

Dijagrami komponenata

- Dijagram komponenata prikazuje fizičku organizaciju softverskog sistema
- Koriste se za modeliranje statičkog pogleda na implementaciju sistema
- Prikazuju se zavisnosti između komponenata sistema

Elementi dijagrama komponenata

- Dijagrami sadrže:
 - stvari: komponente, interfejse, pakete
 - relacije: zavisnosti, generalizacije, asocijacije, realizacije
- **Komponenta** može biti: izvorna datoteka, biblioteka, izvršna datoteka, tabela, dokument

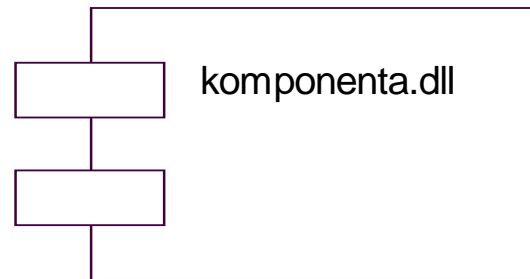


Komponenta

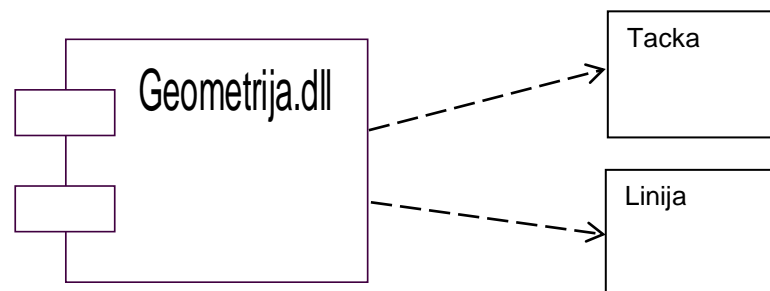
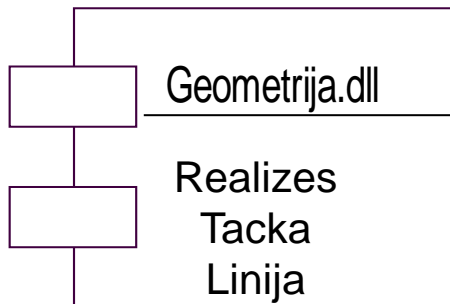
- Komponenta je fizički i zamenjivi deo sistema koji realizuje skup interfejsa
- Ime komponente može biti jednostavno ili sa putanjom

`(ime_paketa::ime_komponente)`

Grafička notacija:



Komponenta može da sadrži i odeljak klasa koje realizuje (Realizes)



Komponente i klase

- **Klasa** reprezentuje logičku apstrakciju, a **komponenta** fizičku stvar
"u svetu bitova"
- Komponenta predstavlja fizičko pakovanje logičkih apstrakcija
- Klase mogu imati atribute i operacije, a komponente samo operacije
- Servisi klase mogu biti pristupačni direktno, a servisi komponente
samo kroz interfejse



Komponente i interfejsi

- Interfejs je skup operacija koji se koristi da bi se specificirao servis klase ili komponente
- Komponenta realizuje jedan ili više interfejsa
- Standardi kao što su COM, CORBA, Enterprise Java Beans koriste interfejse

Primer realizacije interfejsa (skraćena i kanonička forma):



- Interfejs koji realizuje komponenta se naziva **izvoznim** (*export*) interfejsom
- Interfejs koji komponenta koristi se naziva **uvoznim** (*import*) interfejsom

Vrste komponenata

UML razdvaja 3 vrste komponenata:

- komponente za **isporuku** (exe, dll, COM, CORBA, Enterprise Java Beans, html, tabele)
- komponente iz **razvojnog procesa** (izvorne datoteke, datoteke podataka)
- **izvršne** komponente (kreirane kao posledica izvršenja, npr. COM objekat instanciran iz DLL-a)

