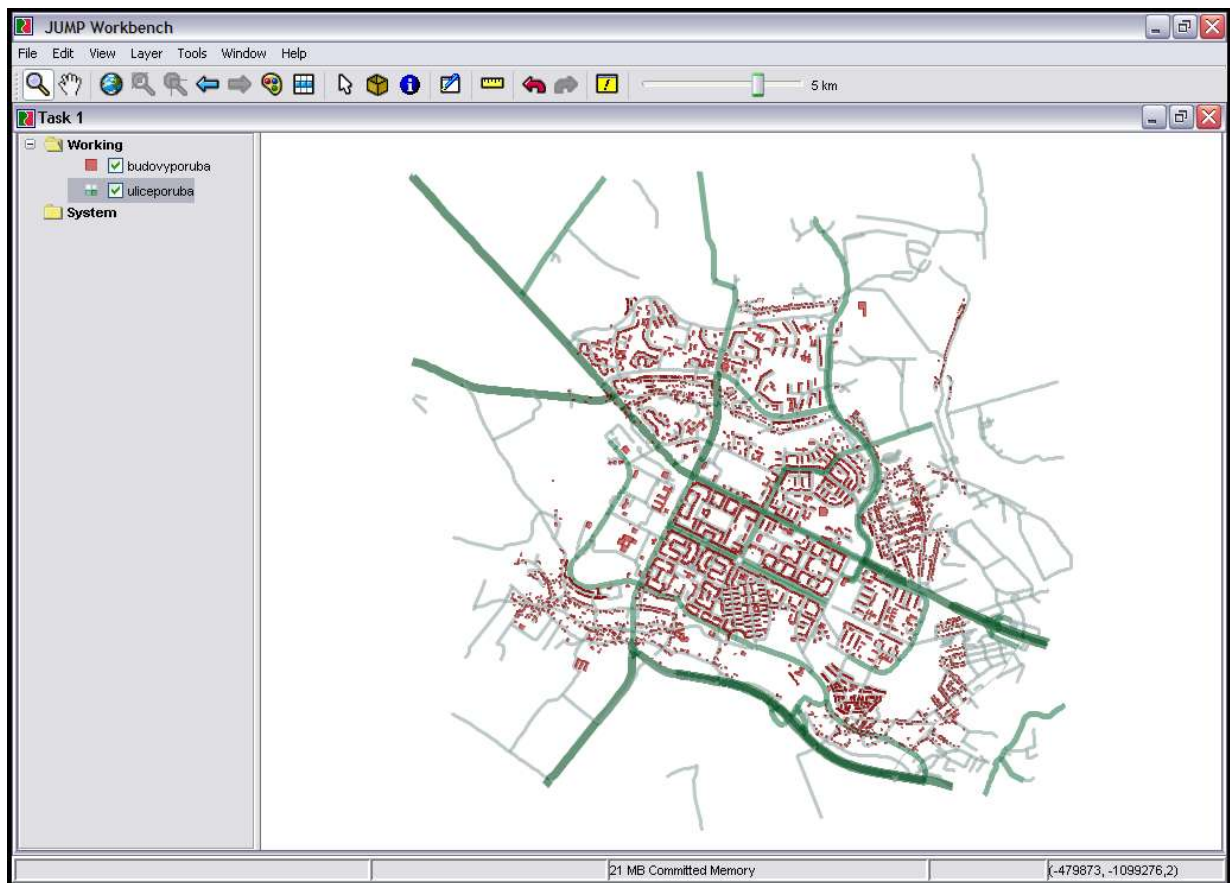
Obr. 1 Vizualizace – Mapové okno



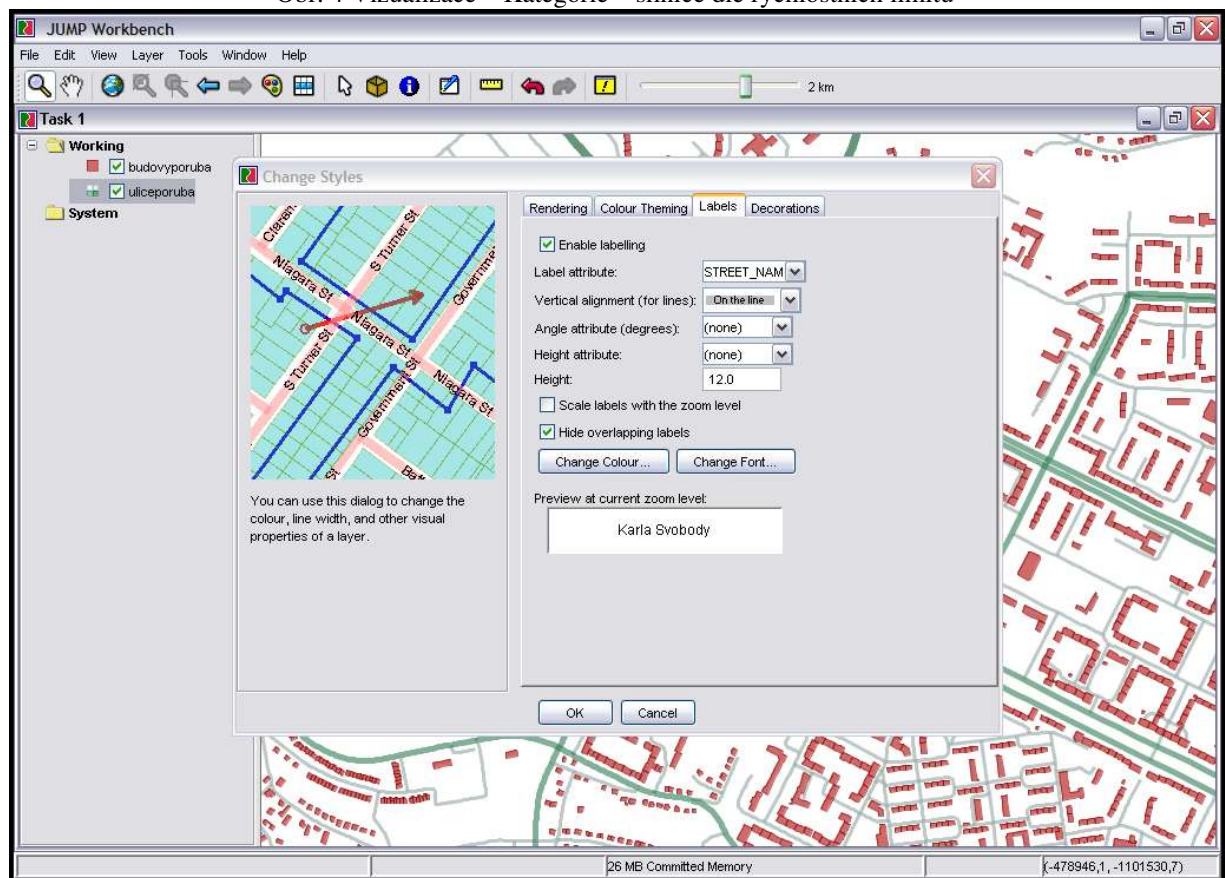
Obr. 2 Vizualizace – Definice symbolu



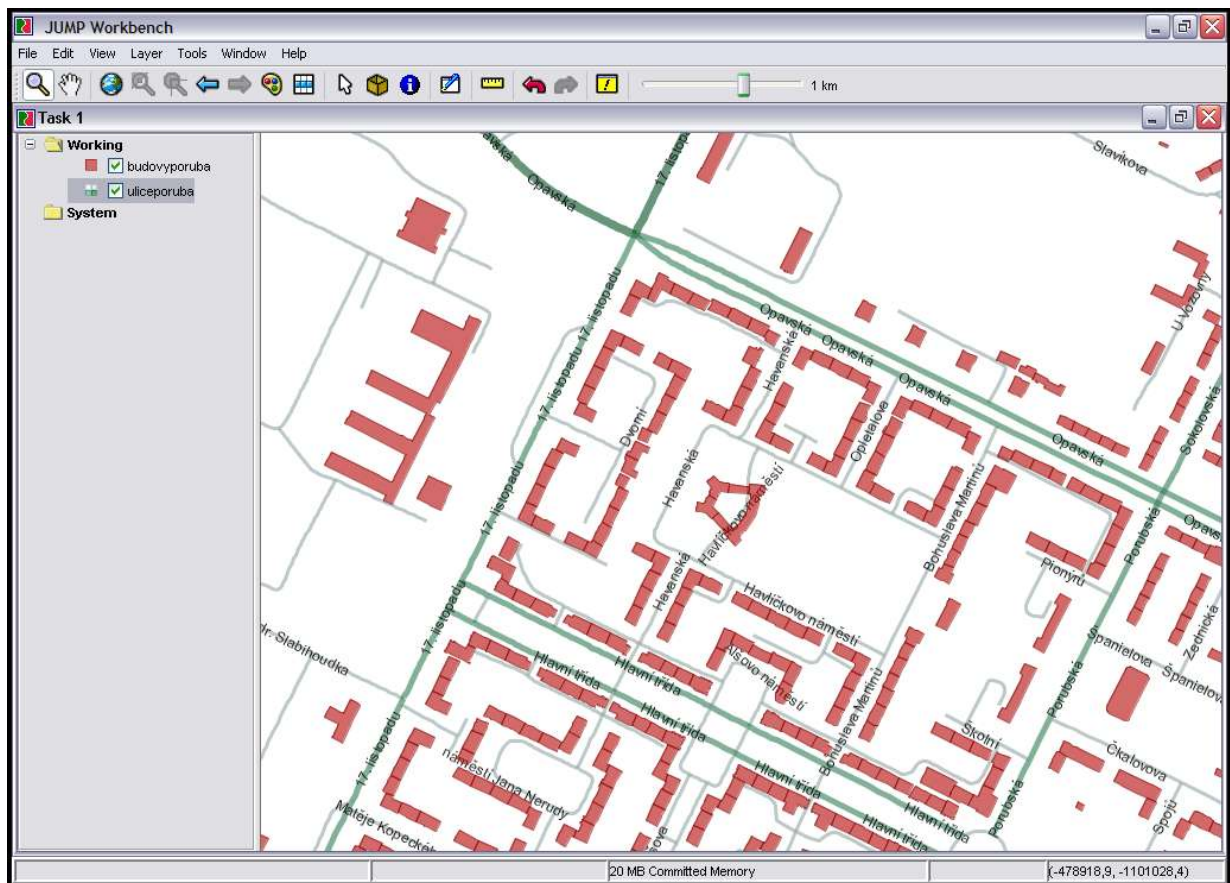
Obr. 3 Vizualizace – Definice kategorií



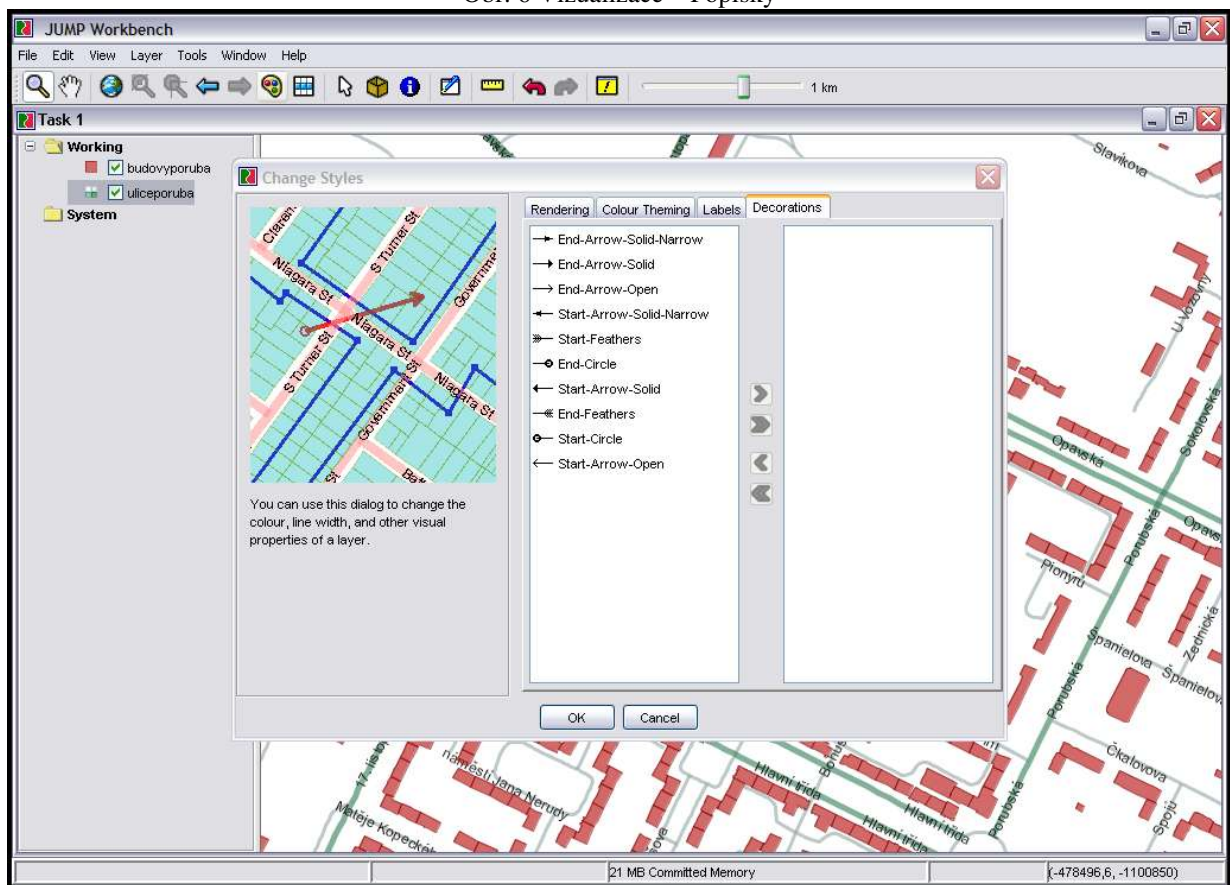
Obr. 4 Vizualizace – Kategorie – silnice dle rychlostních limitů



