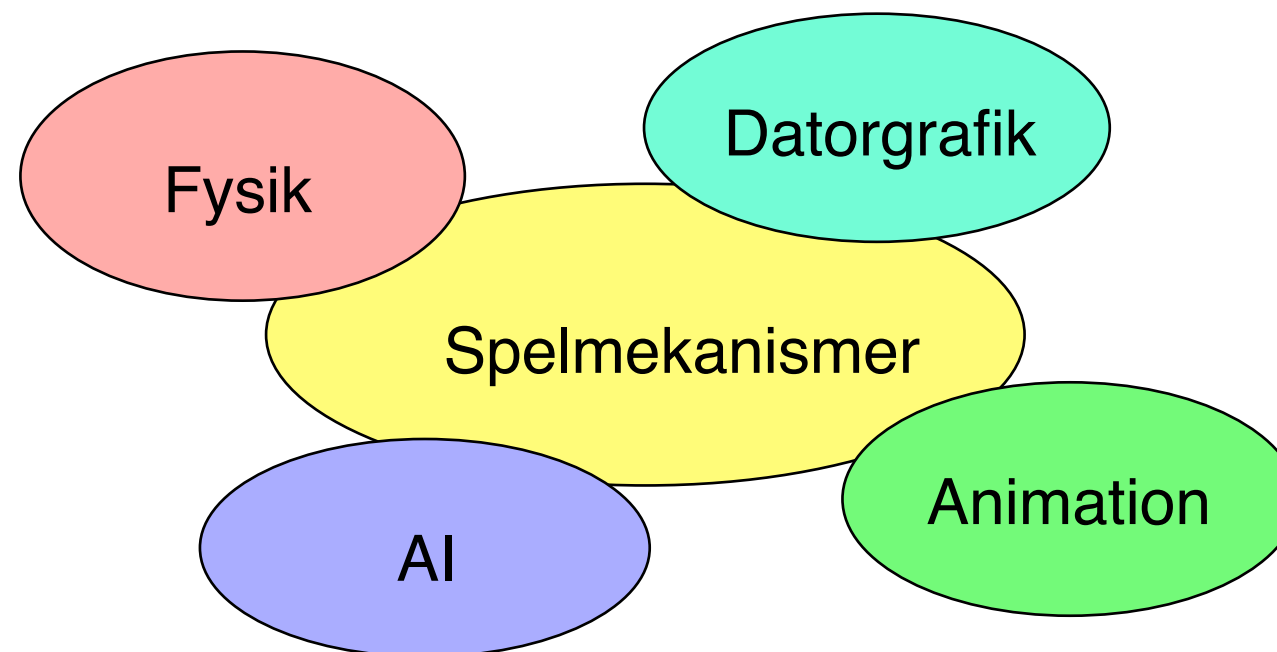




TSBK 03

Teknik för avancerade datorspel
Ingemar Ragnemalm, ISY





Fö 5 (6)

- Volymdata
- Animation: Kinematik, skinning
 - Kvaternjoner



Föreläsningsfrågor

1. Hur kan man göra en voxelvolym av en 3D-modell?
2. Vad är skillnaden mellan framåtkinematik och inverskinematik?
3. Hur kan inverskinematik utföras?
4. Vad är skillnaden mellan stitching och skinning?
5. Hur påförs animationer i skinning?
6. Vad är fördelen med kvaternjoner?



Texturer i 3 dimensioner

Modell till textur

Hur kan du göra en modell eller scen till en 3D-textur?

Textur till modell

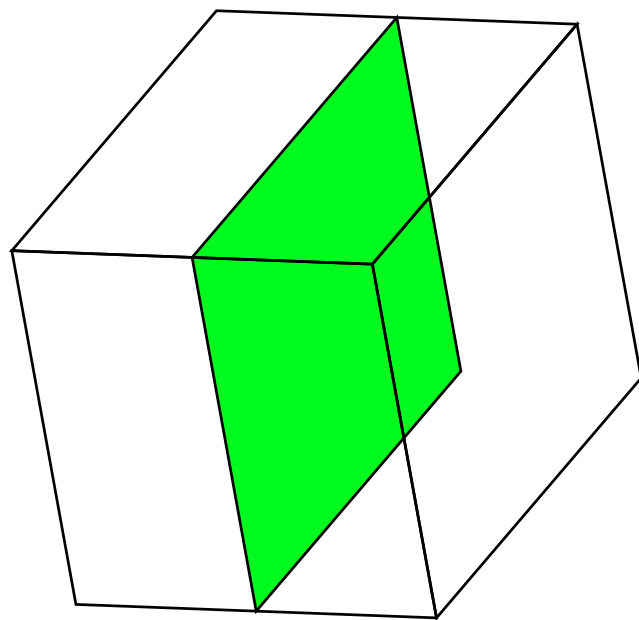
Hur kan du rendera volumetriska data?



