An Interoperable Cartographic Database

Slobodanka KLJUČANIN[1], Zdravko GALIĆ[2]

[1]University of Sarajevo, Faculty of Civil Engineering, Institute for Geodesy and Geoinformatics
Stjepana Tomića 1, 71000 Sarajevo,
Bosnia and Herzegovina
slobodanka_kljucanin@gf.unsa.ba

[2]University of Zagreb,
Faculty of Electrical Engineering and Computing,
Department of Applied Computing
Unska 3, 10000 Zagreb, Croatia
zdravko.galic@fer.hr

***Abstract**: The concept of producing a prototype of interoperable cartographic database is explored in this paper, including the possibilities of integration of different geospatial data into the database management system and their visualization on the Internet. The implementation includes vectorization of the concept of a single map page, creation of the cartographic database in an object-relation database, spatial analysis, definition and visualization of the database content in the form of a map on the Internet.*

***Key words**: interoperability, database, cartography, geoinformation system (GIS)*

1 Introduction

The simple use and the possibility of accessing cartographic data using the Internet have changed the traditional way of map creation and map function. With the discovery of the Internet (Web), the number of geospatial data users has been expanded and so was the possibility of visualization according to users' specific needs. The Internet and the Web make the access to geospatial data and their visualization much easier, that is – they enable geoinformation systems to function in the Web environment. Maps traditionally represent an abstraction of a part of reality, namely the world that surrounds us, and which differ in appearance, content and scale (Kraak and Ormeling, 2003). If they are done well, the user may get series of information concerning the part of Earth's surface that the map represents. Acquisition of geospatial data, which are then stored in databases, is different from user to user, depending on personal needs (different attributes, different accuracy and data measurements). Using different tools for acquiring geospatial data, their processing and storing, the users try to make their everyday work somewhat easier and faster. But, with the use of different tools and different user's needs (which can partly overlap with other users) it was noticed how redundancy of activity and data appears and an unnecessary waste of time and money with it. It is possible to solve this problem by respecting the geoinformation interoperability concept.

The interoperable geoinformation systems (GIS) enable the integration and distribution of geospatial data from different sources as well as different spatial analysis of those geodata. The popularity of geoinformation systems, to which their simplicity of use on the Internet has contributed, significantly affects map creation and quality. The popularity and simplicity of GIS software usage led to more laics using GIS and popularisation of the possibility of map creation, by which geodata use is also popularised, but by which the problem of their quality and appearance emerged.

Popularity of the Internet contributed to the fact that millions of users exchange and distribute geospatial cartographic data, in the simplest way. The simple use and the possibility of accessing data using the Internet has changed the traditional way of map creation.

Interoperabilna kartografska baza podataka

Slobodanka KLJUČANIN[1], Zdravko GALIĆ[2]

[1]Univerzitet u Sarajevu, Građevinski fakultet,
Institut za geodeziju i geoinformatiku
Stjepana Tomića 1, 71000 Sarajevo,
Bosna i Hercegovina
slobodanka_kljucanin@gf.unsa.ba

[2]Sveučilište u Zagrebu,
Fakultet elektrotehnike i računarstva,
Zavod za primijenjeno računarstvo
Unska 3, 10000 Zagreb, Hrvatska
zdravko.galic@fer.hr

Sažetak: *U radu se razmatra koncept izrade prototipa interoperabilne kartografske baze podataka, mogućnosti integracije različitih geoprostornih podataka u sustav za upravljanje bazom podataka i njihova vizualizacija na internetu. Implementacija obuhvaća vektorizaciju sadržaja jednog lista karte, kreiranje kartografske baze podataka u objektno-relacijskoj bazi podataka, prostorne analize, definiranje i vizualizacije sadržaja baze podataka u obliku karte na internetu.*

Ključne riječi*: interoperabilnost, baza podataka, kartografija, geoinformacijski sustav (GIS)*

1. Uvod

Jednostavna uporaba kartografskih podataka i mogućnost pristupanja tim podacima putem interneta promijenili su tradicionalan način kreiranja i funkciju karte. Pojavom interneta, odnosno weba, dodatno je proširen broj korisnika geoprostornih podataka i mogućnosti vizualizacije sukladno specifičnim potrebama korisnika. Internet i web radikalno olakšavaju pristup geoprostornim podacima i njihovu vizualizaciju, odnosno omogućavaju funkcionalnosti geoinformacijskih sustava u web okolišu. Karte tradicionalno predstavljaju apstrakciju dijela realnosti, odnosno svijeta koji nas okružuje; razlikuju se po izgledu, sadržaju i mjerilu (Kraak i Ormeling, 2003). Ako su dobro oblikovane, korisnik može dobiti niz informacija o dijelu Zemljine površine koju karta prikazuje. Prikupljanje geoprostornih podataka, koji se potom pohranjuju u baze podataka, razlikuje se od korisnika do korisnika, zavisno od njegovih potreba (različiti atributi, različita točnost i mjerilo podataka). Uporabom različitih alata za prikupljanje geoprostornih podataka, obradu i pohranjivanje, korisnici nastoje olakšati i ubrzati svoj svakodnevni rad. Međutim, uporabom različitih alata i različitih potreba korisnika (koji se mogu djelomično preklapati s drugim korisnicima) uočeno je kako dolazi do redundancije aktivnosti i podataka, a samim time nepotrebno trošenja vremena i novca. Rješenje tog problema moguće je respektiranjem koncepta geoinformacijske interoperabilnosti.

Interoperabilni geoinformacijski sustavi (GIS) omogućavaju integraciju geoprostornih podataka iz različitih izvora i različite prostorne analize tih geopodataka. Popularnost geoinformacijskih sustava, čemu je pridonijela i njihova jednostavnost korištenja na internetu, znatno utječe na način kreiranja i na kvalitetu karata. Popularnost i jednostavnost korištenja GIS softvera doveli su do toga da se sve više laika koristi GIS-om i mogućnošću kreiranja karata, čime se popularizira korištenje geopodataka, ali javlja se problem u njihovoj kvaliteti i prikazu.

Popularnost interneta pridonijela je da milijuni korisnika na jednostavan način razmjenjuju i distribuiraju geoprostorne, odnosno kartografske podatke. Jednostavna upotreba podataka i mogućnost pristupanja podacima putem interneta promijenili su tradicionalan način kreiranja karte.

2 Interoperability of Geoinformation Systems

Different import and export functions were used for conversion and usage of geodata from different geoinformation systems before introduction of the concept of interoperability in context of GIS/DBMS[1]. A lot of limitations were spotted: loss of data during transmission, problems with their integrity, and the raise of program package complexity, as well as slower processing. The concept of interoperability enables more effective database accessing and usage of geodata by different users (who usually use different applications).

With the existence of different geoinformation technology parallel with the dynamic technological development, the awareness and demand for the interoperability rose, that is for "*complete geospatial data organization as well as geoprocessing resources and widely spread usage of interoperable software for geoprocessing through informatics infrastructure*". That *Open Geospatial Consortium* (OGC) vision has significantly influenced the activities in the sphere of interoperability and led to the specification of the geospatial information exchange technology, enabling geospatial services using the Internet, mainly Web technology.

Interoperability is the ability of communicating, completing the programs or data transfer between different functional units in a way that it doesn't require from the user any kind or amount of knowledge about specific characteristics of those units. Two units (Fig. 2.1) are interoperable if X can send an understandable request Z for services to Y, and if Y can similarly return an answer O to X.

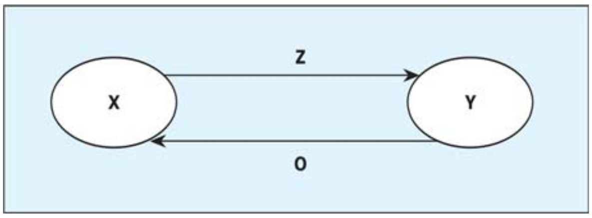


Fig. 2.1. Interoperability (Galić, 2006)
Slika 2.1. Interoperabilnost (Galić, 2006)

Geospatial interoperability is the ability of geoinformation systems to:

1. Freely exchange all geospatial data,
2. Together with computer networks execute program modules that manipulate geospatial data (Galić, 2006).

Geospatial data interoperability incorporates a mutual standard for data exchange and for reducing the number of operations necessary for exchange of different types of data. Some prerequisites have to be satisfied in order to be able to use geoinformation interoperability (Ključanin, 2006).

2.1 Relevant standards

There are many organizations which work on growth of geoinformation standards, and some of those are: *Open Geospatial Consortium* (OGC) and *International Organization for Standardization* (ISO) – *Technical Committee* (TC211), on the basis of whose directions national organizations develop their own geoinformation standards.

OGC initiates the main directions of the development of geoinformation interoperability - the integration and exchange of geospatial data from different sources. The new OpenGIS specification defines a comprehensive software framework for distributed access to geodata and geoprocessing resources. The OpenGIS specification includes an abstract specification and a series of implementation specifications for various *Distributed Computing Platform* (DCP), such as *Common Object Request Broker Architecture* (CORBA), *Object Linking and Embedding/ Component Object Model* (OLE/COM), *Structured Query Language* (SQL) and Java. ISO/TC211 is working on a series of international standards (from ISO 19101 to 19135) that focus on geographic information. The ISO geographic information standard (formally known as the ISO 15046 standard) specifies methods, tools, and services for data management, processing, analysing, accessing, presenting, and transferring geospatial data in digital form between users, systems and locations (Peng and Tsou, 2003).

The recommended OGC geoinformation interoperable model is structured in two parts:

a) *Abstract specification* – is structured in three parts:
   1. *Geospatial data model* (an abstract data model specified by a set of basic data types),
   2. *Service model* (works with operations which define data in the model),
   3. *Common model* (defines semantic interoperability).

b) *Implementation specifications* – define the way of defining the abstract specification in DCP.

OGC has published many specifications, among which are:

- *Simple Features Specification for CORBA*
- *Simple features for SQL* (OGC 2005a)
- *OGC Web Map Service Interfaces Implementation Specification* (ISO 19128, 2005)
- *OGC Web Feature Service Interfaces Implementation Specification* (OGC, 2005c)
- *OGC Geography Markup Language Implementation Specification.* (OGC, 2005b), (ISO/DIS 19136, 2005)

[1] ***Data**base **M**anagement **S**ystem*

2. Interoperabilnost geoinformacijskih sustava

Prije uvođenja koncepta interoperablnosti u GIS/ DBMS[1] kontekstu, za konverziju i uporabu geopodataka iz različitih geoinformacijskih sustava uglavnom su se koristile različite *import* i *eksport* funkcije. Pritom je uočen niz nedostataka: gubitak podataka tijekom prijenosa, problemi njihova integriteta, povećanje kompleksnosti programskih paketa, kao i sporiji rad. Koncept interoperabilnosti omogućava različitim korisnicima (koji obično koriste različite aplikacije) djelotvorniji pristup bazama podataka i uporabu geopodataka.

Uz postojanje različitih geoinformacijskih tehnologija, usporedno s dinamičnim tehnološkim razvojem, rasli su i spoznaja i zahtjevi za interoperabilnošću, tj. za "*potpunom integracijom geoprostornih podataka i resursa za geoprocesiranje i široko rasprostranjenim korištenjem interoperabilnog softvera za geoprocesiranje kroz informatičku infrastrukturu*". Ta vizija *Open Geospatial Consortiuma* znatno je utjecala na aktivnosti u području interoperabilnosti i dovela do specifikacije tehnologija za prijenos geoprostornih informacija i osiguranje geoprostornih usluga putem internetske odnosno web tehnologije.

Interoperabilnost je sposobnost komuniciranja, izvršenja programa ili prijenosa podataka između različitih funkcionalnih jedinica na način koji od korisnika ne zahtijeva nikakvo, odnosno zahtijeva minimalno znanje o specifičnim karakteristikama tih jedinica. Dvije su jedinice (slika 2.1) interoperabilne ako X može poslati međusobno razumljiv zahtjev Z za usluge prema Y, i ako Y može slično vratiti međusobno razumljiv odgovor O prema X.

