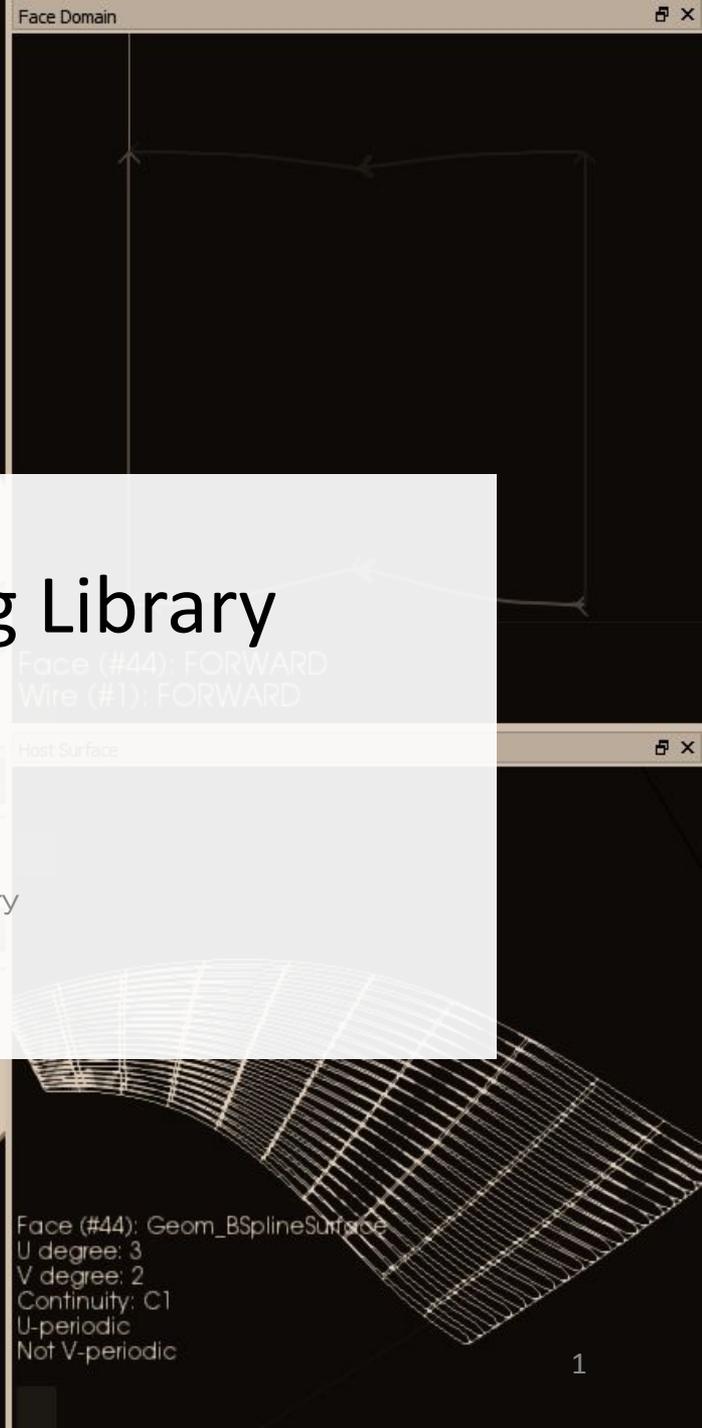




# OpenCascade Modeling Library

## Практический курс Задачник

Лаборатория Компьютерной Графики ННГУ  
Нижний Новгород  
2016



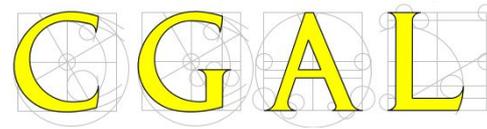
# Урок 1: ВВЕДЕНИЕ

Материалы курса:

<https://sites.google.com/site/modelingpractice>

# OpenCascade – это библиотека (SDK)

Разработка программного обеспечения в наши дни – это редко написание кода с нуля. Для создания коммерчески успешного ПО мирового уровня требуется привлечение сторонних библиотек.



# Цель курса

- Предложить новый инструмент для вашего арсенала разработчика и исследователя
  - Цель 1: вы можете создавать наукоемкое инженерное ПО
  - Цель 2: вы входите в актуальную исследотвальскую область

**ВНИМАНИЕ!**

**Без компьютера здесь ДЕЛАТЬ**

# Что нужно знать?

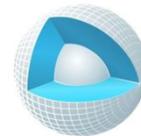
- Язык C++;
- Основы геометрического моделирования:
  - Численные методы;
  - Вычислительная геометрия.
- Базовая «инженерная культура»:
  - Термины «изделие», «сборка», «деталь», САПР и пр. не должны ввергать в ступор.