Rendera till 3D-volym

Görs normalt lager för lager.

Bind ett lager till en FBO. Rita normalt.



```
glFramebufferTexture3D( GL_FRAMEBUFFER,  
Attachment, GL_TEXTURE_3D, TextureID, 0,  
zSlice);
```

↑ Anger lager att rita till



Modell till textur = Voxelisering

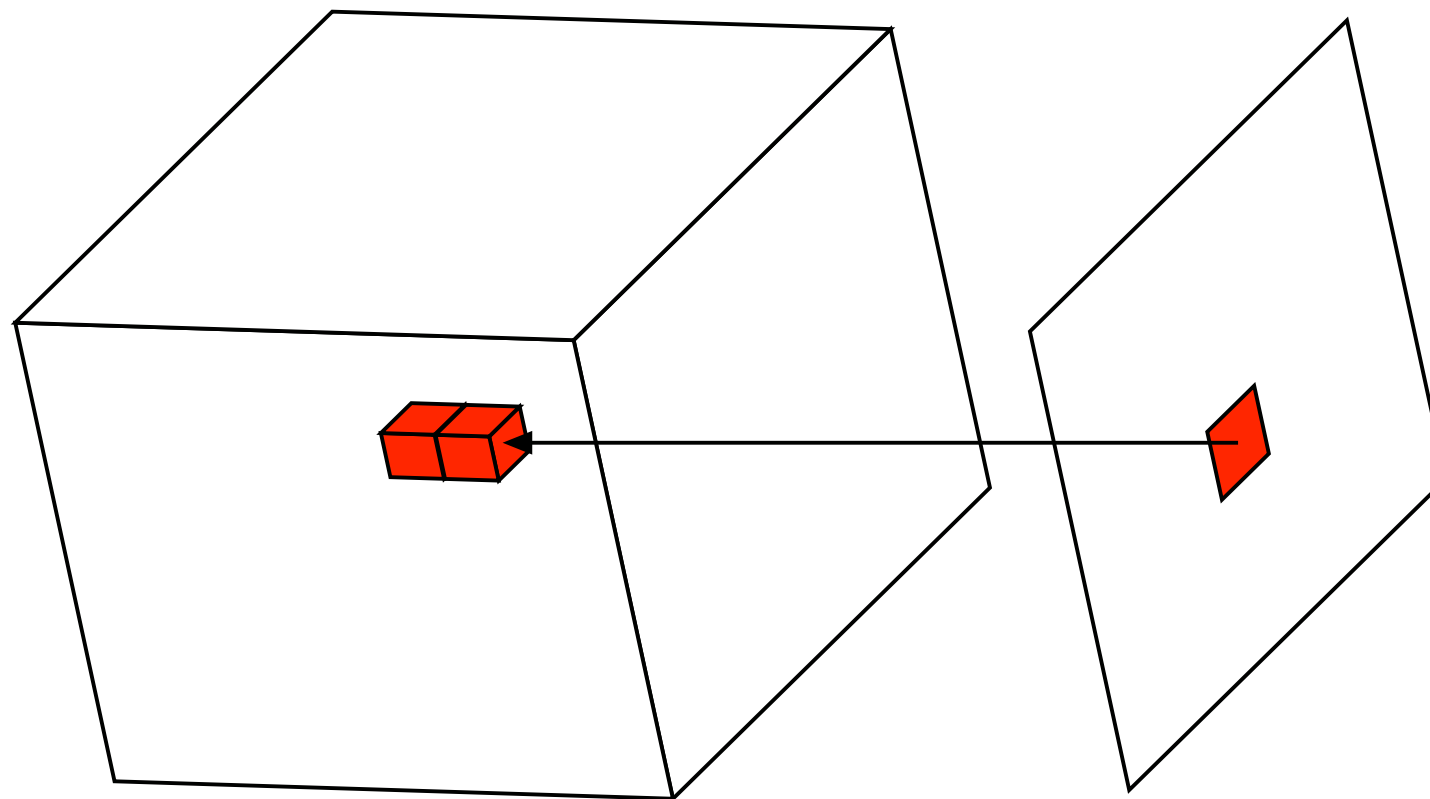
Två metoder:

- Rendering av polygoner till 3D-textur
- Rendering till Z-buffer, skapa 3D-textur från Z-buffer



Voxelisering genom polygonrendering

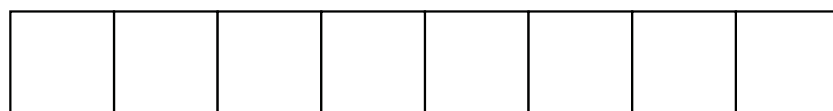
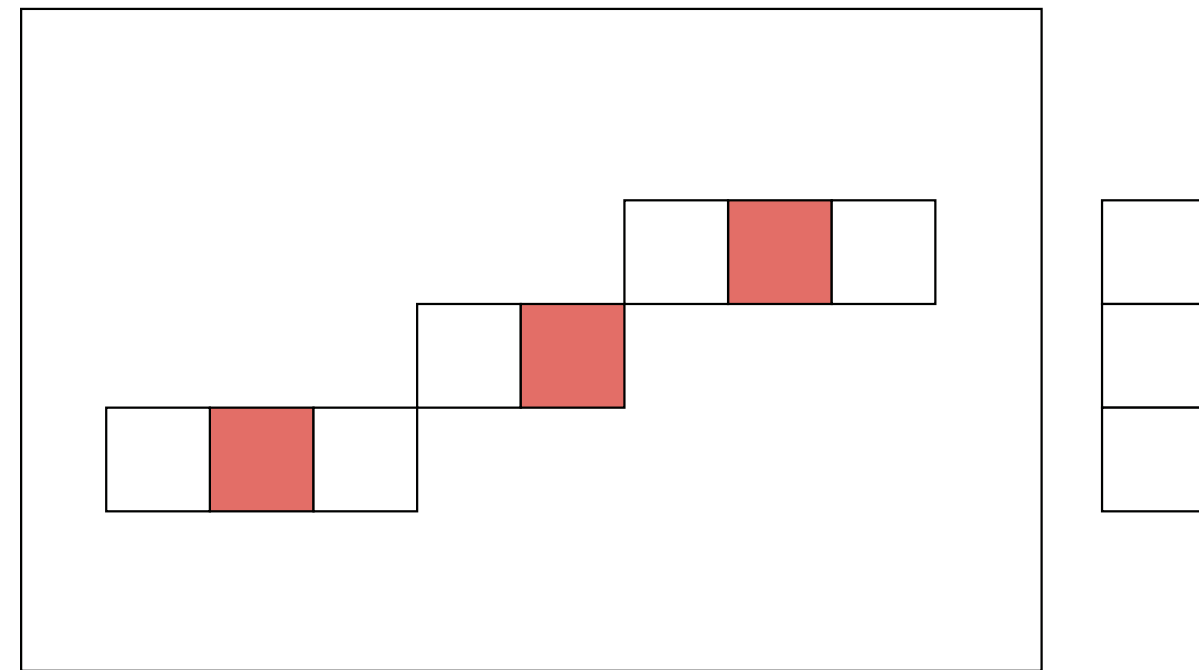
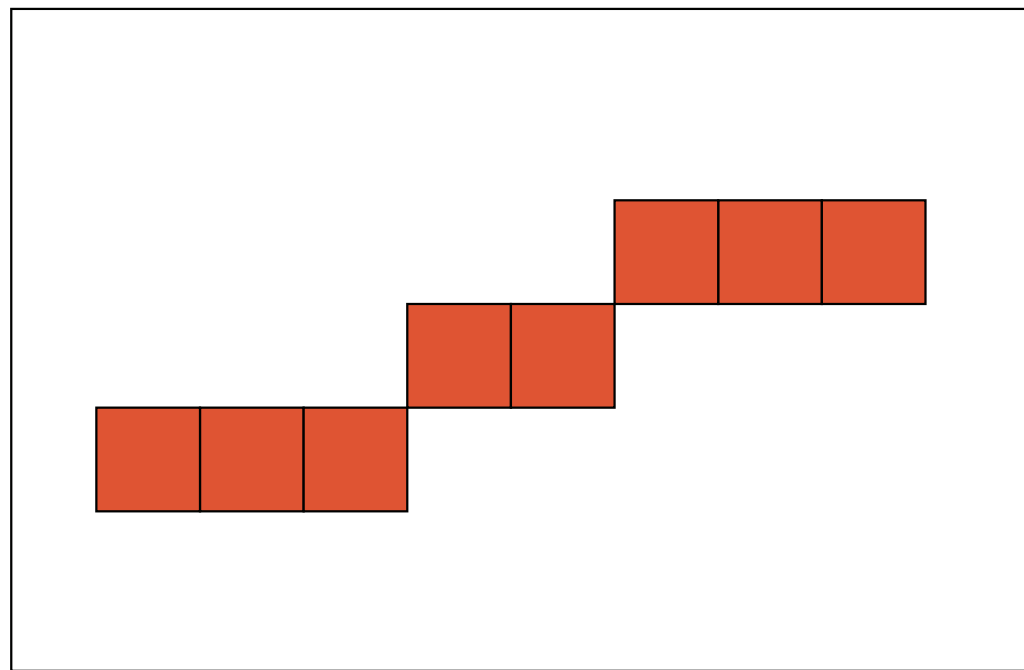
Varje pixel mappas på en eller flera voxlar



