

---

CastNet Vorstellung  
OpenFOAM-Stammtisch Mosbach





## Hintergrund:

- Dienstleister im Bereich CFD/FEA und FSI

## Erfahrung aus vielen Projekten:

- Meist komplexe Probleme (z.B. FSI)
- Kunde will mittelfristig die Probleme selbst lösen können (inhouse)
- Kunde sucht oft Lösung für eine spezielle Anwendung
- Kunde möchte keinen Bericht sondern ein Modell



## Kernkomponente:

CAD-Aufbereitung, Vernetzungstechnologie, abstrakte Modellierung:

Lizenzieren wir (langjährige Entwicklung unseres amerikanischen „OEM“-Partners Simmetrix)

## GUIs, Solver, Exportfunktionalitäten, Modelle:

Entwickeln wir („VAR“)



## Cast Net: Fischernetz

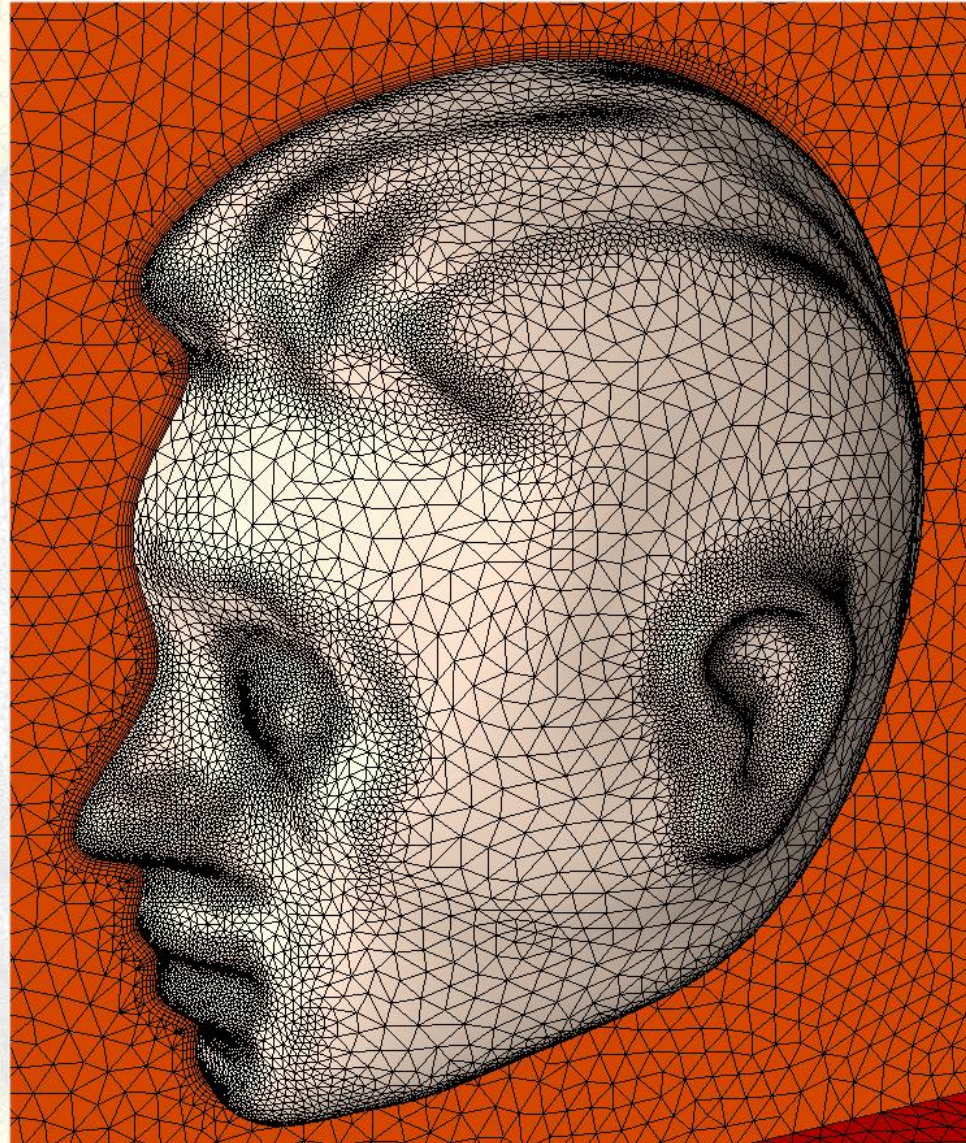
- einfach und effizient
- für die tägliche Arbeit
- zuverlässig
- keine weiteren Hilfsmittel



Vernetzungstechnologie:

- krümmungskontrolliert
- relative Elementgrößen

-> wenige Parameter  
modellgerechte Gitter





## CastNet-Modellierung für OpenFOAM®

### CastNet:

- Modellierungs- und Vernetzungsumgebung für CFD and FEA
- CAD-Modell basierend (Parasolid -xt, Acis –sat oder Granite Pro-E)
- Generiert hybride Gitter
- CFD spezifische Feature (Boundary-layer-Vernetzung, krümmungskontrollierte isotrope/anisotrope Vernetzung)
- Direkte OpenFOAM-Modellierung (Randbedingungen, Zones, Solver, Material...)
- CastNet basiert auf Simmetrix Inc. Vernetzungs-, CAD Aufbereitung- und abstrakter Modellierungstechnologie

**OPENFOAM® is a registered trade mark of OpenCFD Limited**



## **Beispiel 1:**

CastNet Modeling für SimpleFOAM  
(stationär, inkompressibles Fluid)

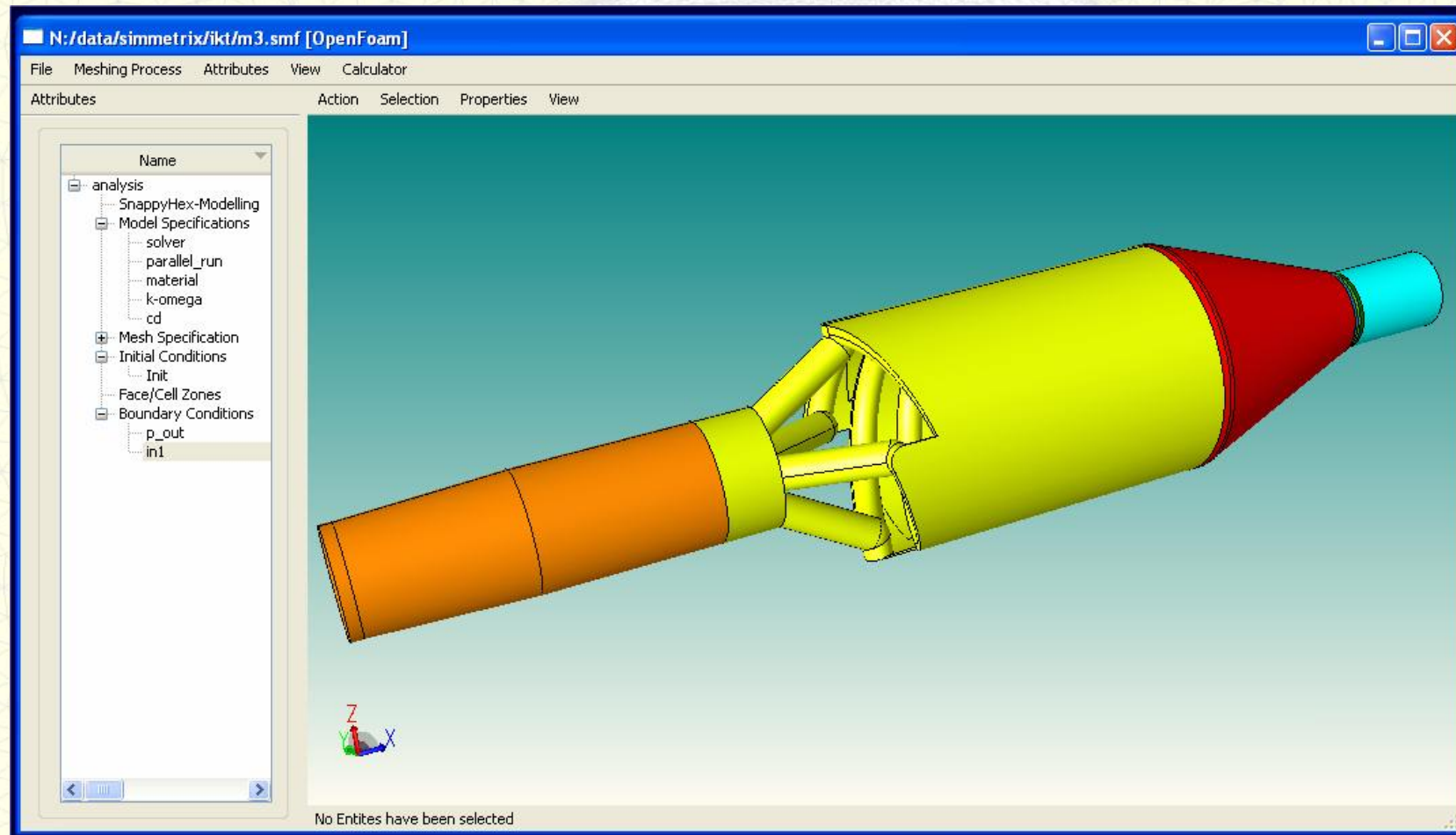
Gitter generiert in CastNet

Vernetzungs-Beispiel: Extrusionswerkzeug

Freundlicherweise zur Verfügung gestellt vom Institut für  
Kunststofftechnik, Universität Stuttgart

