

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA  
LABORATORIJ ZA INTERAKTIVNE SIMULACIJSKE SUSTAVE

**Interaktivni simulacijski sustavi**

**Iscrtavanje**

Prof.dr.sc. Krešimir Ćosić  
Dr.sc. Siniša Popović

LISS, FER, Zagreb

1

**Iscrtavanje**

- ▶ Postupak stvaranja 2D slike iz 3D virtualne scene
- ▶ Obuhvaća niz različitih postupaka
- ▶ Ulaz
  - Virtualna scena, u kojoj se nalaze 3D modeli predstavljeni na odgovarajući način i izvori svjetla
  - Virtualna kamera (ili promatrač) s parametrima poput položaja, orijentacije, širine vidnog polja itd.
- ▶ Izlaz
  - 2D slika na projekcijskom platnu kamere koja se onda može prikazati na raznim prikaznim uređajima

LISS, FER, Zagreb

2

**Iscrtavanje**

- ▶ Značaj u simulatorima – dovoljno brzo generiranje dovoljno kvalitetne slike virtualne scene da bi obuka u simulatoru doprinijela kvaliteti obavljanja zadatka u stvarnom svijetu




LISS, FER, Zagreb

3

**Iscrtavanje**

- ▶ Značaj u simulatorima – dovoljno brzo generiranje dovoljno kvalitetne slike virtualne scene da bi obuka u simulatoru doprinijela kvaliteti obavljanja zadatka u stvarnom svijetu
  - Sjetimo se pojmova **brzine iscrtavanja** i **ukupnog kašnjenja** iz ranijeg predavanja i problema do kojih mogu dovesti u simulatorima
    - Pojava simptoma simulatorske bolesti – dezorijentacije i mučnine
    - Degradacija djelotvornosti obuke zbog neprirodne interakcije
    - Smanjenje osjećaja prisutnosti – problem ako u simulatoru želimo proizvesti jači emocionalni efekt tijekom vježbe

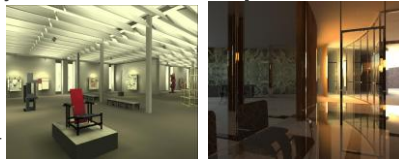
LISS, FER, Zagreb

4

## Sveti gral iscrtavanja

### ► Fotorealistična iscrtana slika

- **Fotorealističan** – odgovara fotografiji stvarnog svijeta
- Izgled fotografije prvenstveno posljedica vrlo složene interakcije vrlo raznovrsnih materijala i svjetala u prirodi
- Računala nailaze na probleme u postizanju fotorealističnosti, posebice zbog ograničene računske moći za simuliranje tih složenih interakcija



LISS, FER, Zagreb

5

## Sveti gral iscrtavanja

### ► Fotorealistična iscrtana slika

- Tijekom iscrtavanja samo se aproksimira interakcija svjetla i materije u stvarnom svijetu; postoje aproksimacije raznih točnosti
- **Phongov model osvjetljenja** – **lokalni** model osvjetljenja, uzima u obzir samo svjetlo koje dolazi direktno iz izvora svjetla. Sve refleksije svjetla u sceni između objekata, koje se u ovom modelu ne promatraju, aproksimiraju se ambijentalnim svjetlom
- **Praćenje zrake** (ray tracing) – jedan **globalni** model osvjetljenja, uzima u obzir i indirektne izvore svjetla kao refleksije svjetla od jednog objekta u sceni na drugi

LISS, FER, Zagreb

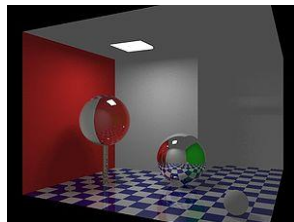
6

## Sveti gral iscrtavanja

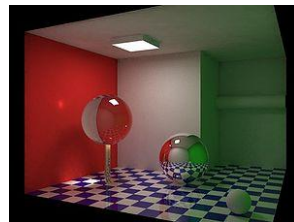
### ► Fotorealistična iscrtana slika

- Igra "uoči 5 razlika"