Geoprostorna interoperabilnost je sposobnost informacijskih sustava da:

1. slobodno razmjenjuju sve vrste geoprostornih podataka
2. zajedno, putem računalnih mreža, izvršavaju programske module koji manipuliraju geoprostornim podacima (Galić, 2006).

Interoperabilnost geoprostornih podataka podrazumijeva zajednički standard za prijenos podataka i smanjenje broja operacija potrebnih za razmjenu različitih vrsta podataka. Kako bi se uopće mogla rabiti geoinformacijska interoperabilnost, moraju biti zadovoljeni određeni preduvjeti: odgovarajući standardi, zadovoljavajuća podrška tehnologije i interoperabilno procesiranje (Ključanin, 2006).

2.1. Relevantni standardi

U svijetu postoji niz organizacija koje se bave razvojem geoinformacijskih standarda, među kojima izdvajamo: *Open Geospatial Consortium* (OGC) i *International Organization for Standardization* (ISO) – *Tehnical Commitee* (TC211), na osnovi čijih uputa nacionalne organizacije razvijaju svoje geoinformacijske standarde.

OGC inicira osnovne smjerove razvoja geoinformacijske interoperabilnosti, odnosno integracije i prijenosa geoprostornih podataka iz različitih izvora. OGC objavljuje specifikacije koje definiraju razumljivi softverski radni okvir za distribucijski pristup geopodacima i izvore obrade geoinformacija. Te specifikacije uključuju apstraktnu specifikaciju i niz implementacijskih specifikacija za radne *Distributed Computing Platform* (DCP), kao što su *Common Object Request Broker Architecture* (CORBA), *Object Linking and Embeding/ Component Object Model* (OLE/COM), *Structured Query Language* (SQL) i Java. ISO/TC211 radi na seriji međunarodnih normi koje se odnose na geoinformacije, odnosno specificiraju metode, alate i servise za upravljanje podacima, njihovu obradu, analiziranje, pristup, prezentaciju i prijenos u digitalnoj formi između korisnika, sustava i položaja (Peng and Tsou, 2003).

Preporučeni geoinformacijski interoperabilni model OGC-a sastoji se od dva dijela:

a) *apstraktne specifikacije* – koja se sastoji se od tri dijela:
   1. *modela geoprostornih podataka* (apstraktni model podataka koji specificira skup osnovnih tipova podataka),
   2. *servisnog modela* (zadužen za operacije koje definiraju podatke u modelu),
   3. *zajedničkog modela* (određuje semantičku interoperabilnost).

b) *implementacijske specifikacije* – određuju način na koji će se definirati apstraktna specifikacija u DCP-u.

OGC je objavio niz specifikacija među kojima i

- *Simple Features Specification for CORBA*
- *Simple features for SQL* (OGC 2005a)
- *OGC Web Map Service Interfaces Implementation Specification* (ISO 19128, 2005)
- *OGC Web Feature Service Interfaces Implementation Specification* (OGC, 2005c)
- *OGC Geography Markup Language Implementation Specification.* (OGC, 2005b), (ISO/DIS 19136, 2005)

2.2. Tehnologije za geoinformacijsku interoperabilnost

Tehnologija potrebna za postizanje geoinformacijske interoperabilnosti može se podijeliti na tri osnovne skupine: softverske komponente, distribuiranu računalnu platformu i mrežnu arhitekturu.

Softverske komponente rabimo u kreiranju aplikacija. Njih kreiraju različiti proizvođači koji koriste različite programske jezike i operacijske sustave. Kako bi funkcionirali zajedno, potrebno je specificirati pravila na osnovi kojih komponente mogu komunicirati, tj. potrebno je da koriste DCP. Za pristupanje klijenata bazama podataka najčešće se koristi troslojna arhitektura (slika 2.2).

[1] Skraćenica za *Database Management System* (*engl.*) - sustav za upravljanje bazom podataka.

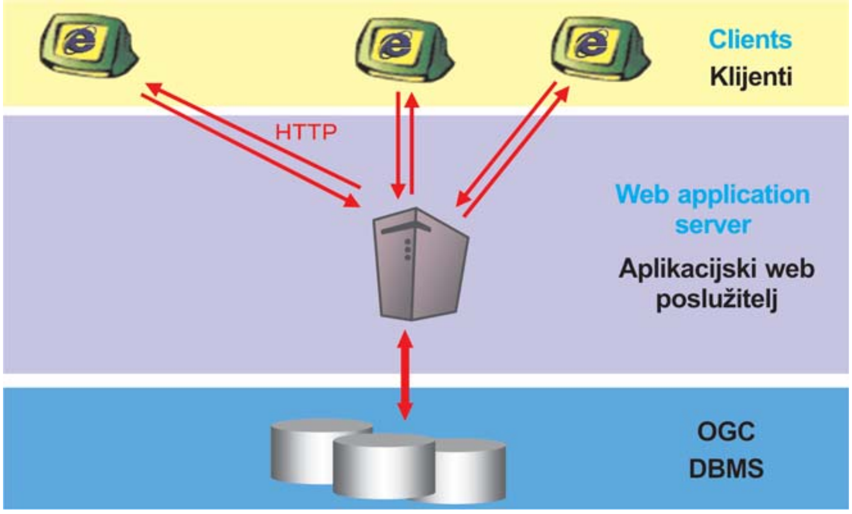


Fig. 2.2. Architecture of three-tier model (Galić, 2004)

Slika 2.2. Arhitektura troslojnog modela (Galić, 2004)

2.2. Technologies for geoinformation interoperability

The technology needed for achieving geoinformation interoperability can be divided into three main groups: software components, distributed computing platforms and network architecture.

Software components are used to create applications. They are created by different producers that use different program languages and operation systems. To make them work together, they need to specify rules on the basis of which the components can communicate, i.e. they need to use DPC. A three-tier architecture is used for client access to database (Fig. 2.2).

For communicating and understanding the components, OGC specifies the work frame, necessary for geoprocessing and semantic interoperability. The main goal of interoperable data processing is to have a geoinformation system that is composed of components, which function as a unique solution. Geoprocessing interoperability includes data access service (catalogue service which offers metadata about geospatial data and objects) and processing data. Interoperable geospatial data processing is enabled by the OGC service model that defines the operation performed on geospatial data (Ključanin, 2006).

2.3 Distribution of geodata over the Internet

The popularity of the Internet and its general usage in everyday life has suddenly risen with the appearance of Web (*World Wide Web*) browsers. Web browsers are based on the hypertext model. With the creation of the standard model and protocol, a wide range of users is given the possibility of undisturbed data access. *Hypertext Transfer Protocol* (HTTP) enables the communication between the Web server and the user, although they use different platforms. That understanding is achieved with the use of the *Hypertext Markup Language* (HTML). Geospatial data are more complex type of data that are stored in the computers, that is in the database and the integration of these heterogeneous data is a challenge and a task of the distributed GIS (Ključanin, 2006).

Considering the way of communication, the distributed geoinformation system could be classified into WebGIS and mobile GIS. WebGIS is composed of four parts: a client, a Web server with an application server, a map server and a data server.

The client usually has the following functions: presents an interface for user interaction with data and functions, formats requests for data or services from a server, and displays data or query results it receives from the server. To increase user interactivity and help users to interact directly with spatial objects on a map, alternative clients that use Web add-ons have been developed. These interactive clients include dynamic HTML and client-side applications such as plug-ins or help programs, Java applets or Java beans and ActiveX Controls (Peng and Tsou, 2003).

The Web server sends a response to the client request - it serves HTML pages and other information. The responses this server sends to the client can be divided into three main types:

- Sending prepared HTML documents or maps,
- Sending Java *applet* and ActiveX *control*,
- Running other programs for query processing.

When the *Web server* gets the request from *the client*, it calls the *application server* service as a translator or a link between the Web server and map server. The main function of application server is to establish, support and terminate relations between these two servers. Its goal is the interpretation of the client's request that is directed to the map server and controls parallel requests. Besides that, it balances the load between the map server and the data server, and manages the transaction and security aspects (Ključanin, 2006).

U svrhu komuniciranja i razumijevanja komponenti, OGC specificira radni okvir, potreban za postizanje tzv. *geoprocesne* i *semantičke interoperabilnosti*. Osnovni je cilj interoperabilne obrade podataka posjedovanje vlastitog geoinformacijskog sustava sastavljenog od komponenti koje funkcioniraju kao jedinstveno rješenje. Geoprocesna interoperabilnost podrazumijeva servisiranje pristupa podacima (katalоški servis koji nudi metapodatke o geoprostornim podacima i objektima) i obrade podataka. Interoperabilna obrada geoprostornih podataka omogućena je preko OGC servisnog modela koji definira operacije nad geoprostornim podacima (Ključanin, 2006).

2.3 Distribuiranje geopodataka na internetu

Popularnost interneta i njegova opća upotreba u svakidašnjem životu naglo se povećala s pojavom web (*World Wide Web*) preglednika. Web preglednici temelje se na modelu hiperteksta. Stvaranjem standardnog modela i protokola omogućeno je širokom krugu korisnika neometano dobivanje informacija. *Hypertext Transfer Protocol* (HTTP) omogućava komunikaciju između web poslužitelja i korisnika, i unatoč tome što koriste različite platforme. To razumijevanje ostvaruje se upotrebom HTML-a (*Hypertext Markup Language*). Geoprostorni podaci smatraju se jednim od najkompleksnijih tipova podataka koji se čuvaju u računalima, odnosno bazama podataka, a integracija heterogenih podataka je i izazov i zadaća distribuiranoga GIS-a (Ključanin, 2006).

S obzirom na način komunikacije, distribuirani geoinformacijski sustav dijeli se na *WebGIS*[2] i *mobilni GIS*. WebGIS se sastoji od četiri komponente: klijenta, web poslužitelja s aplikacijskim poslužiteljem, map poslužitelja i poslužitelja podataka.

Klijent izvodi interakciju korisnika s podacima i funkcijama te vizualizira podatke i rezultate analiza i upita. Kod distribucije podataka na internetu klijenti su web preglednici čiji je zadatak interpretiranje sadržaja HTML dokumenata koje je poslao web poslužitelj, te njihov grafički prikaz. Kako bi se premostila ograničenja standardnog HTML preglednika i korisnicima omogućila izravna interakcija s objektima na karti, razvijeni su klijenti koji koriste web dodatke (dinamični HTML i aplikacije na klijentovoj strani). *Dinamični HTML (DHTML)* na klijentovoj strani su: plug-in, Java applet, Java beans i ActiveX control (Peng and Tsou, 2003).

Web poslužitelj šalje odgovor na zahtjev *klijenta,* odnosno poslužuje HTML stranicama i drugim informacijama. Odgovori koje taj poslužitelj šalje *klijentu* mogu se podijeliti na tri osnovna tipa:

- slanje pripremljenih HTML dokumenata ili planova/karata,
- slanje Java *applet* i ActiveX *control*,
- pozivanje drugih programa za obradu upita.

[2] WebGIS je na webu utemeljeni GIS (Peng i Tsou, 2003).

Kada w*eb poslužitelj* dobije zahtjev od *klijenta* koji je u njemu nepoznatom programu, on poziva servis *aplikacijskog poslužitelja. Aplikacijski poslužitelj* djeluje kao prevodilac ili povezivač između web poslužitelja i map poslužitelja. Glavna mu je funkcija uspostavljanje, održavanje i prekidanje veza između tih dvaju poslužitelja. Zadaća mu je interpretacija zahtjeva klijenata, koje upućuje map poslužitelju, i upravljanje paralelnim zahtjevima. Uz to, balansira opterećenjem između map poslužitelja i poslužitelja podataka, te upravlja transakcijskim i sigurnosnim aspektima (Ključanin, 2006).

Map poslužitelj obrađuje prostorne upite (koje prosljeđuje web poslužitelj), generira i distribuira karte na zahtjev klijenta. Izlazni podaci tih funkcija mogu biti filtrirani podaci koji se šalju klijentu, statična karta (GIF[3], JPEG[4], …) ili podaci u vektorskom obliku (SVG[5], GML[6], …).