# OpenCascade не уникален

- ACIS
- Parasolid
- SMLib
- C3D
- RGK
- OpenCascade



SIEMENS



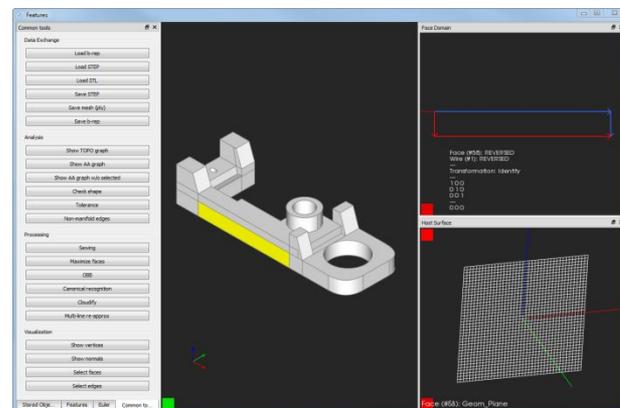
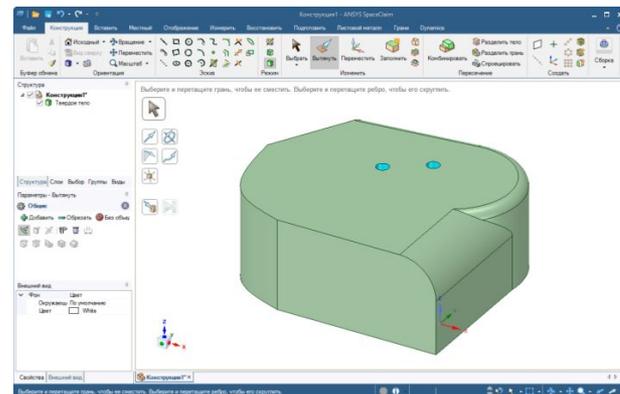
C3D Labs



# Примеры инженерного ПО

- **ACIS:** ANSYS SpaceClaim
- **Parasolid:** SolidWorks
- **C3D:** KOMPAS-3D
- **OpenCascade:** FreeCAD

Зона применения: инженерное и наукоемкое ПО



Проектирование

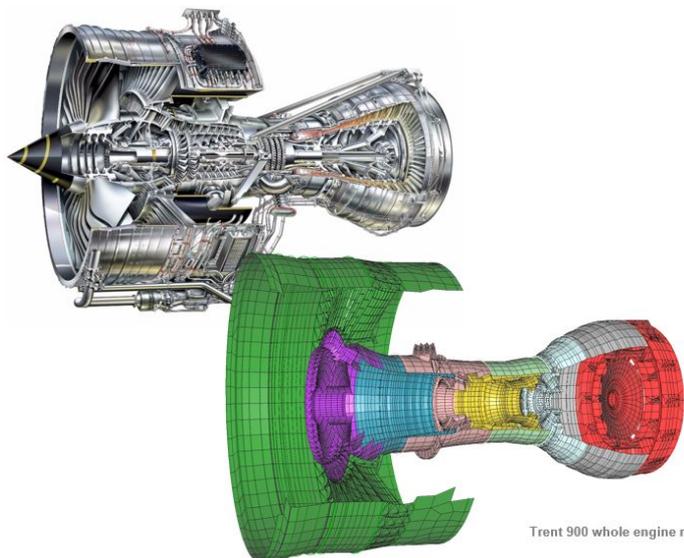
Анализ (прочность, колебания, ...)

Производство

е

о

# Проектирование и Расчет

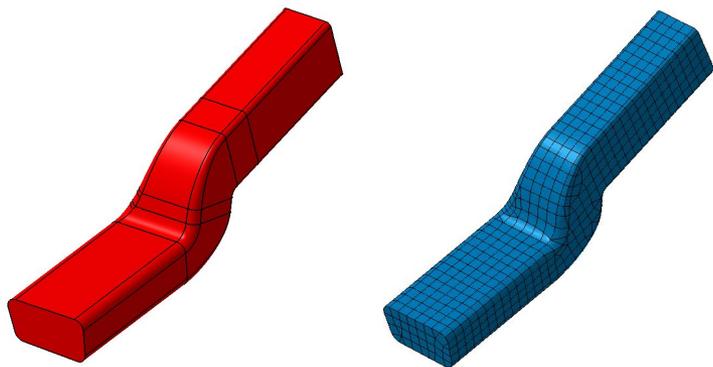


Trent 900 whole engine model: © 2003 Rolls-Royce plc

В любом случае нужна математическая модель самого объекта – геометрическая модель.

