

# Physics – Particle Systems

tum:3D  
computer graphics & visualization

---

# Übersicht/Gliederung

---

- Geschichte und Motivation
- Grundlagen
  - Partikel und Engine
  - Rendering
- Anwendungsbeispiel Fackel
- Komplexere Anwendungen

# Geschichte & Motivation

---

- **Bisher:**

Modelling, Animation und Rendering von  
Objekten mit definiertem Umriss  
Quader, Kugeln, Meshes, ...

- **Heute:**

Was ist mit Rauch, Feuer, Nebel, Wasser, ... ?

# Geschichte & Motivation

---

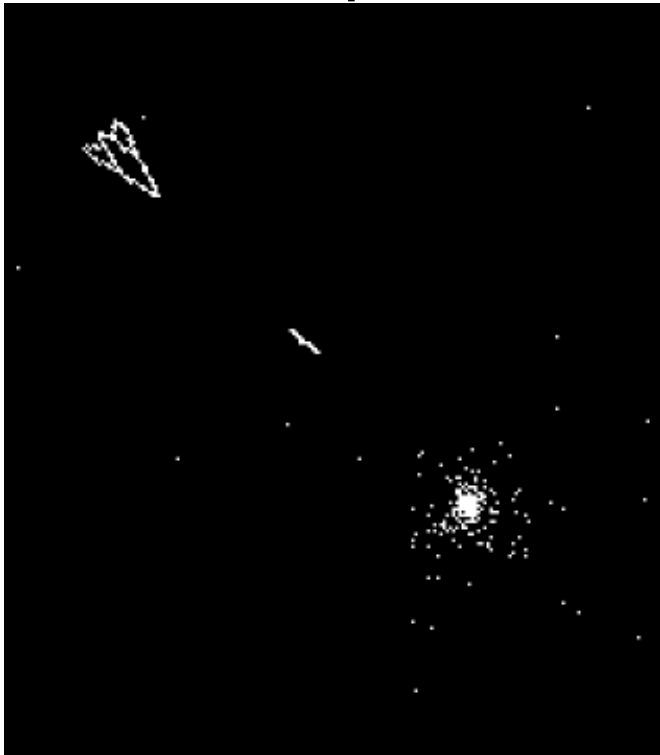
- Wie erzeugt man virtuelles Feuer?



# Geschichte & Motivation

---

- 1962 Spacewar



- 1982 Star Trek II  
„Der Zorn des Khan“  
William Reeves [REEV83]

# Die Idee

---

- physikalisch korrekte Simulation von Explosionen, Feuer, Rauch, ... sehr aufwändig zu modellieren und zu rendern
  - Gesamteindruck ist wichtiger als physikalisch korrekte Details
- ersetze Volumina durch viele generierte punktförmige Objekte

# Das Partikel

---

- Eigenschaften:

- **Bewegung:**

- Position, Geschwindigkeit

- **Aussehen:**

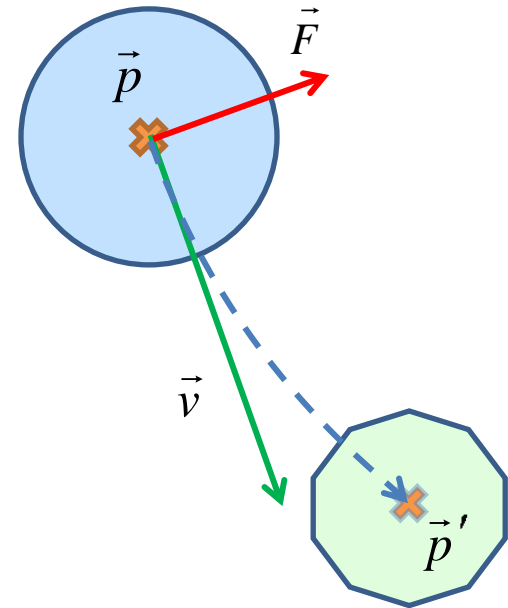
- Farbe, Größe,  
Alpha-Wert(Transparenz), ...

- **Simulation:**

- Alter, Lebensspanne, ...