lokalni model



globalni model

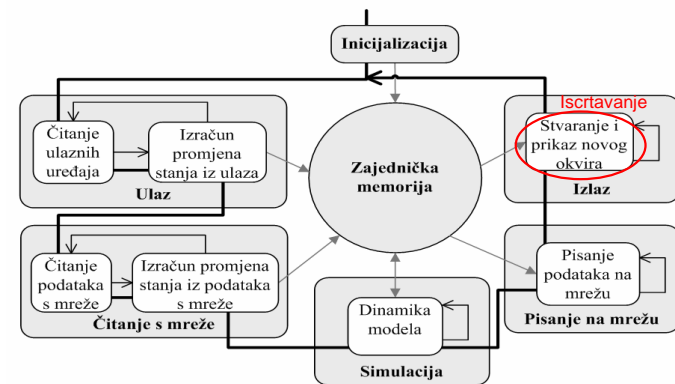


[http://en.wikipedia.org/wiki/Global\\_illumination](http://en.wikipedia.org/wiki/Global_illumination)

LISS, FER, Zagreb

7

## Generička simulacijska petlja otprije – gdje je iscrtavanje?

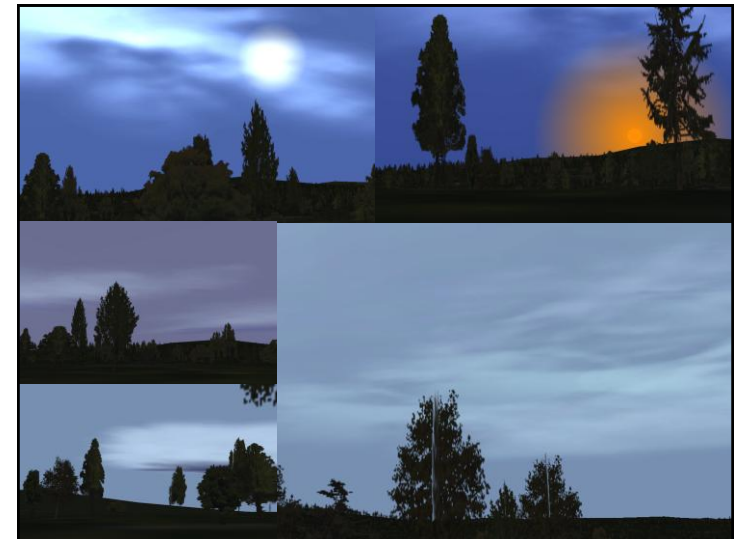
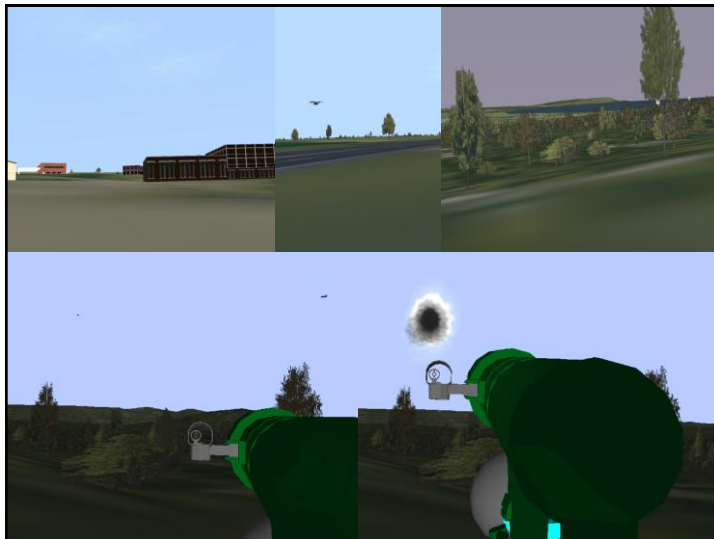
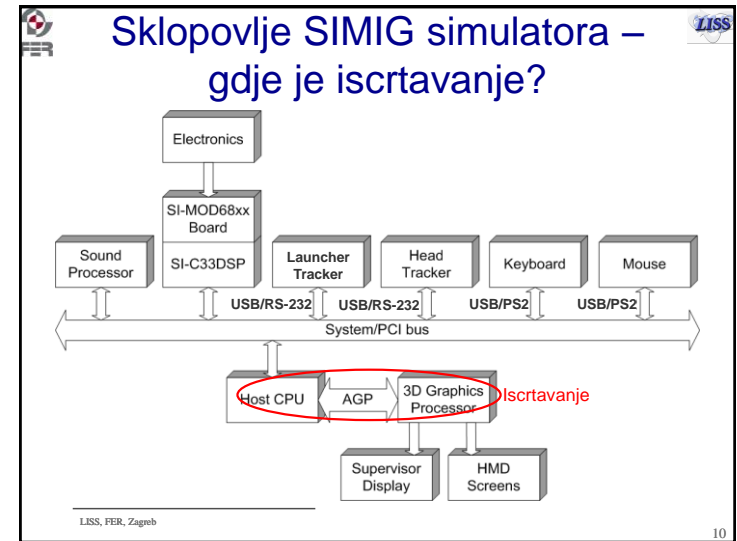
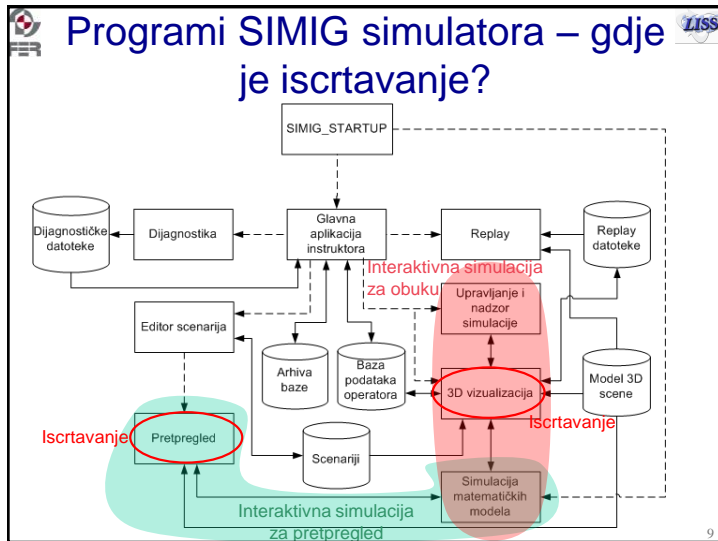