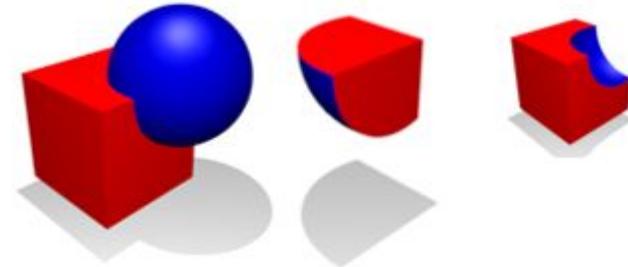


**B-Rep**

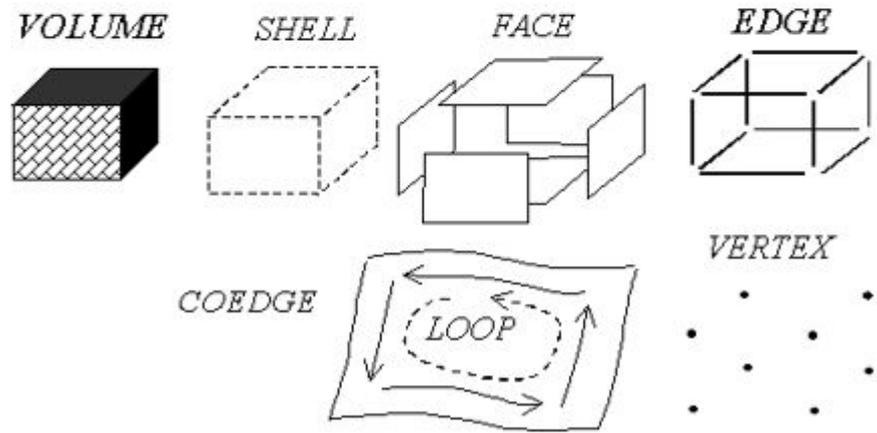


# Точное представление объекта

- Constructive Solid Geometry (CSG)
  - the shape is determined using Boolean set operations (Union, Intersection, Difference) on a set of primitive shapes such as blocks, cylinders, spheres etc.



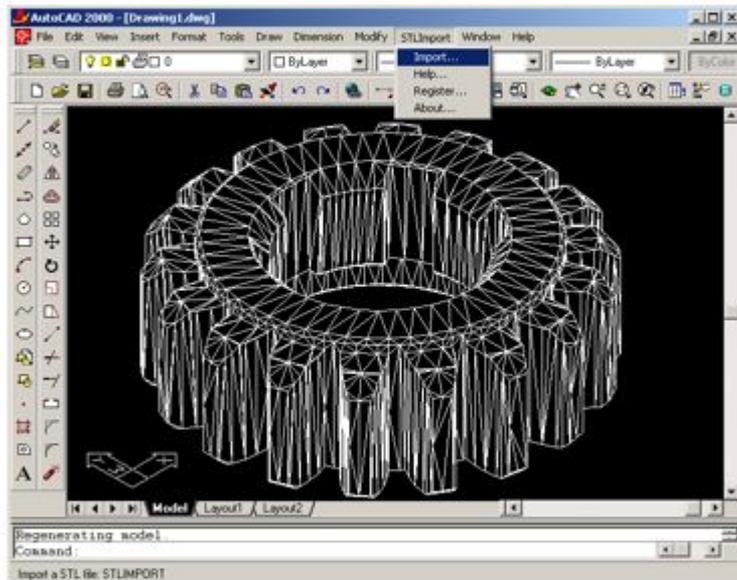
- Boundary representation (B-REP)
  - the shape is represented as a collection of faces, edges, and vertices.



# Неточное представление объекта

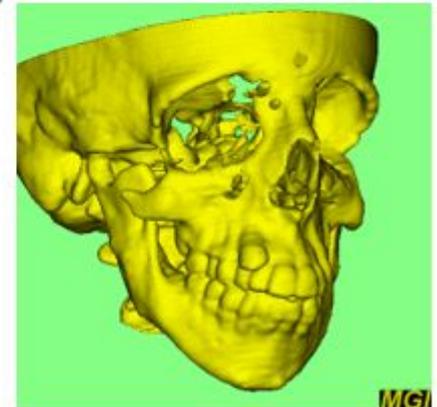
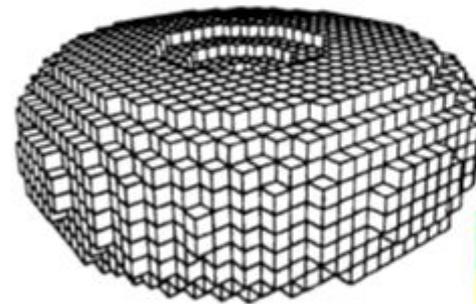
- Facetting

