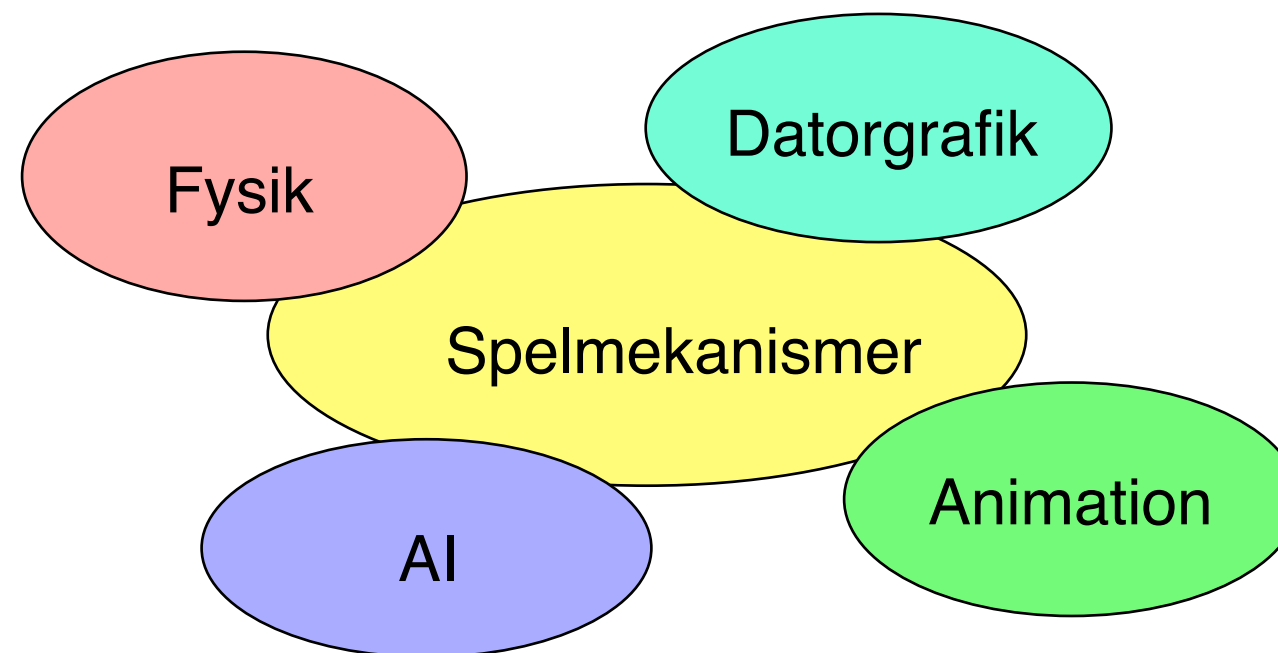




TSBK 03

Teknik för avancerade datorspel
Ingemar Ragnemalm, ISY



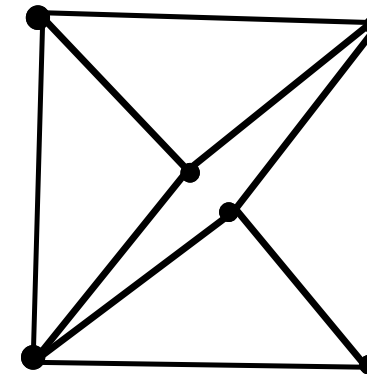
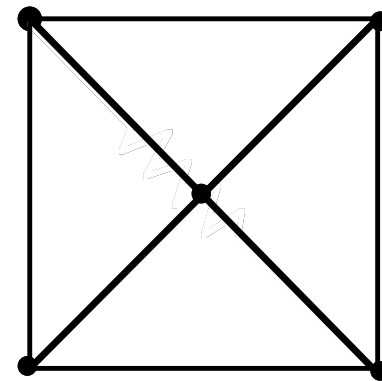
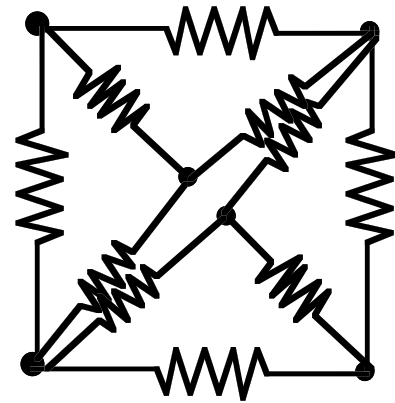
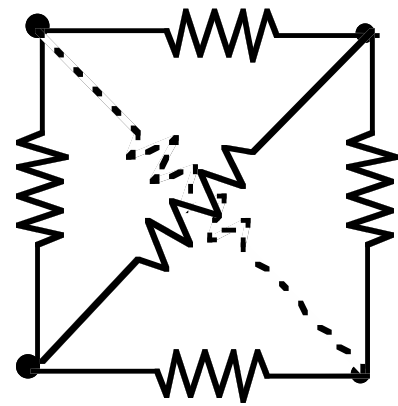


Föreläsning 12

- Förstörbara objekt
- Vätskor och gaser
 - 3D displays
- Några saker till...

