

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

**SEMINAR**

## **PostgreSQL**

*Luka Perkov*

Voditelj: *Boris Vrdoljak*

Zagreb, svibanj, 2009

## Sadržaj

1. Uvod.....	1
2. Povijest PostgreSQL sustava za upravljanje bazama podataka.....	2
3. Karakteristike PostgreSQL sustava za upravljanje bazama podataka.....	3
3.1 Kreiranje nove baze podataka.....	3
3.2 Dodavanje korisnika i grupa.....	3
3.3 Prostor baze podataka.....	4
3.4 Tablice.....	4
3.5 Indeksi.....	5
3.6 Ostalo.....	6
4. PostgreSQL sustav za upravljanje bazama podataka i programski jezici.....	7
4.1 Programski jezici namijenjeni izvršavanju na poslužiteljskoj strani .....	7
4.2 Programski jezici namijenjeni izvršavanju na klijentskoj strani .....	8
5. Osposobljavanje poslužitelja za rad s PostgreSQL sustavom za upravljanje bazama podataka.....	9
6. Alati za rad s PostgreSQL sustavom za upravljanje bazama podataka.....	11
6.1 pgAdmin III.....	11
7. Zaključak.....	13
8. Literatura.....	14
9. Sažetak.....	15

## 1. Uvod

Za dodavanje, pristupanje i procesiranje podataka u bazi podataka potreban je sustav za upravljanje bazama podataka kao što je PostgreSQL. Budući da se računala odlično snalaze pri obradi velikih količina podataka, sustavi za upravljanje bazama podataka igraju centralnu ulogu u računarstvu, kao samostalni alati ili kao djelovi drugih aplikacija.

PostgreSQL je sustav za upravljanje objektno relacijskim bazama podataka. Objektno relacijska baza podataka je baza podataka koja podatke sprema u odvojene tablice, što daje brzinu i fleksibilnost bazama podataka. Tablice su povezane definiranim relacijama što omogućuje kombiniranje podataka iz nekoliko tablica.

PostgreSQL baza podataka je otvorenog kôda, što znači da je dozvoljeno svakome da ju koristi i prilagođava za svoje potrebe. Iako je besplatan, PostgreSQL može se usporediti sa komercijalnim bazama podataka kao što su Sysbase, Oracle i DB2.

U drugom poglavlju je dana povijest PostgreSQL sustava za upravljanje bazama podataka. U trećem poglavlju su opisane karakteristike PostgreSQL sustava i pregled najčešće korištenih SQL operacija. U četvrtom poglavlju dan je pregled mogućnosti interakcije programskih jezika s PostgreSQL bazom podataka. U petom poglavlju dan je postupak postavljanja PostgreSQL sustava za upravljanje bazama podataka što je preduvjet rada sa bazom podataka. pgAdmin III je program s grafičkim sučeljem koji uvelike olakšava administraciju i oblikovanje baza podataka. pgAdmin III opisan je u šestom poglavlju.

## **2. Povijest PostgreSQL sustava za upravljanje bazama podataka**

PostgreSQL sustav za upravljanje bazama podataka ima dugačku povijest. Krajem 70-ih godina prošlog stoljeća na Sveučilištu Berkley počeo je razvoj na Ingres sustavu za upravljanje bazama podataka. Ubrzo nakon završetka razvoja tvrtka Relational Technologies je komercijalizirala Ingres, te nakon brojnih promjena vlasnika pravo na Ingres danas ima Ingres Corporation.

1986. godine na Sveučilištu Berkley Ingres se nadograđuje objektno-orijentiranim svojstvima i dobiva novo ime Postgres. Postgres sustav za upravljanje bazama podataka komercijalizira se kao i Ingres, te postaje privatno vlasništvo tvrtke Illustra.

