


# Računarska grafika

JavaFX – 3D izbor



# Uvod

- Da bi se interagovalo sa nekim objektom u sceni i menjao mu se:
  - položaj
  - orijentacija
  - svojstva materijala
  - način prikaza (žični model/neproziran objekat, lica/naličja)potrebno ga je najpre izabrati
- Izbor se po pravilu vrši mišem
  - kurzor se dovede iznad objekta koji se želi izabrati
  - pritisne se dugme miša za izbor
- Procedura izbora objekta – pik-korelacija (*pick-correlation*)

# Izbor objekta u 3D sceni

- Osnovna ideja:
  - klikom miša dok je kurzor na nekoj poziciji u prikazu scene bira se objekat čija projekcija obuhvata poziciju kurzora
  - ako više projekcija objekata obuhvata poziciju kurzora bira se objekat koji je nablži posmatraču
- Problemi izbora: intuitivnost i određenost
  - ako je prikazan žični model objekta, izbor bliskog može biti neintuitivan kad je kliknuto na objekat u pozadini, vidljiv kroz mrežu bliskog objekta
  - ako je objekat složen, koji nivo objekta u kompoziciji delova se bira: najniži (najbliži listu u stablu kompozicije) ili neki viši?

# Izbor objekta u JavaFX

- Izbor (selekcija) objekata u 3D sceni
- Obraduje se događaj miša prouzrokovan „klikom na objekat“
  - parametar rukovaoca (ili lambda funkcije) je referenca:  
`MouseEvent dog`
- Pozove se metod: `dog.getPickResult()`
- Metod vraća rezultat tipa `javafx.scene.input.PickResult`
- Klasa `PickResult` predstavlja apstrakciju rezultata pik-korelacije (određivanja objekta u 3D sceni na koji je pokazivao kurzor)
- Ukoliko pokazana tačka pripada projekcijama više objekata:
  - rezultat `PickResult rez` će pokazivati na najbliži objekat kameri

# Dohvatanje izabranog objekta

- Za objekat rezultata (klase `PickResult`) pozove se metod `rez.getIntersectedNode()`
- Metod vraća referencu na izabrani čvor (klase `Node`) grafa scene
- Ukoliko su u sceni samo objekti klase `Shape3D`
  - može se konverzijom naniže konvertovati rezultat tipa `Node` u referencu na objekat tipa `Shape3D`
  - nakon toga sa datim objektom može nešto da se radi kao sa 3D oblikom

# Metodi klase `PickResult` (1)

- `Point3D getIntersectedPoint()`
  - vraća tačku preseka u lokalnom koord. sistemu izabranog objekta
  - tačka preseka se određuje tako što projekcioni zrak prolazi kroz poziciju kamere i dodirnutu tačku u prikaznoj ravni
  - tačka preseka je tačka u kojoj zrak prodire stranicu mreže objekta
- `double getIntersectedDistance()`
  - vraća rastojanje između kamere i tačke preseka na površi objekta
- `int getIntersectedFace()`
  - vraća stranicu izabranog čvora, kojoj pripada tačka preseka
  - ako čvor nema korisnički definisane stranice ili ako je čvor dodirnut na ivici, vraća se vrednost `FACE_UNDEFINED`

## Metodi klase `PickResult` (2)

- `Point3D getIntersectedNormal()`
  - vraća normalu na površ u tački preseka
- `Point2D getIntersectedTexCoord()`
  - vraća teksturne koordinate u tački preseka izabranog objekta

# Primer (1)

```
import javafx.scene.shape.Shape3D;
import javafx.scene.input.PickResult;
...
Group koren = new Group(valjak, kvadar, lopta, svetlo);
...
enum Izbor {KVADAR, VALJAK, LOPTA};
private Shape3D telo = null;
private PhongMaterial matStari = new PhongMaterial();
private boolean izabrano=false;
...
```



## Primer (2)

```
EventHandler<MouseEvent> r1 = dog -> { // obrada pritiska
    if (izabrano) return;
    PhongMaterial mat = new PhongMaterial();
    mat.setDiffuseColor(Color.CHARTREUSE);
    Izbor izbor=null;
    PickResult rez = dog.getPickResult();
    if (rez!=null) telo = (Shape3D) rez.getIntersectedNode();
    else telo = null;
```

## Primer (3)

```
if(telo != null){
    if (telo instanceof Box) izbor = Izbor.KVADAR;
    else if (telo instanceof Cylinder) izbor = Izbor.VALJAK;
    else if (telo instanceof Sphere) izbor = Izbor.LOPTA;
    matStari=(PhongMaterial)telo.getMaterial();
    telo.setMaterial(mat);
    izabrano = true;
}
}; // obrada događaja pritiska miša
scena.addEventHandler(MouseEvent.MOUSE_PRESSED, r1);
```

## Primer (4)

```
EventHandler<MouseEvent> r2 = dog -> { // obrada puštanja
    if (telo!=null) telo.setMaterial(matStari);
    izabrano = false;
}; // obrada događaja puštanja miša
scena.addEventHandler(MouseEvent.MOUSE_RELEASED, r2);
```

# Rezultat izvršenja