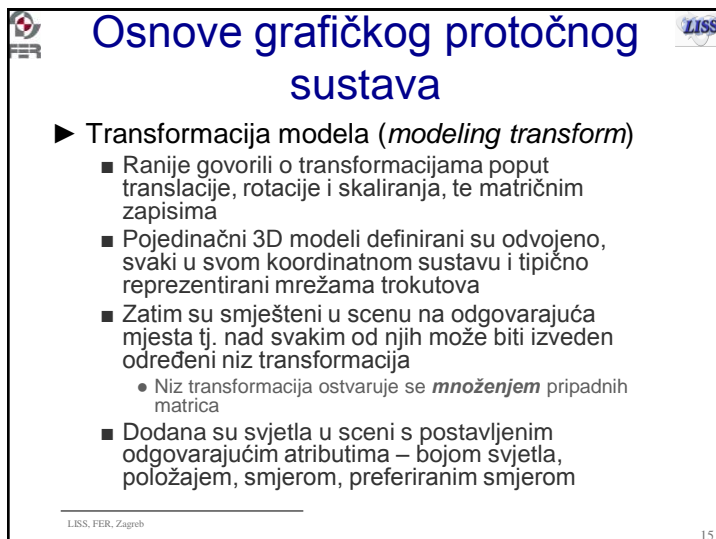
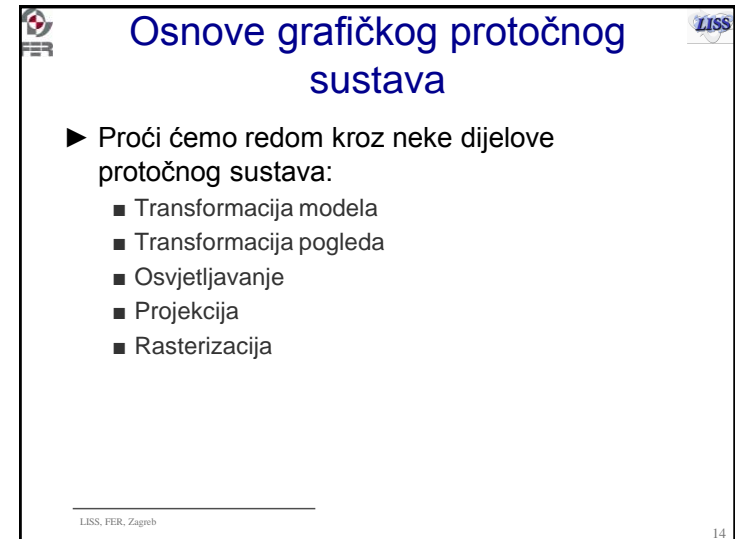
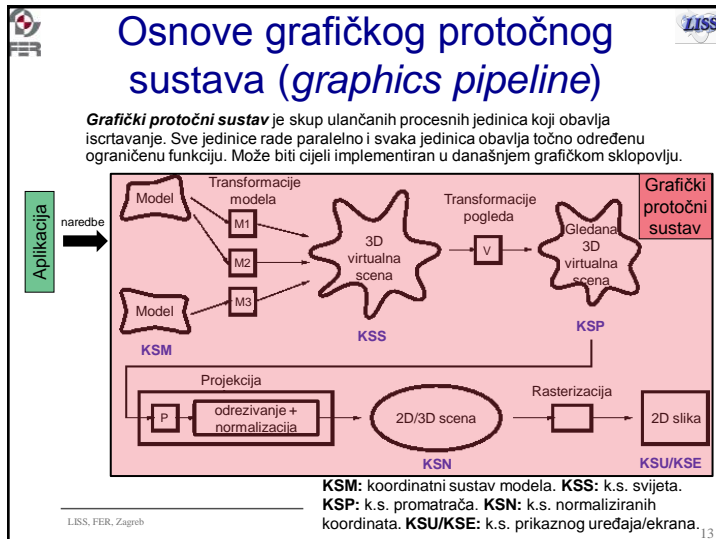
→ tok izvođenja (višedretvena arhitektura)