Postgres nije imao podršku za SQL već se koristio specifični upitni jezik Postquel. Nakon što su Andrew Yu i Jolly Chen dodali Postgres-u podršku za SQL upitni jezik sredinom 90-ih godina, Postgres mijenja ime u PostgreSQL. Danas PostgreSQL razvija internacionalna grupa stručnjaka PostgreSQL Global Development Group.

### 3. Karakteristike PostgreSQL sustava za upravljanje bazama podataka

PostgreSQL je utemeljen na modelu klijent poslužitelj. Jedna PostgreSQL sjednica se sastoji od dvije međusobno povezane aplikacije:

- Poslužiteljske aplikacije, zvane *postgres*, koja je zadužena za rad sa datotekama baze, za prihvaćanje konekcija od klijentskih aplikacija prema bazi podataka i obradu istih,
- Klijentske aplikacije, koja želi obaviti neku radnju s bazom podataka. Klijentska aplikacija može biti tekstualno orijentirana, može imati grafičko sučelje, može biti web server ili specijalizirana aplikacija za nadzor i održavanje baze podataka.

Kao i bilo koja druga aplikacija zasnovana na modelu klijent poslužitelj, i PostgreSQL komunikaciju obavlja preko TCP/IP sučelja. Za svaku sjednicu PostgreSQL stvara novi proces kako bi omogućio paralelnu obradu velikog broja klijentskih zahtjeva. Nakon toga klijentska aplikacija više ne komunicira s glavnim poslužiteljskim procesom, *postgres-om*, nego s novostvorenim procesom upravo za tu klijentsku aplikaciju. PostgreSQL ove akcije obavlja interno i krajnji korisnik ne mora biti upoznat s njima.

#### 3.1 Kreiranje nove baze podataka

PostgreSQL poslužitelj može upravljati s više baza podataka. Uobičajeno je da postoji bar jedna baza podataka po projektu. U PostgreSQL bazi podataka nova baza podataka se stvara koristeći naredbu *createdb*. Naredba za stvaranje baze podataka naziva *privatelibrary* izgleda ovako:

```
$ createdb privatelibrary
```

#### 3.2 Dodavanje korisnika i grupa

Korisnici i grupe u PostgreSQL sustavu za upravljanje bazama podataka imaju važnu ulogu. Administrator baze podataka može stvoriti korisnike i grupe i tako regulirati interakciju pojedinog korisnika s objektima baze podataka. PostgreSQL pohranjuje podatke o

korisnicima i grupama u svoje sistemske kataloge, koji se razlikuju od korisnika i grupa operacijskog sustava na kojem je pokrenut programski paket. Iznimka je korisnik *postgres* koji se prilikom instalacije dodaje i u sam operacijski sustav. Korisnik *postgres* je podrazumijevani super korisnik sa svim ovlastima u radu s bazom podataka. Nakon slijedećih naredbi, koje izvodi korisnik *postgres*, u bazu podataka se dodaje novi korisnik *librarian*:

```
$ ./psql privatelibrary
privatelibrary=# CREATE USER librarian WITH PASSWORD '*****';
privatelibrary=# SELECT username, passwd from pg_shadow;
   username |          passwd
-----+-----
 postgres  | md59f8f64bclaff7b6aa3d780ee3cdd****
 librarian  | md5f350545c60d1d01233ae91d23756****
(2 rows)
```

### 3.3 Prostor baze podataka

Prostor baze podataka (eng. *tablespace*) u PostgreSQL-u omogućuje definiranje alternativnih mjesta za trajno pohranjivanje podataka iz baze. Koristeći prostor baze podataka administrator baze podataka može kontrolirati gdje će u sustavu biti pohranjeni podaci i tako optimizirati performanse sustava. Nakon kreiranja prostora baze podataka može mu se pristupiti preko imena. Slijedeća naredba stvara novi prostor baze podataka *alternative\_tablespace*:

```
privatelibrary=# CREATE TABLESPACE alternative_tablespace LOCATION '/home/postgres/data';
```

### 3.4 Tablice

PostgreSQL kao i svaki drugi sustav za upravljanje relacijskim bazama podataka ima podršku za upravljanje tablicama. Mimo standarda, PostgreSQL podržava napredna obilježja tablica kao što su faktor popunjenosti (eng. *fillfactor*) i identifikatori objekta (eng. *object identifiers*, *OID*).