Poslužitelj podataka *poslužuje prostorne i tematske podatke spremljene u bazi podataka. Obično se koristi objektno-relacijski ili objektni sustav baze podataka, a podacima se pristupa putem:* Object Database Connectivity *(ODBC),* Java Database Connectivity *(JDBC),* Java Data Objects *(JDO),* Object Linking and Embeding Database *(OLE DB) i* ActiveX Data Object *(ADO) (Galić, 2006).*

Veliku ulogu u implementiranju pristupa, upravljanja, manipulacije, predstavljanja i razmjene geopodataka između informacijskih zajednica imaju *Open Geospatial* servisi (Peng i Tsou, 2003).

Web servisi su komponente raspoložive u distribucijskom okruženju (internet, LAN[7] i sl.) koje mogu izvesti određene zadatke, a dizajnirani su tako da aplikacije komuniciraju bez pomoći ljudi. Web servisi predstavljaju arhitekturu (*Service Oriented Architecture* – SOA), a ne određeni skup tehnologija, i funkcioniraju po načelu *zahtjeva* i *odgovora*. Web servisi rješavaju problem nekompatibilnosti različitih formata podataka i računalnih sustava skupom softverskih standarda (XML[8], SOAP[9], UDDI[10], WSDL[11]), koji omogućavaju definiranje, pristupanje i izvođenje operacija nad podacima, pri čem nije potrebno poznavati implementirane tehnologije (Ključanin, 2006).

3. Implementacija kartografske baze podataka

Na primjeru jednog lista topografske karte Bosanska Bijela korišten je model podataka definiran u sklopu *Projekta digitalne topografske karte Bosne i Hercegovine*, koji financira i vodi Japanska agencija za međunarodnu suradnju (JICA), u suradnji s Ministarstvom civilnih

[3] engl. GIF – *Graphics Interchange Format*
[4] engl. JPEG – *Joint Photographic Experts Group*
[5] engl. SVG – *Scalable Vector Graphics*
[6] engl. GML – *Geografpy Markup Language*
[7] engl. LAN – *Local area network*
[8] engl. XML – *eXtensible Markup Language*
[9] engl. SOAP – *Simple Object Access Protocol*
[10] engl. UDDI – *Universal Description, Discovery nad Integration*
[11] engl. WSDL – *Web Services Description Language*

The map server processes spatial queries (sent from the Web server), generates and distributes the maps on client request. Filtered data resulting from these functions can be sent to the client, static map (GIF[2], JPEG[3], …) or vector data (SVG[4], GML[5], …).

The data server serves spatial and thematic data stored in a database. An object-relational or object-oriented database system is used, and the data are accessed through: *Object Database Connectivity* (ODBC), *Java Database Connectivity* (JDBC), *Java Data Objects* (JDO), *Object Linking and Embedding Database* (OLE DB) and *ActiveX Data Object* (ADO) (Galić, 2006).

OpenGIS services (Peng and Tsou, 2003) define the set of services needed to access and process geodata defined in OGM[6] and provide the capabilities for sharing geodata with the GIS community. Web services are components available in a distributing environment (Internet, LAN[7], etc.) that can perform certain tasks, and they are designed in a way the application can communicate without human intervention. Web services represent the architecture (*Service Oriented Architecture* – SOA), not a certain set of technology, and they work on the *request* and *response* principle. Web services solve the problem of incompatibility of different data formats and computer systems by collection of software standards (XML[8], SOAP[9], UDDI[10], WSDL[11]), which enable the definition, access and performing operations on data, without the need to know the implemented technology (Ključanin, 2006).

3 Cartographic Database Implementation

Single topographic map sheet Bosanska Bijela and data model defined within the project of *Digital Topographic Mapping of Bosnia and Herzegovina* were used. The project was led and financed by the Japanese Agency for International Cooperation (JICA) in cooperation with the Ministry of Public Affairs of B&H, Ministry of Foreign Affairs of B&H, Republic Administration for Geodesy and Property-Law Affairs of the Republic of Serbia, Administration for Geodesy and Property-Law Affairs of the Federation of Bosnia and Herzegovina and the Public Government Register of the Brčko District (Ključanin et al. 2006). Logical model of a cartographic database is an object-relational model conformal with SQL:1999 standard. The geospatial analysis is performed by using a set of predefined geometric–topologic functions: *within distance, contain, nearest neighbour, distance, buffer, overlay, etc.* (Galić, 2006).

Oracle Spatial (Kothuri et al., 2004) is an object-relational database, developed according to the (OGC, 2005a) standard – the base standard for SQL schema for storing, accessing, querying, and management of spatial data.

The object-relational model uses tables in which one column stores geometrical attributes. *Oracle Spatial* generic SDO_GEOMETRY type (Figure 3.1) models spatial data occurring in most spatial applications and is conformant with the Open GIS Consortium (OGC) geometry model (Kothuri et al., 2004). Instances of one object class are stored into a database in a single object-relational table, and data in the database are accessed by using standard SQL statements. The database implements well-known textual formats for spatial data, which means geospatial data are stored and accessed very easily. Different applications and different users are able to store and access data (Kothuri et al., 2004), and the goal of database interoperability has been reached.

A map sheet of Bosanska Bijela (Military-Geographic Institute, Belgrade 1975) is used for developing the prototype of the cartographic database. Map scanning was performed by a rotating colour scanner with resolution of 400 dpi. Georeferencing of the scanned map is done in the *CAD Overlay* program package, with the use of the polynomial transformation[12] of the fourth degree.

The *AutoCAD* was used for vectorization of the content, and after that the data are visualized in the *Geomedia Professional* environment. That GIS software enables combining geodata from different sources, different formats and different projections and performs queries over spatial and thematic attributes of object classes. Every cartographic object is defined by a unique identification number (ID) and attributes. Attributes of cartographic objects are taken from an analogue map (example: width and type of the road, type and thickness of the woods, height of geodetic points). The data are exported from the *AutoCAD* format into the Oracle object-relational model, using *Geomedia Professional* functions *Export to Oracle Object Model*. Five different files are generated for every object class (*.DAT – the actual geometrical and thematic data, *.LOG – information about data export, *.PRE – table creation (SQL statement 1), *.CTL – control file, *.POS – metadata import.)

SQL statement No. 1:

```
CREATE TABLE BUNAR
(ID             NUMBER(10,0) PRIMARY KEY,
SPATIALPOINT    MDSYS.SDO_GEOMETRY,
VRSTA_BUNARA    VARCHAR2(50));

CREATE TABLE DALEKOVOD
(ID             NUMBER(10,0) PRIMARY KEY,
SPATIALLINE     MDSYS.SDO_GEOMETRY
NAPON           NUMBER(6,0));
```

[2] GIF – *Graphics Interchange Format*
[3] JPEG – *Joint Photographic Experts Group*
[4] SVG – *Scalable Vector Graphics*
[5] GML – *Geograpfy Markup Language*
[6] OGM – *Open Geodata Model*
[7] LAN – *Local area network*
[8] XML – *eXtensible Markup Language*
[9] SOAP – *Simple Object Access Protocol*
[10] UDDI – *Universal Description, Discovery nad Integration*
[11] WSDL – *Web Services Description Language*

[12] There is no official regulation for map transformation in B&H, therefore polynomial transformation for cadastre's maps was used.

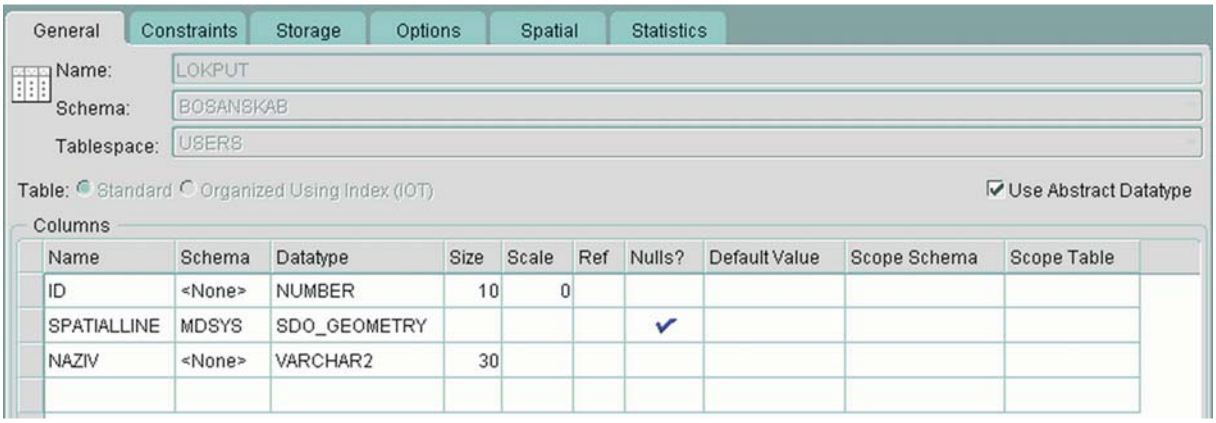


Fig. 3.1. Object-relational table for class "local road"

Slika 3.1. Objektno-relacijska tablica za klasu "lokalni put"

poslova BiH, Ministarstvom vanjskih poslova BiH, Republičkom upravom za geodetske i imovinsko-pravne poslove Republike Srpske, Upravom za geodetske i imovinsko-pravne poslove Federacije Bosne i Hercegovine i Javnim registrom vlade distrikta Brčko (Ključanin i dr. 2006). Logički model kartografske baze podataka je objektno-relacijski model, suglasan s normom SQL:1999. Geoprostorna analiza izvodi se uporabom unaprijed definiranih geometrijsko-topoloških funkcija: *within distance, contain, nearest neigbor, distance, buffer, overlay,* itd. (Galić, 2006).

Oracle Spatial (Kothuri i dr., 2004), je objektno-relacijska baza podataka, razvijena sukladno standardu (OGC, 2005a). Na njegovoj osnovi specificira se standardna SQL shema koja podržava pohranjivanje i pristup podacima, postavljanje prostornih upita i mijenjanje prostornih podatka.

Objektno-relacijski model koristi tablice, u kojima jedan stupac čuva vrijednosti geometrijskih atributa. Generički tip SDO_GEOMETRY *Oracle Spatiala* (slika 3.1) omogućava modeliraje prostornih podataka koji se javljaju u većini prostornih aplikacija, suglasan je s geometrijskim modelom OGC-a. Primjerci jedne objektne klase spremaju se u bazu podataka u objektno-relacijsku tablicu, podacima u bazi pristupa se s pomoću standardnih SQL iskaza. Baza podataka implementira dobro poznate tekstualne formate za specificiranje prostornih podataka, što znači da se takvi podaci mogu spremati i uzimati na jednostavan način. Time je različitim aplikacijama/korisnicima omogućeno pohranjivanje podataka i pristup podacima (Kothuri i dr., 2004), čime je ostvarena interoperabilnost baze podataka.

Za izgradnju prototipa kartografske baze podataka korišten je list karte Bosanska Bijela (Vojno-geografski institut, Beograd 1975). Skeniranje karte izvedeno je na rotacijskom skeneru u boji, rezolucije 400 dpi. Georeferenciranje skenirane karte izvedeno je u programskom paketu *CAD Overlay*, uz uporabu polinomske[12] transformacije četvrtog stupnja.

[12] Kako u BiH ne postoji pravilnik o transformaciji topografskih karata, odabrana je polinomska transformacija, po uzoru na transformaciju katastarskih planova.