"Write to texture". Alltid på unik x,y i volym = konfliktfritt!



Samma problem som vid linjeritning: Måste välja rätt axel att arbeta från!





Ändå problem med luckor!

Lösning: Conservative rasterization: Varje voxel som vidrörs av ett primitiv skall fyllas.

Jfr 4-konnektivitet



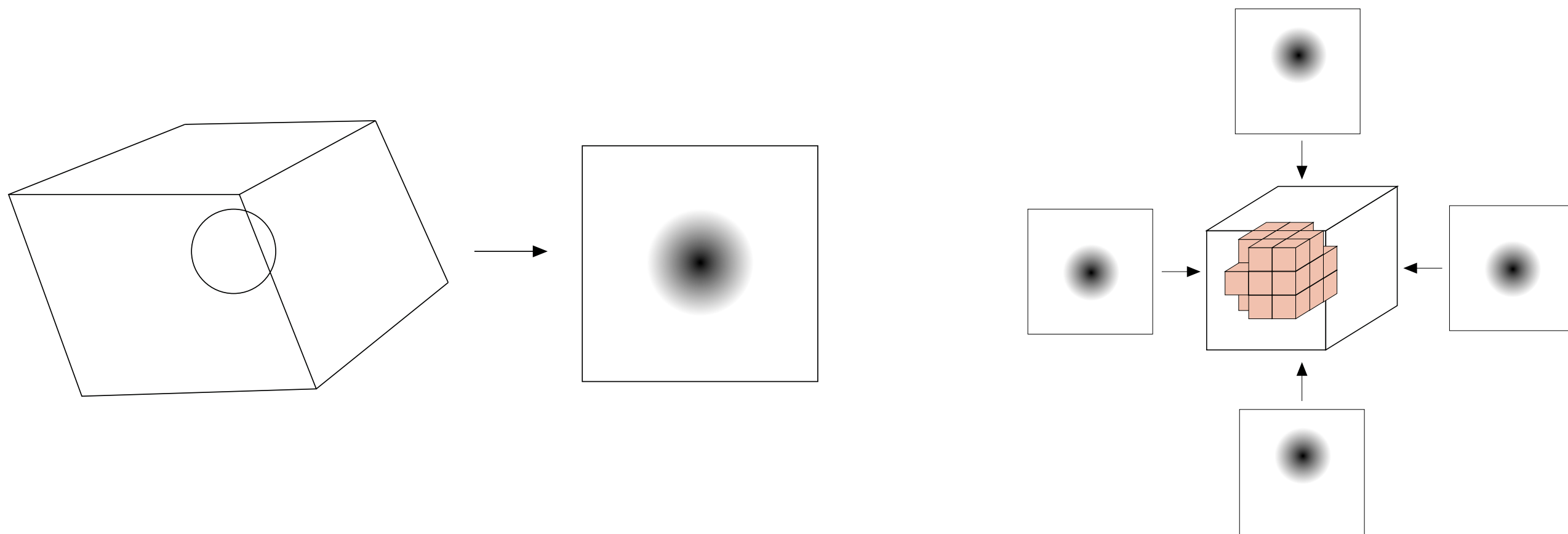
Metod 2: Z-buffer

Rendera scenen/modellen från flera håll (2 till 6)