- e.g. STL mesh file

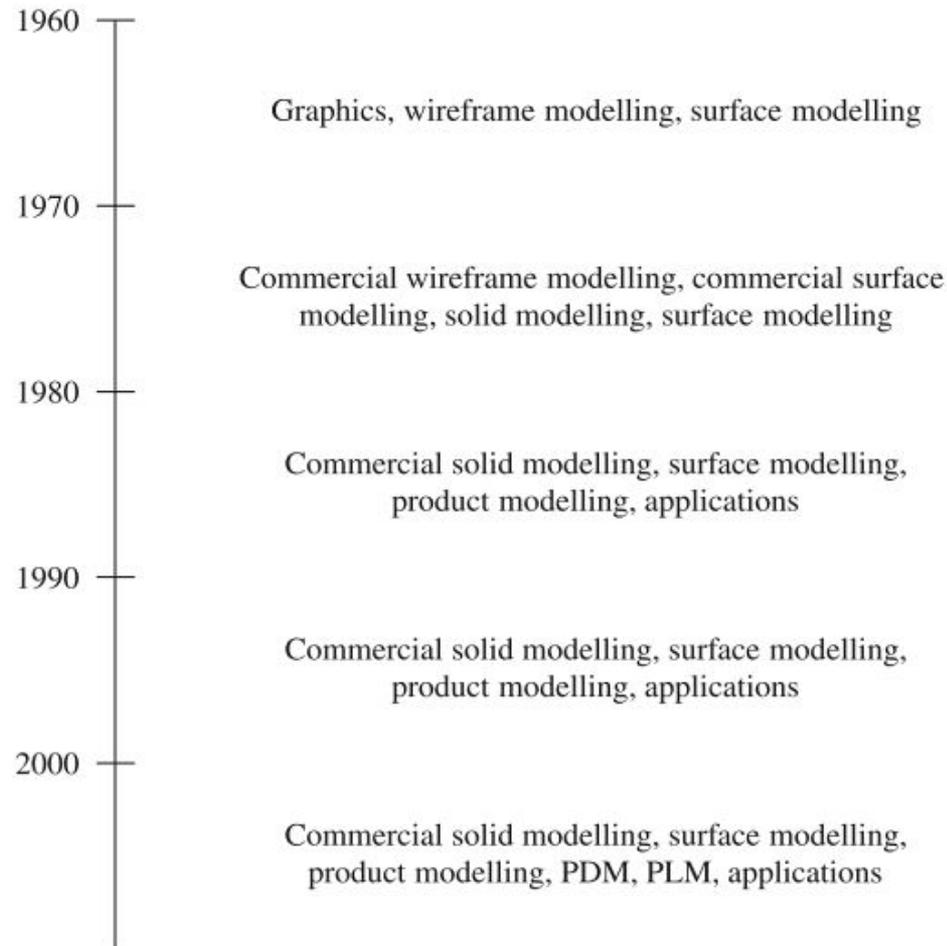


- Voxel

- Applications: medical scans (CAT, NMR, ultrasound)



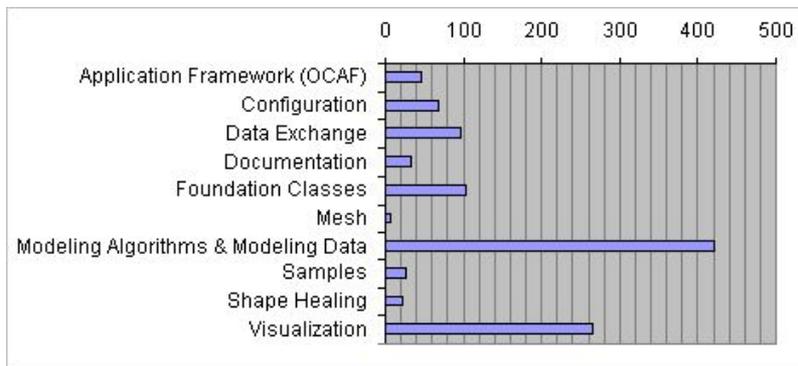
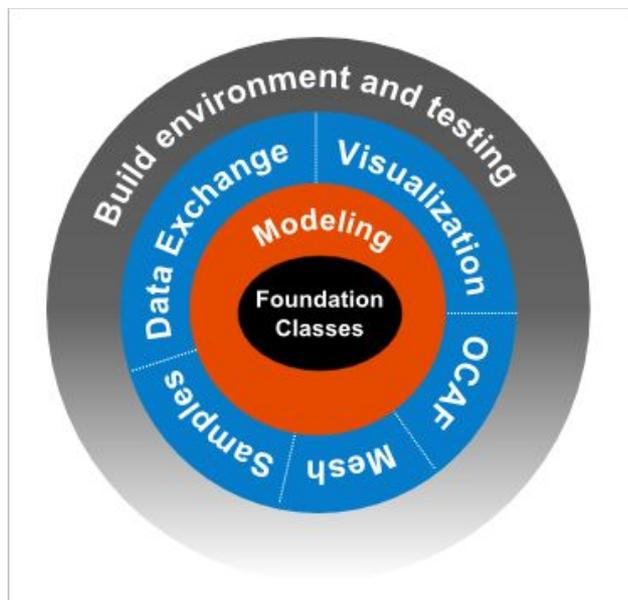
# История CAD