Obr. 5 Vizualizace – Definice popisků



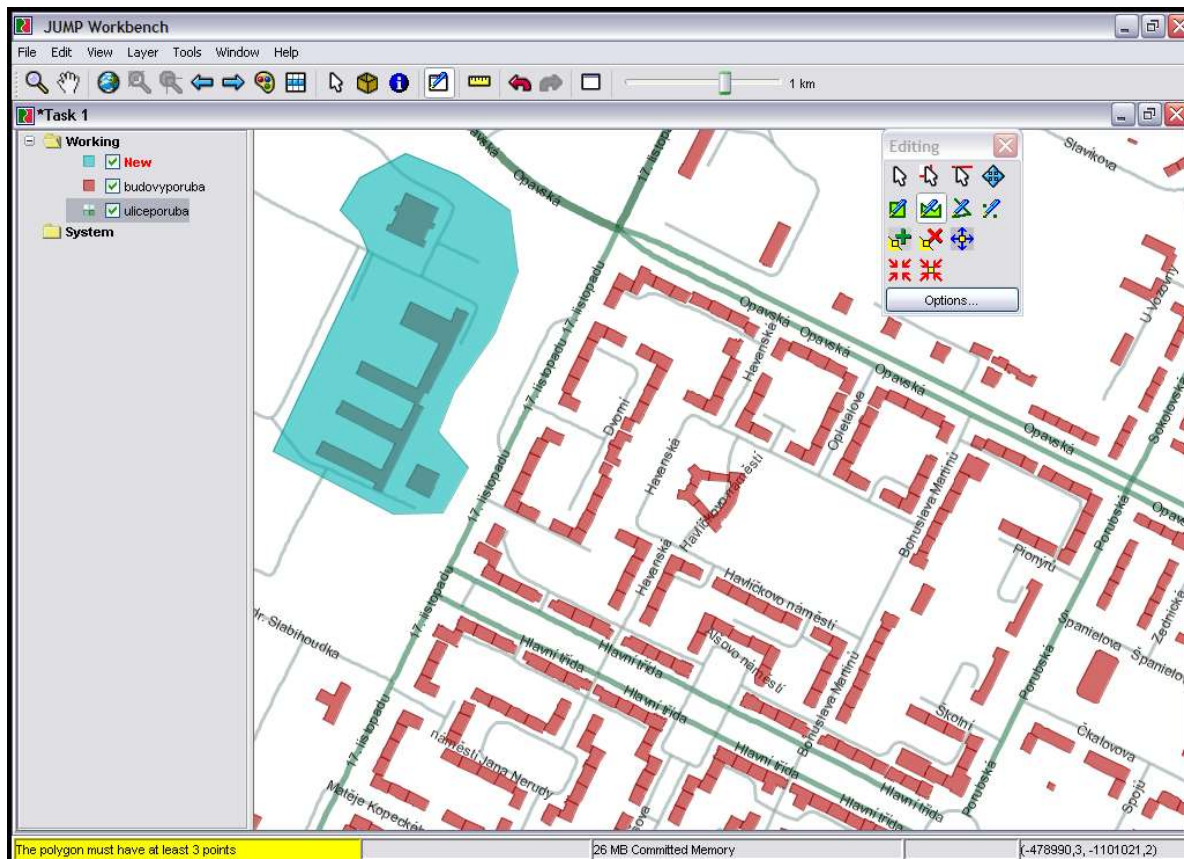
Obr. 6 Vizualizace – Popisky



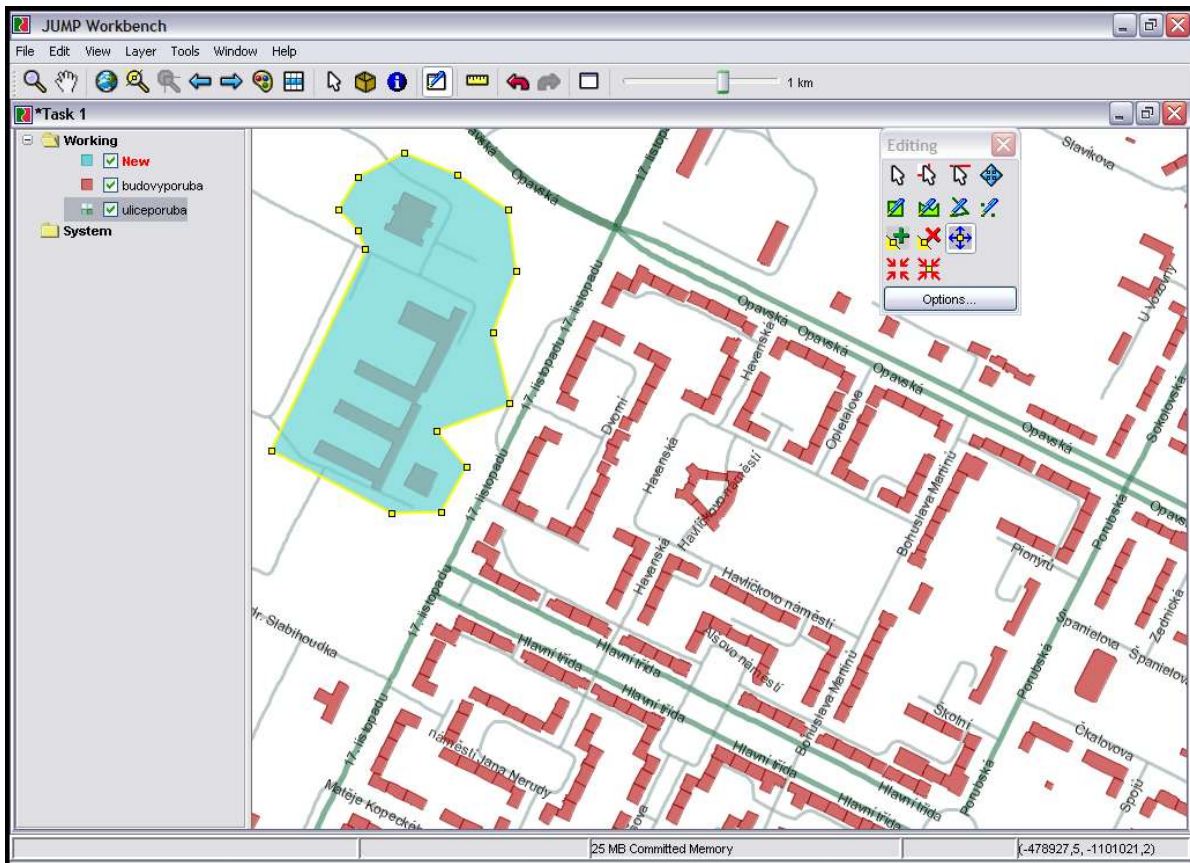
Obr. 7 Vizualizace – Zakončení linií

Editační nástroje

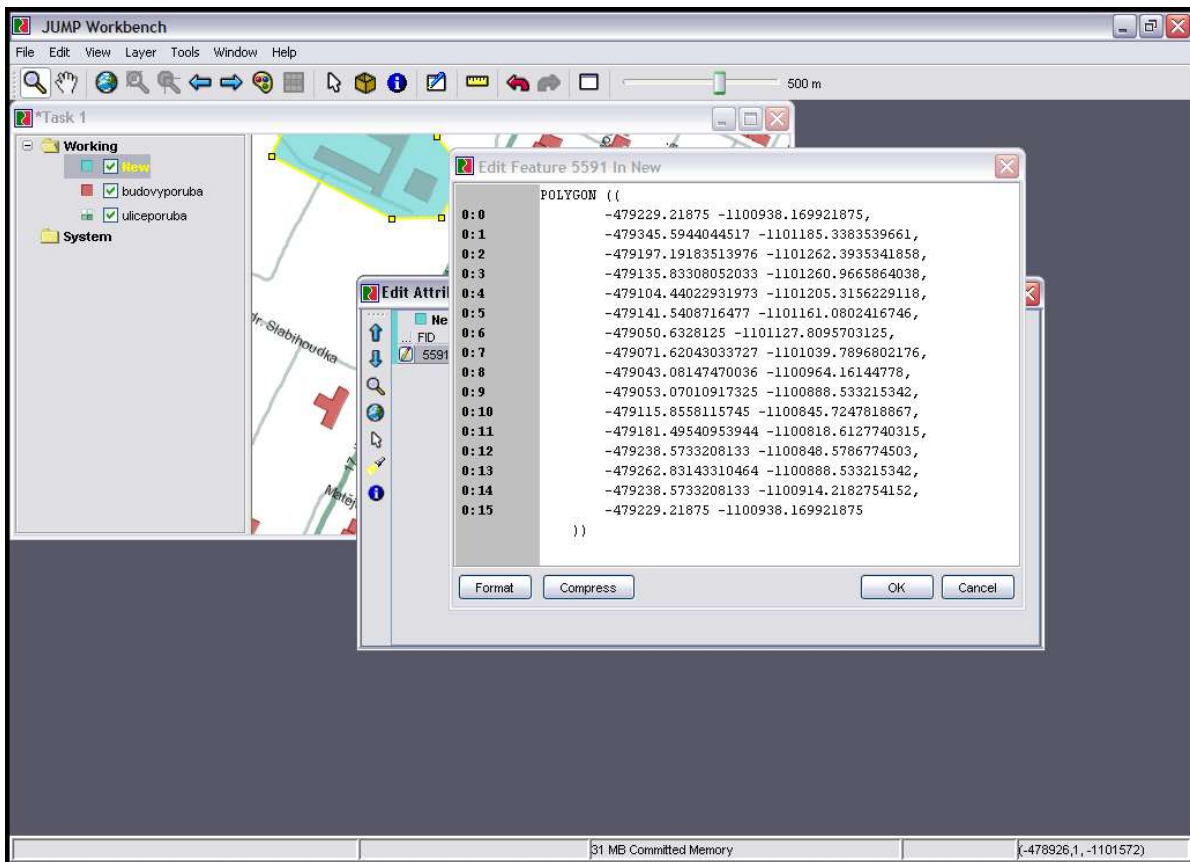
JUMP dovoluje vytvářet a editovat geoprvky. U geoprvků dovoluje editovat jejich prostorovou reprezentaci. Editování lze provádět jak vizuálně pomocí grafické reprezentace geoprvků, tak i textovou editací jejich WKT reprezentace [8]. Pomocí kurzorových dotazů