- ~~Kräfte?~~

- Speicherung nur in komplexen Anwendungen nötig



# Die Engine - Physik

---

- Animation

- Bewegungsgleichungen

$$\vec{v}' := \vec{v}_0 + \int \vec{a} dt$$

$$\vec{p}' := \vec{p}_0 + \int \vec{v} dt$$

- Euler Integration  
mit  $a$  konstant

$$\vec{v}' := \vec{v} + \vec{a} \cdot \Delta t$$

$$\vec{p}' := \vec{p} + \frac{\vec{v} + \vec{v}'}{2} \cdot \Delta t$$

- Wie berechnet man  $a$ ?  
Summe aller wirkenden Kräfte / Partikelmasse

Beeinflussung durch:

- Wind, Luftwiderstand, ...
- Zufall
- Ideen der Designer ( $\rightarrow$  uncanny valley)



# Die Engine - Partikelemitter

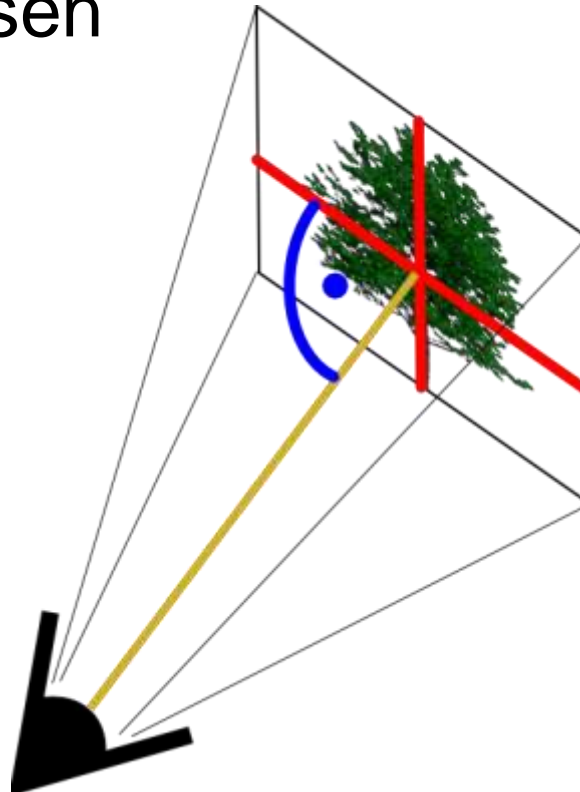
---

- Eigenschaften eines Emitters
  - Form, Verteilungsfunktion
  - Initialeigenschaften der Partikel (Zufallskomponente!)
  - Frequenz (Partikel pro Zeiteinheit)
  - ...