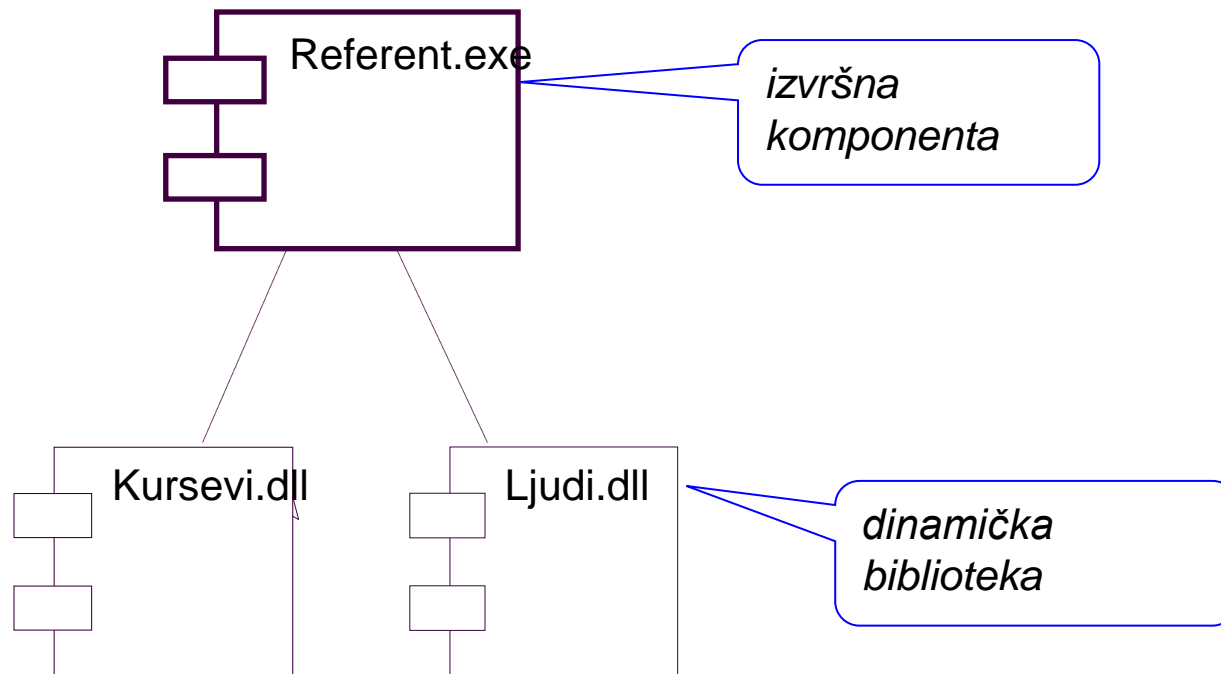
Standardni elementi

UML definiše sledeće standardne stereotipove:

- **executable** - komponenta koja se može izvršavati na čvoru
- **library** - statička ili dinamička objektna biblioteka
- **table** - tabela baze podataka
- **file** - izvorni kod ili podaci
- **document** - dokument



Primer dijagrama



Paketi

- Paketi na dijagramima komponenata sadrže druge pakete i komponente
- Oni se koriste da reprezentuju fizičke podsisteme
- Paketi tipično reprezentuju kataloge u sistemu datoteka
- Logički paketi iz dijagrama klasa se mapiraju u pakete u dijagramima komponenata

Relacija zavisnosti

Paketi i komponente se povezuju **relacijom zavisnosti** koja reprezentuje:

- zavisnost u vreme **prevođenja** kada se radi sa komponentama izvornog koda,
- zavisnost u vreme **povezivanja** kada se radi sa bibliotečkim komponentama, ili
- zavisnost u vreme **izvršenja** kada se radi sa izvršnim komponentama



Najčešće primene

- Modeliranje izvornog koda
- Modeliranje izvršnih izdanja
- Modeliranje fizičkih baza podataka