<http://www.ikt.uni-stuttgart.de>

## Beispiel 1: SimpleFoam-Modellierung

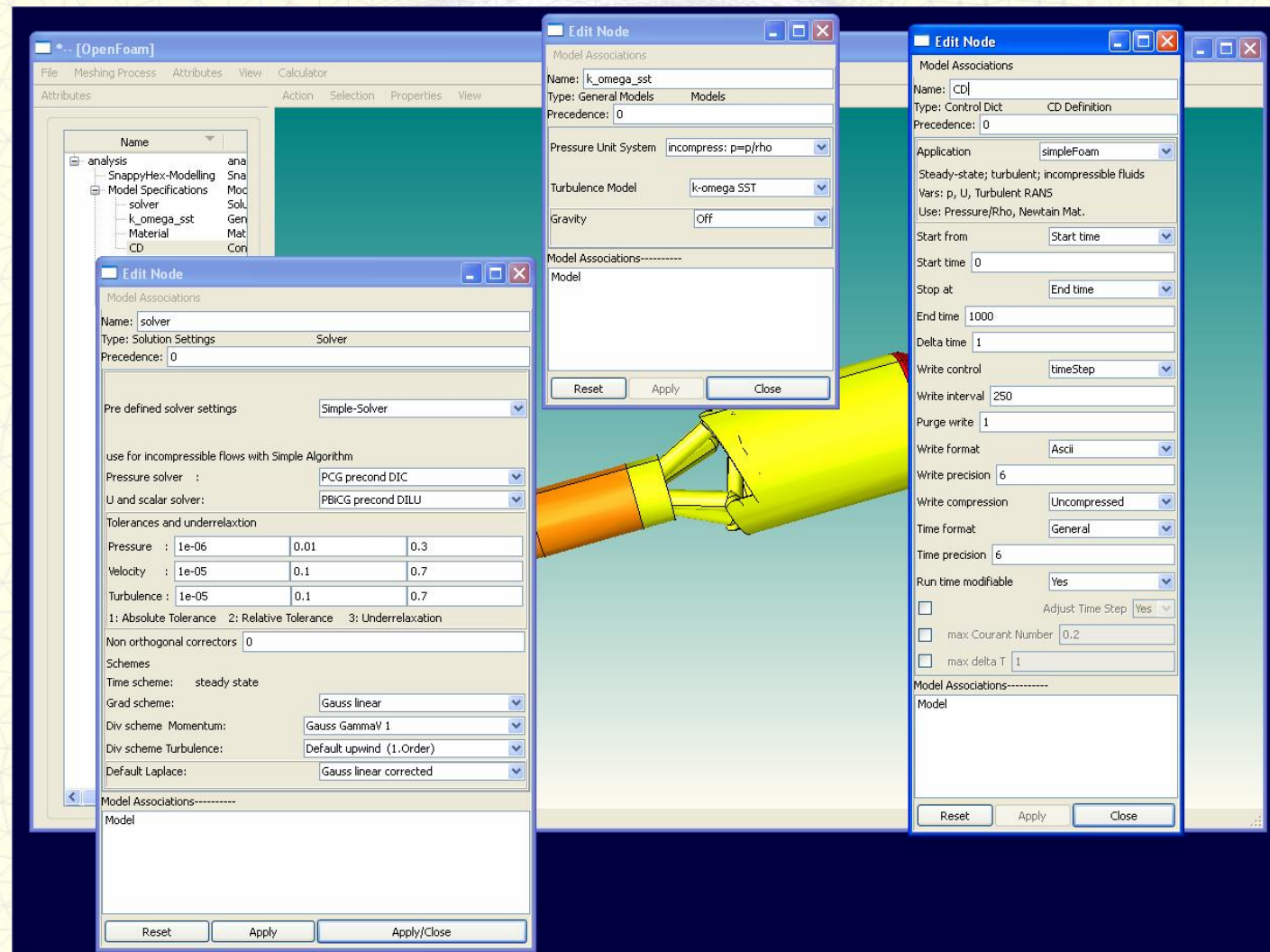




# Beispiel 1: SimpleFoam-Modellierung



Solver Einstellungen,  
Kontrollparameter  
(controlDict) und  
Materialdaten



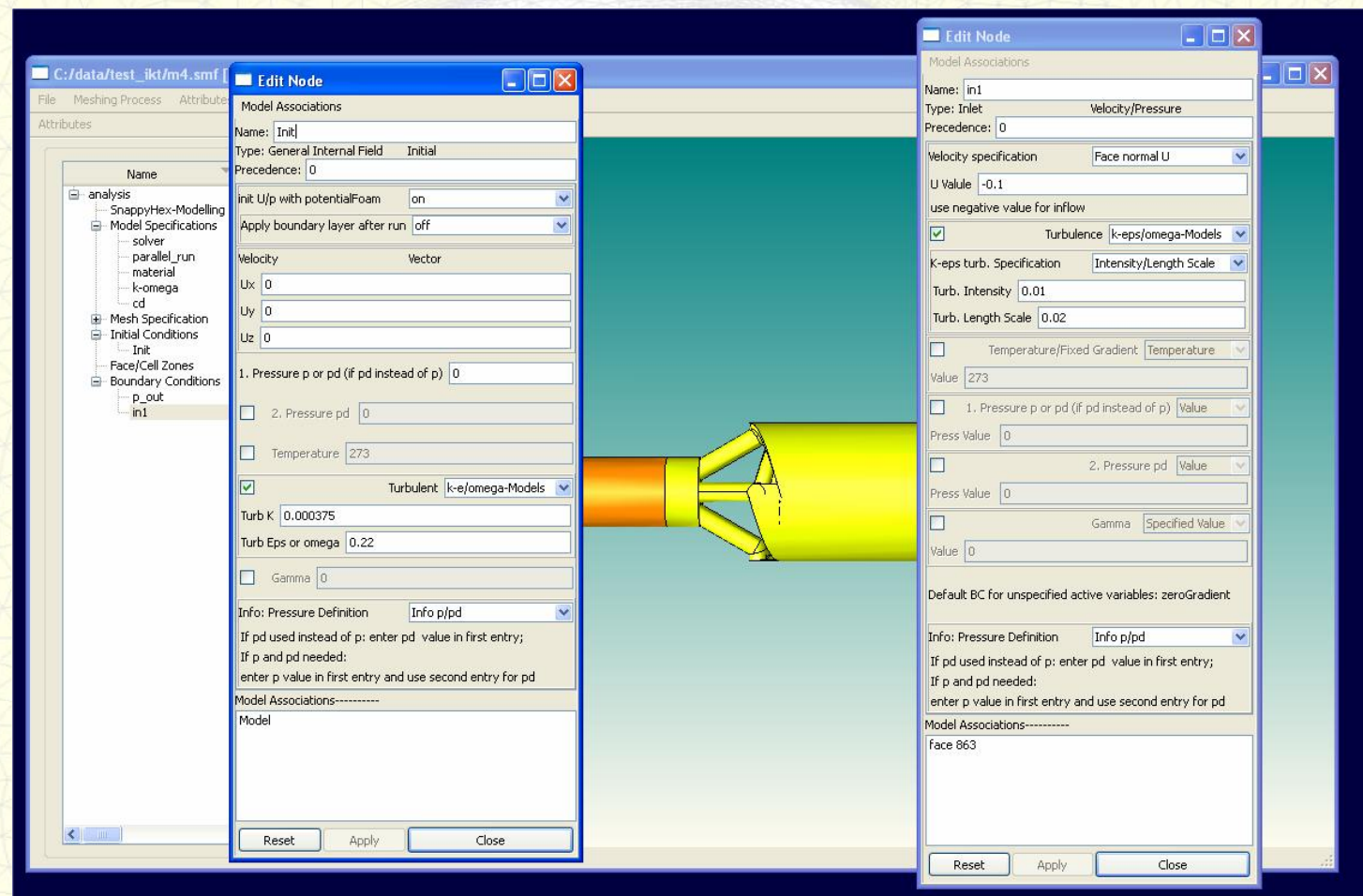


# Beispiel 1: SimpleFoam-Modellierung



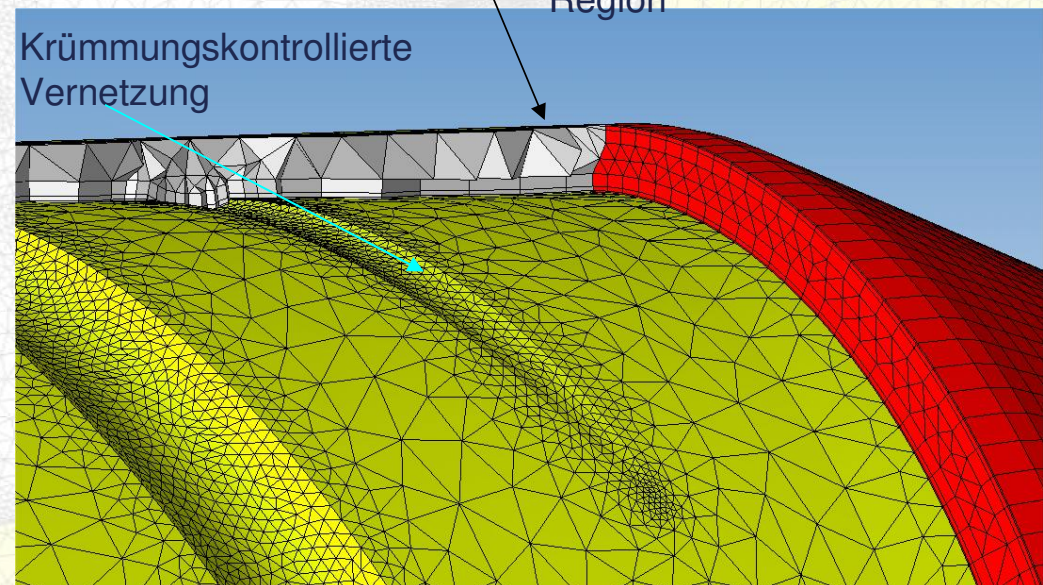
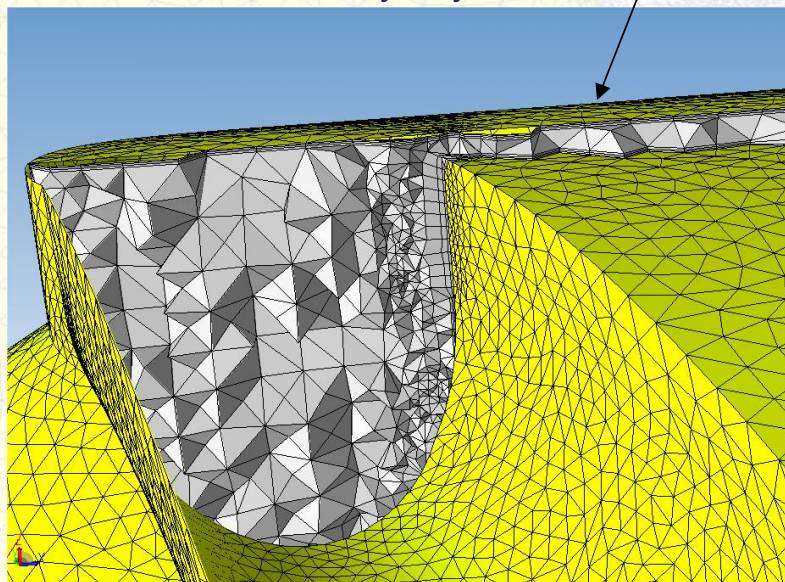
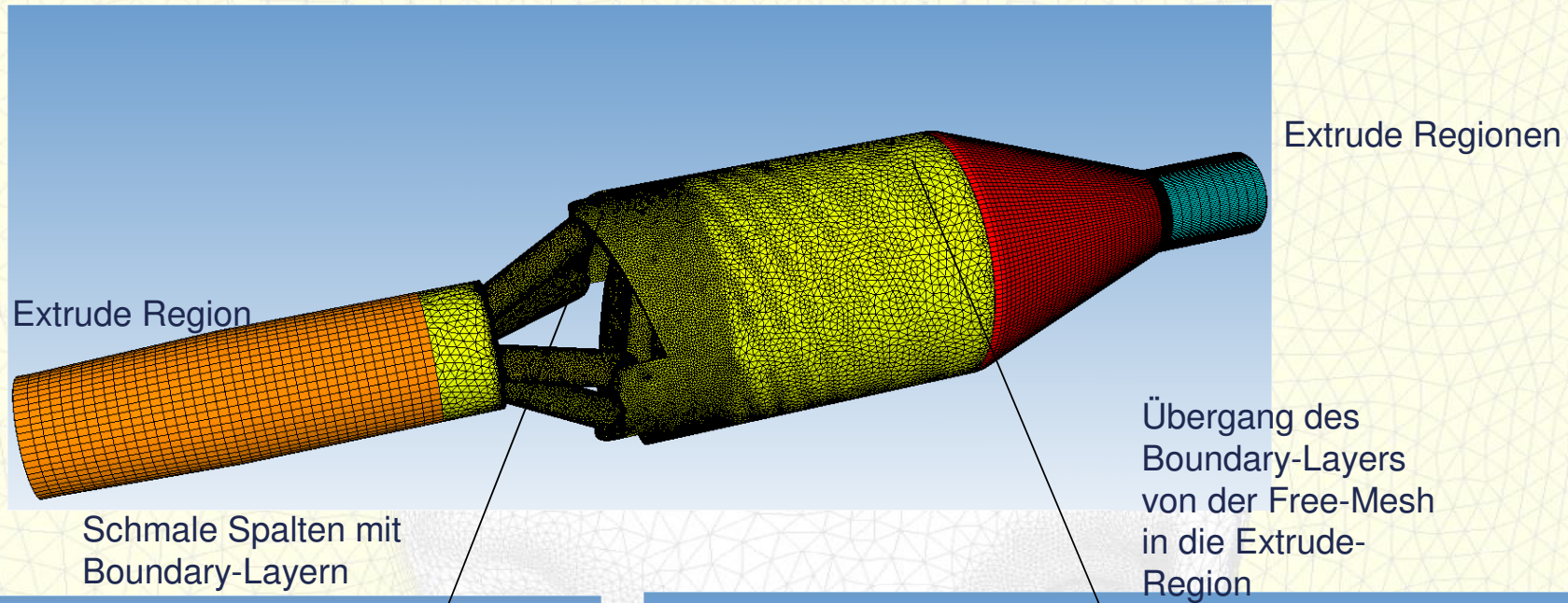
Anfangsbedingungen  
und  
Randbedingungen

Option:  
Initialisierung mit  