se lze jednoduše dotázat buď to na jednotlivé geoprvky nebo skupiny geoprvků.



Obr. 8 Editační nástroje



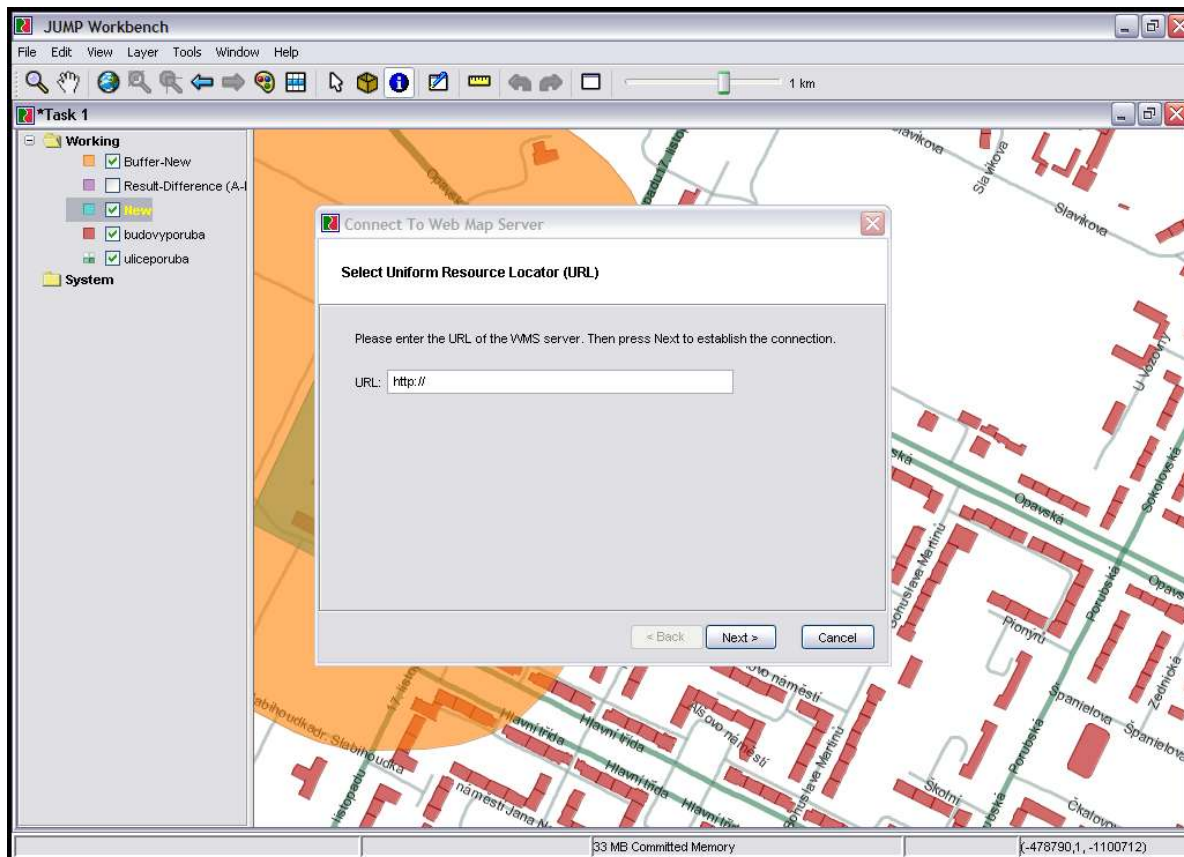
Obr. 9 Posun lomového bodu u polygonu



Obr. 10 Editace souřadnic

Integrovaný WMS klient

JUMP Workbench může také vystupovat jako klient OGC WMS serverů [1]. Poskytuje interaktivní WMS rozhraní dovolující uživateli vytvářet a editovat WMS dotazy. Tyto dotazy lze vytvářet z vrstev uložených na serveru a kombinovat je spolu s lokálně uloženými vrstvami dat.