I.Stroud, 2011

# Состав библиотеки

- Мат. Обеспечение (CAGD)
- Моделирование (CAGD + CAD)
- Обмен данными
- Фасетер → Визуализация
- Сервисы



<https://dev.opencascade.org>

# Еще раз: что такое OpenCascade?

«Although there are facilities for displaying graphics in Open CASCADE, **the real function of the library is to do the math.** There are dozens of graphics libraries (if not hundreds), but **there are very few solid modeling libraries,** and Open CASCADE is the only open source solid modeler»  
(взято с официального форума OCCT).

# Мат. обеспечение

- Базовая линейная алгебра
  - Solvers
  - Eigen values & vectors
  - SVD
- Методы локальной и глобальной оптимизации
  - Newton, BFGS, FRPR, Powell
  - PSO
- Интерполяция и аппроксимация
  - By points or sections (discrete data)
  - By continuous function (e.g. from offset surface to polynomial)

$$|f(x) - \sum_{k=0}^n \alpha_k L_k(x)| < \epsilon.$$

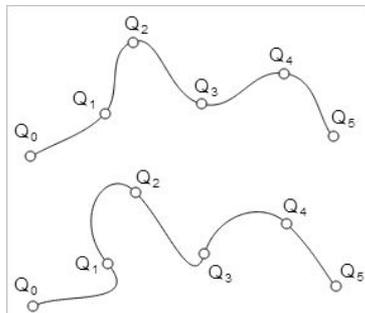
$$L_0(x) = 1,$$

$$L_1(x) = x,$$

$$L_2(x) = \frac{3}{2}x^2 - \frac{1}{2},$$

...

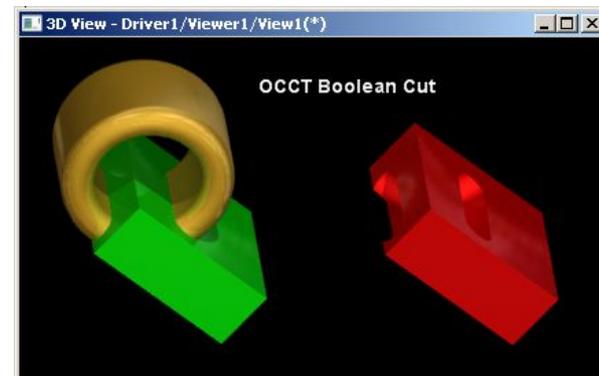
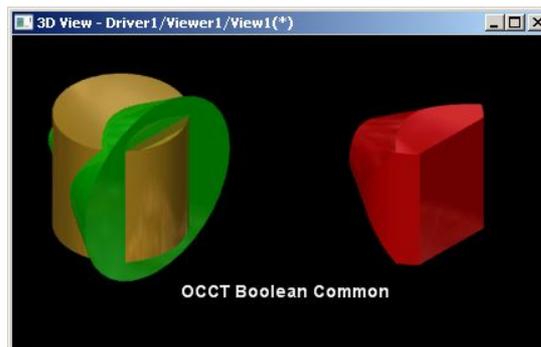
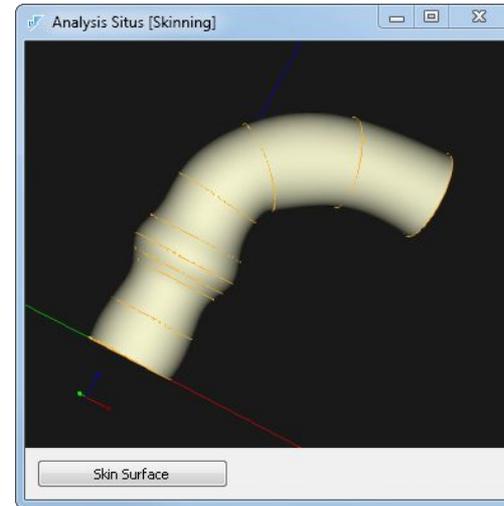
$$L_i(x) = \frac{2i-1}{i}xL_{i-1}(x) - \frac{i-1}{i}L_{i-2}(x).$$



$$E[f] = \sum_{i=1}^N |f(\mathbf{x}_i) - y_i|^2 + \lambda \cdot \int_{\mathbb{R}^n} |D^2 f|^2 dX \rightarrow \min.$$