Za vektorizaciju sadržaja korišten je softver *AutoCAD*, a podaci su potom vizualizirani u okolišu *GeoMedia Professional*. Taj GIS softver omogućava kombiniranje geopodataka iz različitih izvora, različitih formata i različitih projekcija, te izvođenje upita nad prostornim i tematskim atributima objektnih klasa. Nakon vektoriziranja sadržaja karte, svaki je kartografski objekt definiran identifikacijskim brojem (ID) i atributima. Atributi kartografskih objekata preuzeti su s analogne karte (npr. širina i tip kolnika, vrsta i gustoća šume, nadmorska visina geodetskih točaka). Podaci su iz AutoCAD-a prenijeti u Oraclov objektno-relacijski model, uporabom funkcije *Export to Oracle Object Model* Geomedie Professionala. Za svaku objektnu klasu generira se pet različitih datoteka (**.DAT* – aktualni geometrijski i tematski podaci, **.LOG* - informacije o izvozu podataka, **.PRE* – kreiranje tablice (SQL iskaz br. 1), **.CTL* - kontrolna datoteka, **.POS* - unos metapodataka u bazu).

SQL iskaz br. 1:

```
CREATE TABLE BUNAR
(ID               NUMBER(10,0) PRIMARY KEY,
SPATIALPOINT      MDSYS.SDO_GEOMETRY,
VRSTA_BUNARA      VARCHAR2(50));

CREATE TABLE DALEKOVOD
(ID               NUMBER(10,0) PRIMARY KEY,
SPATIALLINE       MDSYS.SDO_GEOMETRY
NAPON             NUMBER(6,0));
```

U bazi podataka oblikuje se novi tablični prostor i korisnik, te se definiraju njegova uloga i privilegiji, kao i shema baze podataka (slika 3.2).

U procesu unosa podataka u bazu, kreiraju se i pune odgovarajuće tablice u shemama GDOSYS i MDSYS, odnosno uvoze se vrijednosti geometrijskih i tematskih atributa objekata i njihovi metapodaci. Geometrija kartografskih objekata sprema se u jedan stupac tablice, tipa SDO_GEOMETRY. Taj unaprijed definirani geometrijski tip posjeduje dvije logičke komponente: prostorni referentni sustav i element Array, odnosno komponete:

A new table space is created in the database, as well as a new user. The user's role and privileges are defined together with the base scheme (Figure 3.2).

During the process of importing data into the database, all relevant tables in GDOSYS and MDSYS schemes are created and loaded, together with all cartographic geometrical and thematic data and their metadata. The geometry of cartographic object is stored into a single table column - the SDO_GEOMETRY type. This predefined geometrical type has two logical components: a spatial reference system and an ARRAY element, with SDO_GTYPE[13], SDO_SRID[14], SDO_POINT[15], SDO_ORDINATES[16], and SDO_ELEM_INFO[17] components.

3.1 Spatial data analysis

One of the basic tasks of database systems is a quick and efficient data analysis. Therefore, a quick access to stored data is enabled, especially if we are talking about databases that store a huge amount and different types of data, such as cartographic databases. The Oracle Spatial uses three main spatial analysis operators:

1. SDO_WITHIN_DISTANCE (finding data within a specified distance)
2. SDO_NN (finding closest neighbours, from the given position in the question, returns all neighbours found, regardless of their distance from the given position and are combined with operators SDO_BATCH_SIZE[18], SDO_NUM_RES[19] and SDO_NN_DISTANCE[20])
3. Spatial interaction operator (finding objects which are in interaction with the given object; these operators are SDO_FILTER and SDO_RELATE)
   a) SDO_FILTER operator is considered to be a simple operator and its task is: To identify every geometry whose MBR (*Minimum Boundaring Rectangle*) is in certain relation with the MBR of the given geometry; to use the spatial index without using the *Geometry Engine* functions.
   b) SDO_RELATE operator finds any geometry that is in an interaction with the given geometry, by specifying the interaction type (*touch, inside, covered-by, on, contains, covers, overlapbdyintersect, equal* or *overlapdisjoint*).

The general operator syntax is presentable as an abstract SQL statement form (Kothuri et al. 2004):

```
<spatial_operator>
(
table_geometry       IN SDO_GEOMETRY,
query_geometry       IN SDO_GEOMETRY
[, parameter_string  IN VARCHAR2
[, tag               IN NUMBER ] ]
)
='TRUE';
```

where:

- table_geometry[21]: is the column type SDO_GEOMETRY in which the operator works,
- query_geometry: is the given position, that is the column type SDO_GEOMETRY of another table,
- parameter_string[22]: specifies the parameter specific for a spatial operator,
- tag: specifies the number used only in a certain spatial operation, this argument is specified together with the parameter_string argument.

3.1.1 Examples of geospatial analysis and visualization

Example 1. SDO_WITHIN_DISTANCE operator

User's spatial query: *"how many buildings (stambenaz) are located within a distance of 1000 m to the religion building (vjerskio) whose id=1"*, is specified by the following SQL query:

```
SELECT COUNT(*)
FROM vjerskio v, stambenaz s
WHERE v.id=1
AND SDO_WITHIN_DISTANCE
(s.spatialpoint, v.spatialpoint,
'DISTANCE=1000 UNIT=M')='TRUE';
```

Result of the query is: "*there are 154 buildings in the database, whose distance is less than 1000 m from the religion building with id=1"*. Fig. 3.3 shows the visualization of the result in the *Geomedia Professional* environment - the buildings are presented in red.

Example 2. SDO_NN operator

If we add parameters SDO_BATCH_SIZE or SDO_NUM_RES to the general SQL statement of the SDO_NN operator (which returns every neighbour objects regardless of their distance), we can speed up the search or limit the number of the closest neighbourhood object. We can use the SDO_NUM_RES parameter in the query "find six buildings which are closest to the religious object whose id=1":

[13] SDO_GTYPE specifies what kind of geometry that object is – point, line, polygon, etc.
[14] SDO_SRID specifies the ID of spatial reference system in which is geometry specified.
[15] SDO_POINT specifies a point as geometry which needs to be saved.
[16] SDO_ORDINATES saves all coordinates of a defined geometry.
[17] SDO_ELEM_INFO specifies where in SDO_ORDINATES series a new element starts, how it is linked and what kind of geometry is specified.
[18] SDO_BATCH_SIZE selects the size of group which we are searching. In this example that is 100.
[19] SDO_NUM_RES defines the number of rows of end result.
[20] SDO_NN_DISTANCE defines the distance between nearest neighbors and given geometry.

[21] table_geometry must be spatially indexed and the syntax must be equal to TRUE.
[22] The sign that this argument is not obligatory.

SDO_GTYPE[13], SDO_SRID[14], SDO_POINT[15], SDO_ORDINATES[16] SDO_ELEM_INFO[17].

3.1. Analiza prostornih podataka

Jedan od osnovnih zadataka sustava baze podataka je brza i djelotvorna analiza podataka. Dakle, podrazumijeva se brz pristup spremljenim podacima, naročito ako govorimo o bazama podataka koje čuvaju velike količine i različite tipove podataka, kao što su to kartografske baze. *Oracle Spatial* koristi tri osnovna operatora za prostorne analize:

1. SDO_WITHIN_DISTANCE (pronalaženje podataka unutar specificirane udaljenosti)
2. SDO_NN (pronalaženje najbližih susjeda, od zadanog položaja u upitu, vraća sve pronađene susjede bez obzira na udaljenost od zadanog položaja, a kombiniraju se s pomoćnim operatorima SDO_BATCH_SIZE[18], SDO_NUM_RES[19] i SDO_NN_DISTANCE[20])
3. operatore za prostornu interakciju (pronalaženje objekata koji su u interakciji sa zadanim objektom; u te operatore ubrajaju se operatori SDO_FILTER i SDO_RELATE)
   a) Operator SDO_FILTER smatra se jednostavnim operatorom, a zadatak mu je da identificira svaku geometriju, čiji MBR (engl. *minimum bounding rectangle*) je u određenom odnosu s MBR-om, zadane geometrije; koristi prostorni indeks bez pozivanja funkcije *Geometry Engine*.
   b) Operator SDO_RELATE pronalazi svaku geometriju koja je u interakciji sa zadanom geometrijom, uz specificiranje tipa interakcije (*touch, inside*, *coveredby, on, contains, covers, overlapbdyintersect, equal* ili *overlapdisjoint*).

Opću sintaksu operatora moguće je prikazati apstraktnim oblikom SQL iskaza (Kothuri i dr., 2004):

```
<spatial_operator>
(
table_geometry       IN SDO_GEOMETRY,
query_geometry       IN SDO_GEOMETRY
[, parameter_string  IN VARCHAR2
[, tag               IN NUMBER ] ]
)
='TRUE';
```

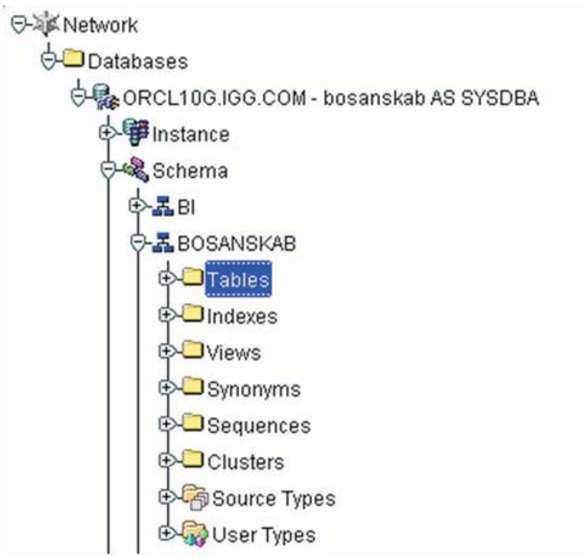


Fig. 3.2. Oracle scheme BOSANSKAB (part)

Slika 3.2. Shema BOSANSKAB (dio) u Oraclu

gdje je:

- table_geometry[21]: stupac tipa SDO_GEOMETRY u kojoj djeluje operator,
- query_geometry: zadani položaj, odnosno stupac tipa SDO_GEOMETRY neke druge tablice,
- parameter_string[22]: specificira parametre specifične za prostorni operator,
- tag: specificira broj upotrijebljen samo u određenoj prostornoj operaciji; taj se argument specificira zajedno s argumentom parameter_string.

3.1.1 Primjeri geoprostorne analize i vizualizacije rezultata

Primjer 1. Operator SDO_WITHIN_DISTANCE

Korisnikov prostorni upit: *˝koliko stambenih jedinica (stambenaz) na udaljenosti do 1000 m gravitira vjerskom objektu (vjerskio) čiji je id=1˝*, specificira se sljedećim SQL upitom:

```
SELECT COUNT(*)
FROM vjerskio v, stambenaz s
WHERE v.id=1
AND SDO_WITHIN_DISTANCE
(s.spatialpoint, v.spatialpoint,
'DISTANCE=1000 UNIT=M')='TRUE';
```

Rezultat upita je: *˝u bazi podataka nalaze se 154 stambena objekta, udaljena najviše 1000 m od vjerskog objekta čiji je id=1˝*. Slika 3.3. prikazuje vizualizaciju rezultata u okolišu *Geomedia Professional*, gdje su ti stambeni objekti prikazani crvenom bojom.

[13] SDO_GTYPE specificira kakvog je oblika geometrija – točka, linija, poligon, kolekcija i dr. koju predstavlja SDO_GTYPE
[14] SDO_SRID specificira ID prostornog referentnog sustava u kojem je geometrija specificirana.
[15] SDO_POINT specificira točku kao geometriju koju treba pohraniti.
[16] SDO_ORDINATES pohranjuje sve koordinate određene geometrije.
[17] SDO_ELEM_INFO specificira gdje u nizu SDO_ORDINATES počinje novi element, na koji način je povezan i koja je vrsta geometrije.
[18] SDO_BATCH_SIZE upotrebljava se za navođenje veličine grupe koja se pretražuje. U ovom je primjeru 100.
[19] SDO_NUM_RES upotrebljava se za određivanje broja redova krajnjeg rezultata.
[20] SDO_NN_DISTANCE upotrebljava se za određivanje udaljenosti najbližih susjeda od zadane geometrije.

[21] table_geometry mora biti prostorno indeksiran (u protivnom se javlja pogreška), a sintaksa mora biti izjednačena s TRUE.
[22] Otvorena pravokutna zagrada znači da taj argument nije obavezan.