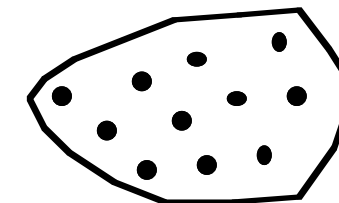
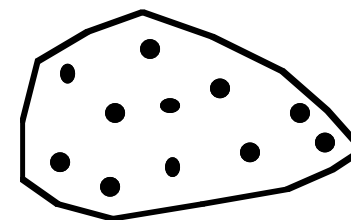
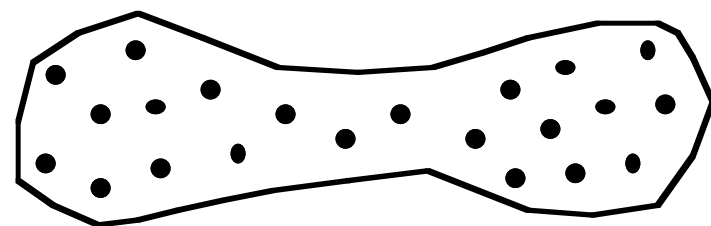


Objekt som kan brytas, krossas mm



Massa-fjäderssystem: Inte så knepigt

FEM: Ganska lätt



Punktbaserat: Inbyggt i systemet



Cloth tearing (Souza mfl 2014)

Över viss kraft-
tröskel "spricker"
en vertex i två!

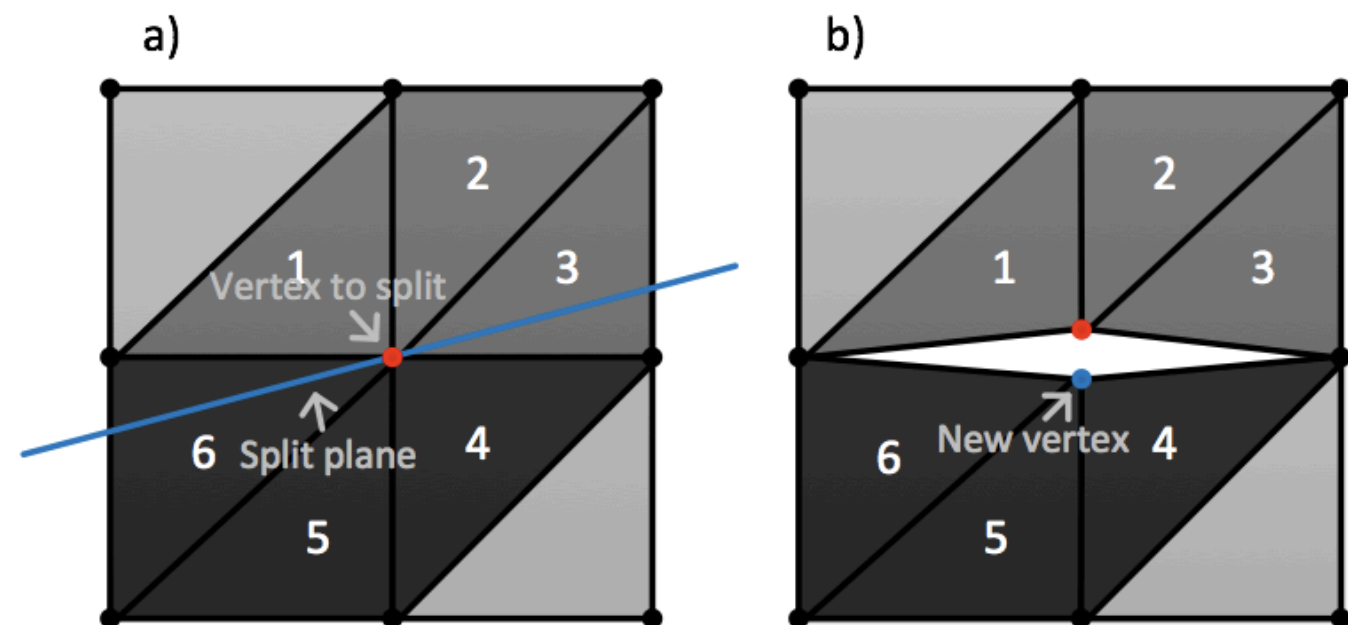


Fig. 1. Vertex splitting algorithm. One vertex is chosen to be split by an arbitrary plane (a) and then a new vertex is created (b). Triangles behind the plane are then assigned to the new vertex.



Resultat



Fig. 6. Cloth tearing. Frames from an interactive application where a cloth patch is ripped off by the user, who grabs one vertex and drags it downward.



Förstörbarhet baserat på händelser

Detektera kraftiga kollisioner (t.ex. projektil mot yta). Beräkna sprickbildning momentant.

1) Fysikalisk beräkning. Propagera genom objektet. Svårt!