potentialFOAM  
und  
applyBoundaryLayer  
für gute Startwerte





## Beispiel 1: SimpleFoam-Gitter

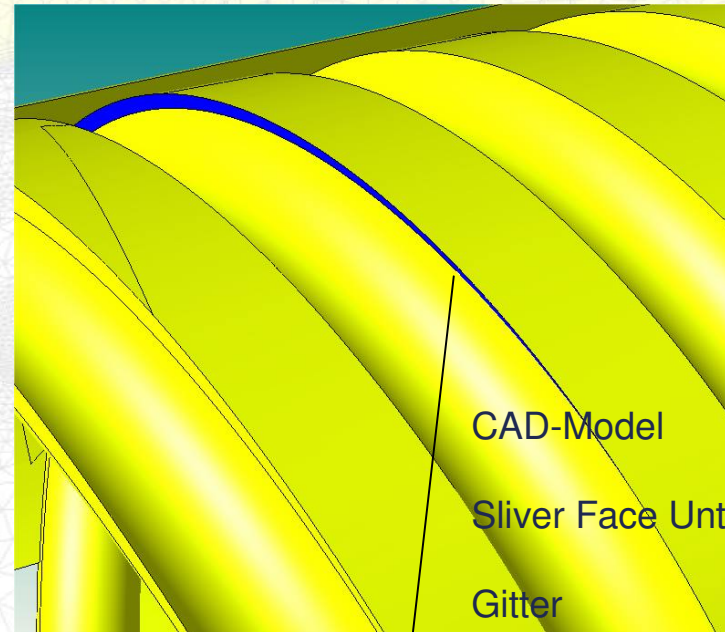
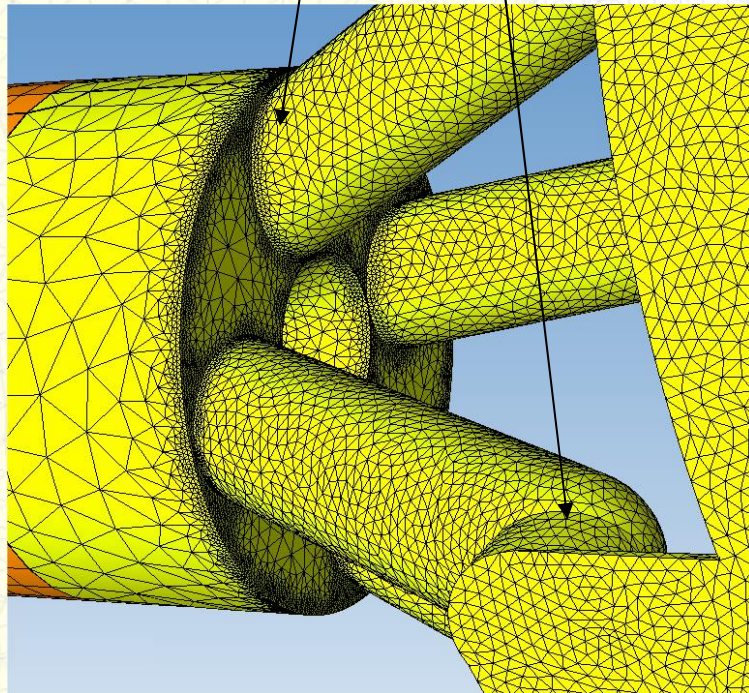




## Beispiel 1: SimpleFoam-Gitter



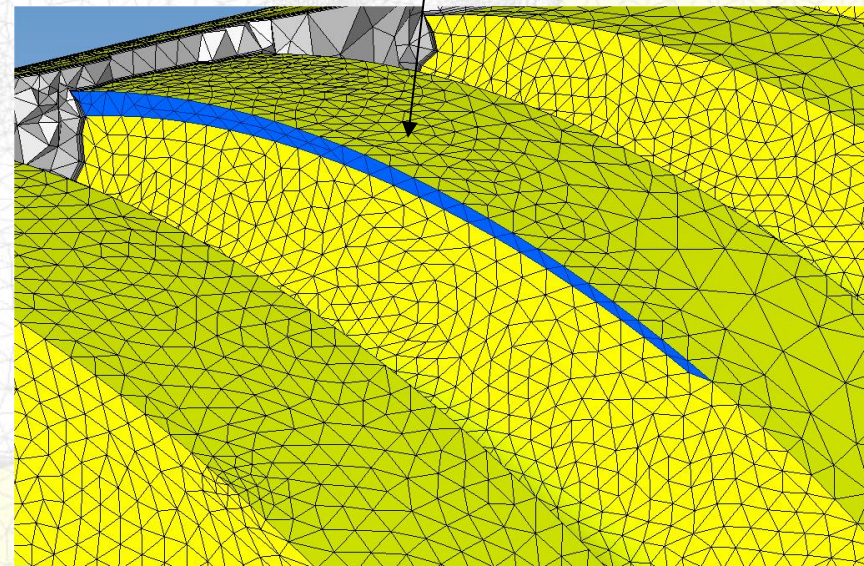
Schwierige Umlenksituationen:  
Alle Flächen mit Boundary-Layern vernetzt



CAD-Model

Sliver Face Unterdrückung

Gitter





## Beispiel 1: SimpleFoam-Dateien



Skript Export um den OpenFOAM-Fall zu generieren, zu lösen und zusammenzuführen.