# Rendering: Billboard

---

- Halbtransparente Flächen, die immer zum Betrachter weisen



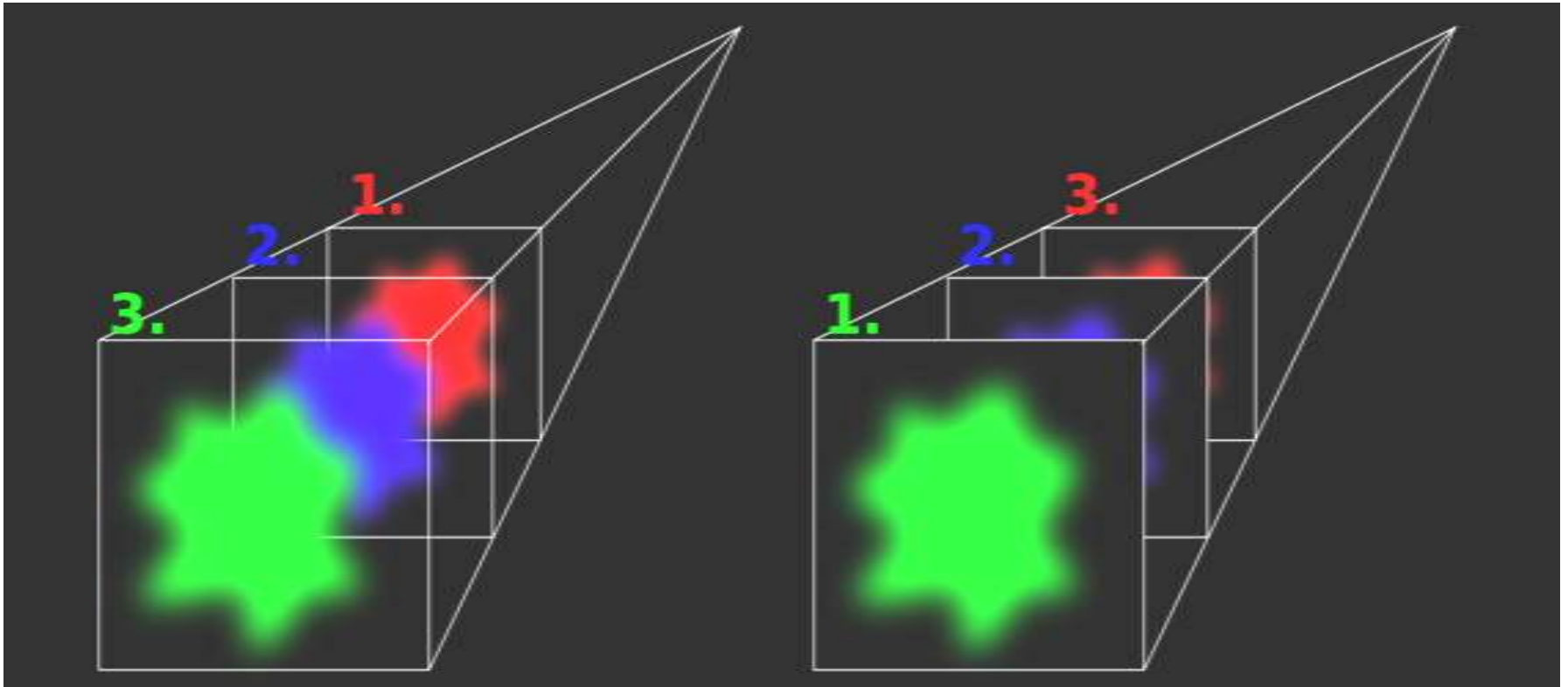
# Exkurs – Transparenz, Blending

- Blending: Kombination von Hintergrund mit dem gerade zu zeichnenden Objekt
  - **Additiv:**  $\text{Neu} = \text{VG} + \text{HG}$
  - **Alpha Blending:**
    - $\alpha$ : 0 unsichtbar – 1 komplett undurchsichtig
    - $\text{Neu} = \text{VG} * \alpha + \text{HG} * (1 - \alpha)$
  - ...



# Exkurs – Transparenz, Blending

- Problem: Reihenfolge



# Exkurs – Transparenz, Blending

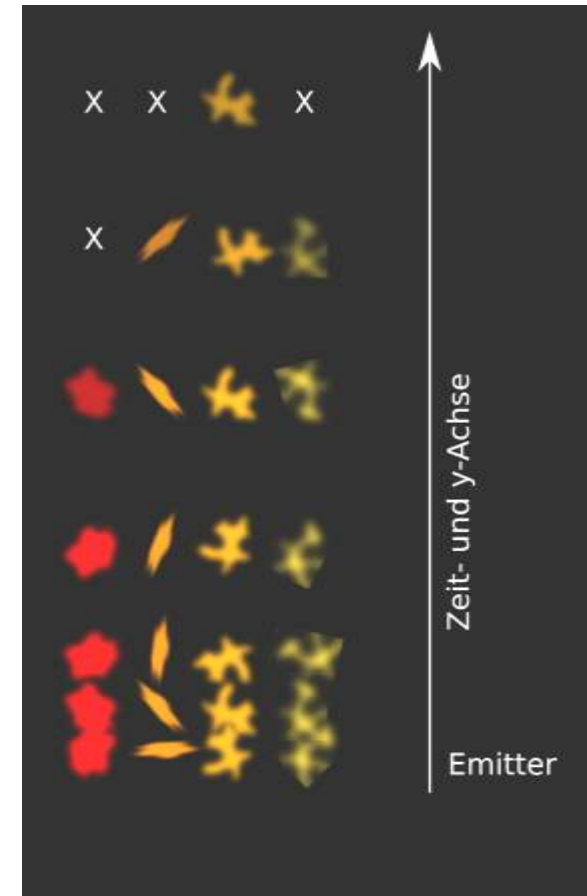
- Lösungen

→ sortieren nach Abstand zur Kamera:

Sortierung teuer, wenn möglich durch anderes Blending vermeiden oder optimieren

# Anwendung – Eine Fackel

- **Emitter**
  - zufällige Position in Halbkugel
  - Lebensdauer höher für Mitte
  - zufällige Farbe (viel rot, etwas grün, ein wenig blau)
- **Animation**
  - Beschleunigung entlang Brennrichtung
  - Abbremsen durch Luftwiderstand
  - gegen Lebensende durchsichtig werden
  - zufällig drehen
- **Rendering**
  - additiv → keine Sortierung nötig