Faktor popunjenosti označava popunjenost pojedinog objekta baze podataka (npr. tablice ili indeksa). Valjane vrijednosti za faktor popunjenosti nalaze se u intervalu [10, 100]. Manje popunjavanje nekog objekta se preporuča ukoliko se vrijednosti u tom objektu često mijenjaju (npr. ukoliko postoji puno INSERT naredbi), dok se potpuno popunjavanje

preporuča samo u objektima čije se vrijednosti nikada ili vrlo rijetko mijenjaju. Za tablice podrazumijevani faktor popunjenosti je 100, dok za indeks koji koristi b-stablo podrazumijevani faktor popunjenosti je 90.

Identifikatore objekata (eng. object identifiers), OID, koristi PostgreSQL baza podataka kao primarne ključeve za svoje sistemske kataloge i tablice. OID se mogu dodati u korisnički definirane tablice, ali treba imati na umu da je OID trenutno implementiran kao 4 bajtna cjelobrojna vrijednost te se njegova upotreba ne preporuča.

Slijedećom SQL naredbom se stvara nova tablica *book*:

```
CREATE TABLE book(  
    id integer NOT NULL,  
    "ISBN" character(13) NOT NULL,  
    authors character varying(64) [],  
    short_review text,  
    edition smallint NOT NULL,  
    publish_date date,  
    number_of_pages smallint,  
    CONSTRAINT "Primary Key table book" PRIMARY KEY (id) USING INDEX TABLESPACE  
    alternative_tablespace,  
    CONSTRAINT "Unique ISBN table book" UNIQUE ("ISBN") USING INDEX TABLESPACE  
    alternative_tablespace  
    ) WITH (FILLFACTOR=80, OIDS=FALSE)  
TABLESPACE alternative_tablespace;  
ALTER TABLE book OWNER TO librarian;
```

### 3.5 Indeksi

PostgreSQL pruža 4 različita načina indeksiranja:

- B-stablo,
- Hash,
- GiST i
- GIN.

Svaki indeks koristi drugi algoritam koji je optimiziran za pojedinu vrstu upita. Podrazumijevani algoritam je B-stablo budući da se on najčešće koristi.

B-stablo je pogodno za optimizaciju upita koji koriste operatore uspoređivanja (npr. operatori  $<$ ,  $<=$  u *SELECT* naredbi) i za operatore poput *BETWEEN* i *IN*. Hash indeksi su pogodni samo za jednostavne komparacije (npr. operator  $=$  u *SELECT* naredbi). GiST indeksi se najčešće koriste za optimiziranje tablica koje reprezentiraju dvodimenzionalno geometrijsko tijelo, dok su GIN indeksi namijenjeni za optimiziranje pretrage po objektima koji mogu sadržavati više od jedne vrijednosti, kao npr. polje.

Slijedećom SQL naredbom se stvara indeks *index\_book\_privatelibrary* nad tablicom *book*:

```
CREATE INDEX index_book_privatelibrary
  ON book USING btree ("ISBN");
```

### 3.6 Ostalo

PostgreSQL baza podataka ima još mnogo karakteristika čije bi detaljno opisivanje nadmašilo okvire ovog teksta. Ovdje su nabrojane neke od mogućnosti PostgreSQL sustava za upravljanje bazama podataka:

- PostgreSQL velikim dijelom podržava SQL:2008 standard. Na nekim mjestima postoji razlika u sintaksi zbog specifičnosti PostgreSQL baze podataka.
- Podržane su transakcije uz pomoć MVCC-a (eng. multi version concurrency control).
- Podržani su primarni i strani ključevi kao i okidači.
- Uz standardne tipove podataka podržani su i geometrijski tipovi podataka. Također korisnik može stvoriti vlastite tipove podataka.