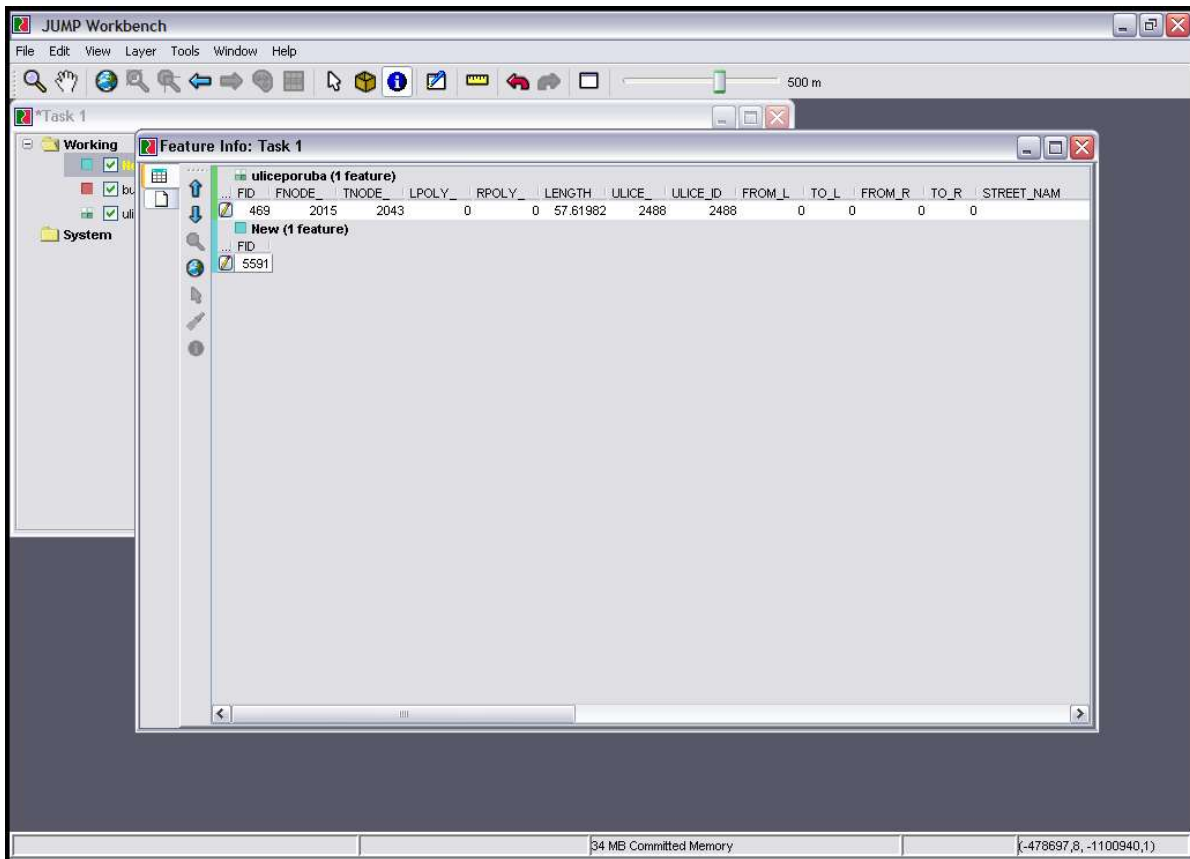


Obr. 11 Definice připojení k WMS

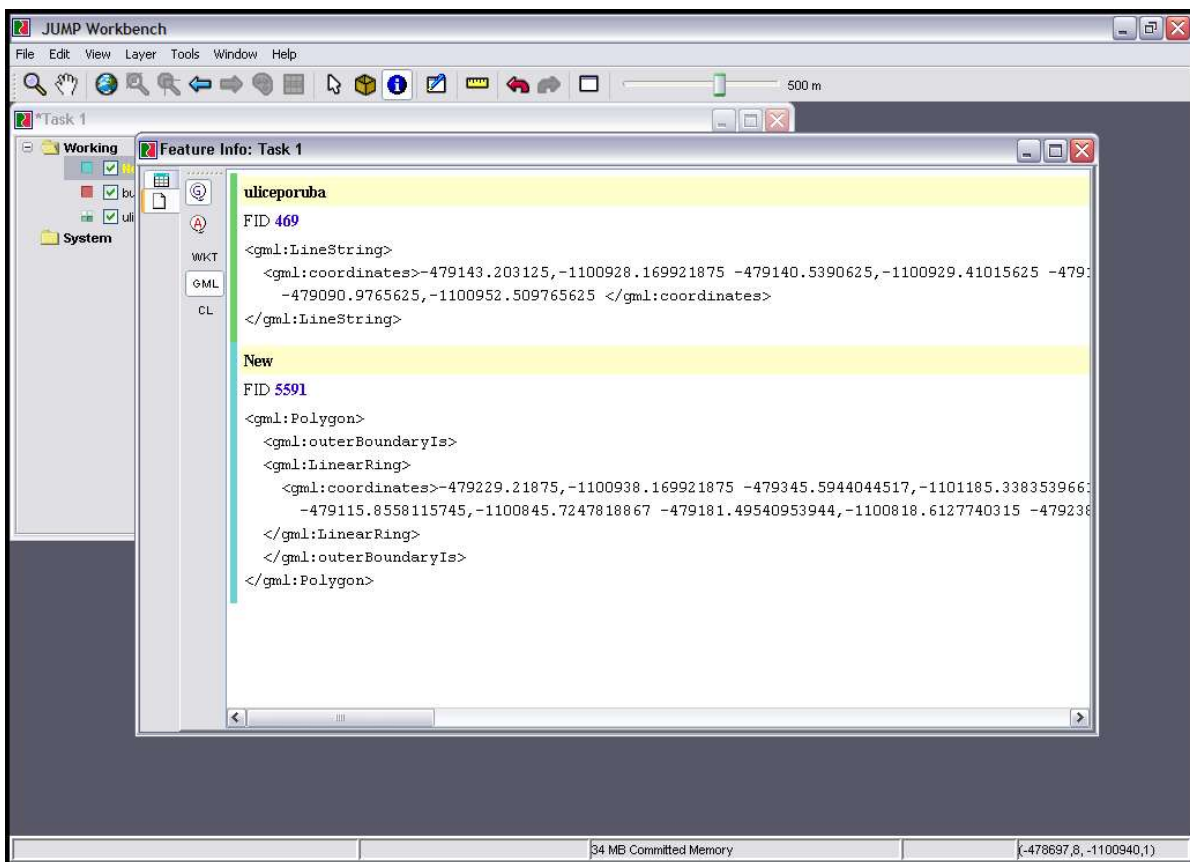
Nástroje pro prostorové analýzy

JUMP poskytuje rozsáhlou sadu nástrojů pro provádění prostorových analýz. Mezi nástroje patří:

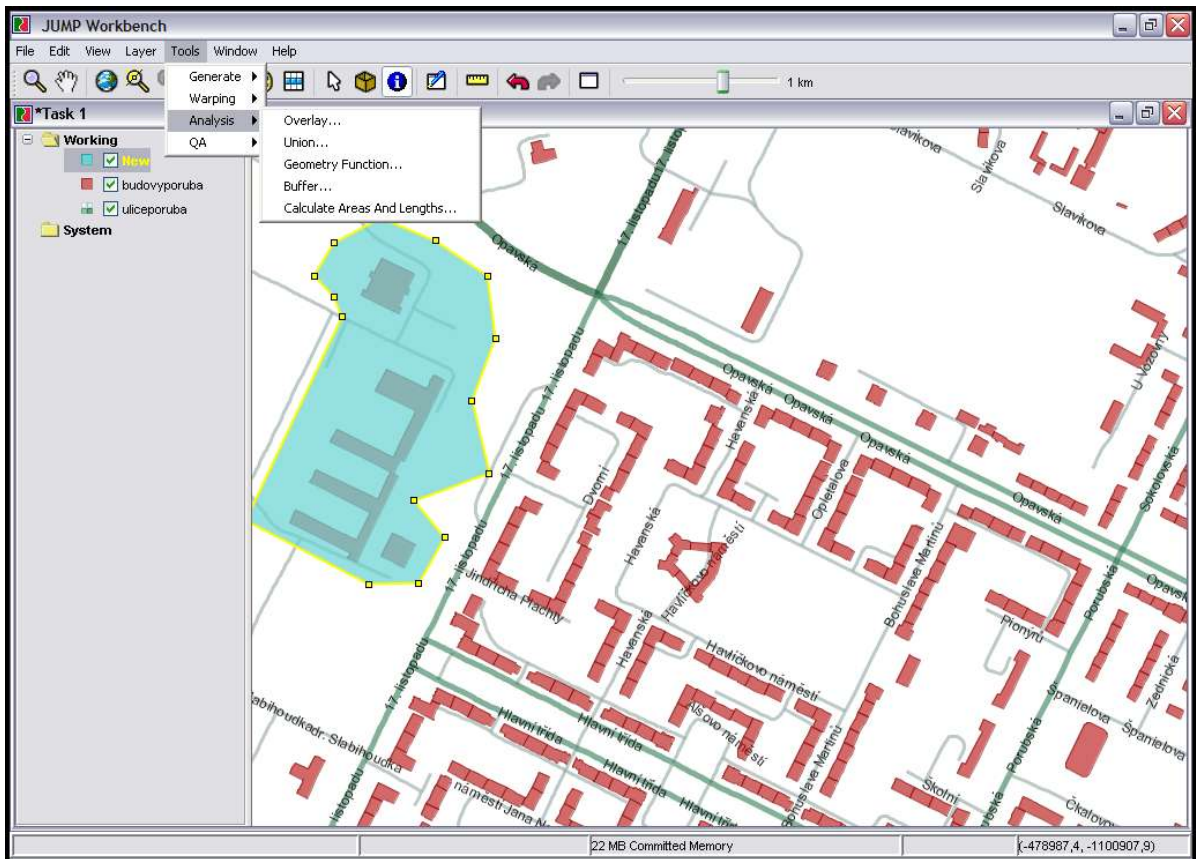
- Souhrnná statistika obsahu vrstev zobrazující počet všech bodů, komponent, ploch, délek
- Statistiky pro jednotlivé vrstvy zobrazující počet všech bodů, komponent, ploch, délek
- Nástroj pro měření vzdáleností
- Výpočet obsahu a délek
- Překryv dvou datových sad
- Sjednocení geometrických prvků datových sad
- Vytvoření obalových zón