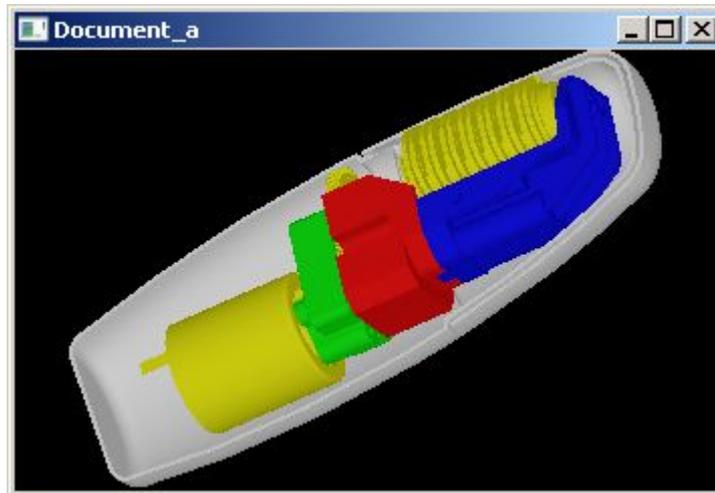
# Моделирование

- Поверхностное
  - Кинематические поверхности
  - Скиннинг
  - ...
- Твердотельное
  - Призма
  - Уклон
  - Тонкостенное тело
  - Булевы операции
  - ...



# Обмен данными

- Форматы:
  - Нейтральные форматы: STEP (ISO 10303), IGES
  - Полигональные форматы: STL, VRML
- Данные:
  - Геометрия
  - Мета-данные ( сборки, цвета, имена, слои, специальные атрибуты)

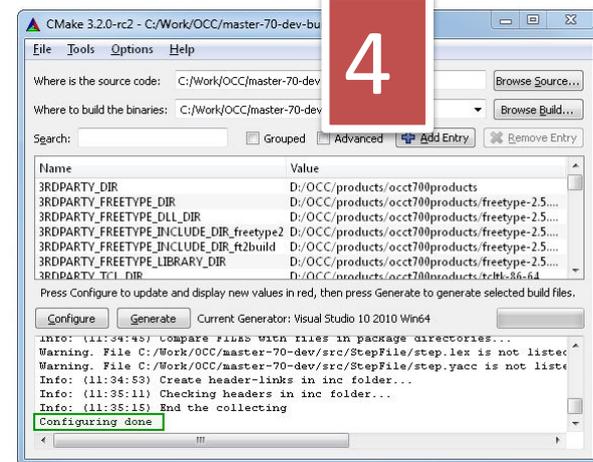
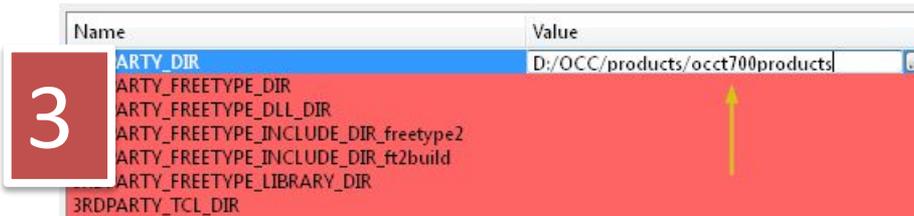
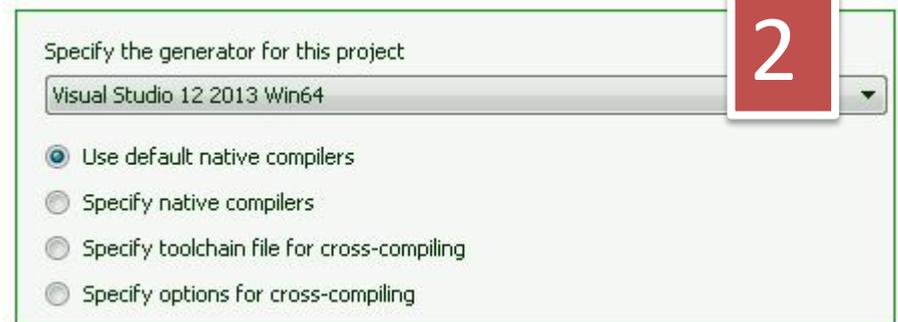
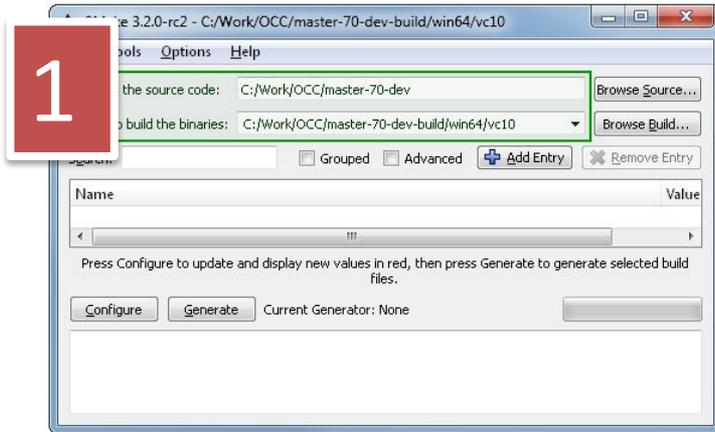