# Anwendung – Eine Fackel

---



# Häufige Schwierigkeiten

---

- Integration in Szene
- Wechselwirkung mit Beleuchtung
- Kombinationen (Feuer+Rauch)
- Performance

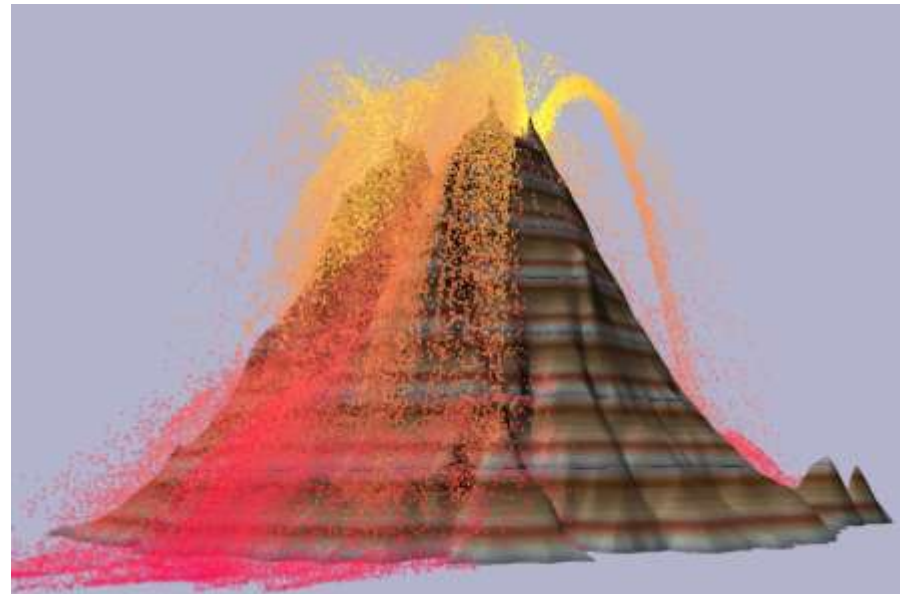
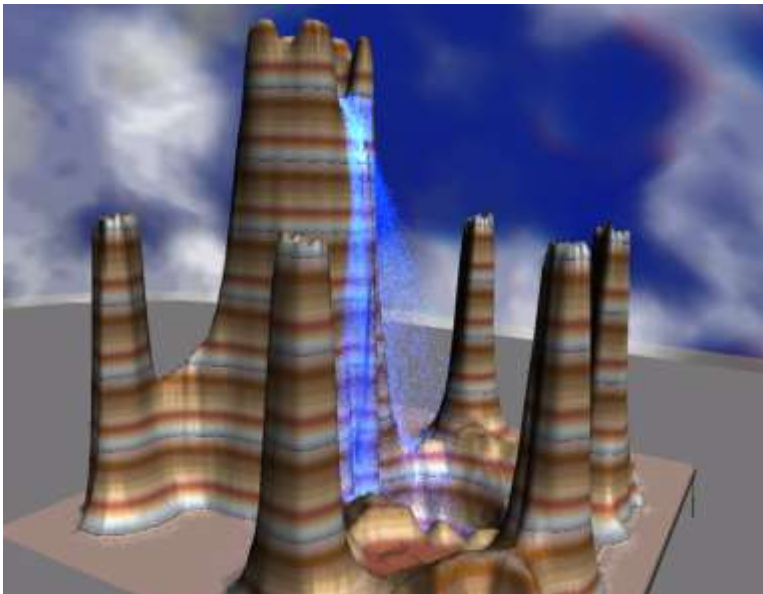