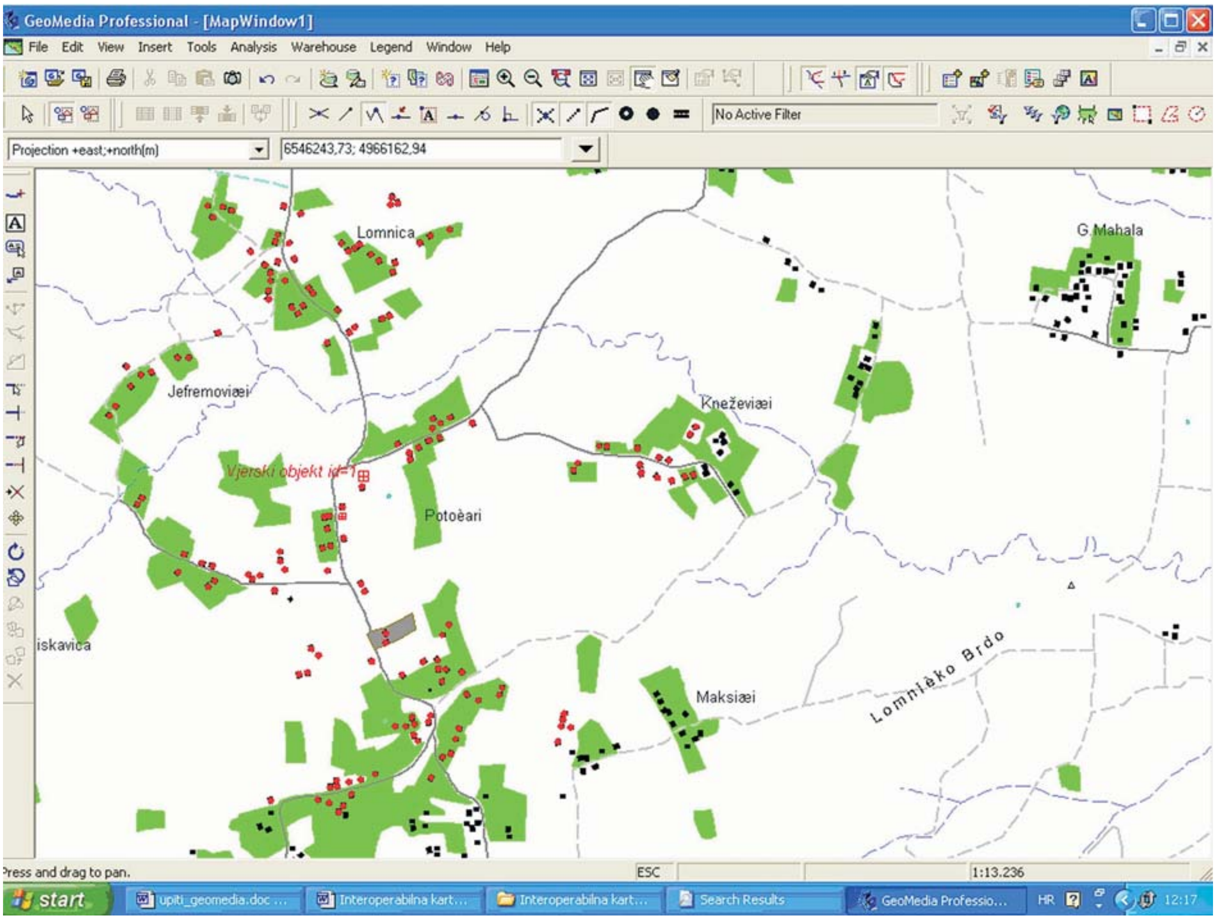


Fig. 3.3. Visualization query (GIS GeoMedia Professional)

Slika 3.3. Vizualizacija upita (GIS GeoMedia Professional)

```
SELECT s.id
FROM vjerskio v, stambenaz s
WHERE v.id=1
AND SDO_NN(s.spatialpoint, v.spatialpoint,
'SDO_NUM_RES=6')='TRUE';
```

The response to the query is the list of identifiers of six building, nearest to the religious building whose id=1.

The spatial operator SDO_NN can be used in combination with the SDO_NN_DISTANCE operator. We could also apply the SDO_NN_DISTANCE operator to the former example:

```
SELECT s.id, SDO_NN_DISTANCE(1) dist
FROM vjerskio v, stambenaz s
WHERE v.id=1
AND SDO_NN(s.spatialpoint, v.spatialpoint,
'SDO_BATCH_SIZE=100 UNIT=M',1)='TRUE'
AND ROWNUM<=6
ORDER BY dist;
```

The result of the query is the list identifiers of the nearest civil building and their distance (in meters) from the religious building with id=1:

Example 3. Spatial interaction operators

To analyse possible negative influences (e.g. toxin substance in the air, radiation etc.) of one transformer station on objects in a certain area zone (e.g. 250 m), it is necessary to create a *buffer* zone around the given object (transformer station):

```
CREATE TABLE tstanica_b AS
SELECT id, SDO_GEOM.SDO_BUFFER
(b.spatialpoint, 250, 0.5) geom
FROM tstanica b;

INSERT INTO USER_SDO_GEOM_METADATA
SELECT 'TSTANICA_B', 'GEOM', DIMINFO, SRID
FROM USER_SDO_GEOM_METADATA
WHERE TABLE_NAME='TSTANICA'

CREATE INDEX tsb_sidx ON tstanica_b(geom)
INDEXTYPE IS MDSYS.SPATIAL_INDEX;
```

The query: "*is there any interaction between the buildings and the buffer zone of the transformer station*", is specified with:

```
SELECT s.id
FROM tstanica_b t, stambenaz s
WHERE t.id=1
AND SDO_RELATE(s.spatialpoint, t.geom,
'MASK=ANYINTERACT')='TRUE'
ORDER BY s.id;
```

The result of the query consists of identifiers of 10 civil objects that are in an interaction with the buffer zone of the transformer station.

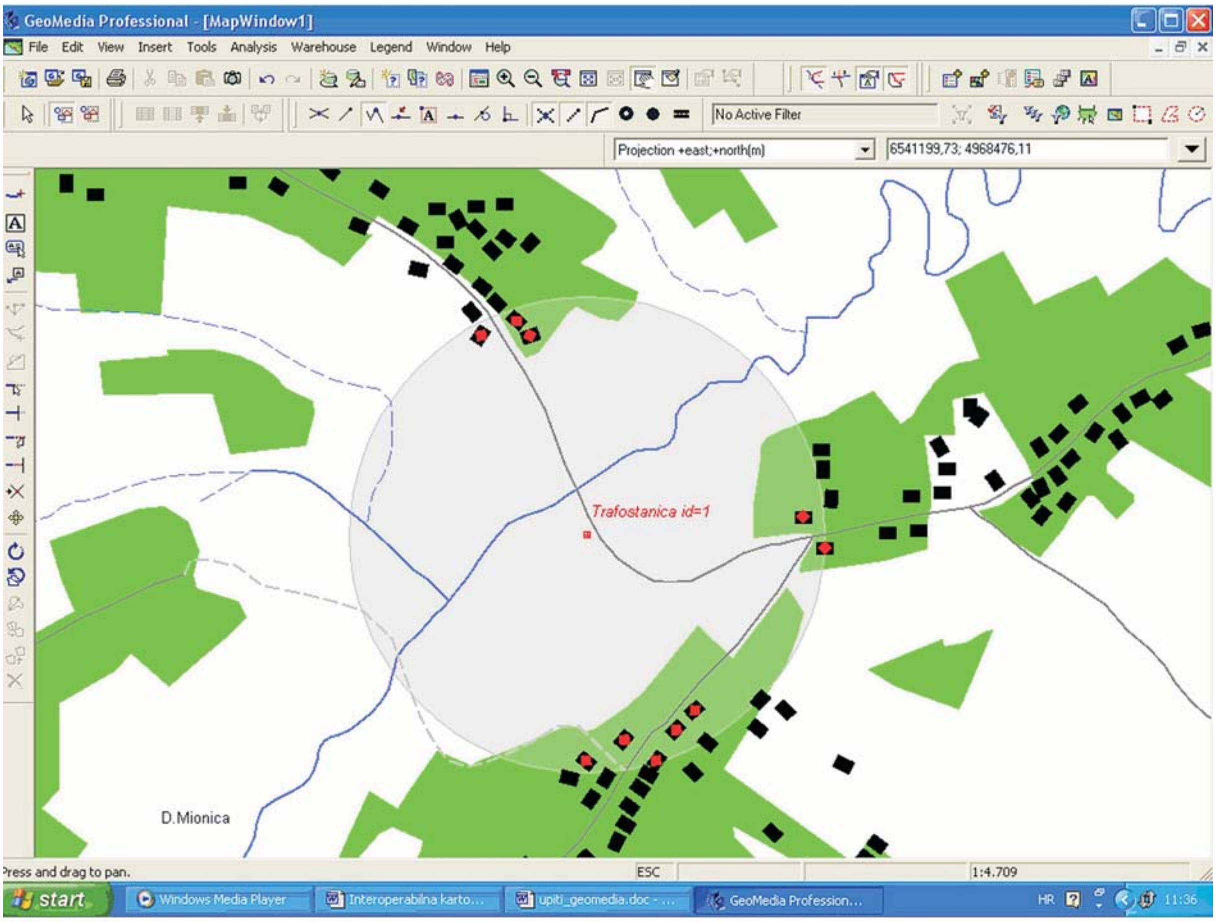


Fig. 3.4. Query visualization (GIS GeoMedia Professional)

Slika 3.4. Vizualizacija upita (GIS GeoMedia Professional)

Primjer 2. Operator SDO_NN

Ako se općem SQL iskazu operatora SDO_NN (koji vraća sve susjedne objekte, bez obzira na njihovu udaljenost) dodaju parametri SDO_BATCH_SIZE ili SDO_NUM_RES, može se ubrzati pretraživanje ili ograničiti broj najbližih susjednih objekata.

Ako se koristi parametar SDO_NUM_RES u upitu *˝pronađi šest stambenih objekata koji su najbliži vjerskom objektu čiji je id=1˝*, slijedi:

```
SELECT s.id
FROM vjerskio v, stambenaz s
WHERE v.id=1
AND SDO_NN(s.spatialpoint, v.spatialpoint,
'SDO_NUM_RES=6')='TRUE';
```

Odgovor na upit je popis identifikatora šest stambenih objekata, najbližih vjerskom objektu čiji je id=1.

Prostorni operator SDO_NN može se koristiti u kombinaciji s operatorom SDO_NN_DISTANCE. Ako se na prethodni primjer primijeni operator SDO_NN_DISTANCE, onda se specificira sljedeći SQL upit:

```
SELECT s.id, SDO_NN_DISTANCE(1) dist
FROM vjerskio v, stambenaz s
WHERE v.id=1
AND SDO_NN(s.spatialpoint, v.spatialpoint,
'SDO_BATCH_SIZE=100 UNIT=M',1)='TRUE'
AND ROWNUM<=6
ORDER BY dist;
```

Rezultat upita je popis identifikatora najbližih stambenih objekata i njihova udaljenost u metrima od vjerskog objekta id=1.