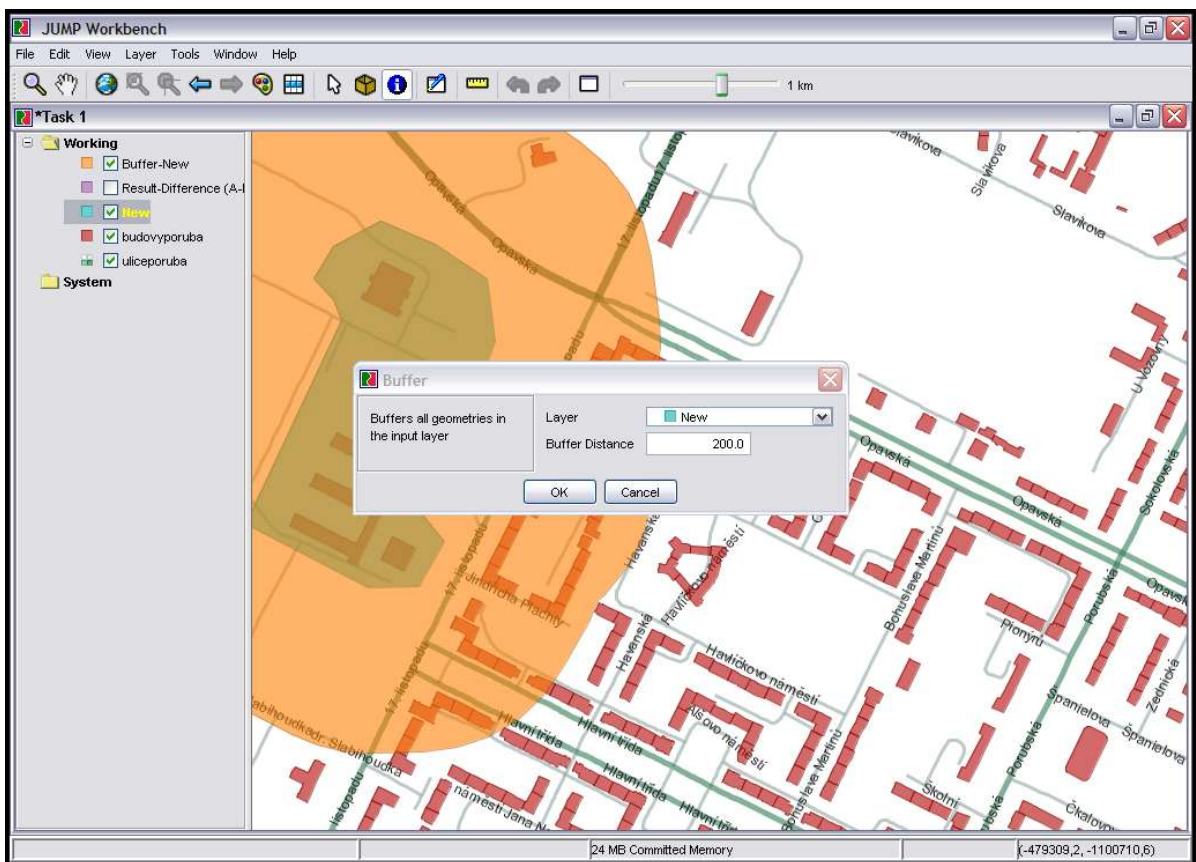
Obr. 12 Identifikace geopršku – zobrazení atributů



Obr. 13 Identifikace geopršku – zobrazení GML



Obr. 14 Nabídka některých analýz



Obr. 15 Obálka

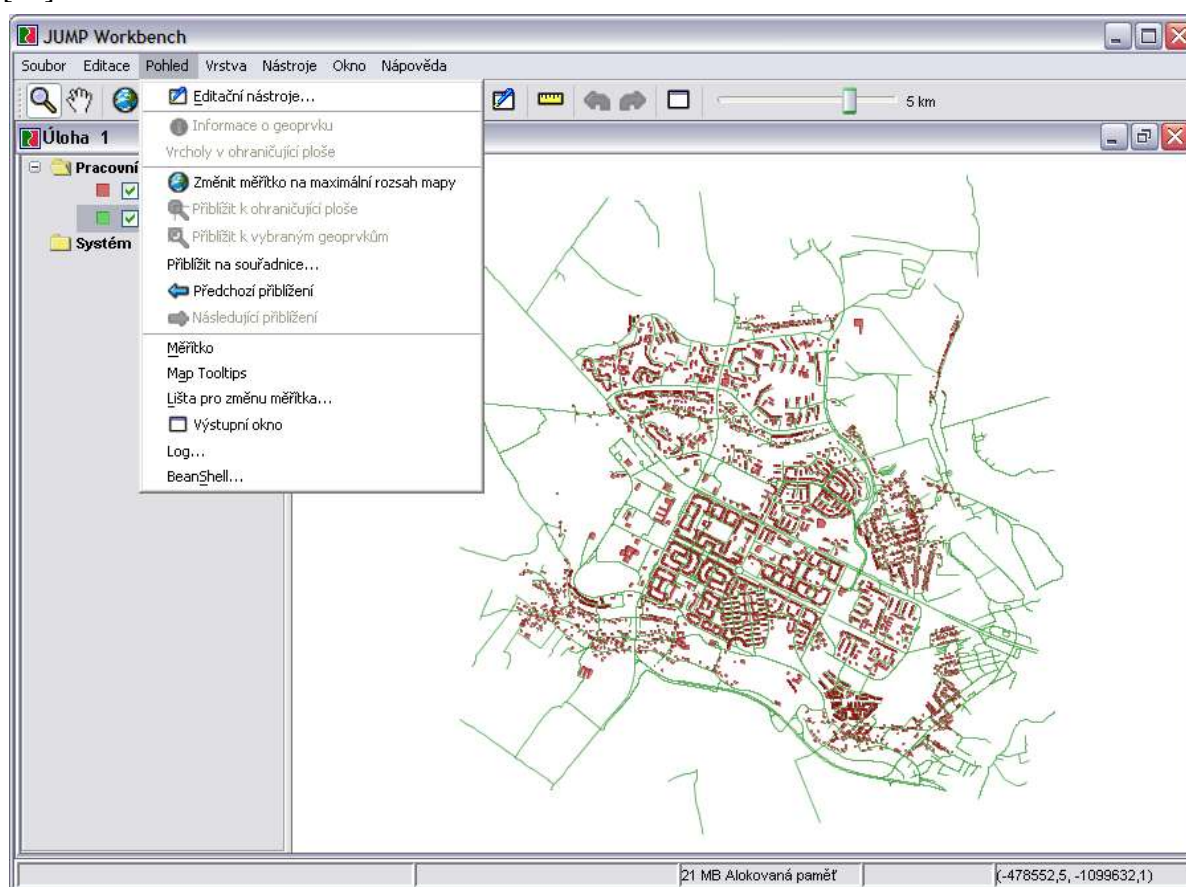
Ověření správnosti geometrie

JUMP poskytuje nástroje k ověření a opravě chybných dat v datové sadě tzv. Quality Assurance / Quality Control (QA/QC). Nástroje zahrnují procedury:

- Ověření správnosti geometrie
- Detekce opakujících se bodů
- Detekce nedokončených úseků
- Detekce malých úhlů
- Detekce špatných geometrických prvků

Úprava do národního prostředí

Ve verzi 1.1.2 byla přidána podpora standardu I18N [14], která umožňuje překlad uživatelského prostředí do jiných jazyků. Lokalizaci do českého jazyka je možné získat na [12].