# Организация в FS

- Пакет 1 / Класс 1
- Пакет 1 / Класс 2
- Пакет 2 / Класс 1
- Пакет 2 / Класс 2
- ...

n	Name	Name
...		BRep
Adaptor2d		BRepAdaptor
Adaptor3d		BRepAlgo
AdvApp2Var		BRepAlgoAPI
AdvApprox		BRepApprox
AIS		BRepBlend
APIHeaderSection		BRepBndLib
AppBlend		BRepBuilderAPI
AppCont		BRepCheck
AppDef		BRepClass
AppParCurves		BRepClass3d
Approx		BRepExtrema
ApproxInt		BRepFeat
AppStd		BRepFill
AppStdL		BRepFilletAPI
Aspect		BRepGProp
BinDrivers		BRepIntCurveSurface
BinLDrivers		BRepLib
BinMDataStd		BRepLProp
BinMDataXtd		BRepMAT2d
BinMDF		BRepMesh
BinMDocStd		BRepOffset
BinMFunction		BRepOffsetAPI
BinMNaming		BRepPrim
BinMXCAFDoc		BRepPrimAPI
BinObjMgt		BRepProj
BinObjDrivers		BRepSweep
BinTools		BRepTest
BinXCAFDrivers		BRepToIGES
Bisector		BRepToIGESBRep
BitGte		BRepTools
Blend		BRepTopAdaptor
BlendFunc		BSplCLib
Bnd		BSplSLib
BndLib		BVH
BOPAlgo		CDF
BOPCol		CDM
BOPDS		ChFi2d
BOPTest		ChFi3d
BOPTools		ChFiDS

# Сборка (CMake)



# Литература к уроку 1

1. An Introduction to Solid Modeling. M. Mantyla. 1988.
2. The NURBS Book. L. Piegl, W. Tiller.
3. Solid Modelling and CAD Systems. How to Survive a CAD System. I. Stroud. 2011.
4. Обзор Open CASCADE Technology. С. Сляднев // [isicad.ru](http://isicad.ru)