→ tok podataka

→ tok izvođenja (jednodretvena arhitektura)

8





## Osnove grafičkog protočnog sustava

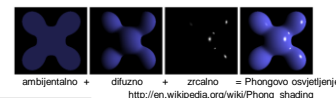
### ► Transformacija pogleda (*view transform*)

- U sceni je pozicioniran promatrač/kamera, zadan sa svojim položajem i orijentacijom (i drugim parametrima kao što je npr. vidno polje koje je nebitno za transformaciju pogleda)
- Iz koordinatnog sustava svijeta (KSS), koordinate točaka svih 3D modela te položaji i smjerovi svjetala transformiraju se u koordinatni sustav promatrača (KSP)

## Osnove grafičkog protočnog sustava

### ► Osvjetljavanje (*lighting*)

- Kad su poznata svjetla i svi objekti u sceni (dakle i materijali), moguće je izračunati boju za pojedini vrh ili po pojedinom trokutu
  - Moguće je specificirati da li želimo računati po vrhu ili trokutu
  - Bitno – u ovom dijelu se boja samo računa i zapamti, ali trokut se ne ispunjava sada
- Koristi se **Phongov model osvjetljenja** (*Phong reflection model*)



## Osnove grafičkog protočnog sustava

### ► Osvjetljavanje

#### ■ Phongov model osvjetljenja

$$I_p = k_a i_a + \sum_{\text{svi izvori svjetla u sceni}} (k_d (L \cdot N) i_d + k_s (R \cdot V)^{\alpha} i_s)$$

boja točke p u sceni      ambijentalna      difuzna      zrcalna

- Boja točke  $p$  = zbroj interakcija **ambijentalne**, **difuzne** i **zrcalne** komponente **refleksivnosti materijala** ( $k$ ) i **svjetala** ( $i$ ) u sceni
- Ambijentalna komponenta – utjecaj sveprisutnog svjetla, koje **ne dolazi iz posebnog smjera niti posebnog izvora**
- Difuzna komponenta – svjetlo koje dolazi iz određenog izvora doprinosi boji u mjeri u kojoj se smjer upada svjetla poklapa s normalom objekta u točki  $p$  (**ne ovisi o promatraču**)
- Zrcalna komponenta – svjetlo koje dolazi iz određenog izvora doprinosi boji točke  $p$  u mjeri u kojoj reflektirana zraka svjetla ide prema promatraču (stvara odsjaj)

## Osnove grafičkog protočnog sustava

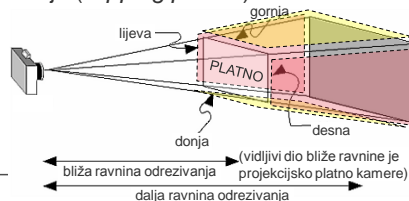
### ► Projekcija

- Zanima nas posebno samo perspektivna projekcija, ima i drugih tipova projekcije
- **Perspektivna projekcija** – scena se projicira na ravninu po pravcima koji svi prolaze kroz oko promatrača/kamere
- Što je objekt dalje od kamere, to će njegova projicirana slika biti manja