Es müssen nur 3 Kommandos eingegeben werden:

**Foam\_pre** (Erstellen des Falls)

**Foam\_solve** (Lösen)

**Foam\_post** (Zusammenführen der Ergebnisse bei paralleler Rechnung)

Es werden nur Linux und reine OpenFOAM-Funktionalitäten verwendet.

```
foam_pre.sh
1 dos2unix *
2 mkdir constant/
3 mkdir constant/polyMesh/
4 mkdir constant/polyMesh/sets
5 mv points owner faces neighbour boundary co
6 mv RASProperties constant/
7 mv transportProperties constant/
8 mkdir system/
9 mv controlDict system/
10 mv createPatchDict system/
11 mv fvSchemes system/
12 mv fvSolution system/
13 mv setFieldsDict system/
14 createPatch -overwrite
15 mv U 0/
16 mv p 0/
17 mv k 0/
18 mv omega 0/
19 cp -r 0 0.org
20 renumberMesh -overwrite
21 cp system/fvSchemes system/fvSchemes_2
22 cp system/fvSolution system/fvSolution_2
23 mv fvSchemes_pot system/
24 mv fvSolution_pot system/
25 cp system/fvSchemes_pot system/fvSchemes
26 cp system/fvSolution_pot system/fvSolution
27 potentialFoam -writep
28 cp system/fvSchemes_2 system/fvSchemes
29 cp system/fvSolution_2 system/fvSolution
30

omega
1 /*----- C++ -----*/
2 | ===== |
3 | \ \ / F i e l d | OpenFOAM: The Open Source CFD
4 | \ \ / O p e r a t i o n | Version: 1.5
5 | \ \ / A n d | Web: http://www.OpenFOAM
6 | \ \ / M a n i p u l a t i o n |
7 |-----|
8 FoamFile
9 {
10     version      2.0;
11     format        ascii;
12     class         volScalarField;
13     object        omega;
14 }
15 dimensions      [0 0 -1 0 0 0 0];
16
17 internalField uniform 0.22;
18
19 boundaryField
20 {
21     {
22         in1
23         {
24             type      turbulentMixingLengthFrequencyInlet;
25             mixingLength 0.02;
26             k          k;
27             value      $internalField;
28         }
29
30         p_out
31         {
32             type      zeroGradient;
33         }
34
35         default_wall
36         {
37             type      zeroGradient;
38         }
39     }
40 }
```

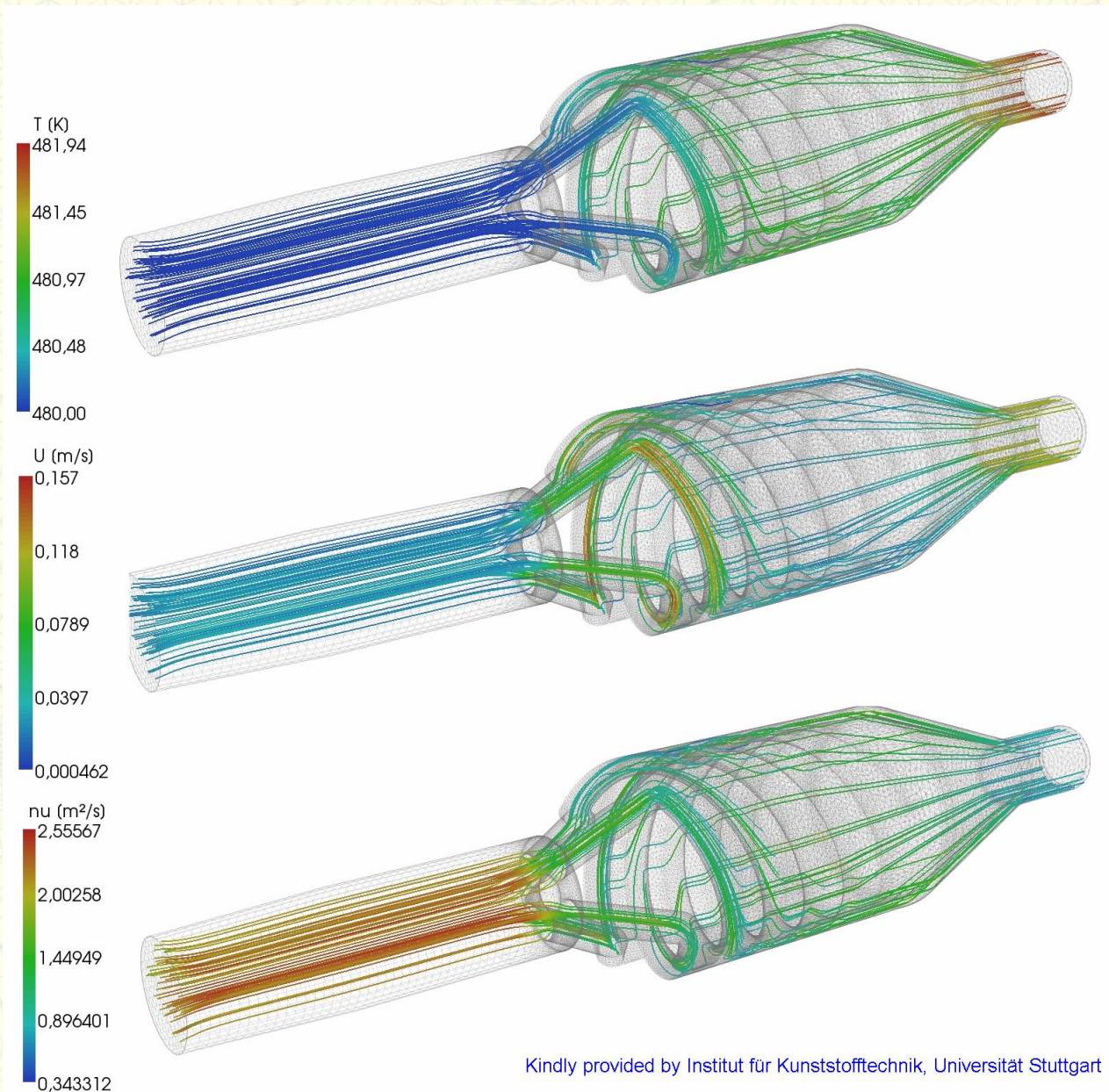


## Beispiel 1: SimpleFoam + Wärmetransport/-Ergebnisse



Rheologische  
Untersuchungen von  
Kunststoffschmelzen am  
IKT, Stuttgart

Visualisiert mit  
ParaView (Open Source)



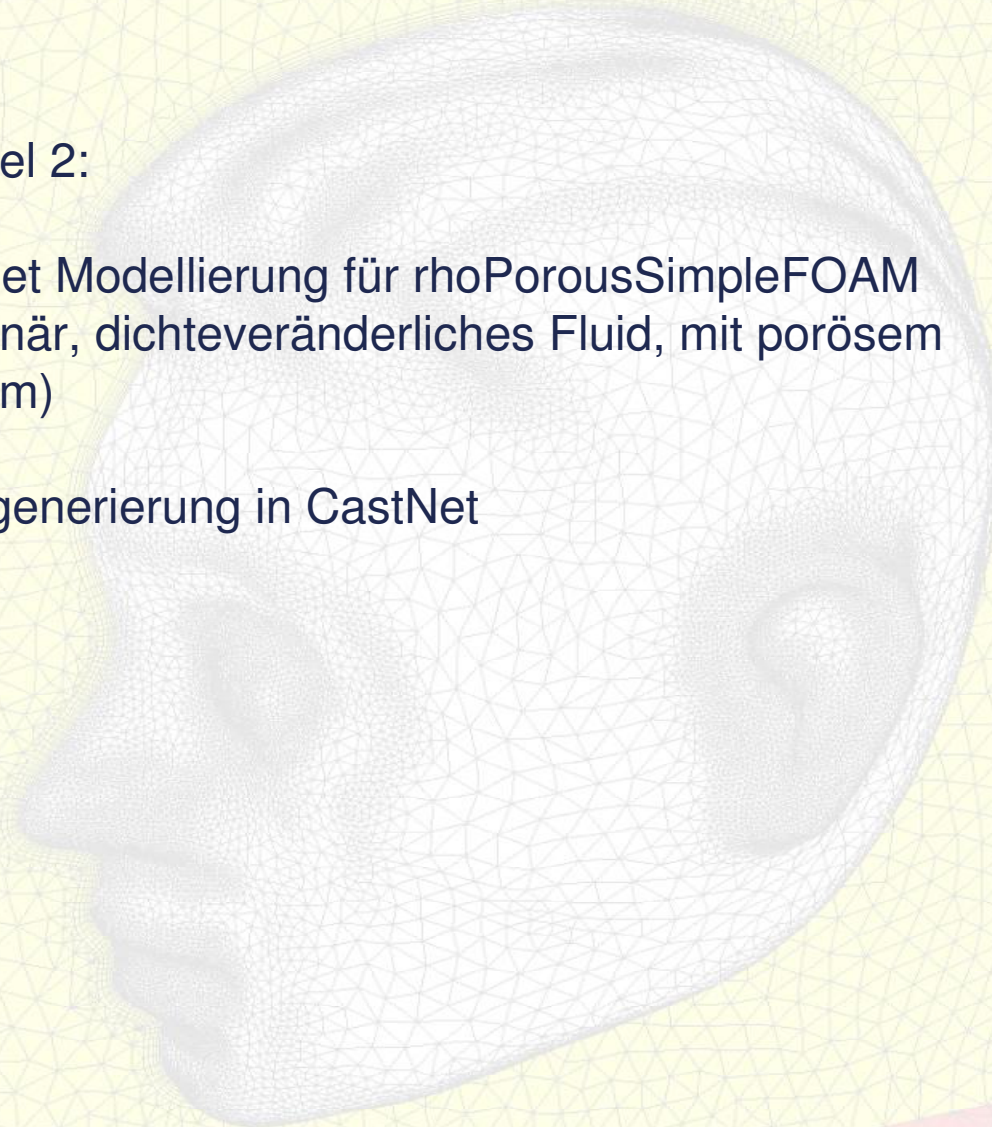
Kindly provided by Institut für Kunststofftechnik, Universität Stuttgart



Beispiel 2:

CastNet Modellierung für rhoPorousSimpleFOAM  
(stationär, dichteveränderliches Fluid, mit porösem  
Medium)