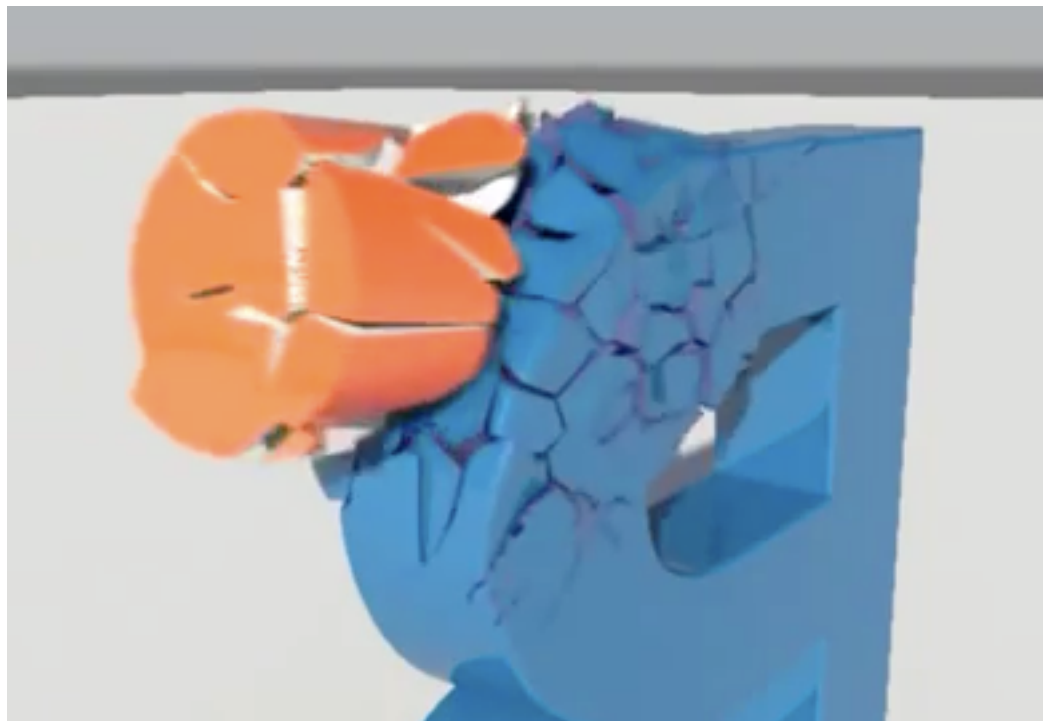
2) Geometrisk simulering. Mycket lättare! Exempel: Voronoi shattering.

En mängd fragmentcentra sprids ut över objektet. Uppdelning görs med Voronoitesselering.



Voronoi shattering

En mängd fragmentcentra sprids ut över objektet.
Uppdelning görs med Voronoitesselering.

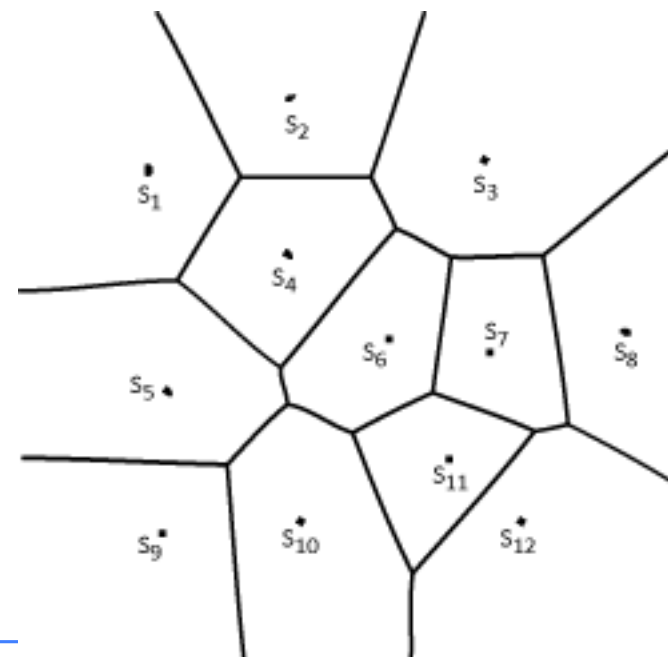




Voronoidiagram

Givet en mängd punkter, så definieras Voronoidiagrammet som en uppdelning (2D eller 3D) i områden som är närmast respektive punkt.

Dual till Delaunaytriangulering. Närbesläktad med avståndstransformer.