Primjer 3. Operatori za prostornu interakciju

Za analiziranje mogućih negativnih utjecaja (npr. zračenje) neke trafostanice na okolne stambene objekte koje se nalaze u određenoj zoni (npr. 250 m), potrebno je kreirati koridor (tzv. *buffer* zonu) oko zadanog objekta (trafostanice):

```
CREATE TABLE tstanica_b AS
SELECT id, SDO_GEOM.SDO_BUFFER
(b.spatialpoint, 250, 0.5) geom
FROM tstanica b;

INSERT INTO USER_SDO_GEOM_METADATA
SELECT 'TSTANICA_B', 'GEOM', DIMINFO, SRID
FROM USER_SDO_GEOM_METADATA
WHERE TABLE_NAME='TSTANICA'

CREATE INDEX tsb_sidx ON tstanica_b(geom)
INDEXTYPE IS MDSYS.SPATIAL_INDEX;
```

Upit: "*postoji li bilo kakva interakcija među stambenim objektima i koridora trafostanice*", specificiramo sljedećim SQL iskazom:

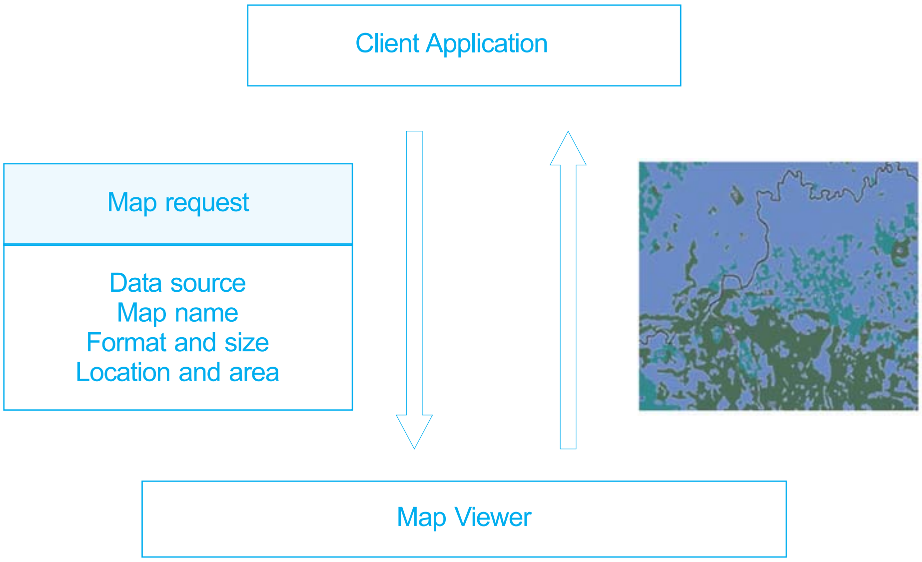


Fig. 3.5. Processing MapViewer requests and responses (Kothuri et al., 2004)

Fig. 3.4 shows us the transformer station (red colour), its buffer zone (250 m – grey colour), and ten buildings (red colour).

3.2 Visualization on the Internet

We have already seen the visualization of cartographic data in desktop GIS software, in this chapter we will consider in detail visualization on the Internet, using the *Oracle MapViewer*. *MapViewer* is a Java application for visualization of geospatial data, and can be executed on every computer platform. It can be used either in *Oracle Application Server* environment or as independent *Oracle Application Server Container* for *Oracle Container for Java* (J2EE – OC4J*)* – a key *Oracle* application server component.

MapViewer is a pure Java server-side component included with Oracle Application Server. Its main components are as follows:

1. *MapViewer servlet*[23] *running inside the Oracle Application Server*: The servlet processes requests sent by client applications, fetches proper information from spatial tables and constructs maps in a variety of graphical formats (GIF, PNG or JPEG), which it then returns to the requesting client.
2. *Map definitions:* Map definitions are stored inside the database. This is where you describe your maps: which tables to use, how the maps should be rendered (colours, line, thickness, fonts, etc.), and so on.
3. *Java client Application programming interface* – API: Java applications will use this API to simplify their development by avoiding the need for manually constructing and parsing XML requests and responses. The Java API also includes JavaServer Pages (JSP) tags to ease the inclusion of maps in JSP.
4. *Map Definition Manager Tool:* This is a standalone program that helps users manage map definitions stored inside the database. (Kothuri et al., 2004.).

Fig. 3.5 illustrates the request and answer process between the user and the *MapViewer* server. The flow of operation for requesting a map is as follows:

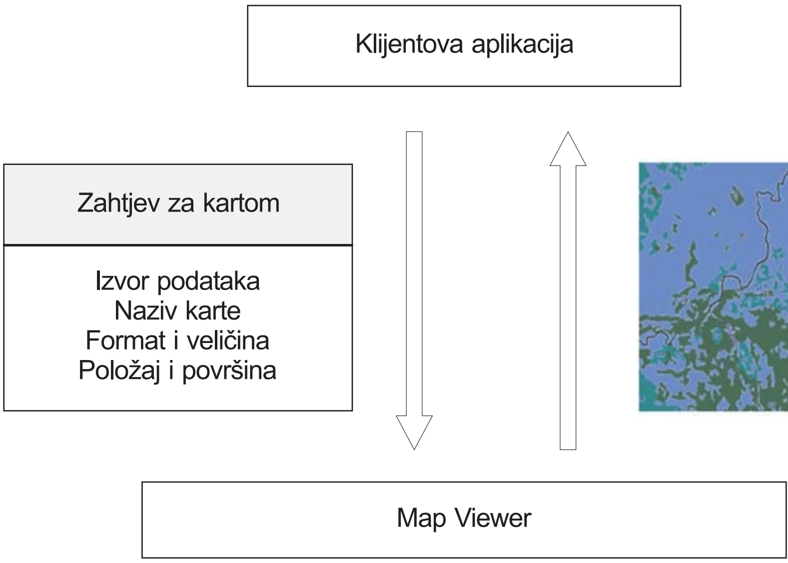
1. The client application constructs a web service request to obtain a map. That request can be generated directly in XML or via a number of Java methods. The request contains the name of the data source (database) to read, the name of the map to generate, its format (GIF, PNG[24], or JPEG), its size in pixels, and the area covered by the map. The application can construct the XML request manually, or it can use the Java client API to generate it.
2. The client calls the *MapViewer servlet* over HTTP, passing the XML request as a parameter.
3. The *MapViewer servlet* parses the request, reads necessary map definitions from the database and generates a map in GIF, PNG, or JPEG format. The map is written out as a file. Note that map definitions are cached by the servlet. The servlet can also optionally cache part or all of the spatial data it reads.
4. The *servlet* constructs an XML response that includes the URL to the generated image file and returns it to the client.
5. The client then parses the XML and extracts the map image URL, which it then forwards to the client browser (Kothuri et al., 2004).

3.2.1 Constructing map content

Fig. 3.6 shows map construction and links between styles, themes, tables and map.

[23] *Servlet* can remember part of or all data read.

[24] PNG – *Portable Network Graphics (format)*



Slika 3.5. Obrada zahtjeva i odgovora MapViewera (Kothuri i dr., 2004)

```
SELECT s.id
FROM tstanica_b t, stambenaz s
WHERE t.id=1
AND SDO_RELATE(s.spatialpoint, t.geom,
'MASK=ANYINTERACT')='TRUE'
ORDER BY s.id;
```

Rezultat upita su identifikatori 10 stambenih objekata koji su u interakciji s koridorom trafostanice.

Na slici 3.4. vidi se trafostanica (crvena boja), njezin koridor (250 metara - siva boja) i deset stambenih objekata (crvena boja).

3.2 Internetska vizualizacija

Vidjeli smo kako je vizualizacija kartografskih podataka moguća u jednom konkretnom desktop GIS softveru, a u ovom poglavlju opširnije ćemo razmotriti vizualizaciju na internetu, uporabom *Oracle MapViewera*. To je Java aplikacija za vizualizaciju geoprostornih podataka, te se može pokrenuti na bilo kojoj računalnoj platformi. Može se koristiti dvojako: u okolišu *Oracle Aplication Servera* (aplikacijski poslužitelj Oracla) ili kao samostalni *Oracle Aplication Server Container* za *Oracle Container for Java* (J2EE – OC4J*)*, ključne komponente aplikacijskog poslužitelja Oracle.

MapViewer se sastoji od sljedećih komponenti:

1. *MapViewer servleta*[23] (obrađuje zahtjeve poslane od klijentove aplikacije, dohvaća odgovarajuće informacije iz prostornih tablica i konstruira karte u različitim formatima, a potom ih vraća klijentu);
2. Definiranja sadržaja karte (pohranjuje se u bazu podataka; označava gdje se nalazi opis karte odnosno koje tablice koristi, koje boje koristi, fontove naziva, simbole i sl.);
3. *Java client Application programming interface* - API (aplikacija koja koristi API za pojednostavnjenje njezina razvoja, bez potrebe za ručnom konstrukcijom ili raščlanjivanjem XML zahtjeva i odgovora); uključuje i *JavaServerPages* (JSP) za lakše uključivanje karte u JSP;
4. *Map Definition Manager Tool* (samostalni program za pomoć korisnicima u definiranju vizualizacije, tj. sadržaja karte na temelju podataka u *Oracle Spatial* bazi) (Kothuri i dr., 2004).

Slika 3.5. ilustrira proces zahtjeva i odgovora između korisnika i poslužizelja *MapViewera*. Proces zahtjeva za kartom odvija se u sljedećim koracima:

1. Klijentova aplikacija formira zahtjev za dobivanje karte putem web servisa, koji može biti generiran putem XML-a ili metodama Jave. Zahtjev treba sadržavati: izvorne baze podataka, naziv karte, format, veličinu u pikselima i površinu koju će obuhvaćati karta. Aplikacija može izravno konstruirati XML zahtjev ili koristiti *Java client API*;
2. Klijent poziva *MapViewer servlet* preko HTTP-a, povezujući XML;
3. *MapViewer servlet* analizira zahtjev, čita potrebnu definiciju karte iz baze podataka i generira kartu u jednom od rasterskih formata (GIF, PNG[24] ili JPEG). Karta je definirana kao datoteka i pamti se s pomoću servleta;
4. *Servlet* konstruira XML odgovor koji uključuje Uniform Resource Locator (URL), kako bi generirao kartu (datoteku slike) i vratio ju klijentu;
5. Klijent potom analizira XML i izdvaja URL karte, koju potom šalje klijentovu pregledniku (Kothuri i dr., 2004).

[23] *Servlet* može pamtiti dio prostornih podataka koje pročita ili sve.

[24] engl. PNG - *Portable Network Graphics (format)*

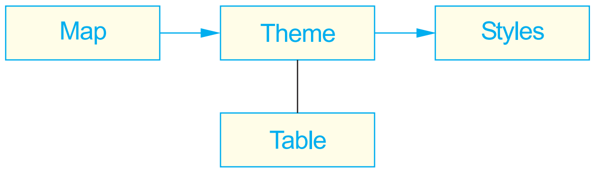


Fig. 3.6. Components of map

By running the program *Map Definition Manager Tool*, and linking it with database in which geospatial (cartographic) data are saved, one is able to define metadata of a map. Defining metadata of the map is executed by defining styles (selection of colour, marks, lines, areas, texts and advanced style), themes (taking data from table saved in database) and defining look of maps.

In implementation of example of B&H's cartographic database, two types of marks were used: image mark for complex types of symbols (đeram, izvori, etc.), and vector mark for simple types of point objects (zgrade, bunari, etc.). The same symbols are saved in the table ALL_SDO_STYLES, and we can use them (vjerski objekti, geodetske točke, etc.). By selecting the option *line*, we can define lines, i.e. select kind or colour of the line, and we can use lines from ALL_SDO_STYLES. Style for area defines the pattern that fills an area on the map. Type, size and colour of the text on the map are also defined. Defined styles are saved in the USER_SDO_STYLES views.

The *Oracle Map Definition Tool* is a tool that helps the user to define the theme, select tables (from which data are taken), select geometry column, and style used for presentation of the themes (Figure 3.7.). Defined themes are saved in USER_SDO_THEMES views.

Definition of a map consists of name specification, selected content - theme for visualization, and minimum/maximum scale. Defined map content is stored in the USER_SDO_MAPS views.

3.2.2 XML requests for the visualization of database content

Map visualization is possible only after map content definition. The application server has the role of a mediator between the Web server and the database server. Its task is to begin and support, and later end the relationship between the Web and the map server. Their main functions are taking the client's requests and their direction to the server, as well as balancing the procedures between the map server and the data server. The map server performs spatial analyses, generates and delivers maps to the client on his or her request. The data server serves data (spatial and thematic) that are stored in the database.