Använd Z-buffern för att fylla 3D-volymer



Rendera till Z-buffer

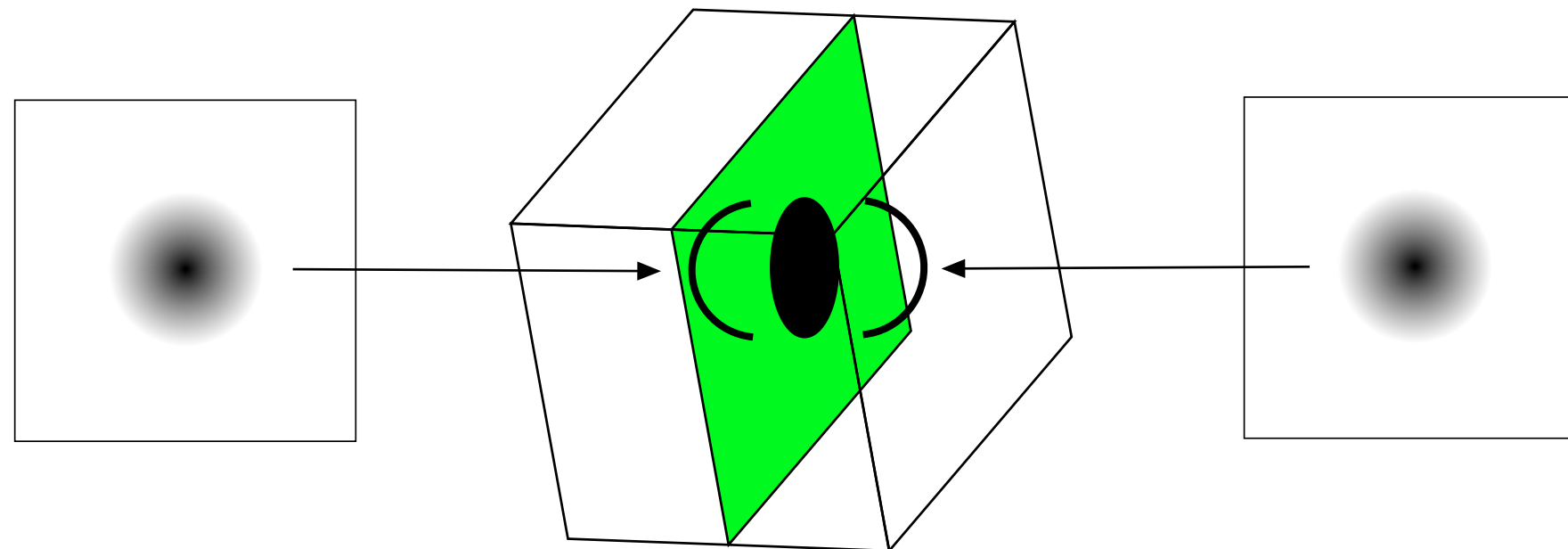


Rendera med parallellprojektion för enklast geometri



Implementation:

- 1) Rendera till 3D-texturen lager för lager. Fyller hela volymen.
- 2) "Write to texture", fyll enbart "skalet". Kräver minst 6 riktningar.





Rendera från volymdata till utbild:

- Marching cubes
- Polygonsnitt genom volym
 - Ray-marching



Marching cubes

Voxlar med täthetsvärden

Gränsvärde i täthet

Voxlar hörn i kuber

Bilda polygoner som skär i gränsvärdet

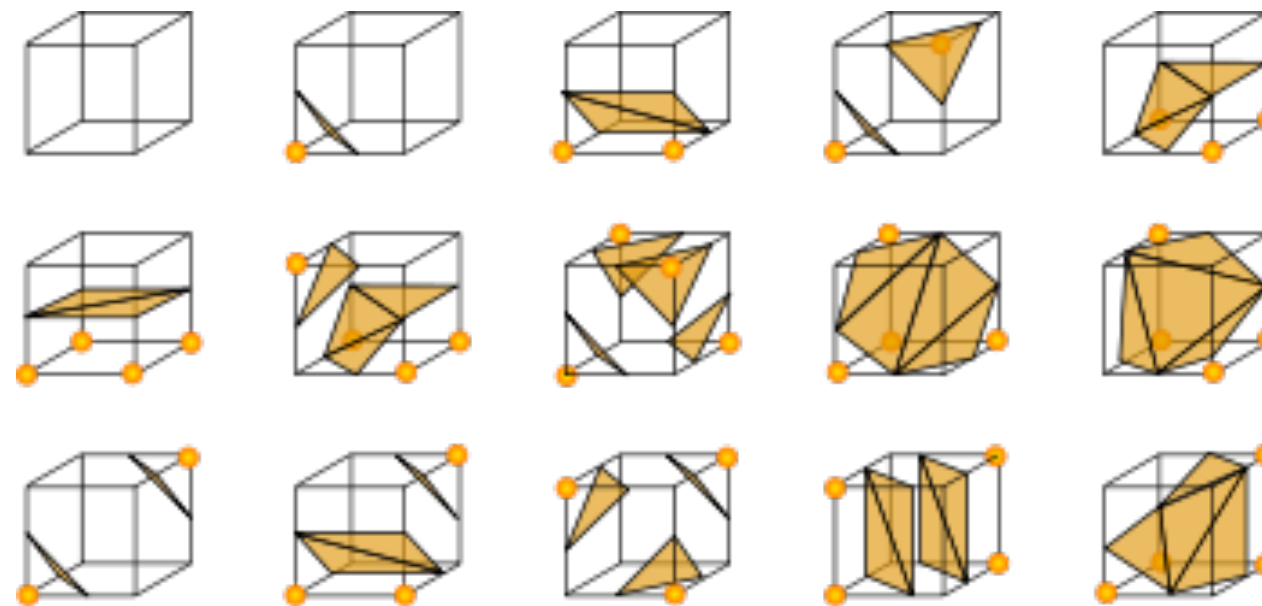
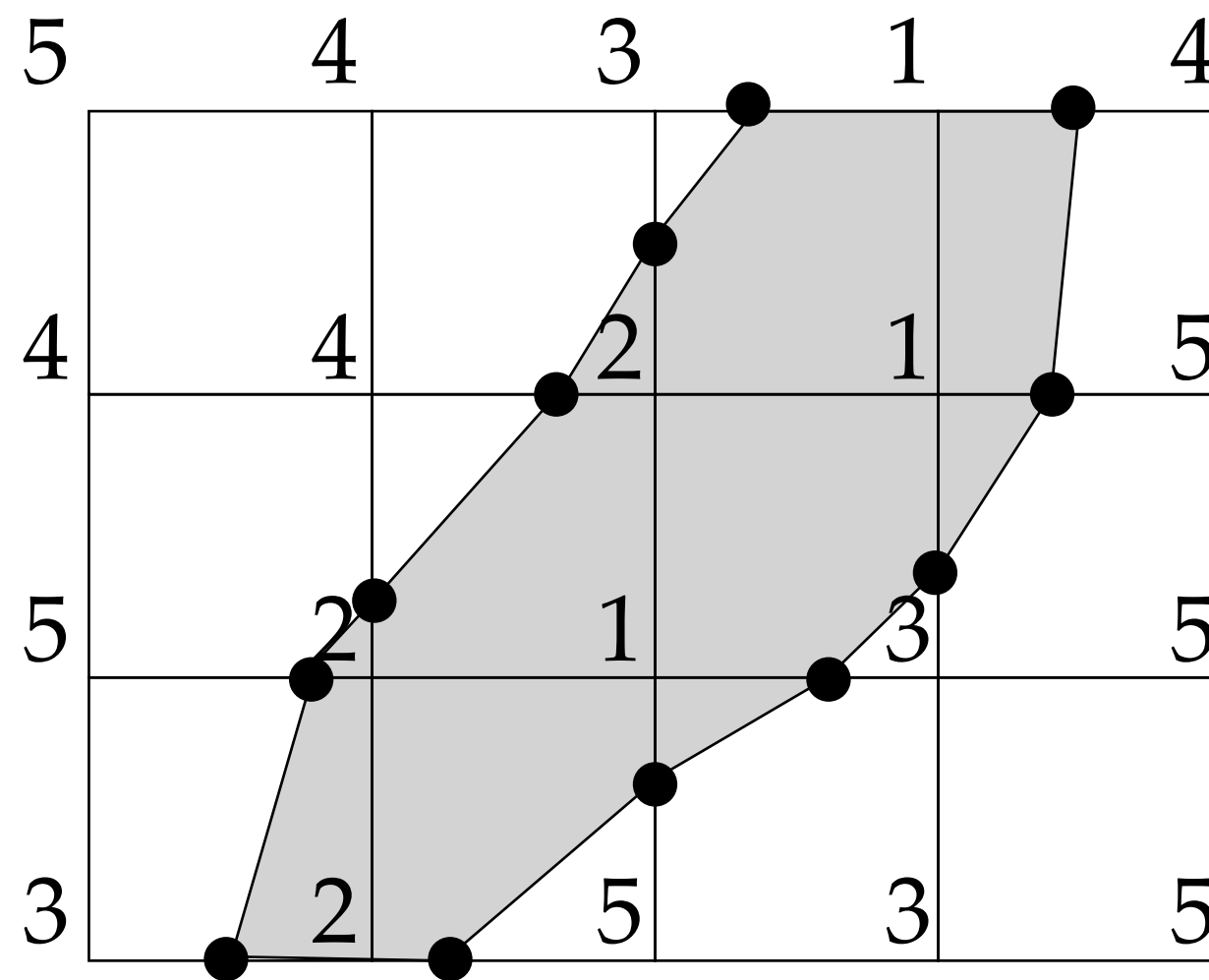