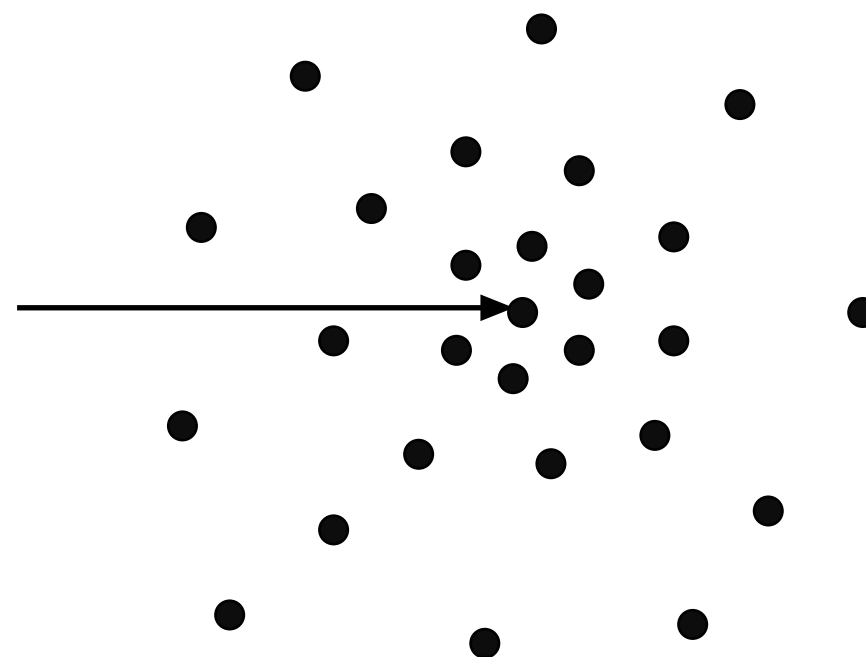




Voronoi shattering

En ren fråga om att sprida punkterna rätt (samt dela upp modellen)! Tätt närmast kollisionspunkten, avtagande täthet utåt.

Ger ett såående realistiskt resultat fast det är rent "fusk", en ren approximation.





Projekt inom deformerbara kroppar

Brett och populärt område. Många möjligheter.

Massa-fjäder: Bra för tyg, utmanande för volymer.
Tyganimationsprojekt bör ha något extra.

FEM: Rättfram och kapabel metod, men rätt svår att få ihop.
Intressant om du redan är bekant med FEM.

Formmatching eller tryckmodellen: Hanterbara.

Villkorssystem: Lätt i 2D, mer utmanande i 3D.

Destruerbara objekt: Många varianter att testa! Fysikaliskt korrekt är svårt. Voronoi shattering enklare.



Simulering av vätskor och gaser

- Partikelsystem

Beroende partiklar, enkla kollisioner mellan sfärer.

- Gridbaserat, Navier-Stokes ekvationer

Kan explicit simulera virvlar mm.

- Vatten simuleras ofta med enbart ytan/staplar



Navier-Stokes ekvation(er)

Analytiska ekvationer för att beskriva vätskor och gaser.
"Klassisk Navier-Stokes" uttrycks ofta så här:

$$\frac{\delta \mathbf{v}}{\delta t} = -(\mathbf{v} \cdot \nabla) \mathbf{v} + \gamma \Delta \mathbf{v} - \frac{1}{\rho} \nabla p + \mathbf{f}$$

Dessutom finns en kompletterande formel för tryckvariation.



Operatorerna ovan: Del (nabla) och Laplace

Kompakta uttryck från vektoranalys för att hantera differentialer i vektorform.

$$\text{Del: } \nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)$$

$$\text{Laplace: } \Delta = \nabla^2 = \nabla \cdot \nabla = \left(\frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial z^2} \right)$$



Navier-Stokes ekvation

Men vad betyder den?

$$-(\boldsymbol{v} \cdot \nabla) \boldsymbol{v}$$

Divergens av hastighet.

$$\gamma \Delta \boldsymbol{v}$$

Viskositet

$$-\frac{1}{\rho} \nabla p$$

Rörelse pga tryckvariation

$$\boldsymbol{f}$$

Externa krafter



$$- (\mathbf{v} \cdot \nabla) \mathbf{v}$$

Divergens av hastighet.

Hastigheten ökar vid konvergens

Jfr smala delar av floder

(Convection)



$\gamma \Delta v$

Viskositet

Förmåga att "dra med" omgivande material i rörelse

(Drag)



$$-\frac{1}{\rho} \nabla p$$

Rörelse pga tryckskillnad

Lokalt högt tryck skapar rörelse utåt.

Lokalt lågt tryck skapar rörelse inåt.

Gradienten av trycket styr.



Implementeras som en 2D- eller 3D-grid.