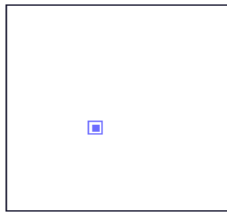
Obr. 16. Uživatelské prostředí v českém jazyce

JTS Topology Suite (JTS)

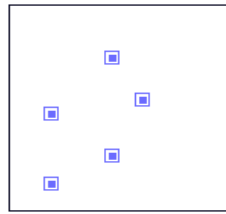
Projekt Topology Suite (sada funkcí pro topologii) je souborem API funkcí sloužících k popsání prostorového modelu a základních geometrických funkcí. Prostorový model je navržen v souladu se standardy OpenGIS konsorcia jež specifikují základní prostorové prvky a prostorové dotazy v SQL [10]. Definované funkce zahrnují:

- Prostorové predikáty (založené na modelu DE-9IM)
- Překryvné funkce (průnik, rozdíl, sjednocení, symetrický rozdíl)
- Obalové zóny

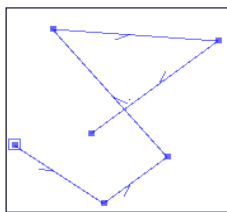
- Konvexní obaly
- Funkce pro výpočet délek a obsahů
- Kontrola topologie



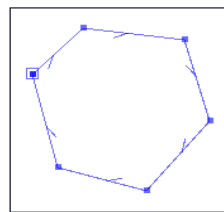
Point



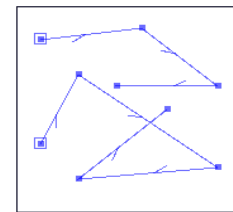
Multi-Point



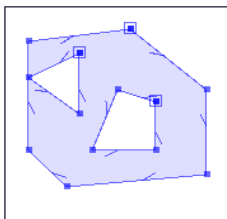
Line-String



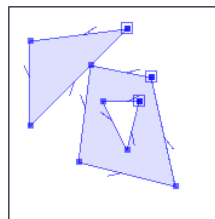
Linear-Ring



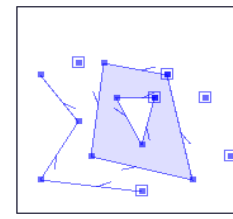
Multi-Line-String



Polygon



Multi-Polygon



Geometry-Collection

Obr 17. Seznam podporovaných geometrických typů [upraveno z 10]

Conflation Suite JCS Project (řešení problému sjednocení dat)

Problémem při správě geografických dat je integrace dvou nebo více datových sad, jež jsou z různých zdrojů. Geoprvky ležící na stejném místě na zemi mohou mít v případě takových dat různou geografickou reprezentaci, jiné označení a různou chybu určení polohy. Kombinace takto rozdílných datových sad vyvolává kolize.

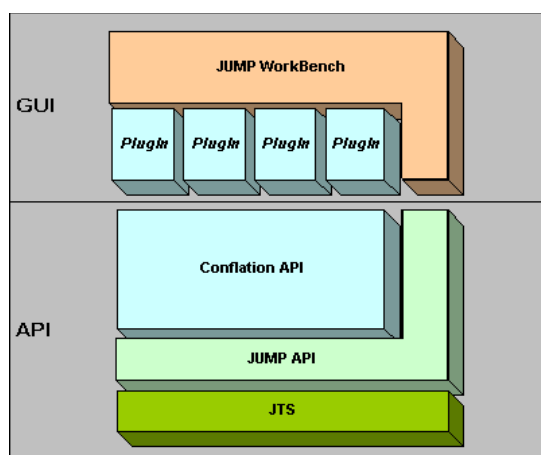
Integraci geografických dat je věnována sada nástrojů projektu JCS. Spolu s rozvojem projektu věnovanému topologii JTS a pokročilému prostorovému rozhodování, je možné vytvořit nové nástroje jež budou integraci řešit robustním, efektivním a opakovatelným způsobem. Projekt JCS je postavený na projektu JTS.

JCS projekt je navržen tak aby byl nezávislý na jednotlivých komerčních programových řešeních. Nepodporuje nativní formáty, pro načtení a uložení dat využívá výhradně formátu GML [9]. Pro převod dat na tento formát využívá programových řešení třetích stran.

Architektura JCS

JCS je vytvořen tak aby poskytoval snadný přístup k algoritmům pro sjednocování dat, jak uživatelským grafickým způsobem, tak i přístupem programátorským. Všechny algoritmy jsou implementovány do podoby API funkcí s nimiž se počítá při automatizovaném zpracování a snadném využití v dalších aplikacích.

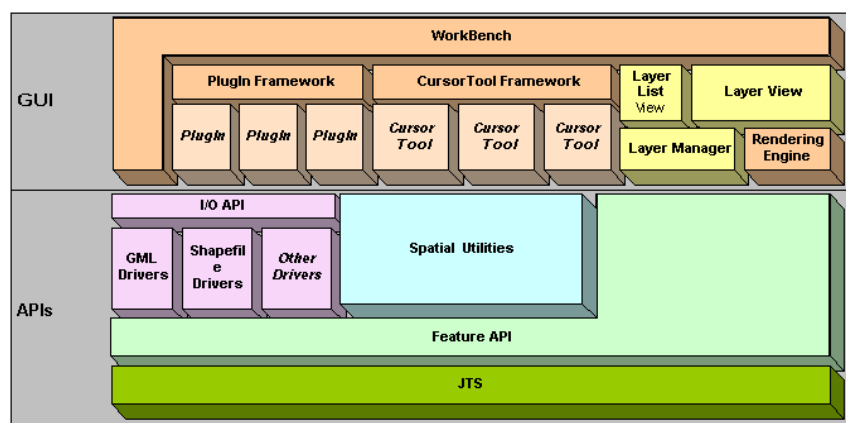
Projekt JCS využívá grafického uživatelské rozhraní projektu JUMP a také jeho API funkcí ke zpracování geografických dat. Na obrázku č. 18 je znázorněna architektura projektu JCS.



Obr. 18 architektura JCS [převzato z 3]

Architektura aplikace JUMP

Architektura aplikace JUMP je postavena na modularitě, mnohonásobné využitelnosti a přizpůsobitelnosti. Jádro funkcionality představuje soubor API funkcí jež jsou napsány tak, aby šly jednoduše využít v různých aplikacích. Právě včleněním těchto API funkcí do aplikace nezávisle na grafickém rozhraní je dosaženo vysokého stupně modularity. Díky tomu lze jak samotné API funkce tak grafické uživatelské rozhraní jednoduše využít i v jiných aplikacích.



Obr. 19 Architektura aplikace JUMP [převzato z 5]

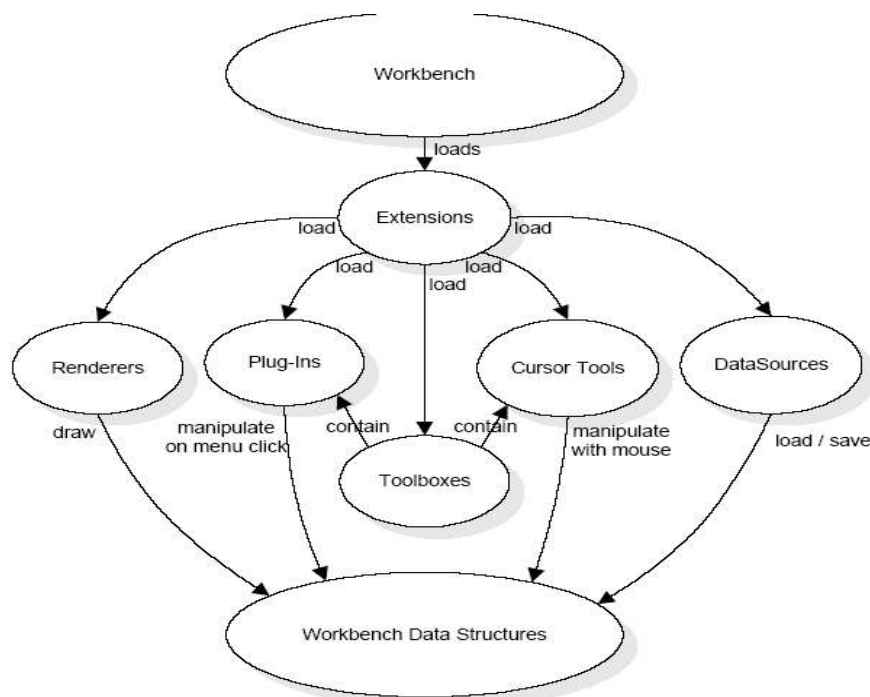
Rozšířitelnost aplikace JUMP

Nové moduly pro rozšíření lze jednoduše přidávat a navíc tyto moduly mohou spolupracovat s ostatními.

K dispozici je již mnoho existujících rozšíření např. čtení rastrových dat, konektor na ArcIMS server [12], Find Route, PostGIS connector [11]. Všechny externí rozšíření se snaží pod jednu střechu umístit projekt **JUMP Pilot Project** [2].