## 4. PostgreSQL sustav za upravljanje bazama podataka i programski jezici

Programski jezici koji se koriste za proširivanje PostgreSQL-a se mogu podijeliti u dvije skupine:

- programski jezici namijenjeni izvršavanju na poslužiteljskoj strani i
- programski jezici namijenjeni izvršavanju na klijentskoj strani.

### 4.1 Programski jezici namijenjeni izvršavanju na poslužiteljskoj strani

PostgreSQL sustav za upravljanje bazama podataka može proširiti svoju osnovnu funkcionalnost ovisno o potrebama korisnika. Proširivanje funkcionalnosti se vrši na poslužiteljskoj strani, i ono može obuhvatiti definiranje korisničkih tipova podataka i funkcija, okidača...

PostgreSQL može izvršavati korisnički definirane funkcije koje nisu napisane u SQL-u ili C-u. Takvi jezici se zovu proceduralni jezici (eng. *procedural languages*, *PL*). PostgreSQL ne zna protumačiti tekst funkcije napisane u proceduralnom jeziku. Umjesto toga taj se zadatak prosljeđuje posebnom rukovoditelju koji taj zadatak može obaviti. Rukovoditelj nije ništa drugo nego obična C funkcija koja se učitava i koristi po potrebi.

PostgreSQL standardno podržava slijedeće proceduralne jezike:

- PL/pgSQL,
- PL/Tcl,
- PL/Perl i
- PL/Python.

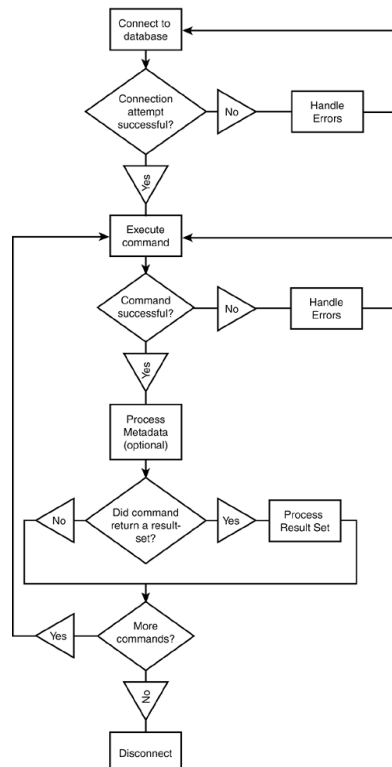
Postoje i drugi proceduralni jezici, ali oni nisu standardno uključeni u PostgreSQL bazu podataka te ih je potrebno naknadno postaviti.

## 4.2 Programski jezici namijenjeni izvršavanju na klijentskoj strani

Za izradu klijentske aplikacije koja će pristupiti PostgreSQL sustavu za upravljanje bazama podataka, potrebno je koristiti aplikacijsko programsko sučelje (eng. *application programming interface, API*). Klijentske aplikacije najčešće imaju korisnička sučelja i zbog toga ih najčešće koriste krajnji korisnici. Za svaki programski jezik programer može odabrati jedan ili više API-ja koje može koristiti:

- za jezik C - libpq, libpqeasy, ecpg, ODBC;
- za jezik C++ - libpq, libpq++, libpqeasy, ecpg, ODBC;
- za jezik Java - JDBC;
- za jezik Python - PyGreSQL...

Na slici 1. je prikazan osnovni tijek komunikacije između baze podataka i klijentske aplikacije.