- Lägg på yttre kraft
- Transportera material efter v (eller snarare hämta bakåt)
 - Diffusion

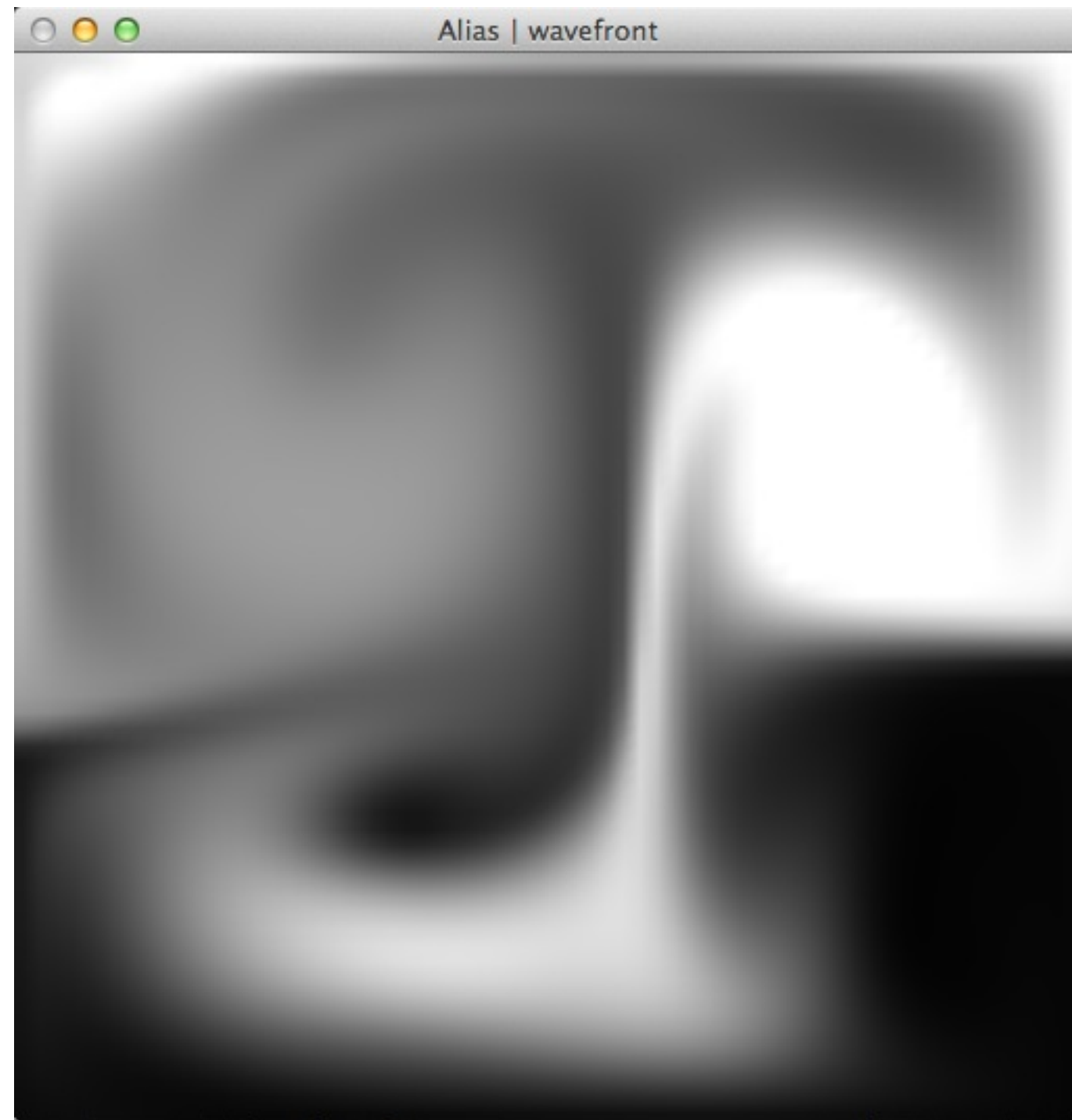
Vanligt att arbeta med icke-kompressibla flöden (även om de egentligen är det)



Jos Stams demo

Alias|Wavefront

GDC 2003





Några teman till...