# Verbesserungen

---

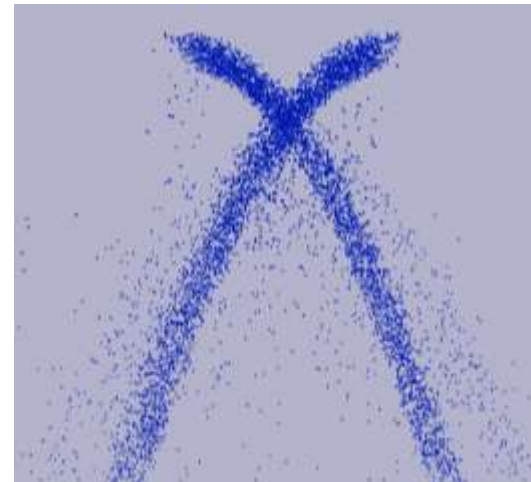
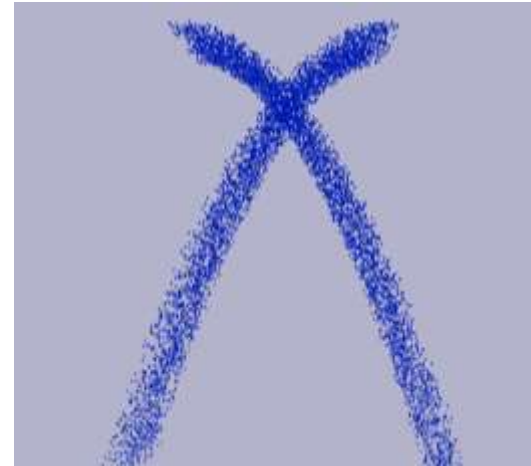
- Einsatz der GPU für Rendering **und** Animation
  - CPU: < 100.000 Partikel → GPU: Millionen Partikel
  - Inzwischen auch für Kollisionserkennung, Tiefensortierung etc.



# Verbesserungen

---

- Partikel abhängig voneinander  
z.B. Partikel-Partikel Kollisionen
- Shader-Einsatz für realistischere Darstellung



# Flüssigkeiten

---

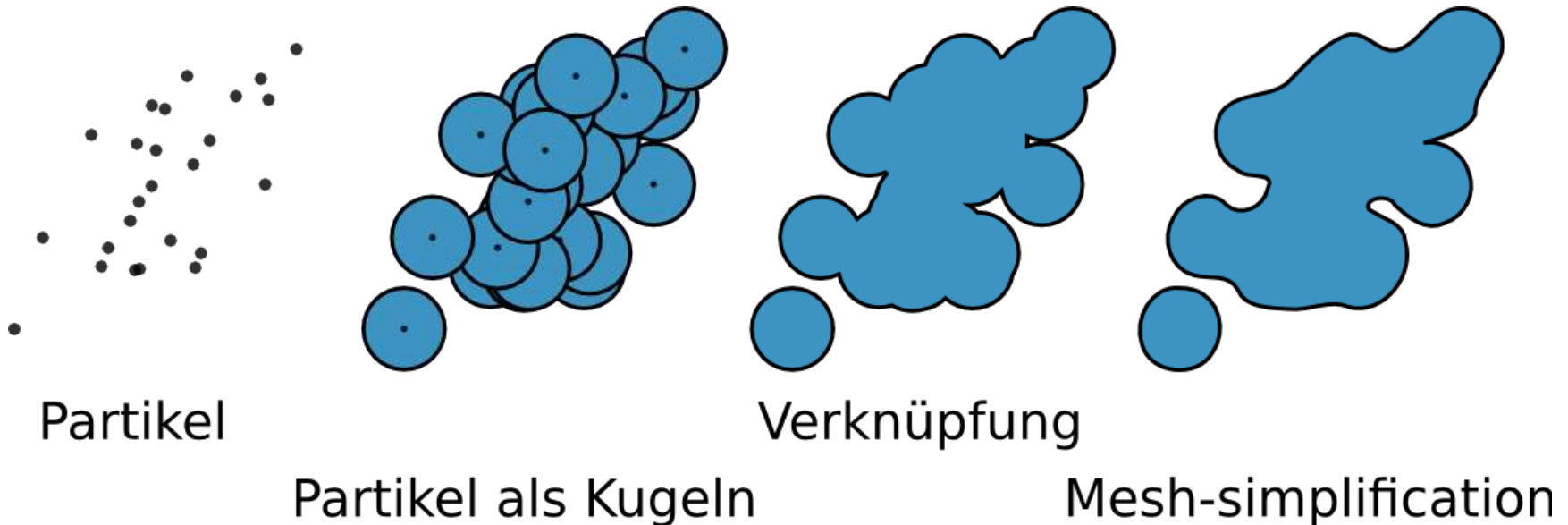
- Simulation bewegter Flüssigkeiten mit Partikeln