Slika 1. Komunikacija između PostgreSQL-a i klijentske aplikacije

## 5. Osposobljavanje poslužitelja za rad s PostgreSQL sustavom za upravljanje bazama podataka

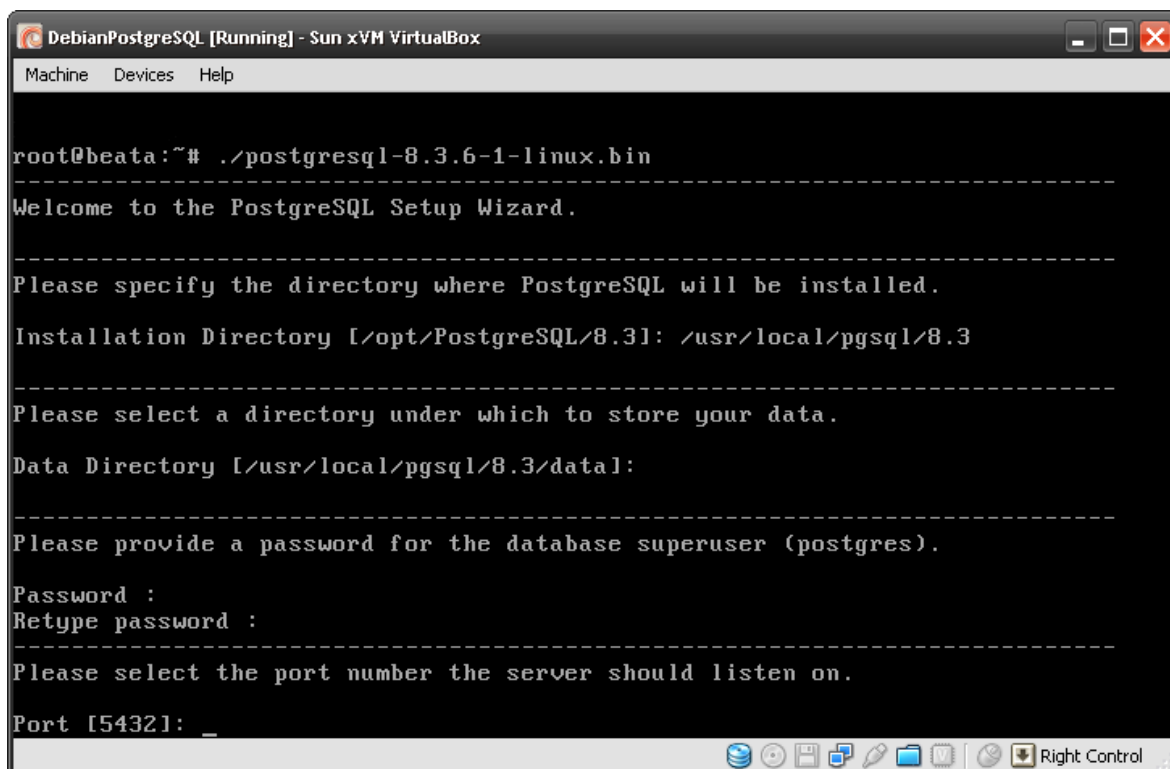
U ovom poglavlju opisan je postupak instalacije i konfiguracije PostgreSQL (verzija 8.3.6-1) sustava za upravljanje bazama podataka. Računalo na kojem će se izvršiti instalacija pokrenut je 32-bitni linux operativni sustav, Debian 5.0 Lenny. Prije početka instalacije potrebno je skinuti instalacijsku datoteku sa web stranice:

- <http://www.postgresql.org/download/linux>.