Moln

Växter

Hår, päls och gräs

Ljuseffekter

Deferred shading



Moln

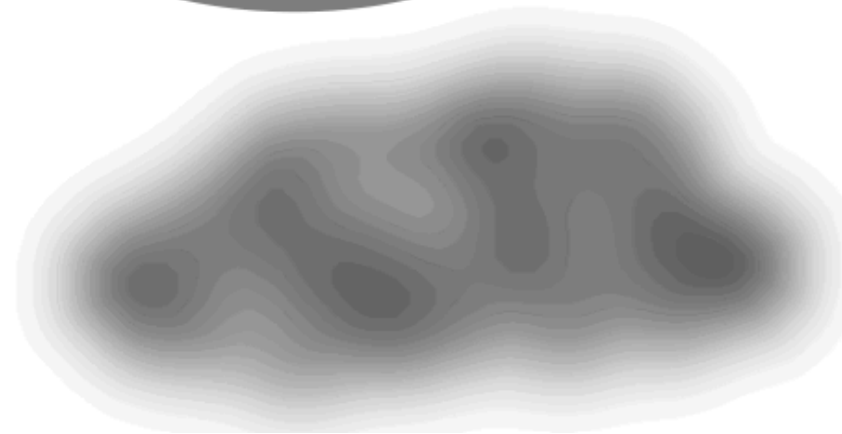
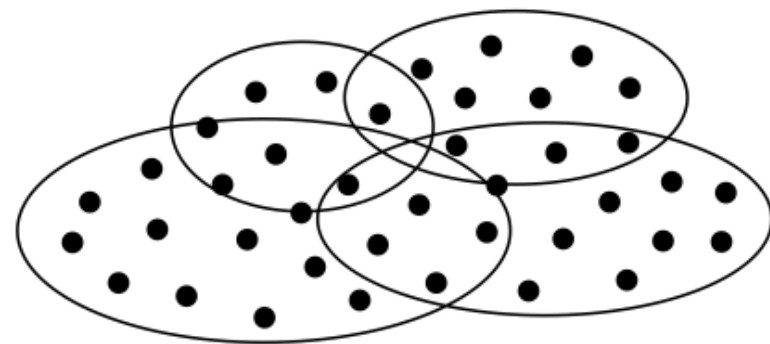
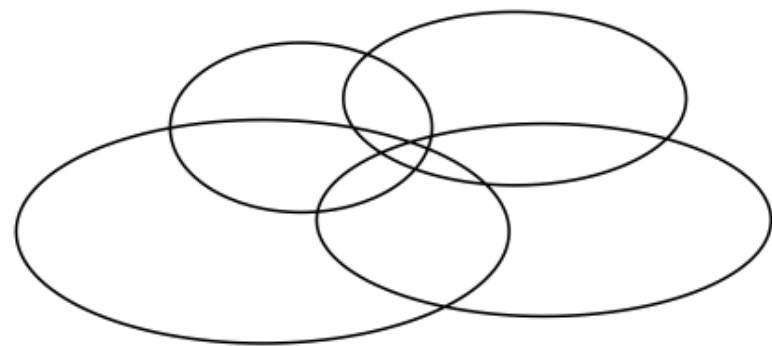
Lätt: Fast i skybox/skydome

Men vi vill att de skall kunna:

- ändra form över tiden
- ha olika form från olika håll
- reflektera och bryta ljus korrekt
- se bra ut i en flygsimulator



Moln från punkter renderade med billboards





Moln från volymetriska data

Arbeta från 3D-textur.

Ljussätt t.ex. via raycasting

Beräkningstungt!



Vegetation

Procedurella träd

Mekanistiska eller deskriptiva

Local-to-global eller global-to-local

Level-of-detailsystem



Mekanistiska eller deskriptiva

Mekanistiska = utgår från trädets tillväxt

Deskriptiva = bekräftar utseendet



Level-of-detail för träd

Extremt detaljerade!

LOD på flera nivåer, typiskt med billboards
på yttersta nivån

Ytterst kan vara ett löv, en gren, hela trädet,
eller hela grupper av träd!



Ett träd i fyra detaljnivåer



Figur 21 levelOfDetail = 0.



Figur 22 levelOfDetail = 1.



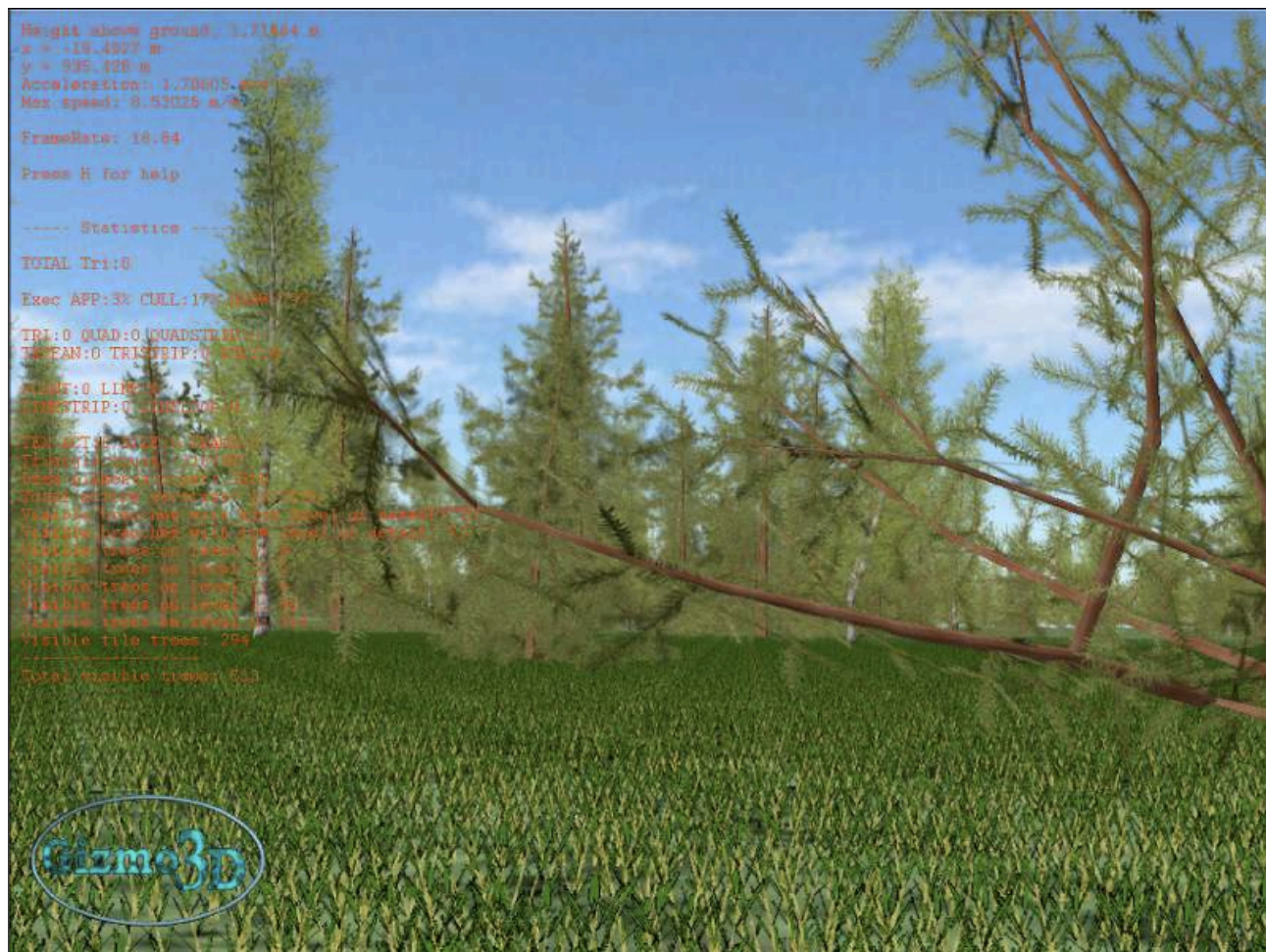
Figur 23 levelOfDetail = 2.