## Osnove grafičkog protočnog sustava

### ► Projekcija

- Primjenjuje se pojednostavljen model promatrača/kamere tzv. **pinhole camera** – nema leća, “zjenica”/“otvor blende” je veličine jedne točke
- Vidno polje kamere može se zadati definiranjem **projekcijskog volumena** (*view frustum*) pomoću ravnina odrezivanja (*clipping planes*)



LISS, FER, Zagreb

21

## Osnove grafičkog protočnog sustava

### ► Rasterizacija

- Postupak pretvaranja projicirane slike iz **vektorske** reprezentacije (matematička reprezentacija pojedinih projiciranih trokuta) u **rastersku** reprezentaciju (pokrivenost trokuta pikselima uređaja za prikaz).
- Obuhvaća sjenčanje, teksturiranje, razne testove poput testa dubine, stapanje na temelju prozirnosti (*alpha blending*) itd. Nešto detaljnije promotrit ćemo sjenčanje i testiranje dubine
- **Sjenčanje** (*shading*) – određivanje boje svih piksela koji pripadaju projiciranom trokutu na temelju boja pripisanih vrhovima/trokutima tijekom osvjetljavanja

LISS, FER, Zagreb

22

## Osnove grafičkog protočnog sustava

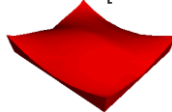
### ► Rasterizacija

- Zapravo se u sjenčanju ne određuje odmah boja piksela, već tzv. *fragmenti* koji predstavljaju kandidate za piksele i u sebi sadrže još informacija osim koordinata i boje. U okviru ovog kolegija nije neophodno raditi razliku, pa sve zovemo ‘pikseli’
- Dvije jednostavne vrste sjenčanja – vizualne razlike između njih dosta su očite:

konstantno



Gouraudovo [Gu ró ovo]



LISS, FER, Zagreb

[http://freespace.virgin.net/hugo.elias/graphics/x\\_polygo.htm](http://freespace.virgin.net/hugo.elias/graphics/x_polygo.htm)

23



## Osnove grafičkog protočnog sustava

### ► Rasterizacija

- Konstantno sjenčanje (*flat shading*)
  - Tijekom osvjetljavanja ili direktnim zadavanjem boje, jedna konstantna boja određena je za cijeli trokut
  - Nakon projekcije, svi pikseli slike koji pripadaju projiciranom trokutu obojit će se tom bojom
- Gouraudovo sjenčanje
  - Tijekom osvjetljavanja ili direktnim zadavanjem boje, boja je određena za svaki pojedini vrh trokuta – različiti vrhovi istog trokuta stoga općenito imaju različite boje
  - Nakon projekcije, pikseli slike koji pripadaju projiciranom trokutu obojit će se interpoliranjem između boja vrhova
  - Složenije računanje od konstantnog, ali nije problem današnjem grafičkom sklopovlju – Gouraud se često koristi

LISS, FER, Zagreb

24

## Osnove grafičkog protočnog sustava



► Rasterizacija

- **Z spremnik** – jedna od tehnika uklanjanja skrivenih ploha tj. onih koje nisu vidljive promatraču
- Također poznat kao **spremnik dubine scene** (*depth buffer*) – na mjesto svakog piksela sprema treću koordinatu promatraču najbliže točke u sceni, u odnosu na sve druge točke koje su se iz scene projicirale u taj piksel
  - Zašto Z spremnik? Iz običaja da u ravni projekcije govorimo o x i y koordinatama pa je z onda treća koordinata (tj. dubina). Npr. OpenGL ima tu konvenciju – u KSP-u pogled je usmjeren duž negativne z-osi pa je ravnina projekcije paralelna s xy ravninom

---

LISS, FER, Zagreb

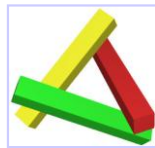
25

## Osnove grafičkog protočnog sustava

► Rasterizacija



- Z spremnik omogućuje točno iscrtavanje objekata koji se međusobno djelomično zakrivaju
- U tom smislu, predstavlja bitno poboljšanje spram jednostavnog **slikarskog algoritma** (*painter's algorithm*)
  - Taj algoritam pokušava osigurati eliminaciju skrivenih ploha crtajući objekte od najdaljeg prema najbližem promatraču
  - Slikarskim algoritmom npr. ne bi bilo moguće korektno iscrtati objekte na slici, a Z spremnikom da