Nakon skidanja datoteke potrebno joj je promijeniti dozvole, i nakon toga se može započeti s instalacijom:

```
$ chmod +x postgresql-8.3.6-1-linux.bin
$ ./postgresql-8.3.6-1-linux.bin
```

Nakon zadavanja gornjih naredbi pokreće se instalacijski postupak. Prvo se od korisnika traži direktorij u kojem želi postaviti PostgreSQL sustav za upravljanje bazama podataka. U idućem koraku korisnik treba unijeti direktorij u kojem će se čuvati podaci. U sustav se dodaje novi superkorisnik postgres koji ima sve ovlasti za upravljanje nad svim bazama podataka u sustavu. Podrazumijevana vrata za uspostavljanje udaljenih konekcija prema sustavu su 5432 i ona se u postupku instalacije mogu promijeniti. Na slici 2. je prikazan prvi dio instalacije PostgreSQL sustava za upravljanje bazama podataka.



Slika 2. Početak instalacije PostgreSQL sustava za upravljanje bazama podataka

U instalacijskom postupku je još potrebno samo odrediti lokalizacijske postavke i tu završava postupak instalacije. Da bi se omogućilo spajanje s udaljenog računala na PostgreSQL sustav za upravljanje bazama podataka potrebno je u datoteci *postgresql.conf* (*/data/postgresql.conf*) promijeniti liniju "#listen\_addresses = 'localhost'" u "listen\_addresses = 'xyz.xyz.xyz.xyz'", pri čemu je niz 'xyz.xyz.xyz.xyz' potrebno promijeniti u stvarne vrijednosti. Isto tako u datoteci *pg\_hba.conf* (*/data/pg\_hba.conf*) potrebno je umjesto postojećih linija dodati slijedeće (potrebno je unijeti stvarne vrijednosti):

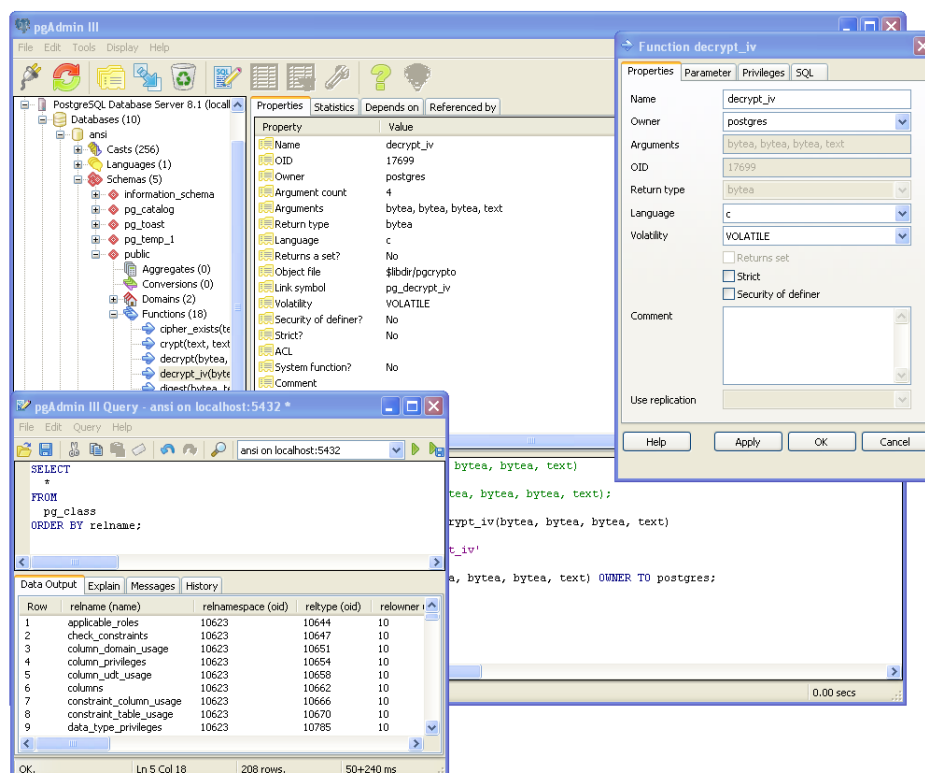
```
host    all    all    127.0.0.1/32    password
host    all    all    xyz.xyz.xyz /24    password
```