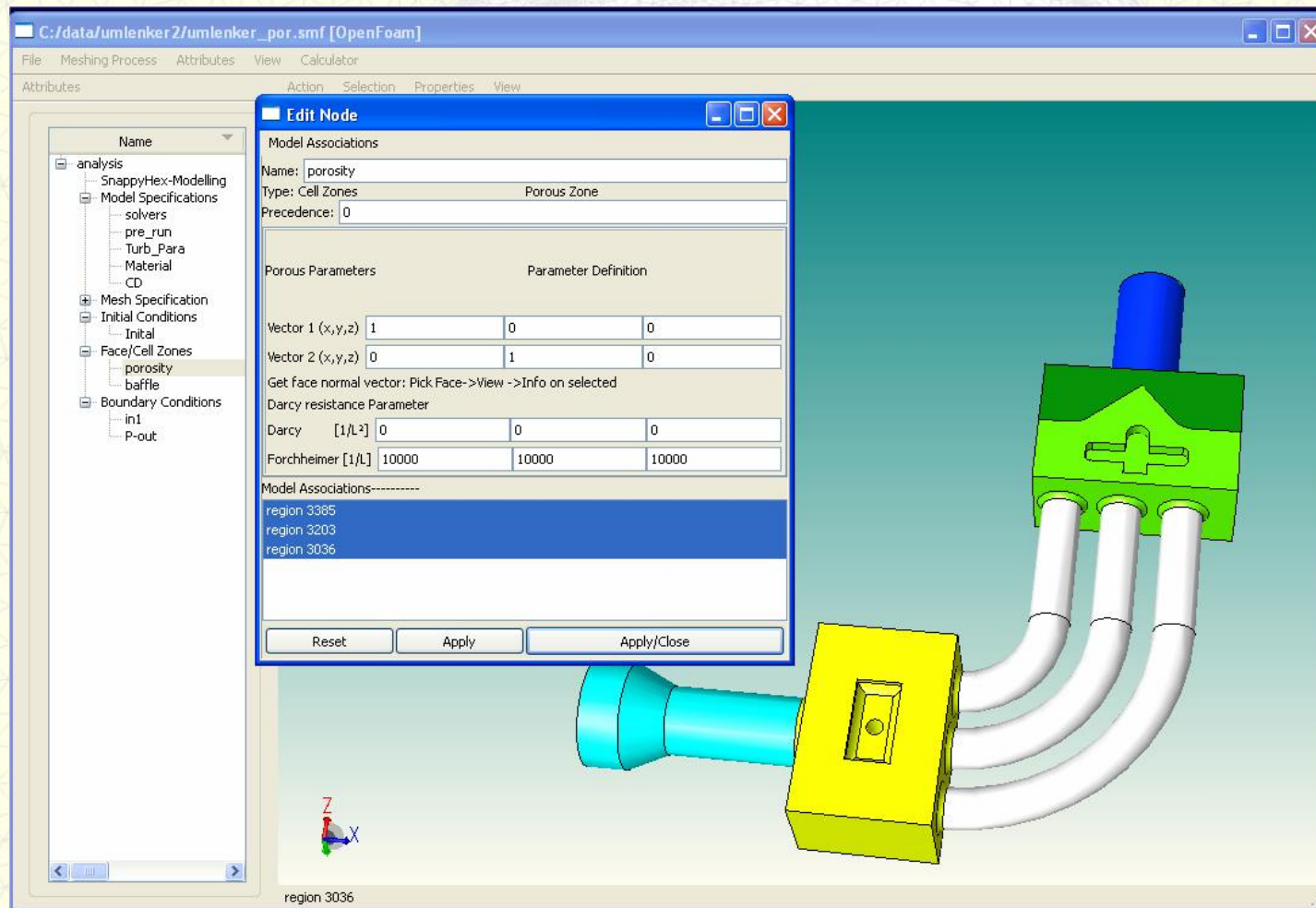
Gittergenerierung in CastNet





## Beispiel 2: rhoPorousSimple-Foam-Modellierung

Poröse Regionen können im CAD-Model direkt angewählt werden (weiße Regionen)

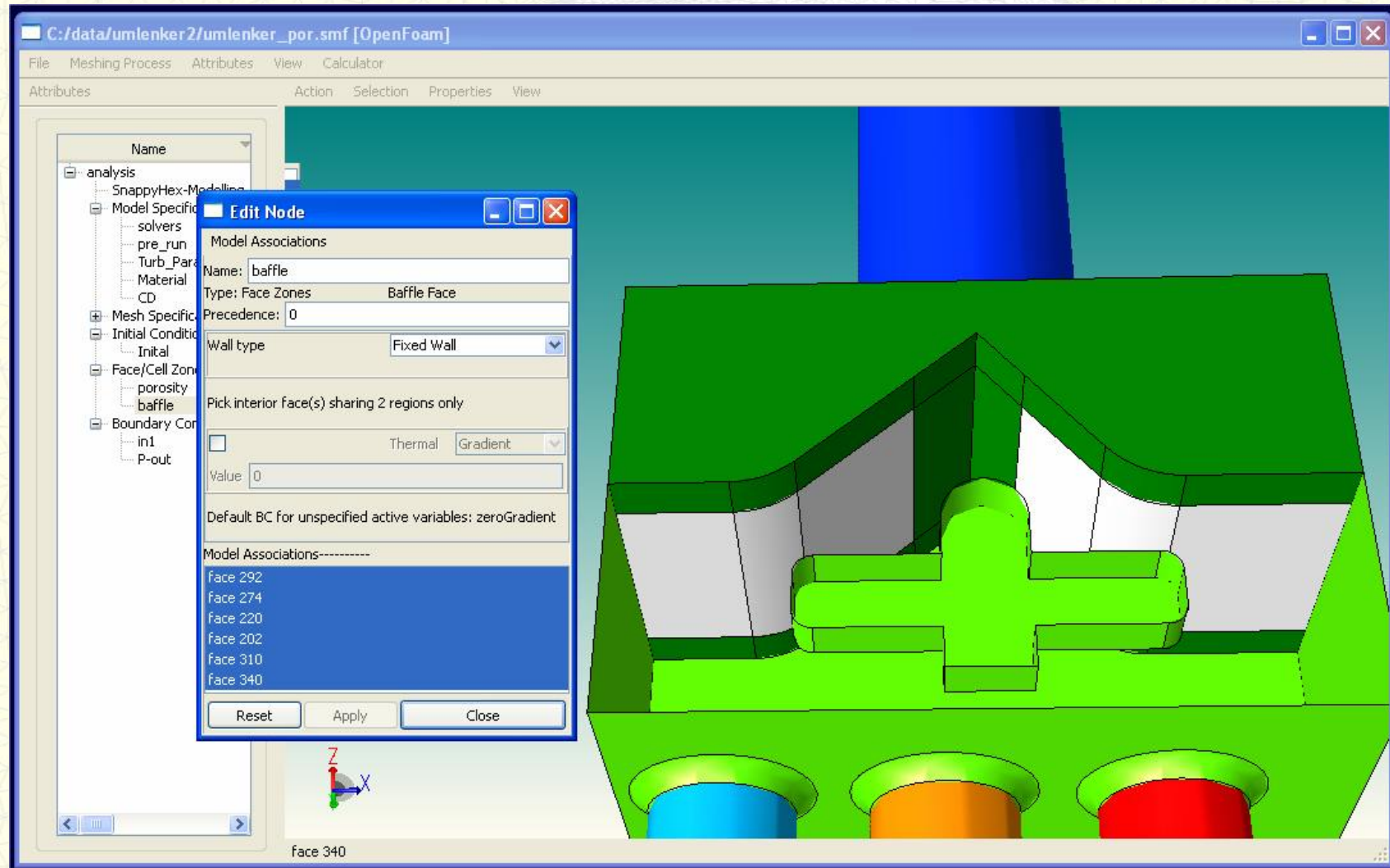




## Beispiel 2: rhoPorousSimple-Foam-Modellierung

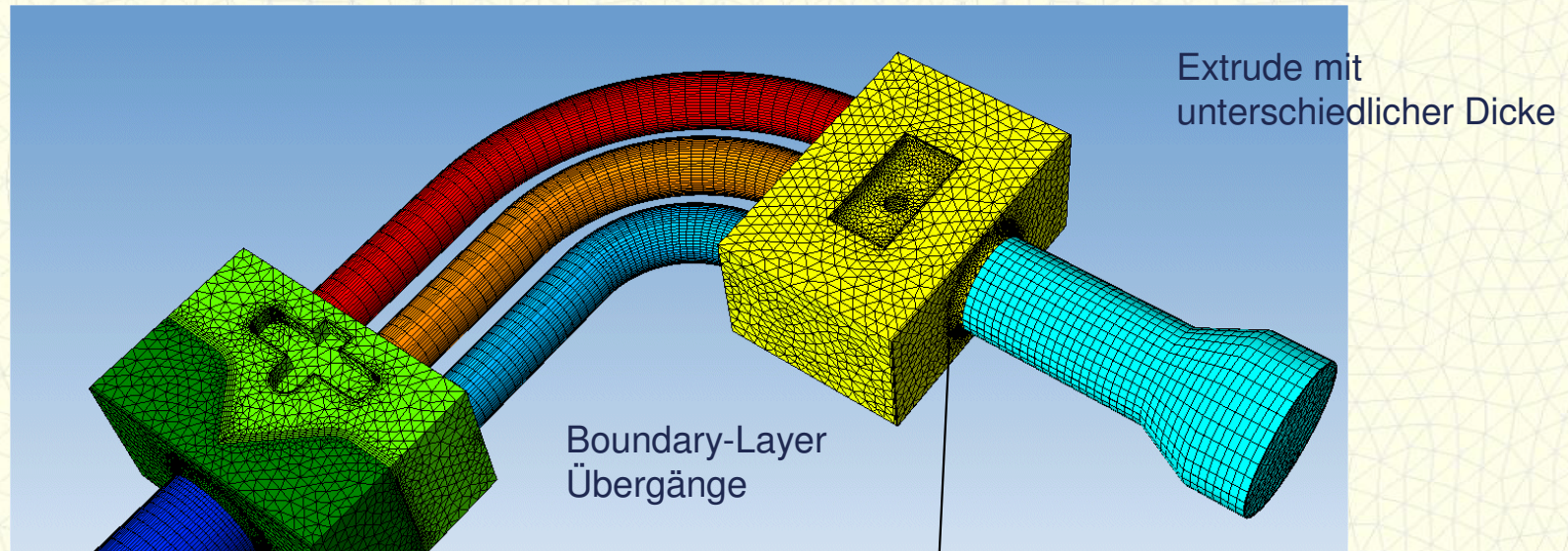


Baffle-faces können im CAD-Model direkt angewählt werden (weiße Flächen)