Následující prvky umožňují pro snadnou rozšířitelnost aplikace:

- Systém pro rozšiřování nabídky hlavního menu o položky spouštějící přidané pluginy
- Plugin nástrojové lišty pro rychlé spouštění pluginů
- Pluginy mohou využít kurzorových dotazů
- Datové zdroje mohou být rozšířeny o nové formáty
- Pluginy uloženy do formátu jar spouští aplikace automaticky



Obr 20. Vazby mezi JUMP Workbench a rozšířeními [převzato z 13]

Příklad vytvoření pluginu

Aby byla aplikace schopna komunikovat s jednotlivými funkčními částmi (pluginy) musí každý plugin obsahovat třídu *PluginNamePlugIn*, která rozšiřuje abstraktní třídu *AbstractPlugIn*.

Třída *AbstractPlugIn* definuje dvě abstraktní metody *initialize* a *execute* jež musí každý plugin implementovat:

- Metoda *initialize* definuje název položky z hlavního menu aplikace, která spouští plugin.
- Metoda *execute* vykonává vlastní funkci pluginu.

Kromě třídy plugin se musí vytvořit rovněž třída rozšiřující třídu *Extension*, která se stará o zavedení pluginu do aplikace JUMP.

Pokud plugin vykonává časově náročné operace měl by místo třídy *AbstractPlugIn*

implementovat rozhraní *ThreadedPlugIn*. Rozhraní definuje metodu *Run*, pomocí níž je chod pluginu spuštěn ve zvláštním vlákně a po dobu své činnosti neblokuje chod grafického rozhraní.

HelloWord Plugin

Jednoduchým příkladem, který byl převzat z [13], je příklad vypsání textu HelloWorld do okna aplikace JUMP Workbench. Kód č. 1 představuje soubor HelloWordPlugIn.java a kód č. 2 představuje soubor MyExtension.java.

```
package example;
import com.vividsolutions.jump.workbench.plugin.AbstractPlugIn;
import com.vividsolutions.jump.workbench.plugin.PlugInContext;

public class HelloWordPlugIn extends AbstractPlugIn {

    public void initialize(PlugInContext context) throws Exception{
        context.getFeatureInstaller().addMainMenuItems(this, new String
[] { "Tools", "Test" }, getName(), false, null, null);
    }

    public boolean execute(PlugInContext context) throws Exception {
        context.getWorkbenchFrame().getOutputFrame().createNewDocument
();
        context.getWorkbenchFrame().getOutputFrame().addText("Hello,
World!");
        context.getWorkbenchFrame().getOutputFrame().surface();
        return true;
    }
}
```

Kód 1 HelloWordPlugIn.java

```
package example;
import com.vividsolutions.jump.workbench.plugin.Extension;
import com.vividsolutions.jump.workbench.plugin.PlugInContext;

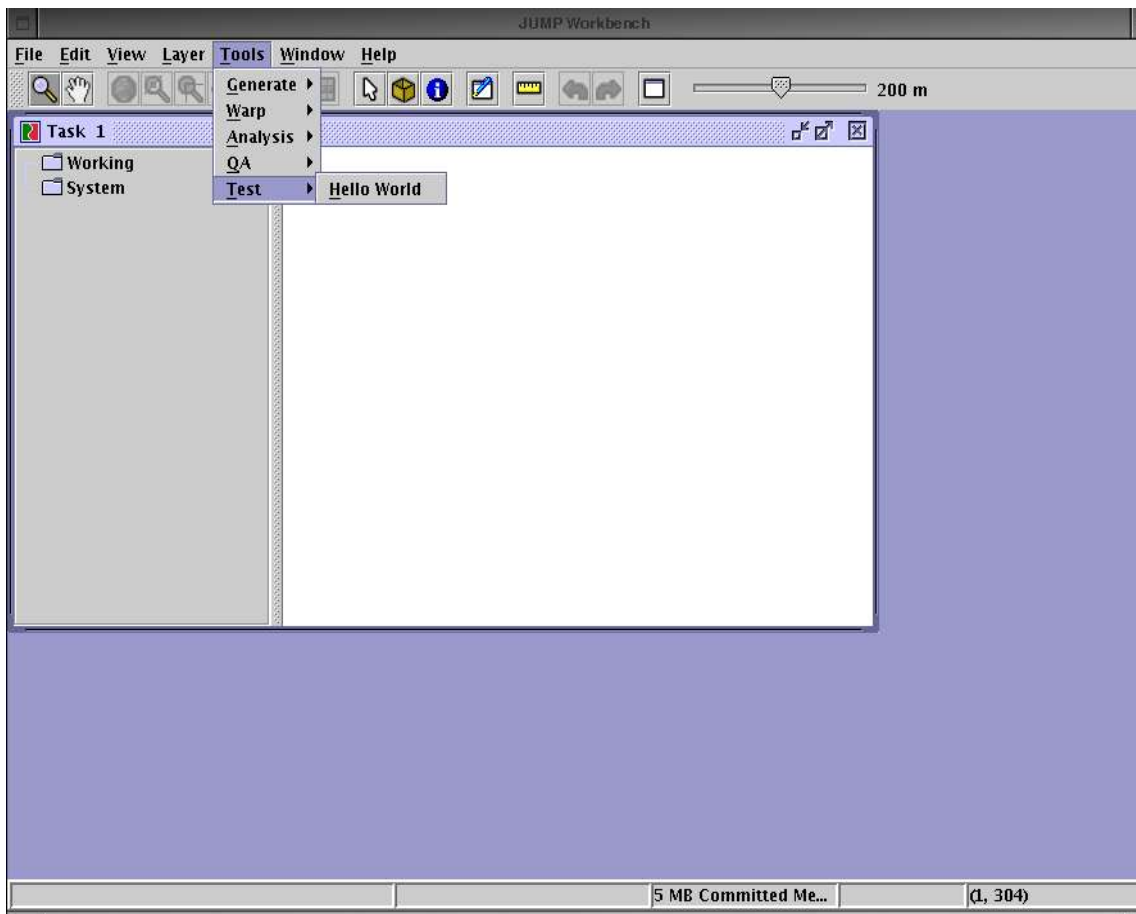
public class MyExtension extends Extension {
    public void configure(PlugInContext context) throws Exception {
        new HelloWordPlugIn().initialize(context);
    }
}
```

Kód 2 MyExtension.java

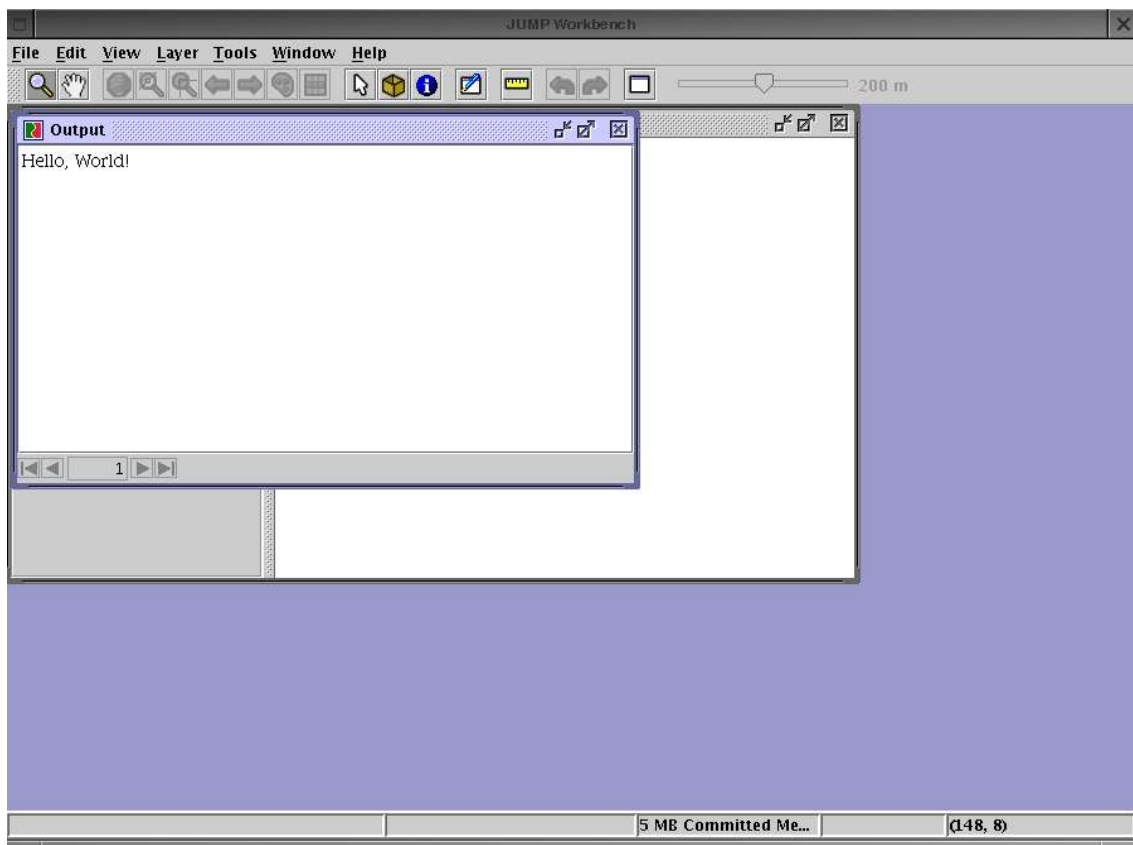
Kompilace, instalace a spuštění vytvořeného pluginu

Samotná kompilace i instalace je velmi jednoduchá. V případě použití OS Linux k tomu stačí tři příkazy. O spuštění se pak postará samotný JUMP Workbench (*inicialize*) a uživatel (*execute*).

