



Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

# 3D-Algebra Version 2

Thomas Behr

Fakultät für Mathematik und Informatik  
Datenbanksysteme für neue Anwendungen



**FernUniversität in Hagen**

19.November 2015

©2015 FernUniversität in Hagen



# Das Ziel

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige



- Repräsentation von Punkten, Flächen und Körpern im  $\mathbb{R}^3$
- Import/Export von Standardformaten
- Objekterzeugung
- Manipulation einzelner Objekte (Drehen, Verschieben usw.)
- Mengenoperationen



# Wozu und warum jetzt?

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

- Darstellung von Objekten der realen Welt
- 3D-Geräte für Heimgebrauch erschwinglich
  - 3D-Scanner
  - 3D-Drucker
- größere Datenmengen verfügbar
- Verwaltung in einer Datenbank sinnvoll



# Warum Version 2?

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

- Algebra3D seit dem letzten FaPra-Durchlauf vorhanden
- Gleiche Funktionalität wie Version 2
- Viewer funktioniert gut
- Algebra selbst ist nicht robust
  - Rundungsfehler
  - logische Fehler?
- Reimplementierung des C++-Teils
- Konzentration auf robuste Implementierung
- Vermeiden von Rundungsfehlern



# 3D-Daten – Woher kommen die?

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

- in Programmen konstruiert
- 3D-Scanner
  - Laser-Scanner, z.B. Makerbot Digitizer
  - Strukturiertes Licht, z.B. David SLS-2
  - Infrarotraster, z.B. Sense3D, Kinect
  - Ultraschallscanner
- Photogrammetrie



`point3d` Element aus dem  $\mathbb{R}^3$

`surface3d` Oberfläche (Menge von Dreiecken)

`volume3d` Körper (Menge von Dreiecken)

weitere Typen

`vector3d` Richtungsvektor

`plane3d` Ebene



# Oberflächen

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

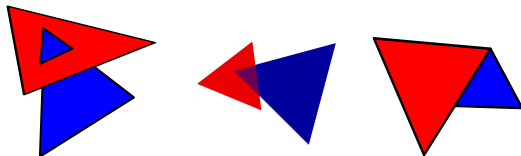
Verbinden

Zerlegen

Sonstige

- dargestellt als Menge von Dreiecken
- müssen nicht zusammenhängend sein
- überlappungsfrei
- Dreiecke haben paarweise entweder
  - einen gemeinsamen Eckpunkt
  - eine gemeinsame Kante, d.h. zwei gemeinsame Eckpunkte
  - gar keine gemeinsamen Punkte

Nicht erlaubt:





- ebenfalls als Dreiecksmenge dargestellt
- Bedingungen für Oberflächen gelten auch hier
- zusätzlich:
  - Die Oberflächen sind wasserdicht, d.h. jede Kante eines Dreiecks ist auch Kante eines anderen Dreiecks.
  - Im Inneren eines Körpers darf kein weiterer Körper sein.
  - Jedem Dreieck ist ansehbar, auf welcher Seite das Innere des dargestellten Körpers liegt.





# Import von Körpern

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

**Import**

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

- STL-Format (Surface Tessellation Language)
- binär oder ASCII
- stellt Mengen von Dreiecken dar
- Datenquelle  
`www.thingiverse.com`



# STL-ASCII-Beispiel

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

```
solid liberty
  facet
    normal 2.03e-01 -8.889e-01 -4.10e-01
  outer loop
    vertex 3.94e+01 -7.22e+00 1.24e+02
    vertex 3.91e+01 -7.68e+00 1.25e+02
    vertex 3.87e+01 -7.45e+00 1.25e+02
  endloop
endfacet
...
endsolid
```



# STL-BINARY-AUFBAU

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

## Bytes Inhalt

0-79 Header, Inhalt nicht festgelegt

80-83 vorzeichenloser Integer: Anzahl der Dreiecke

84-... Dreiecke

Jedes Dreieck ist wie folgt aufgebaut:

## Bytes

0-3, 4-7, 8-11

12-15, 16-19, 20 -23

24-27, 28-31, 32-35

36-39, 40-43, 44-47

48-49

50-.. ?

## Inhalt

( $n_1, n_1, n_2$ ): Normalenvektor

( $x_1, y_1, z_1$ ): erster Punkt

( $x_2, y_2, z_2$ ): zweiter Punkt

( $x_3, y_3, z_3$ ): dritter Punkt

vorzeichenloser Integer: Byteanzahl eines am Dreieck hängenden Attributs, im Normalfall 0

optional: zusätzliches Attribut



# Normalenvektor und Knotenorientierung

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

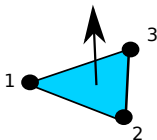
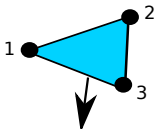
Transformieren

Verbinden

Zerlegen

Sonstige

- Normalenvektor
  - steht senkrecht auf dem Dreieck
  - zeigt nach außen
  - hat Länge 1
- Knoten sind entgegengesetzt zum Uhrzeigersinn orientiert, wenn man von außen auf das Dreieck sieht
- Außeninformation ist redundant gespeichert





# Importkorrektur

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

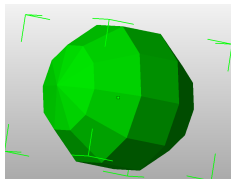
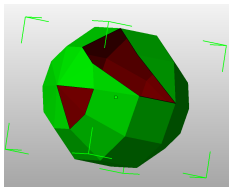
Transformieren