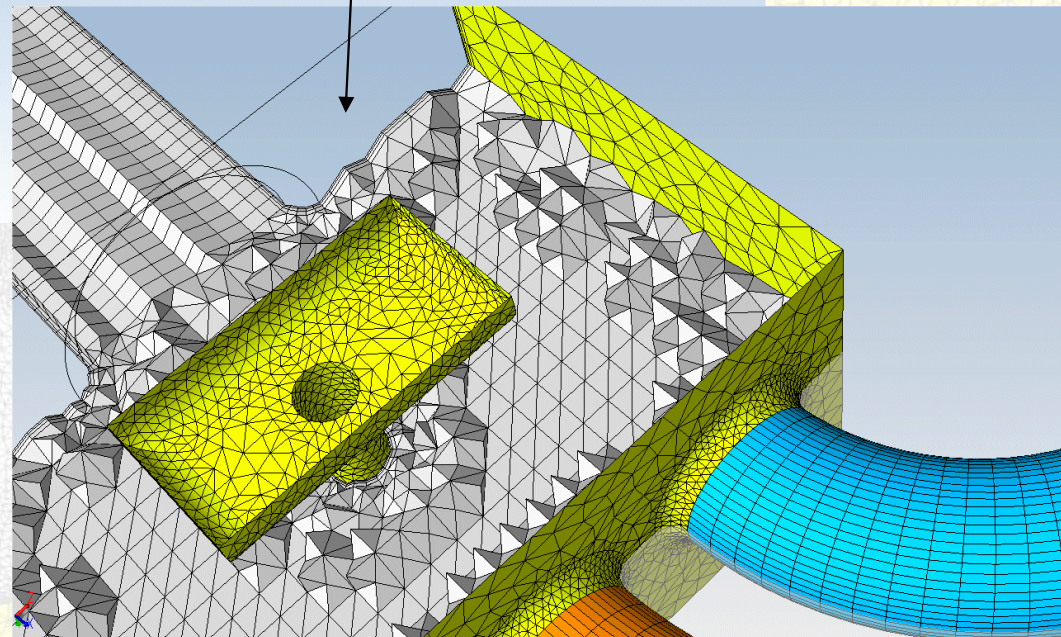


## Beispiel 2: rhoPorousSimple-Foam-Gitter



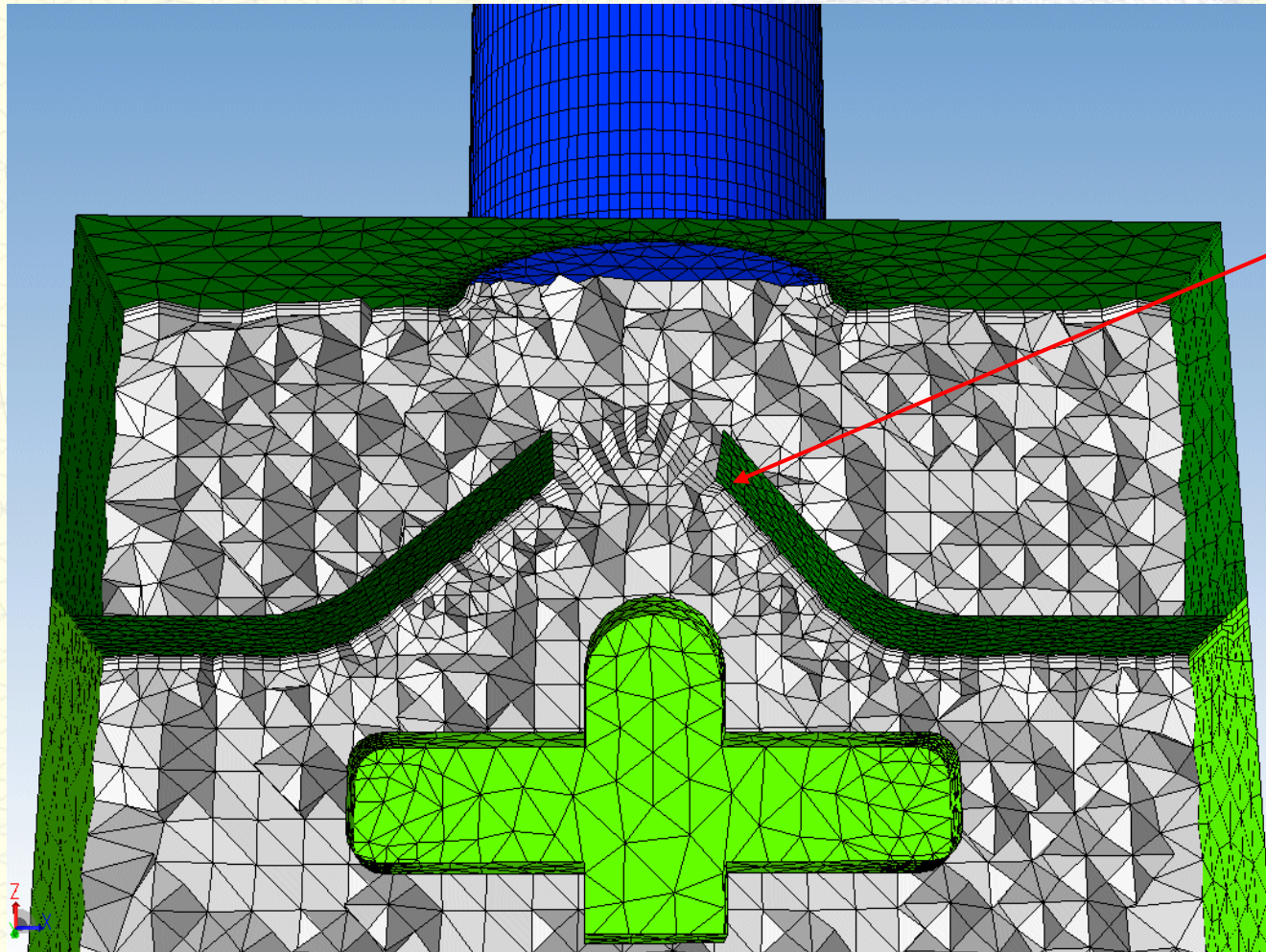
Boundary-Layer  
Übergänge

Extrude mit gestaffelter  
Zellenverteilung





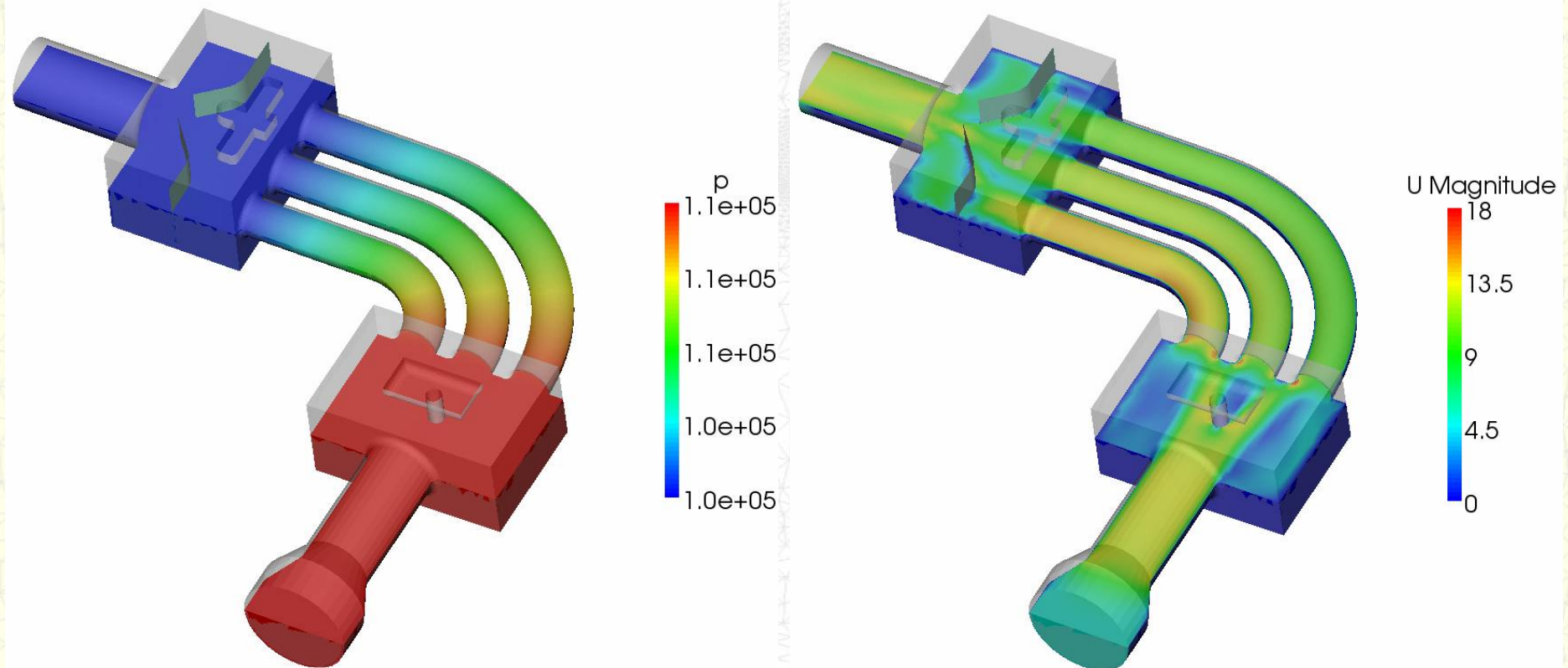
## Beispiel 2: rhoPorousSimple-Foam-Gitter



Boundary-Layer  
Übergang von  
Baffle Face



## Beispiel 2: rhoPorousSimple-Foam-Ergebnisse



Visualisiert mit ParaView (Open Source)



CastNet bietet neben der hybriden internen Vernetzung auch die Modellierung für den Skript-basierenden OpenFOAM-Vernetzer SnappyHexMesh (Hex-dominante Gitter) an.