Bild fr Wikipedia



Marching squares - Marching cubes i 2D





Information Coding / Computer Graphics, ISY, LiTH





GPU-accelererad Marching squares

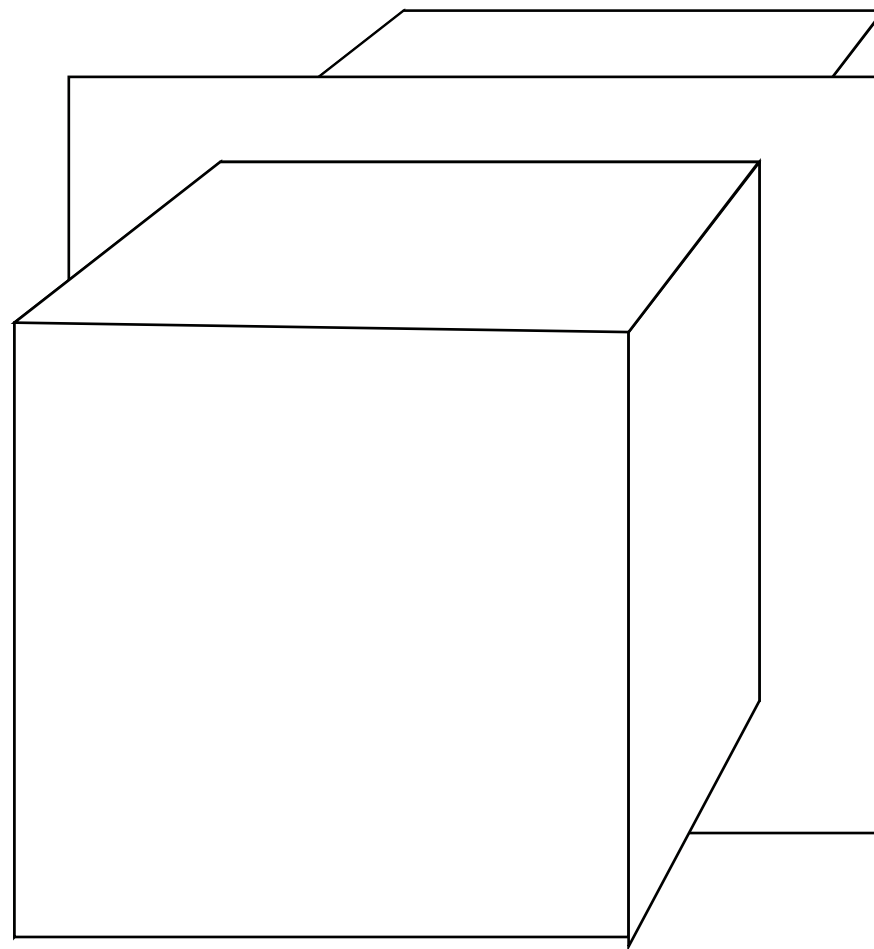
Geometry shaders!