# Oberflächenextraktion

---

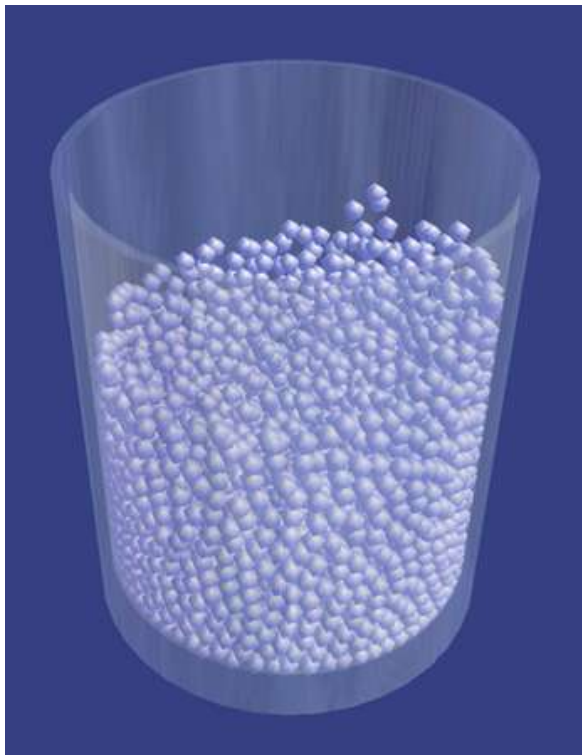
- Wie kommt man von einer Punktmenge zu einem Oberflächenmesh?
  - naiver Ansatz:



# Oberflächenextraktion

---

- besser:  
marching cubes, „3D-Rasterisieren“



# Flüssigkeiten

---

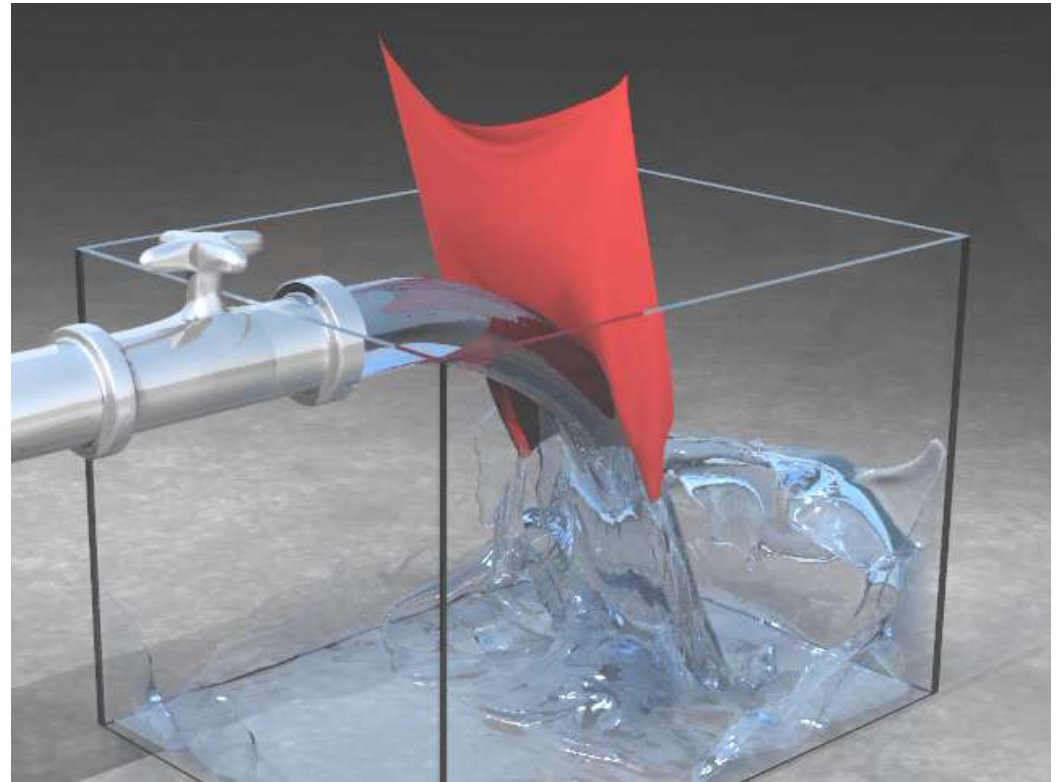




# Was gibt es noch?

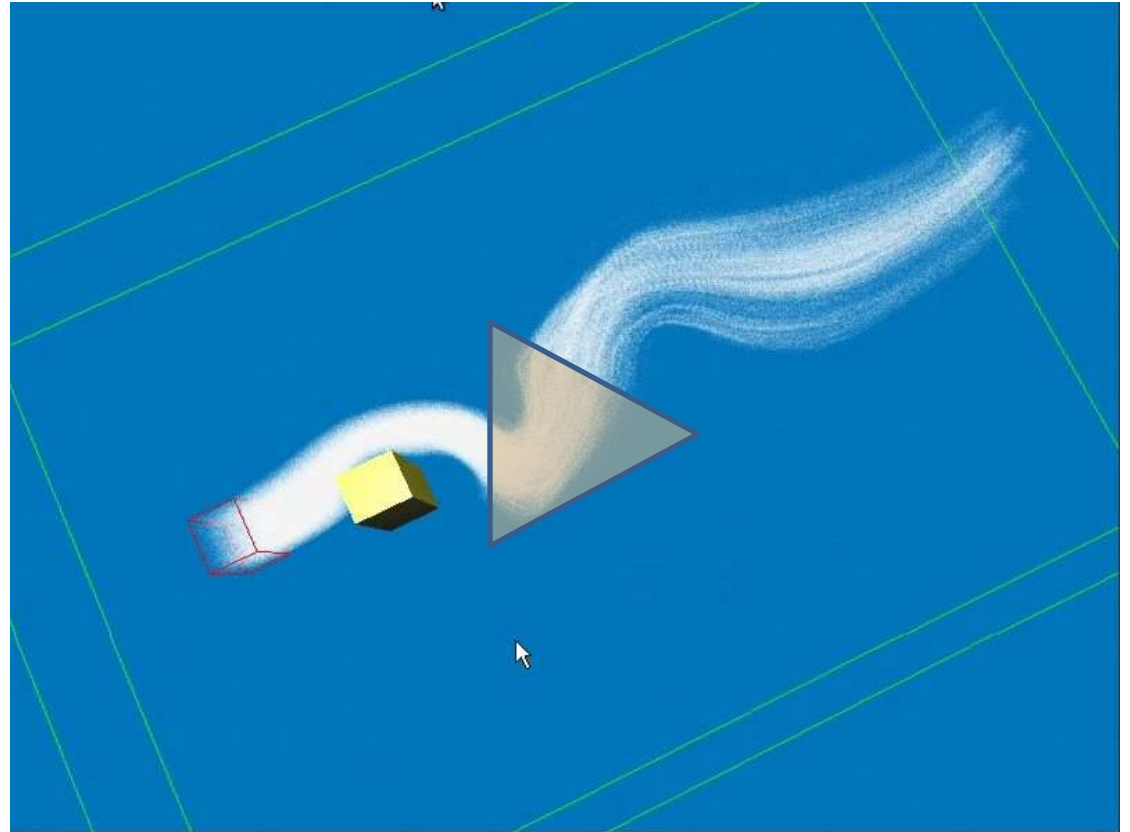
---

- Vegetation:  
Gras, Bäume, ...
- Textilien
- deformierbare  
Körper
- Tier-Schwärme
- ....



# Wissenschaftliche Anwendungen

- Berechnung und Visualisierung von Strömungen





# Partikelsysteme - Auf einen Punkt

- Ersatz für volumetrisches Rendering
- nicht nur für Feuer und Rauch
- ermöglichen faszinierende Effekte
- erleichtern die Arbeit von Designern
- stochastischer Prozess als zentrales Element
- auch für wissenschaftliche Anwendungen  
nützlich

# Effekte

---



# Effekte

---



# Effekte



---

Vielen Dank für eure  
Aufmerksamkeit.

Fragen?

# Quellen

---

- Particle Systems – a technique for rendering a class of fuzzy objects, *William T. Reeves, 1983*
- Advanced Animation and Rendering Techniques, *Alan Watt/Mark Watt*
- Partikelsysteme, *Robert Selzmann, TU Dresden*
- Building a Million Particle System, *Lutz Latta, Massive Development GmbH*
- Particle-Based Fluid Simulation for interactive Applications, *Matthias Müller et. al.*
- Simulating Whitewater Rapids in Ratatouille, *Eric Froemling & Tolga Gotekin & Darwyn Peachey, PIXAR Animation Studios*
- Partikelengine in Kürze unter [www.kernelz.de](http://www.kernelz.de) (inkl Source)