### **Generellen Vorteile**

- Modellerstellung in einer Umgebung
- Komfortable Definition einer SnappyHexMesh-Vernetzung
- Gitterstudien leicht durchführbar: Umschalten von CastNet zur SnappyHexMesh Vernetzung, beibehalten von Randbedingungen, Solvereinstellungen etc.
- Design- und Konzeptstudie mit abstrakter Modellierung für beide Vernetzungsverfahren

### **Vorteile der SnappyHexMesh-Modellierung:**

- Bestimmte Solver (z.B. \*InterFOAM die Volume-of-Fluid Solver) arbeiten besser mit hex-dominanten Gittern.
- Eine SnappyHexMesh Volumenvernetzung ist auch möglich, wenn das CAD-Modell defekt ist und nur ein lückenhaftes Oberflächengitter erstellt werden kann.

### **Nachteile bei SnappyHexMesh:**

- Kanten können nicht exakt eingehalten werden
- Boundary-Layer sind schwieriger zu kontrollieren

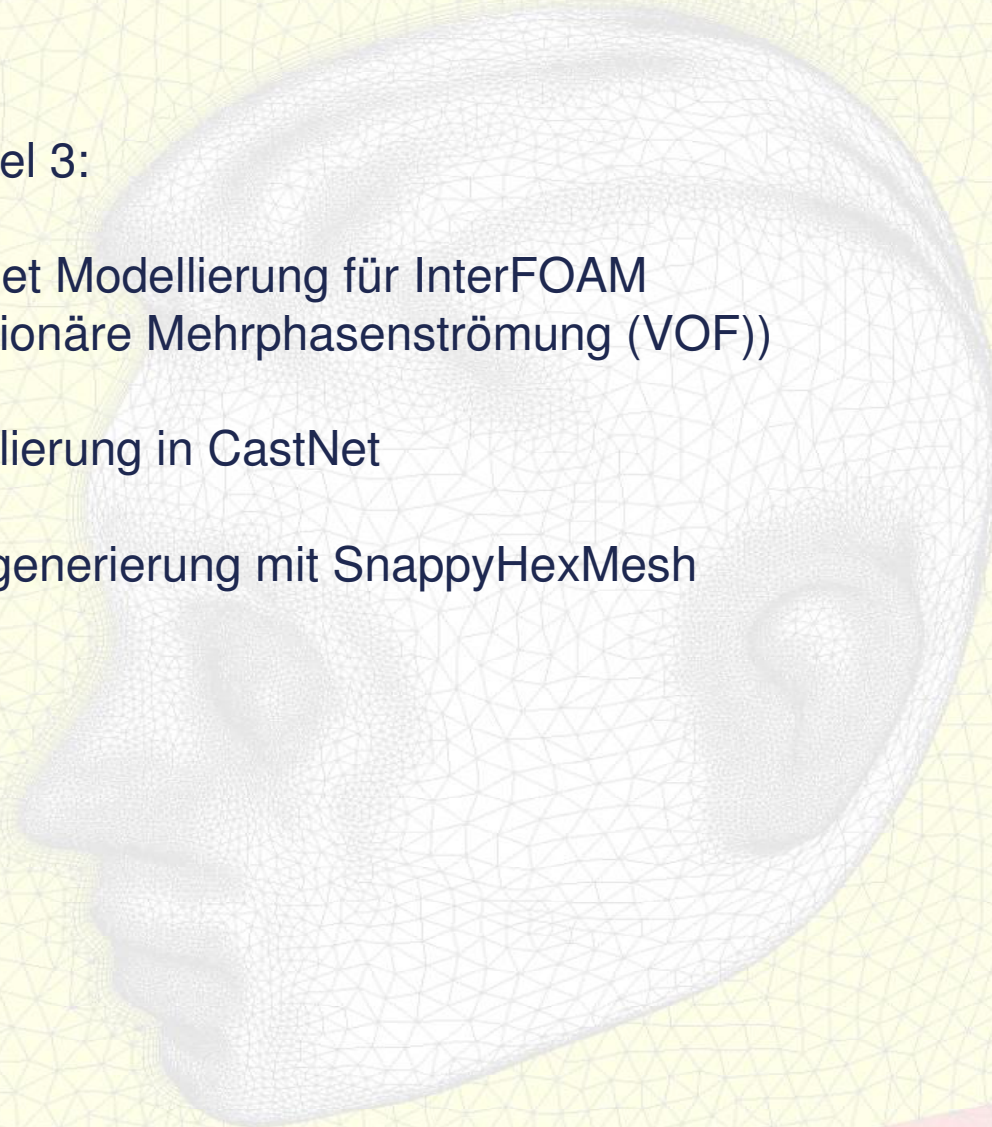


Beispiel 3:

CastNet Modellierung für InterFOAM  
(instationäre Mehrphasenströmung (VOF))