Figur 24 levelOfDetail = 3.



Skogsmiljöer i stor detalj



(Emil Janssons examensarbete)



Hår, päls och gräs

Billboards längs objektkanter "fins"

Polygoner (ev från Geometry shader)



Ljuseffekter

Genomlysta objekt

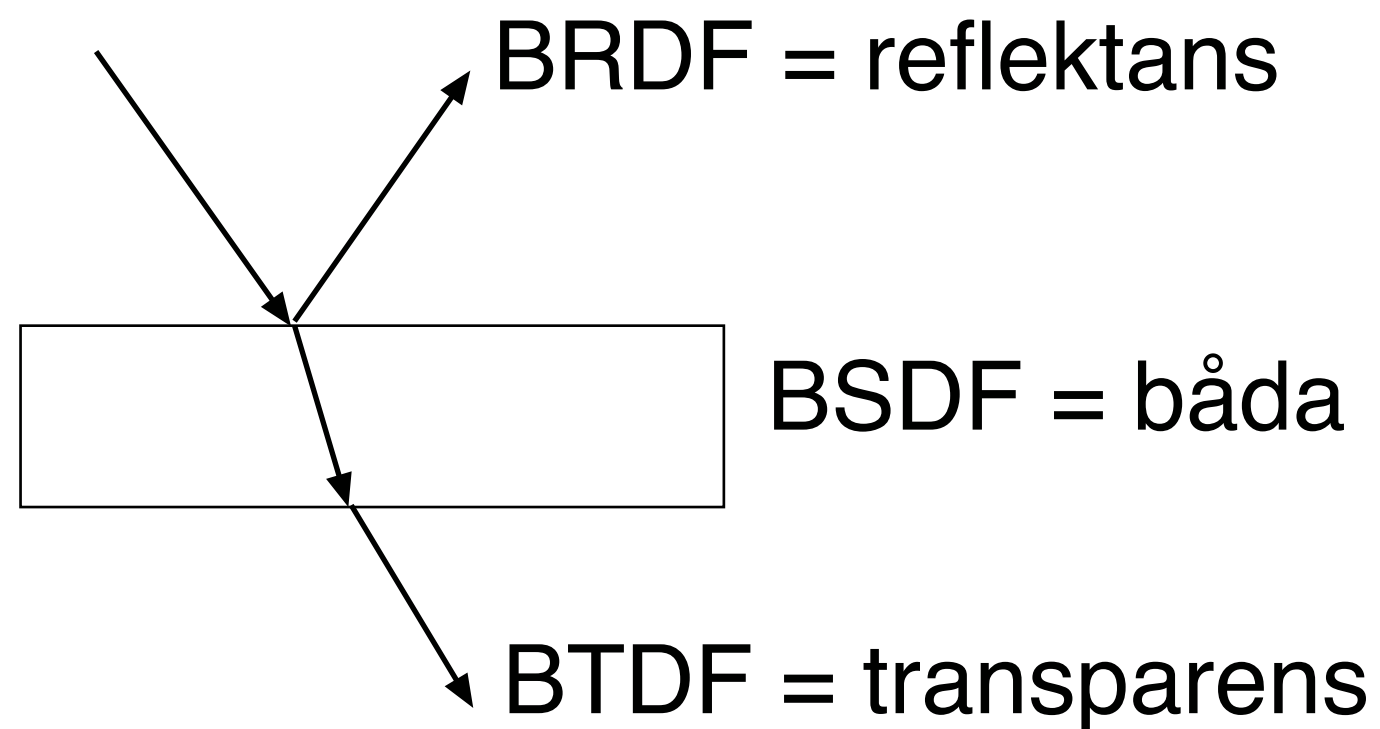
Filtereffekter, "God rays"

BRDF för mer generell belysningsmodell



BRDF, BSDF, BTDF..