Verbinden

Zerlegen

Sonstige

- Viele STL-Dateien teilweise fehlerhaft
- Der **importSTL**-Operator korrigiert Fehler soweit möglich
- dies umfasst
  - Teilen sich schneidender oder überlappender Dreiecke
  - Entfernen doppelter Dreiecke
  - Schließen von Löchern
  - Umdrehen der Orientierung von Dreiecken
  - Entfernen von Dreiecken aus dem Objektinneren





- **exportSTL:** volume3d × text × string × bool → bool  
Körper × Dateiname × Objektname × binär → Erfolg  
exportiert einen Körper
- Prüfen der Korrektheit, z.B. mittels `net fabb`  
Basisversion kostenlos



# Erstellung einfacher Körper

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

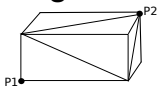
Sonstige

**Operator**

**Ergebnis**

**Argumente**

**createCube**



*point3d* × *point3d*

**createCylinder**



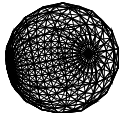
*point3d* × *real* × *real* × *int*

**createCone**



*point3d* × *real* × *real* × *int*

**createSphere**



*point3d* × *real* × *int*



# Konvertierung von Regionen

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

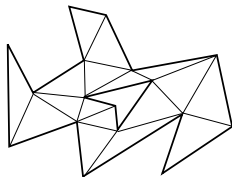
Transformieren

Verbinden

Zerlegen

Sonstige

- **region2surface:** region  $\rightarrow$  surface3d
  - Triangulation einer Region
- **region2volume:** region  $\times$  real  $\rightarrow$  volume3d
  - Region mit einer Dicke
- **mregion2volume:** mregion  $\times$  real  $\rightarrow$  volume3d
  - bewegte Region
  - „Dicke“ eines Tages







# Affine Transformationen

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

- Rotation, Skalierung, Verschiebung
- ändern die Lagebeziehungen von Objekten nicht
- lassen sich auf Matrixmultiplikationen zurückführen
- Operatoren sind nur im Aufbau der Matrix verschieden
- jeder Punkt des Objekts wird mit der Transformationsmatrix multipliziert
- Verwendung von 4-dim. Punkten und  $4 \times 4$  Matrizen für  $\mathbb{R}^3$

Beispielmatrix (Rotation um die  $x$ -Achse mit Winkel  $\phi$ ):

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi & 0 \\ 0 & \sin \phi & \cos \phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



# Operatoren

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

**rotate** rotiert Objekt um bestimmten Punkt und Achse mit geg. Winkel

**mirror** spiegelt Objekt an einer Ebene

**translate** verschiebt Objekt um einen Vektor

**scaleDir** skaliert ein Objekt aus einem Punkt in gegebene Richtung

**scale** gleichmäßige Skalierung aus einem geg. Punkt



# Mengenoperationen

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

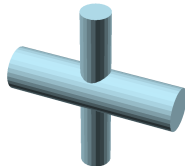
Zerlegen

Sonstige

**union** Vereinigung zweier Objekte

**minus** Differenz zweier Objekte

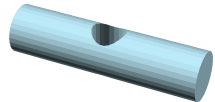
**intersection** Schnitt zweier Objekte



union



intersection



difference

zweier Zylinder



# Zerlegen

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

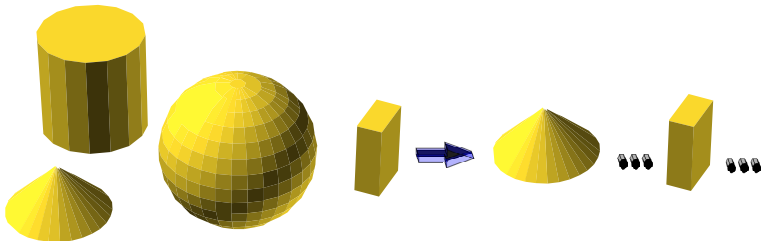
Transformieren

Verbinden

Zerlegen

Sonstige

- Oberflächen und Körper können aus mehreren, nicht verbundenen Komponenten bestehen
- Operator **components** zerlegt Objekt in diese Komponenten
- Ergebnis ist ein Strom von Objekten





# Sonstige Operatoren

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

**size** liefert die Anzahl der enthaltenen Dreiecke  
**bbbox** liefert die Bounding Box eines Objekts als  
*rect3d*



# Robustheit

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

- einige Operatoren erfordern komplexe Berechnungen auf Dreiecken, z.B. Import, Mengenoperationen
- Berechnungen im 3D-Raum
- Rundungsfehler
- Verwendung der `gnu multiple precision library`
  - Manual unter:  
<https://gmplib.org/gmp-man-6.0.0a.pdf>
  - Wichtige Klasse `mpq_class` zur Darstellung von rationalen Zahlen beliebiger Genauigkeit
  - Anpassung von Berechnungsergebnissen notwendig, um durch `double` darstellbares Ergebnis zu erhalten (Entfernen zu kleiner Dreiecke und Auffüllen durch Nachbardreiecke)



# Geschwindigkeit

Aufgabe  
Phase2

Thomas Behr

Motivation

Typen

Import

Export

Erzeugen

Transformieren

Verbinden

Zerlegen

Sonstige

- häufig Suche nach sich schneidenden Dreiecken notwendig
- große Mengen an Dreiecken (mehrere 10 bis 100 Tausend)
- Temporärer Aufbau eines R-Baums (im Hauptspeicher) sinnvoll
- Implementierung unter `secondo/include/mmrtree.h` vorhanden



Aufgabe  
Phase2

Thomas Behr

- Motivation
- Typen
- Import
- Export
- Erzeugen
- Transformieren
- Verbinden
- Zerlegen
- Sonstige



Für Fragen bereit ich bin.