---

LISS, FER, Zagreb


26

## Osnove grafičkog protočnog sustava

► Rasterizacija

- Velika prednost Z spremnika je što ne trebamo voditi računa kojim redoslijedom iscrtavamo
- Primjer iscrtanog automobila u OpenGL-u bez i sa upotrebom Z spremnika





<http://kyrah.net/scratch/opengl/cg-intro-2-reloaded.pdf>

---

LISS, FER, Zagreb

27

## Osnove grafičkog protočnog sustava

► Rasterizacija

- Algoritam Z spremnika je jednostavan za implementaciju i brzo se izvodi
- Z spremnik očito zauzima memorijski prostor (dubinska informacija se čuva za svaki piksel)
- Na osobnim računalima ta količina memorije nije problematična i stoga se Z spremnik široko upotrebljava u današnjem grafičkom sklopovlju

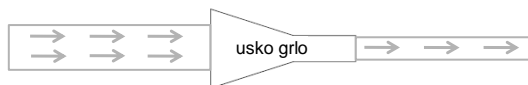
---

LISS, FER, Zagreb

28

## Problemi kod iscrtavanja

- ▶ Svaka obrada u grafičkom protočnom sustavu traje određeno vrijeme
- ▶ Protočni sustav radi brzinom najsporije procesne jedinica, koja predstavlja **usko grlo** sustava (*bottleneck*)
- ▶ Za optimiziranje performansi unutar grafičkog protočnog sustava treba ustanoviti gdje je usko grlo



LISS, FER, Zagreb

29

## Što utječe na brzinu iscrtavanja?

- ▶ Prijenos podataka u grafički protočni sustav (vrhovi, teksture) – to je dio prije grafičkog protočnog sustava (slanje naredbi iz aplikacije)
- ▶ Transformacije
- ▶ Izračun osvjetljenja
- ▶ Projekcija
- ▶ Sjenčanje
- ▶ Teksturiranje
- ▶ Test dubine (ažuriranje stanja Z spremnika)
- ▶ Drugi testovi i upotreba nekih naprednijih mogućnosti

LISS, FER, Zagreb

30

## Postupci optimizacije

- ▶ Ne ulazimo detaljnije u načine ustanovljavanja uskih grla unutar protočnog sustava
- ▶ Opisat ćemo samo nekoliko tehnika kojima se u grafički protočni sustav može poslati manja količina podataka iz aplikacije **ali bez narušavanja percepcije složenosti scene**
  - Odbacivanje geometrije (*culling*)
  - Razine detalja
  - Straničenje (*paging*) terena

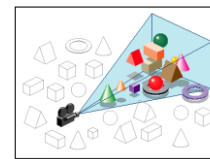
LISS, FER, Zagreb

31

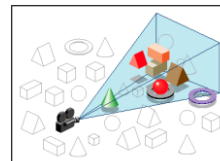
## Odbacivanje geometrije



Bez odbacivanja svi vrhovi i trokuti ulaze u grafički protočni sustav



Odbacivanje na temelju projekcijskog volumena (*view frustum culling*) – u grafički protočni sustav šaljemo samo dio scene vidljiv promatraču  
**Ovo ćemo detaljnije promotriti**



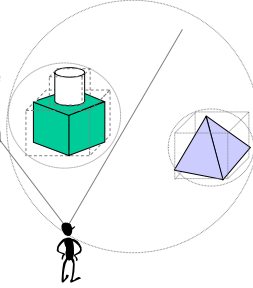
*View frustum culling* + odbacivanje zaklonjenih objekata (*occlusion culling*)

32



## Odbacivanje na temelju projekcijskog volumena

- Odbacivanje na temelju projekcijskog volumena
  - Obavlja se na razini grafa scene, prije nego se vrhovi šalju u grafički protočni sustav
  - Upotrebljavaju se jednostavne obujmice i one se testiraju spram projekcijskog volumena
    - Sfera obujmica (*bounding sphere*)
    - Kvadar obujmica (*bounding box*) – kako ga naći ako znamo koordinate svih vrhova objekta?