To achieve map visualization, it is necessary to direct an XML request to the server. In the following example, the request of the visualization of the object class of waters presented (rivers, springs, temporary water and streams) with the sending of the XML requests:

```
<?xml version="1.0" standalone="yes"?>
<map_request
  title="Interoprabilna kartografska baza
```

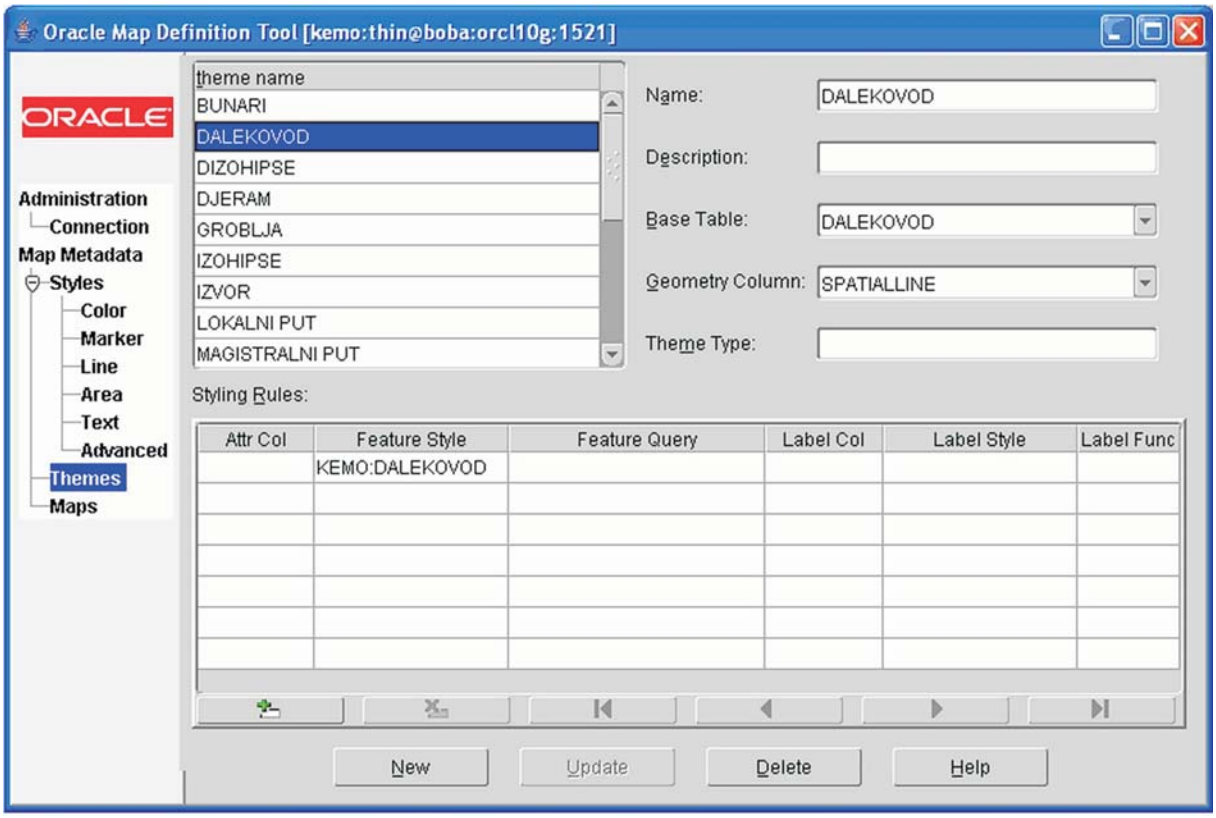
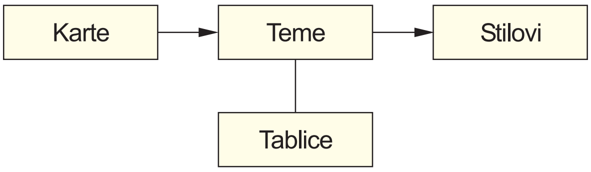



Fig. 3.7. Theme definition by the Map Definition Tool

Slika 3.7. Definiranje tema s pomoću alata za definiranje karata (Map Definition Tool)



Slika 3.6. Komponente karte

3.2.1 Konstruiranje sadržaja karte

Slika 3.6. prikazuje načelo konstruiranja karte, odnosno veze između stila, tema, tablica i karte.

Pozivanjem programa *Map Definition Manager Tool* i njegova povezivanja s bazom podataka u kojoj su spremljeni geoprostorni (kartografski) podaci, omogućeno je definiranje metapodataka karte. Definiranje metapodataka karte izvodi se putem definiranja stilova (odabir boje, oznaka, linija, površina, teksta i naprednog stila), tema (dohvaćanje podataka iz tablica spremljenih u bazi podataka) i definiranja izgleda karata.

U implementaciji primjera kartografske baze podataka BiH korištena su dva tipa oznaka: rasterski za složenije tipove simbola (đeram, izvori i sl.) i vektorski za jednostavnije tipove točkastih objekata (zgrade, bunari i sl.). Korištene su i ponuđene gotove oznake (pohranjene u tablici ALL_SDO_STYLES): vjerski objekti, geodetske točke itd. Odabirom opcije *Line* mogu se definirati linije, odnosno odabrati vrsta, debljina i boja linije, a mogu se koristiti i linije ponuđene u ALL_SDO_STYLES. Stil za površine definira uzorak kojim je ispunjena površina na karti. Definiraju se također vrsta, veličina i boja teksta naziva na karti. Definirani stilovi spremaju se u pogledu USER_SDO_STYLES.

Oracle Map Definition Tool je alat koji korisniku olakšava imenovanje teme, pronalaženje odgovarajuće tablice (iz koje se uzimaju podaci), odabir geometrijskog stupca, kao i stil koji se koristi za prikaz teme (slika 3.7). Definirana tema sprema se u pogledu USER_SDO_THEMES.

Definiranje izgleda karte obuhvaća specifikaciju naziva, odabir sadržaja, odnosno tema za vizualizaciju, te minimalno i maksimalno mjerilo u rasponu kojega je tema vidljiva. Definirani sadržaj karte sprema se u poglede USER_SDO_MAPS.

3.2.2 XML zahtjevi za vizualiziranje sadržaja baze podataka

Nakon definiranja sadržaja karte moguća je njezina vizualizacija. Aplikacijski poslužitelj ima ulogu posrednika između web poslužitelja i poslužitelja baze podataka. Zadaća mu je uključivanje, uspostava i održavanje, odnosno prekidanje veze između web i map poslužitelja, preuzimanje klijentovih zahtjeva i njihovo upućivanje map poslužitelju, kao i balansiranje procedura između map poslužitelja i poslužitelja podataka. Map poslužitelj izvodi prostorne analize, generira i dostavlja karte klijentu na njegov zahtjev. Poslužitelj podataka poslužuje podatke (prostorne i tematske) spremljene u bazi podataka.

Za vizualizaciju karte potrebno je uputiti XML zahtjev poslužitelju. U sljedećem primjeru zahtijeva se vizualizacija objektnih klasa vodâ (rijeke, potoci, povremeni vodotokovi i izvori), slanjem XML zahtjeva:

```
<?xml version="1.0" standalone="yes"?>
<map_request
  title="Interoprabilna kartografska baza
        podataka "
  datasource="orcl10g.igg.com"
  width="600"
  height="800"
  format="PNG_STREAM">
  <center size="13000">
    <geoFeature>
      <geometricProperty>
        <Point>
          <coordinates>6545000.0,4965000.0
         </coordinates>
        </Point>
      </geometricProperty>
    </geoFeature>
  </center>
  <themes>
    <theme name="IZVOR"  />
    <theme name="POTOK" />
    <theme name="RIJEKE" />
    <theme name="POVREMENI VODOTOK" />
  </themes>
</map_request>
```


gdje je:

title – naslov karte koja se prikazuje s pomoću MapViewera,

basemap – naziv osnovne karte sadržaj koje je definiran s pomoću *Oracle Map Definition Tool*,

datasource – logički naziv baze podataka (Oracle servis),

width – širina vizualizirane karte u pikselima,

height – visina vizualizirane karte u pikselima,

format – odabrani format prikaza karte,

center size – mjerilo (ili bolje rečeno visina karte u metrima),

coordinates – koordinate centra karte (originalne veličine u metrima),

theme name – naziv teme.

Poslužitelj kao odgovor na prethodni XML zahtjev vraća sadržaj dijela baze podataka u obliku rasterske slike (karte) s odabranim temama (slika 3.8):

Idući primjer ilustrira XML zahtjev za vizualizaciju izvora u obliku specifičnih kartografskih simbola (crveni križići):

```
<?xml version="1.0" standalone="yes"?>
<map_request
  title="Interoperabilne kartografske baze
        podataka"
  basemap="BOSANSKAB"
  datasource="ORCL10G.IGG.COM"
  width="600"
  height="800"
  format="PNG_STREAM">
  <center size="9000">
    <geoFeature>
      <geometricProperty>
        <Point>
          <coordinates>6543100, 4958000
```

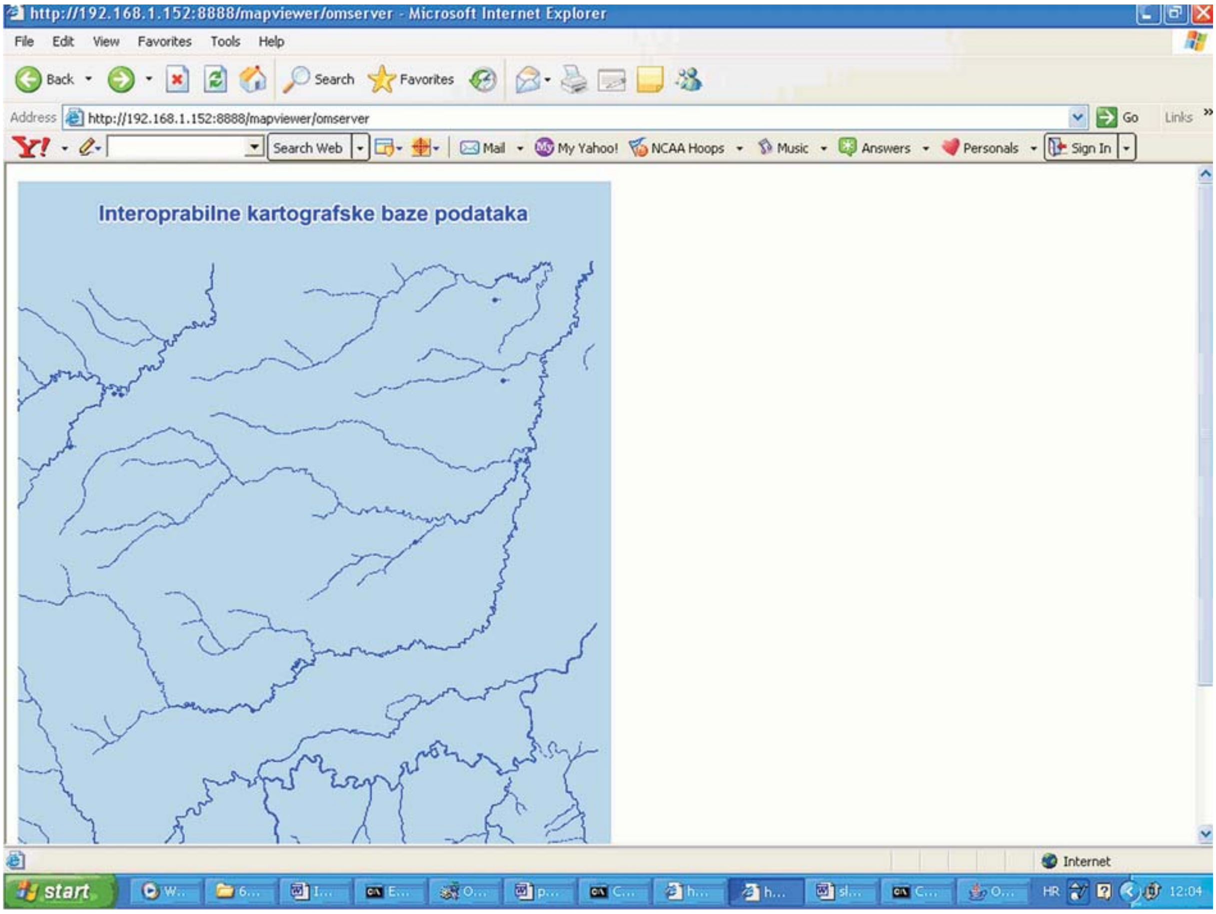


Fig. 3.8. Example of visualization of database content in the form of a multi-theme map