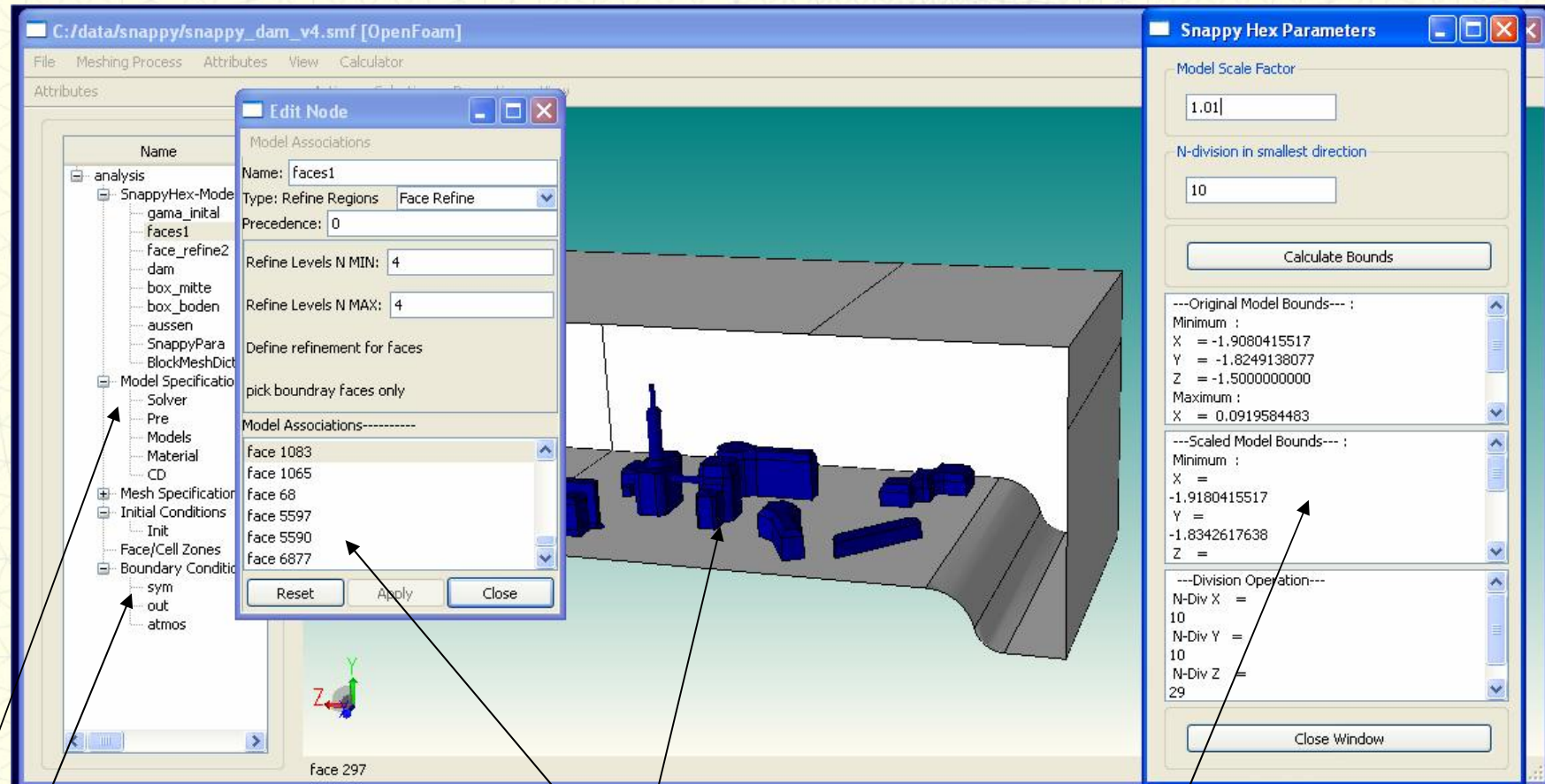
Modellierung in CastNet

Gittergenerierung mit SnappyHexMesh





## Beispiel 3: InterFOAM with SnappyHexMesh-Modellierung



Solver-Setup und  
Randbedingungsdefinition  
bleiben bestehen

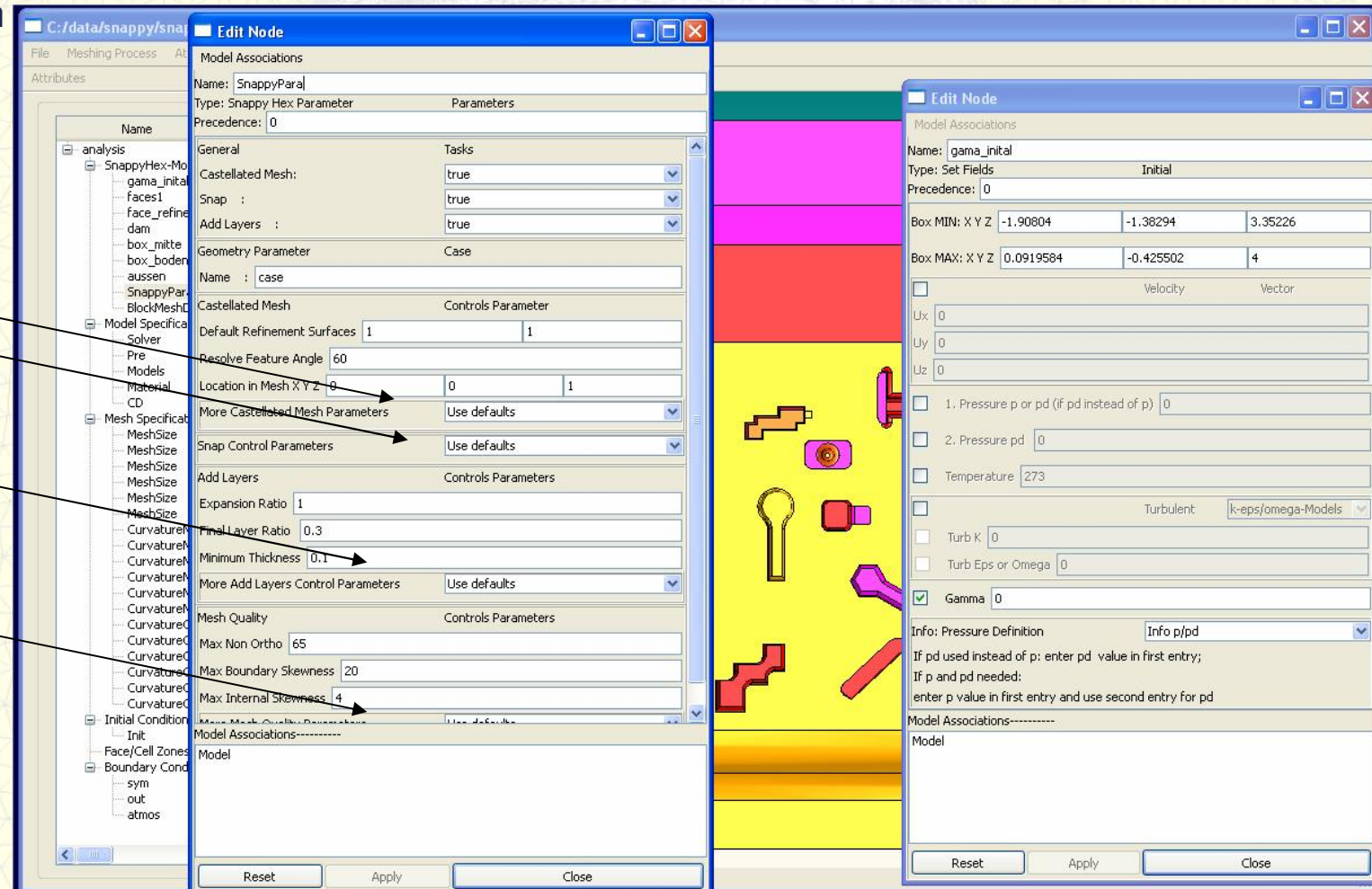
Verfeinerungsflächen  
und Layer können  
einfach angewählt  
werden

Der SnappyHex-Mesh-  
Calculator bestimmt die  
Parameter für das  
Hintergrundgitter



## Lokale Initialisierungen möglich

Optionen alle  
Parameter  
einzustellen





## Beispiel 3: InterFOAM mit SnappyHexMesh-Modeling



Export:  
SnappyHexMeshDict  
BlockMeshDict  
Spezielles stl-File

+ Skripte um den Job  
zu generieren

Beispiel:  
Flächen

Exportiertes  
snappyHexMeshDict:  
Alle Flächen sind  
verfügbar und es  
können in CastNet  
individuelle Layer und  
Verfeinerungen  
definiert werden.

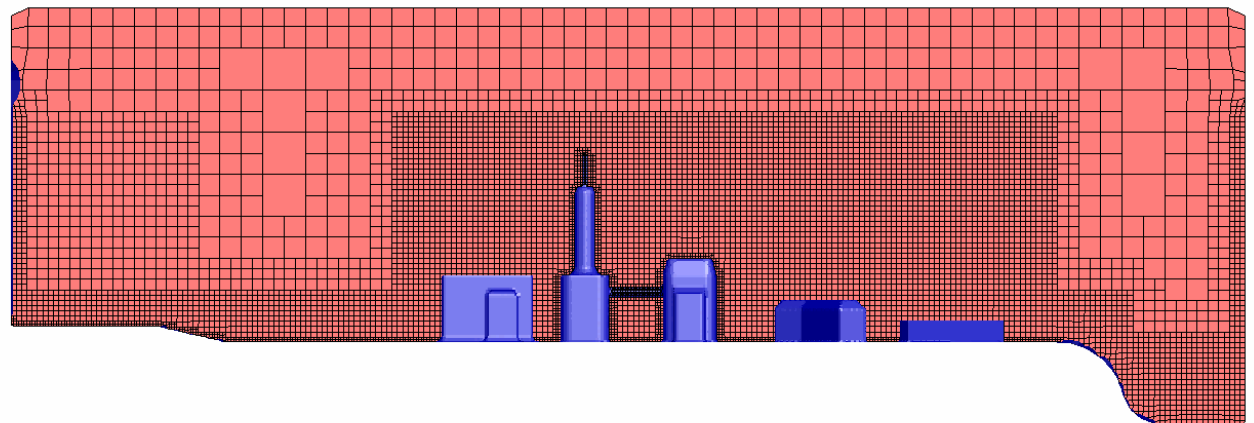
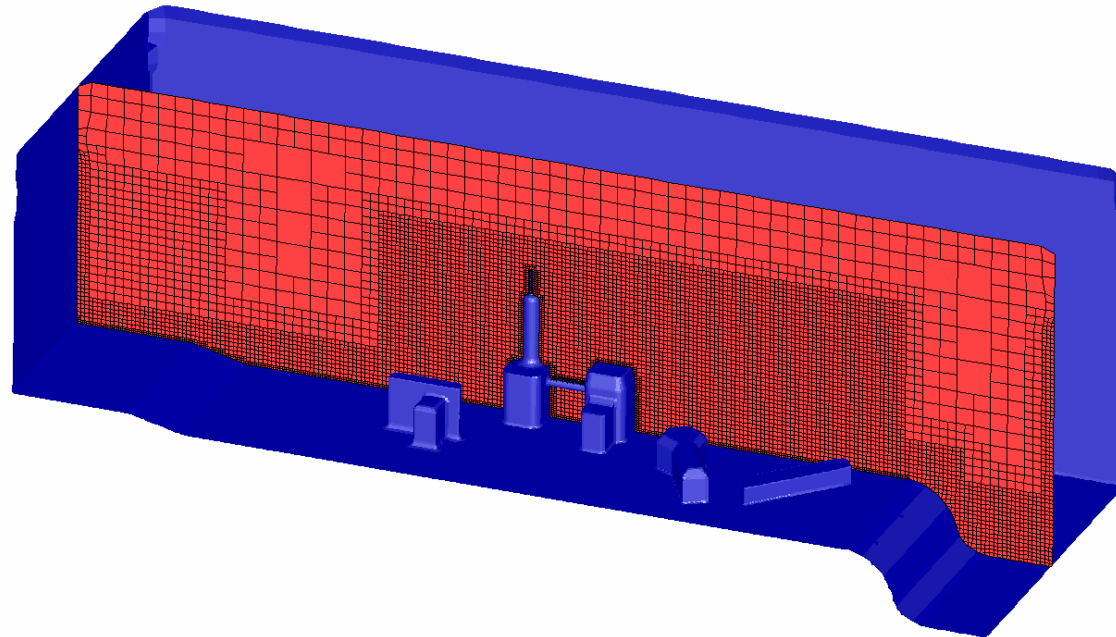
```
1 /*----- C++ -----*/
2 |=====|
3 | \ \ / F i e l d | OpenFOAM: T
4 | \ \ / O p e r a t i o n | Version: 1
5 | \ \ / A n d | Web: h
6 | \ \ / M a n i p u l a t i o n |
7 |-----|
8 FoamFile
9 {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     object        autoHexMeshDict;
14 }
15
16 // *****
17 castellatedMesh true;
18 snap            true;
19 addLayers       true;
20
21 geometry
22 {
23     case.stl
24     {
25         type triSurfaceMesh;
26         regions
27         {
28             face-6432
29             {
30                 name face-6432;
31             }
32             face-6439
33             {
34                 name face-6439;
35             }
36             face-6453
37             {
38                 name face-6453;
39             }
40             face-6462
41             {
42                 name face-6462;
43             }
44             face-5597
45             {
46                 name face-5597;
47             }
48         }
49     }
50 }
```

```
1 /*----- C++ -----*/
2 |=====|
3 | \ \ / F i e l d | OpenFOAM: The Open Sc
4 | \ \ / O p e r a t i o n | Version: 1.5
5 | \ \ / A n d | Web: http://www.
6 | \ \ / M a n i p u l a t i o n |
7 |-----|
8 FoamFile
9 {
10     version      2.0;
11     format        ascii;
12     class         dictionary;
13     object        createPatchDict;
14 }
15 // *****
16
17 matchTolerance 1E-3;
18 pointSync true;
19 patches
20 {
21     {
22         name out;
23         type patch;
24         constructFrom patches;
25         patches (
26             face-6094
27             face-267
28         );
29     }
30 }
31 {
32     name sym;
33     type symmetryPlane;
34     constructFrom patches;
35     patches (
36         face-6901
37         face-5625
38         face-287
39         face-6462
40         face-6078
41         face-6439
42         face-6870
43         face-6064
44         face-6453
45         face-5639
46         face-6892
47         face-254
48     );
49 }
```

Zur  
Randbedingungs-  
definition werden  
die Flächen  
automatisch  
zusammengefasst



### Beispiel 3: InterFOAM mit SnappyHexMesh





**Stand:**

Verstärkter Bedarf an CastNet von Firmen ohne CFD-Infrastruktur (OpenFOAM alleiniger CFD-Einsatz)

Verstärkter Bedarf an CastNet von Universitäten, kleineren Ingenieurbüros und OpenFOAM-Einsteigern

**Ausblick:**

Ausbau der Stärken von CastNet für OpenFOAM-Modellierung: Einfaches Adressieren von 2d (faces/CAD-Flächen) und 3d (cells/ CAD-Regionen) Bereichen

Umsetzung weiterer Solver:

MRF\* und chtMultiRegion

Umsetzung weiterer Feature

GGI für rotierende Systeme

Suche nach spezialisierten Partnern für Kooperationen in der OpenFOAM-Modell-Entwicklung