Voxeldata in, polygoner ut

(Mer på senare föreläsning)



2D-plan genom 3D-textur

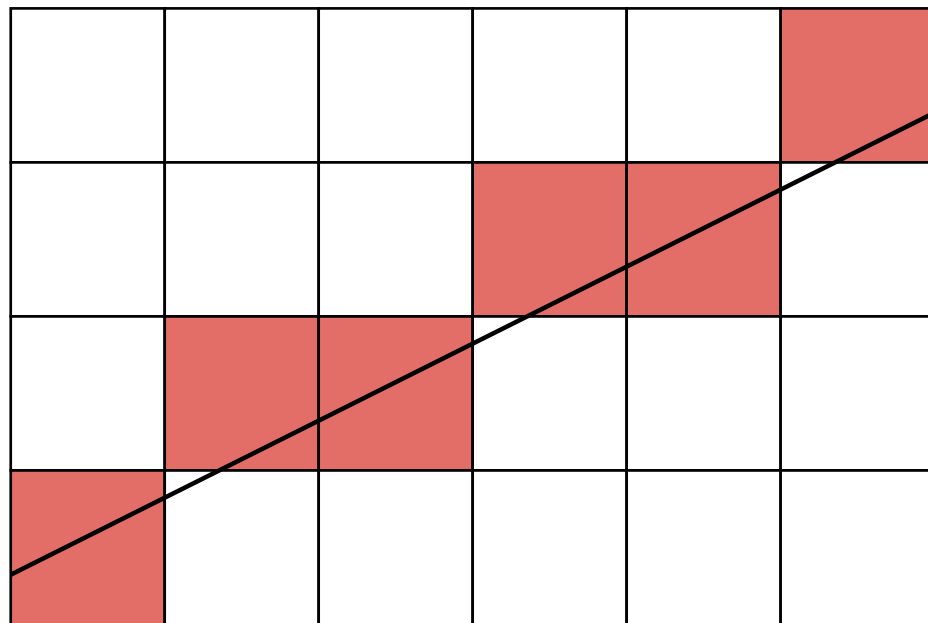


Bilda ett antal plan, lämpligvis med transparens, texturera med olika djup i texturen

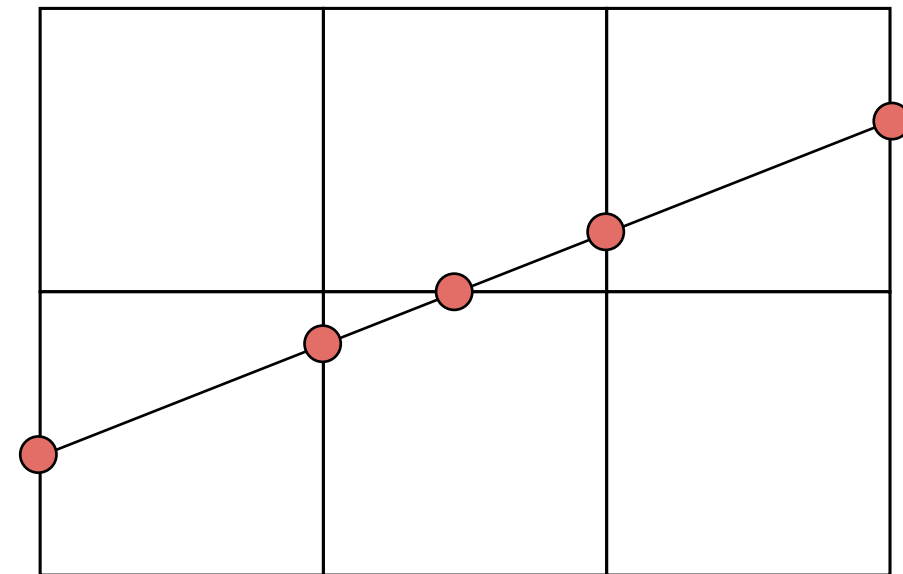


Ray-marching

En form av strålkastning för reguljärt samplade data. Stega på linjeritningsvis genom texturen, voxel till voxel eller kant till kant.



Voxel till voxel: Som Bresenham eller DDA.



Kant till kant: Alltid 2 (3) möjliga fall, tag det närmaste.
Liknar 4-konnektiv linjeritare.



Tillämpningar

- Voxelisera en scen för att underlätta belysningseffekter (t.ex. global ljussättning, object space AO mm)
- Voxelisera en modell för att analysera modellen som volymdata
- Representation och visualisering av volymdata