LISS, FER, Zagreb

33

## Odbacivanje na temelju projekcijskog volumena

- Za ovakvo odbacivanje bitna je prostorna organizacija baze podataka

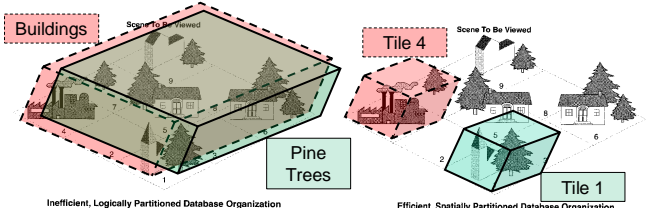


LISS

34

## Odbacivanje na temelju projekcijskog volumena

- Čvorovi odmah ispod korijena u prostorno organiziranoj bazi imaju disjunktne obujmice – to povećava vjerojatnost da će se neki od njih odbaciti jer mu obujmica neće presjecati projekcijski volumen (za razliku od čvorova pod korijenom u logički organiziranoj bazi)

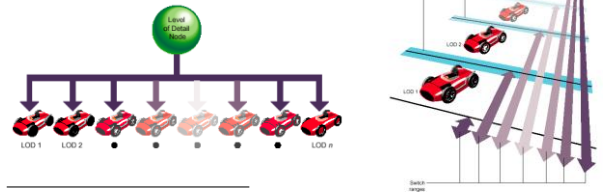


LISS

35

## Razine detalja (level of detail, LOD)

- LOD čvorovi postoje u grafu scene – sadrže 3D modele istog objekta u različitim razinama detalja – ranije vidjeli kako se takvi modeli mogu izgraditi
  - Tijekom iscrtavanja LOD čvorovi omogućuju odabir odgovarajućeg modela tj. djeteta, na temelju trenutne udaljenosti modela od promatrača/kamere
  - Može se koristiti stapanje (*blending*) radi manje uočljivog prijelaza između susjednih razina detalja



LISS, FER, Zagreb

Slika preuzeta iz IRIS Performer™ Programmer's Guide

36

## Straničenje terena

- ▶ Optimizacija iscrtavanja terena koji su preveliki da bi čitavi stali u memoriju računala i imaju previše poligona da bi ih cijele bilo moguće prikazati interaktivnom brzinom iscrtavanja
- ▶ Straničenje terena je varijacija na temu razina detalja
  - 3D model terena izradi se u više razina detalja
  - Ali ima još...
 

Za razliku od "običnih", ove razine detalja nikad nisu sve učitane u memoriju – potrebne razine detalja učitavaju se s tvrdog diska kada je potrebno tj. kada im se promatrač u sceni dovoljno približi

LISS, FER, Zagreb

37

## Straničenje terena

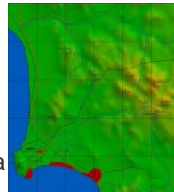
- ▶ S diska se dohvaćaju dijelovi kojima se korisnik približava u sve većoj razlučivosti
- ▶ Previsoke razlučivosti onih dijelova spram kojih se korisnik udaljava vraćaju se na disk te se umjesto njih učitavaju niže razine detalja
- ▶ Teren se tipično dijeli na osnovne ploče, koje se onda mogu predstaviti različitim razinama detalja

LISS, FER, Zagreb

38

## Primjeri u SIMIG simulatoru

- ▶ Odbacivanje je ugrađeno u OpenSceneGraph programsku podršku koja implementira graf scene simulatora
- ▶ LOD čvorovi također su ugrađeni u OpenSceneGraph
- ▶ Straničenje terena podržano je od strane:
  - Terra Vista alata tijekom 3D modeliranja terena
    - 6 x 5 ploča, 3 LOD-a po ploči
  - OpenSceneGrapha tijekom iscrtavanja



LISS, FER, Zagreb

39

## Razine alata za iscrtavanje

ALATI ZA ISCRTAVANJE

SIMIG 3D vizualizacija, Serious Sam, UT, Quake, web aplikacija

aplikacije

OpenSceneGraph, VRML, Serious Engine, Unreal Engine

međuprogrami (middleware):  
grafovi scene,  
pogoni računalnih igara

OpenGL, Direct3D

programska  
sučelja za grafiku

nVidia GeForce \*, ATI Radeon \*

GRAFIČKO SKLOPOVLJE  
(graphics processing units, GPUs)

LISS, FER, Zagreb

40



## Zahvale



- ▶ Dio materijala u ovom predavanju preuzet je ili prilagođen iz nastavnih materijala na sljedećim adresama
  - <http://graphics.stanford.edu/courses/cs248-07/>
  - <http://medialab.di.unipi.it/web/IUM/Waterloo/html.html>