## 6. Alati za rad s PostgreSQL sustavom za upravljanje bazama podataka

PostgreSQL sustav za upravljanje bazama podataka ima otvoreni kôd i upravo tome može zahvaliti postojanje velikog broja ekstenzija i programa za administraciju. Postoji jedna cijela zajednica odvojena od samih PostgreSQL programera koja nastoji na jednom mjestu okupiti sve nadogradnje PostgreSQL sustava za upravljanje bazama podataka, riječ je o <http://pgfoundry.org/>. Jedan od poznatijih programa za administriranje i održavanje PostgreSQL-a je pgAdmin III.

### 6.1 pgAdmin III

pgAdmin III je jedna od najpopularnijih i najpotpunijih aplikacija s grafičkim sučeljem za administraciju sustava za upravljanje bazama podataka. pgAdmin III podržava PostgreSQL, EnterpriseDB, Mammoth PostgreSQL, Bizgres i Greenplum sustave.



Slika 3. pgAdmin III

pgAdmin III je dizajniran na način da može udovoljiti različitim potrebama korisnika, od pisanja jednostavnih SQL upita do razvoja kompleksnih baza podataka. Grafičko sučelje podržava sve funkcije PostgreSQL-a i uvelike olakšava administraciju i razvoj.

Aplikaciju razvija zajednica PostgreSQL programera diljem svijeta. Dostupna je u više jezika i naravno otvorenog je kôda i besplatna za korištenje.

## 7. Zaključak

Prema nekim predviđanjima 80% programskih paketa će do 2012. godine biti otvorenog kôda, što znači da će se moći koristiti i mijenjati bez naknade autoru. Danas, kada postoji mnogo različitih sustava za upravljanje bazama podataka, ponekad je teško odabrati onaj koji će zadovoljiti kvalitetom i cijenom. Današnji besplatni sustavi poput PostgreSQL-a mogu se uspoređivati i s komercijalnim sustavima za upravljanje bazama podataka. To najbolje potvrđuju velike tvrtke kao što su Skype i Yahoo koje koriste modificirane inačice PostgreSQL sustava za upravljanje bazama podataka za svoje potrebe.

PostgreSQL je do svoje 7. inačice bio na zlu glasu zbog svoje nestabilnosti i brzine. Za potrebe ovog Seminara postavio sam inačicu 8.3 s kojom nisam imao nikakvih problema. Iako je PostgreSQL stvarno odličan sustav za upravljanje bazama podataka, na kraju ipak moram spomenuti i neke stvari koje PostgreSQL nema u potpunosti riješene kao što su potpuna podrška za XML i objektno relacijske tipove podataka (npr. multiset).

U radu su opisane osnovne mogućnosti PostgreSQL baze podataka. S osnovnim znanjem SQL upita i općenitog poznavanja baza podataka ovaj seminar pruža dobre temelje za rad na PostgreSQL bazi podataka.

## 8. Literatura

1. izvor podataka o PostgreSQL sustavu za upravljanje bazama podataka

<http://www.postgresql.org/>

<http://www.postgresql.org/docs/8.3/static/index.html>

<http://pgfoundry.org/>

2. Douglas K., Douglas S. PostgreSQL: The comprehensive guide to building, programming, and administering PostgreSQL databases .Drugo izdanje. Internet izdanje. Sams Publishing, 2005.
3. Matthew N., Stones R. Beginning Databases with PostgreSQL: From Novice to Professional. Drugo izdanje. Internet izdanje. Apress, 2005.



## **9. Sažetak**

Cilj ovog seminara bio je dati pregled osnovnih mogućnosti PostgreSQL sustava za upravljanje bazama podataka. U seminaru dan je pogled na PostgreSQL sustav za upravljanje bazama podataka iz perspektive administratora sustava, administratora baze podataka i programera. Tim osobama bi trebao olakšati početne korake s PostgreSQL bazom podataka.