Slika 3.8. Primjer vizualizacije sadržaja baze podataka u obliku višetematske karte


```
    podataka "
  datasource="orcl10g.igg.com"
  width="600"
  height="800"
  format="PNG_STREAM">
  <center size="13000">
    <geoFeature>
      <geometricProperty>
        <Point>
          <coordinates>6545000.0,4965000.0
          </coordinates>
        </Point>
      </geometricProperty>
    </geoFeature>
  </center>
  <themes>
    <theme name="IZVOR"  />
    <theme name="POTOK" />
    <theme name="RIJEKE" />
    <theme name="POVREMENI VODOTOK" />
  </themes>
</map_request>
```

where these are presented:

title – meaning the title of the map that is presented using the *MapViewer*,

basemap – stands for the basic map whose content is defined by *Oracle Map Definition Tool*,

datasource – logical database term (Oracle service),

width – the width of the visualized map presented in pixels,

height – the height of the visualized map in pixels,

format – the chosen format of map display,

centre size – scale of map (or better, map height in meters),

coordinates – coordinates of the centre of the map (original sizes in meters),

theme name – theme's name.

As a response to the former XML request, the server returns the content of the part of the database in the form of an image (map) with the selected themes (Fig. 3.8).

The next example illustrates the XML request for visualization of the source in the form of specific cartographic symbols (red crosses):

```
<?xml version="1.0" standalone="yes"?>
<map_request
  title="Interoperabilne kartografske baze
        podataka"
  basemap="BOSANSKAB"
  datasource="ORCL10G.IGG.COM"
  width="600"
  height="800"
  format="PNG_STREAM">
  <center size="9000">
    <geoFeature>
      <geometricProperty>
```

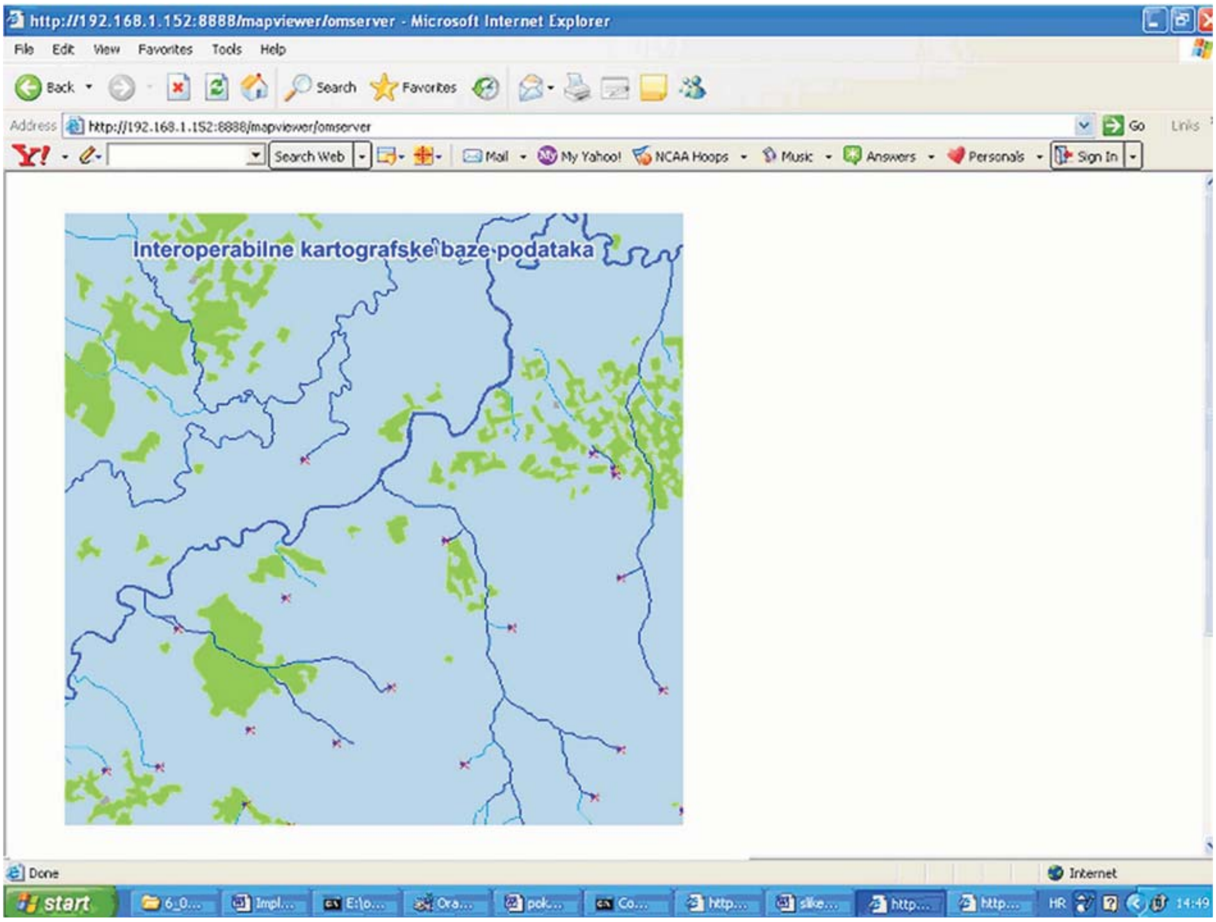


Fig. 3.9. Example of visualization of the object class "izvor"

Slika 3.9. Primjer vizualizacije objektne klase "izvor"

```
          </coordinates>
        </Point>
      </geometricProperty>
    </geoFeature>
  </center>
  <themes>
    <theme name="Izvor">
      <jdbc_query
        datasource="ORCL10G.IGG.COM"
        spatial_column="SPATIALPOINT"
        render_style="M.CYAN PIN">
          select * from izvor
      </jdbc_query>
    </theme>
  </themes>
</map_request>
```

Poslužitelj, kao odgovor na XML zahtjev, vraća kartu prikazanu na slici 3.9.

4. Zaključak

Geoprostorne podatke prikupljaju, obrađuju i distribuiraju različite organizacije/institucije, koje obično koriste različite računalne sustave i modele baza podataka, te se stoga javlja niz problema u procesu uporabe i prijenosa tih podataka. Ti se problemi mogu riješiti respektiranjem i uporabom koncepta *geoinformacijske interoperabilnosti*.

U radu je prikazano kreiranje prototipa interoperabilne kartografske baze podataka, koja omogućava pohranjivanje kompleksnih geoprostornih, odnosno kartografskih tipova podataka, suglasno otvorenim standardima OGC-a. Takva, interoperabilna baza podataka omogućava korisnicima otvoren i brz pristup kartografskim objektima, postavljanje prostornih upita utemeljenih na SQL-u, analizu i ažuriranje, *različitim* produktima. Kartografski podaci spremljeni u interoperabilnoj bazi podataka mogu se vizualizirati *različitim* sučeljima (desktop GIS, WebGIS) na temelju korisnikovih, aplikacijski specifičnih zahtjeva.

U BiH još uvijek ne postoji nacionalni kartografski model podataka, a ni norme i pravila usuglašena s odgovarajućim OGC/ISO normama, te je predočena implementacija prototipa interoperabilne kartografske baze podataka bila otežana. Međutim, iz iskustva stečenih tijekom implementacije može se zaključiti kako se primjenom moderne tehnologije i OGC/ISO standarda može kreirati interoperabilna kartografska baza podataka. Kako bi kreirani prototip interoperabilne baze podataka zaista bio dostupan širokom krugu korisnika, nužno je poboljšati računalnu infrastrukturu, prihvatiti ili preuzeti odgovarajuće geoinformacijske/kartografske norme te radikalno promijeniti i poboljšati izobrazbu geodetskih kadrova u svrhu njihove aktivne uloge i rada na tržištu geoinformacija, koje se uspostavlja.

```
        <Point>
          <coordinates>6543100, 4958000</
coordinates>
        </Point>
      </geometricProperty>
    </geoFeature>
  </center>
  <themes>
    <theme name="Izvor">
       <jdbc_query
         datasource="ORCL10G.IGG.COM"
         spatial_column="SPATIALPOINT"
         render_style="M.CYAN PIN">
           select * from izvor
       </jdbc_query>
    </theme>
  </themes>
</map_request>
```

The server, as a response to the XML request, returns the map represented on Fig. 3.9.

4 Conclusion

Different organizations/institutions collect, process and distribute geospatial data and they usually use different computer systems and database models, which causes a lot of problems in the process of usage and exchange of these data. These problems can be solved by respecting and using the geoinformation interoperability concept.

In this paper, we presented the creation of an interoperable cartographic database, which enables the storing of complex geospatial-cartographic data types, according to the OGC Open standards. This type of the interoperable database offers the users an open and quick access to cartographic objects, enables the specification of SQL-based spatial queries, as well as the analysis and the update using *different* GIS products. The cartographic data that are stored in the interoperable database can be visualized with *different* interfaces (GIS desktop, WebGIS) on the bases of the user's application–specific requests.

There is still no official cartographic data model in B&H, neither are there any norms and rules in accordance with the suitable OGC/ISO norms. This causes the prototype implementation of an interoperable cartographic database to be difficult. However, using the experience gathered during the implementation, we can conclude that by using modern technology and the OGC/ISO standards, the creation of an interoperable cartographic database is possible. To make the created interoperable database prototype fully accessible to a wide range of users, it is necessary to improve the computer infrastructure, adopt or take over suitable/corresponding geoinformation cartographic norms, and also to radically change and improve the geodetic personnel is education, having in mind their active participation and their role in the geoinformation market which is being formed.

References / Literatura

Galić, Z. (2004): WebGIS (pisana predavanja), Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu, Zagreb.

Galić, Z. (2006): Geoprostorne baze podataka, Golden marketing – Tehnička knjiga, Zagreb.

ISO 19128 (2005): Geographic Information – Web Map Server Interface, International Organization for Standardization, Geneva, 2005.

ISO/DIS 19136 (2005): Geographic Information – Geography Markup Language (GML), International Organization for Standardization, Geneva, 2005.

Kraak, M. J., Ormeling, F. (2003): Cartography, the Visualisation of Geospatial Data, Pearson Education, 2003

Ključanin, S. (2006): Interoperabilne kartografske baze podataka, magistarski rad. Građevinski fakultet, Sarajevo.

Ključanin, S., Tuno, N., Mulahusic, A. (2006): Internetska vizualizacija topografske baze podataka. Intergeo East 2006, Beograd.

Kothuri, R., Godfrind, A., Beinat, E. (2004): Pro Oracle Spatial, Apress, Berkeley.

OGC (2005a): OpenGIS Implementation Specification for Geographic Information - Simple Feature Access – Part 2: SQL option, Version 1.1.0, Open Geospatial Consortium

OGC (2005b): Geography Markup Language (GML) Encoding Specification, Open Geospatial Consortium

OGC (2005c): Web Feature Service Implementation Specification, Version 1.1.0, Open Geospatial Consortium

OGC (2006): OpenGIS Web Map Server Implementation Specification, Version 1.3.0, Open Geospatial Consortium

Peng, Z. R., Tsou, M. H. (2003): Interent GIS – Distributed Geographic Information Services for the Interenet and Wireless Networks, John Wiley & Sons, Inc., Hoboken, New Jersey.