1. javac example/*.java
2. jar cvf example.jar example/*
3. cp example.jar jumppath/lib/ext/example.jar



Obr 21. Spuštění nového pluginu



Obr 22. Výstup z nového pluginu

Drobná úprava pro HelloWorldPlugIn

Upravený kód č. 3 vypíše do okna seznam vybraných geoprkvů (vybraných standardním nástrojem programu JUMP) a u každého uvede velikost plochy. Výsledek je zobrazen na obr. č. 23

```
package example;
import com.vividsolutions.jump.workbench.plugin.AbstractPlugIn;
import com.vividsolutions.jump.workbench.plugin.PlugInContext;

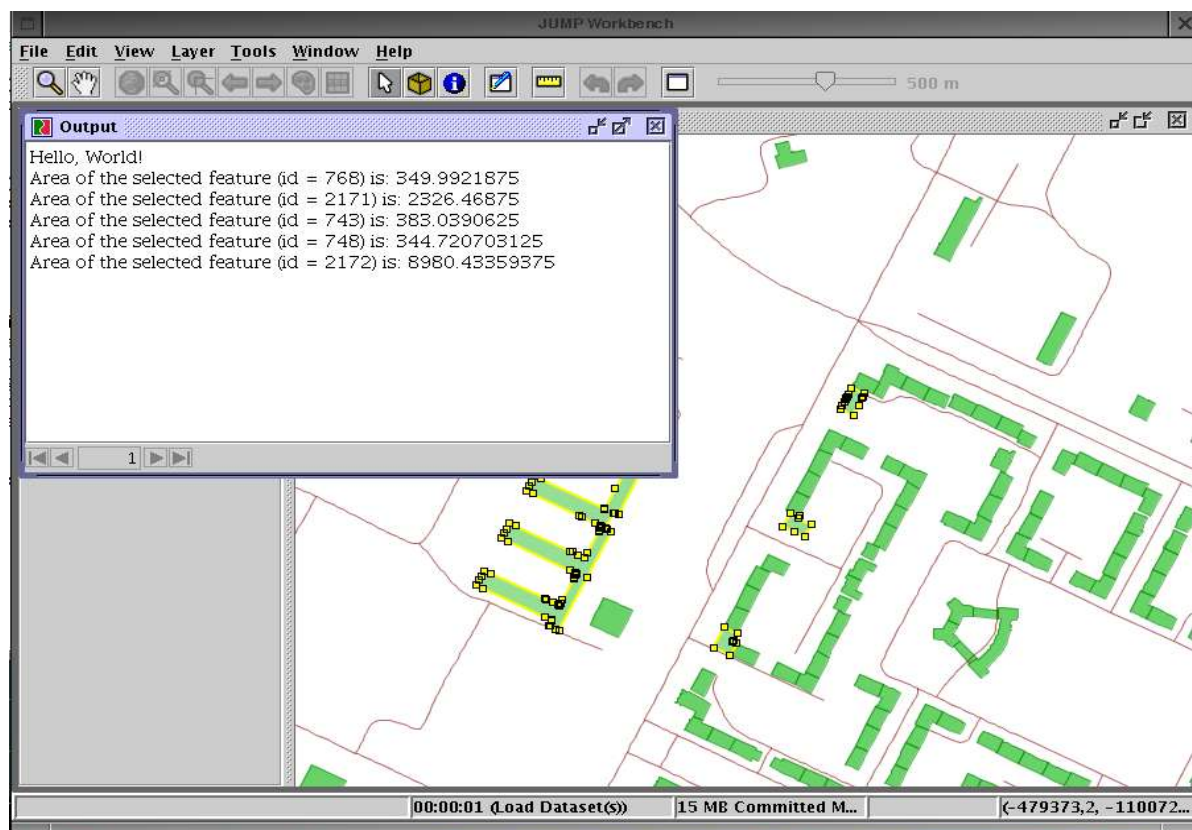
import com.vividsolutions.jts.geom.*;
import com.vividsolutions.jump.feature.*;
import com.vividsolutions.jump.workbench.ui.*;
import java.util.*;

public class HelloWorldPlugIn extends AbstractPlugIn {

    public void initialize(PlugInContext context) throws Exception{
        context.getFeatureInstaller().addMainMenuItems(this, new String
[] { "Tools", "Test" }, getName(), false, null, null);
    }

    public boolean execute(PlugInContext context) throws Exception {
        context.getWorkbenchFrame().getOutputFrame().createNewDocument
();
        context.getWorkbenchFrame().getOutputFrame().addText("Hello,
World!");
        String area;
        String id;
        Collection fcA = ((SelectionManagerProxy)
context.getActiveInternalFrame()).getSelectionManager().
createFeaturesFromSelectedItems();
        for (Iterator ia = fcA.iterator(); ia.hasNext(); ) {
            Feature fa = (Feature) ia.next();
            id = String.valueOf(fa.getAttribute(1));
            Geometry ga = fa.getGeometry();
            area = String.valueOf(ga.getArea());
            context.getWorkbenchFrame().getOutputFrame().addText
("Area of the selected feature (id = " + id + ") is: " + area);
        }
        context.getWorkbenchFrame().getOutputFrame().surface();
        return true;
    }
}
```

Kód 3 Upravený kód HelloWorldPlugIn.java



Obr 23. Výstup z nového pluginu

Závěr

Príspevek krátce popisuje projekt JUMP se zaměřením na popis možností nástroje JUMP Workbench. Výstupy z projektu JUMP jsou publikovány pod licenci **GPL** což umožňuje jejich snadné získání a úpravu. Nástroj JUMP Workbench je sice v začátcích vývoje, ale již dnes je možné jej využít pro projekty menšího rozsahu, případně pro výuku problematiky GIS.

Otevřená koncepce projektu se zaměřením na modularitu dává naději, že bude i nadále rozšiřována jeho funkcionalita a nový projekt **Jump Pilot Project** dává naději, že se bude možné ze strany uživatele v záplavě vznikajících pluginů vyznat.

Věříme, že projekt JUMP může být důstojným doplňkem kolekce open source nástrojů pro oblast GIS.

Poděkování

Děkujeme Magistrátu města Ostrava za provozování webových mapových služeb. Pro demonstraci funkcionality nástroje JUMP Workbench bylo využito dat ze systému GISMO, které jsou dostupné na [7].

Dále děkujeme všem, kteří se zasloužili o používání open source nástrojů.

Literatura

1. ISO/TC 211: ISO/DIS 19128. ISO/TC 211 Secretariat, Geneva, Switzerland, 2004, 83 s. Dostupný na WWW: <http://portal.opengis.org/files/?artifact_id=5316&version=1>
2. JPP Team. JUMP Pilot Project. 2005. Dostupné na WWW: <<http://jump->

pilot.sourceforge.net/index.php>

3. JUMP Project ORG. JCS Conflation Suite. 2005. Dostupné na WWW: [<http://www.jump-project.org/project.php?PID=JTS&SID=OVER#overview>](http://www.jump-project.org/project.php?PID=JTS&SID=OVER#overview)
4. JUMP Project ORG. JTS Topology Suite. 2005. Dostupné na WWW: [<http://www.jump-project.org/project.php?PID=JCS&SID=OVER#overview>](http://www.jump-project.org/project.php?PID=JCS&SID=OVER#overview)
5. JUMP Project ORG. JUMP Unified Mapping Platform. 2005. Dostupné na WWW: [<http://www.jump-project.org/project.php?PID=JUMP&SID=OVER>](http://www.jump-project.org/project.php?PID=JUMP&SID=OVER)
6. JUMP Project ORG. The JUMP Project. 2005. Dostupné na WWW: [<http://www.jump-project.org/>](http://www.jump-project.org/)
7. Městský Informační Systém, oddělení GIS. Mapový server Magistrátu města Ostravy. 2002 – 2005. Dostupné na WWW: [<http://gisova.mmo.cz/>](http://gisova.mmo.cz/)
8. MySQL AB. Well Known Text (WKT). Dostupné na WWW: [<http://dev.mysql.com/doc/mysql/en/gis-wkt-format.html>](http://dev.mysql.com/doc/mysql/en/gis-wkt-format.html)
9. OGC. Geography Markup Language. 2003. Dostupné na WWW: [<https://portal.opengeospatial.org/files/?artifact_id=7174>](https://portal.opengeospatial.org/files/?artifact_id=7174)
10. OGC. Simple Features - SQL. 1999. Dostupné na WWW: [<http://www.opengis.org/docs/99-049.pdf>](http://www.opengis.org/docs/99-049.pdf)
11. PostGIS ORG. PostGIS. 2004. Dostupné na WWW: [<http://www.postgis.org/>](http://www.postgis.org/)
12. Růžička J. a kol. JUMP Plugins from GISAK.VSB.CZ. 2005. Dostupné na WWW: [<http://gis.vsb.cz/ruzicka/Projekty/jump/index.php>](http://gis.vsb.cz/ruzicka/Projekty/jump/index.php)
13. Vivid Solutions, Inc. JUMP Developer's Guide. 2003. Dostupné jako součást dokumentace produktu na WWW: [<http://www.jump-project.org/>](http://www.jump-project.org/)
14. W3C. W3C Internationalization (I18N) Activity. 2005. Dostupné na WWW: [<http://www.w3.org/International/>](http://www.w3.org/International/)