Funktioner mer fler inparametrar än Phong,
tillåter mer frihet i uttryck.





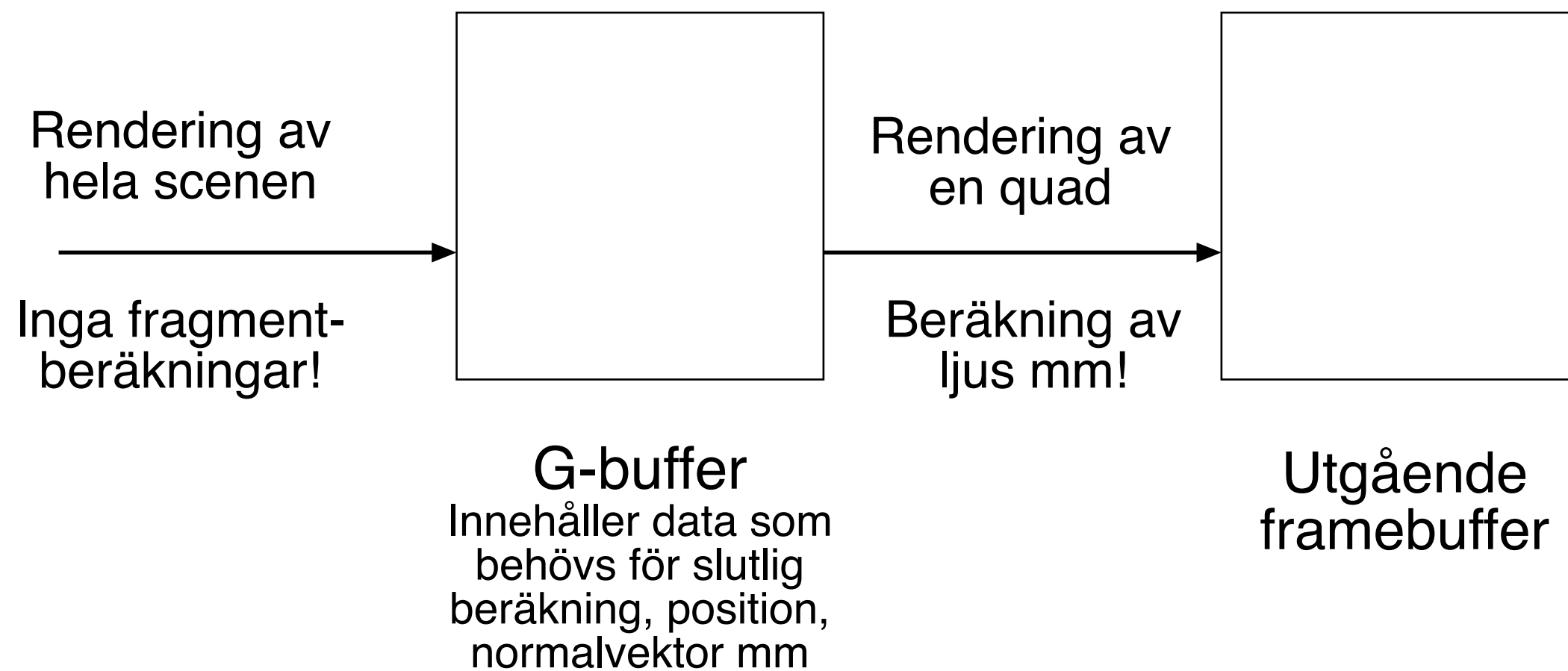
Deferred shading

Optimeringsmetod som reducerar
fragmentberäkningar

Räkna bara ut fragment som syns!



Deferred shading





När är deferred shading en fördel?

- Multipla ritningar per pixel
- Tung fragment shader-processning:
 - Många ljuskällor
 - Avancerad bump mapping
 - Ray casting etc per pixel



Problem med deferred shading

- Transparens är svårt
- Ljusberäkningar måste göras per pixel - svårt att göra optimeringar för enkla objekt.
- Prestanda kan försämrars för scener utan multipla lager (lindrigt problem)



Ljud

Enkelt ljud: CallMeAL

Spela detta ljud nu!

- Ljudeffekter
- Bakgrundsljud/musik

Räcker långt!



Avancerat ljud

Stereoplacering. Stöds av OpenAL

Ljudspridning. Kan göras med ray-tracing-liknande analys av en scen.