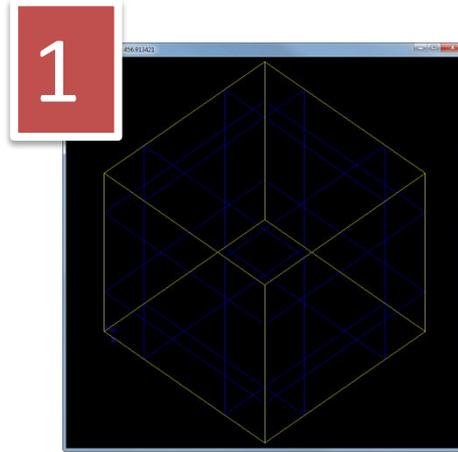
# Урок 2: Hello World

# Точка входа: Draw

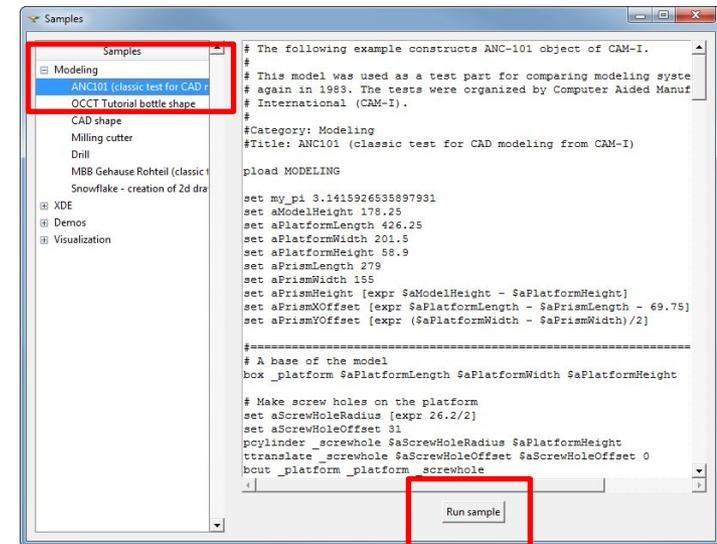
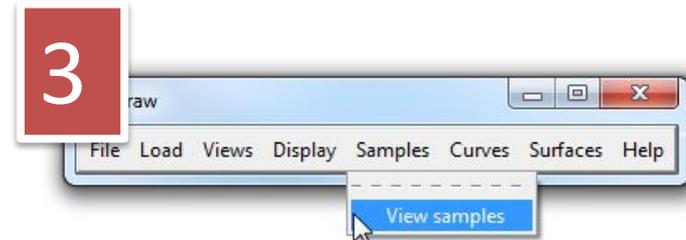
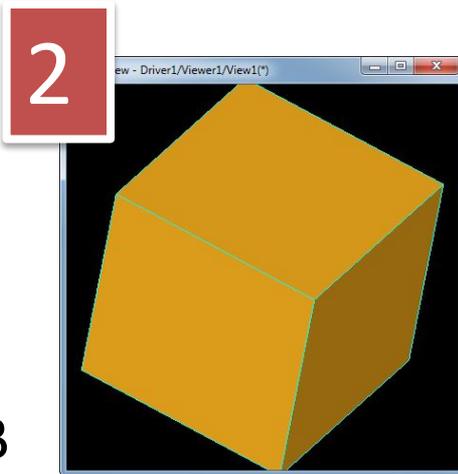
- Интерпретатор Tcl с пользовательскими расширениями
- Доступ ко всей базовой функциональности ядра
- Быстрое прототипирование

# Базовые команды

- > **pload ALL**
- > **box a 1 1 1**
- > **axo; fit**

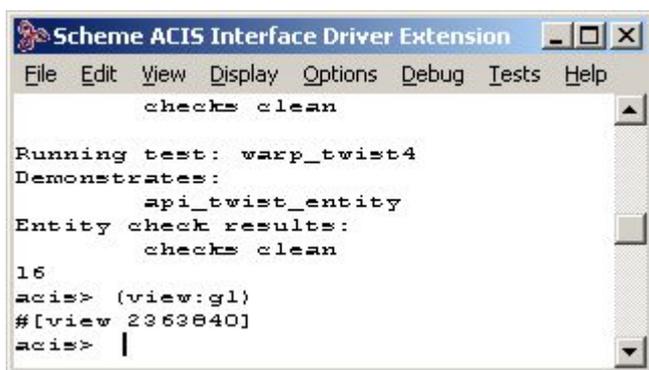


- > **vinit**
- > **vdisplay a**
- >> **f**
- >> **s**
- >> **Ctrl + RMB**



# Аналоги

- ACIS Scheme AIDE

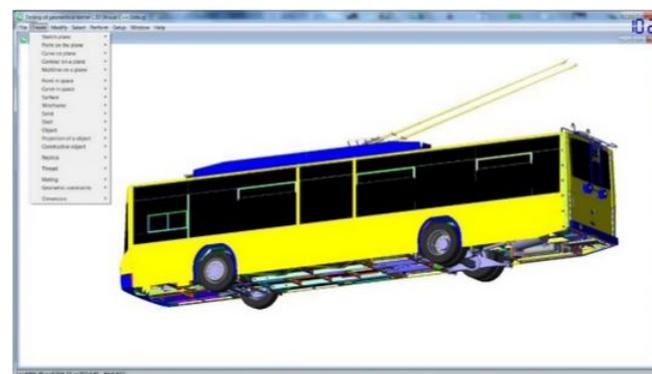


```
Scheme ACIS Interface Driver Extension
File Edit View Display Options Debug Tests Help
checks clean

Running test: warp_twist4
Demonstrates:
  api_twist_entity
Entity check results:
  checks clean

16
acis> (view:gl)
#[view 2969840]
acis> |
```

- C3D Test Application



Речь всегда идет о простейшей «точке входа», а не о полноценном решении.

# Создание простейшего приложения

- Варианты
  - Ваша функция `main()` // Нет визуализации
  - Приложение с интерпретатором Draw // `.exe`
  - Приложение как плагин для Draw // `.dll`
- Скачайте решение с сайта курса:
  - <https://sites.google.com/site/modelingpractice/>

# Снова к теории: Geometry vs Topology

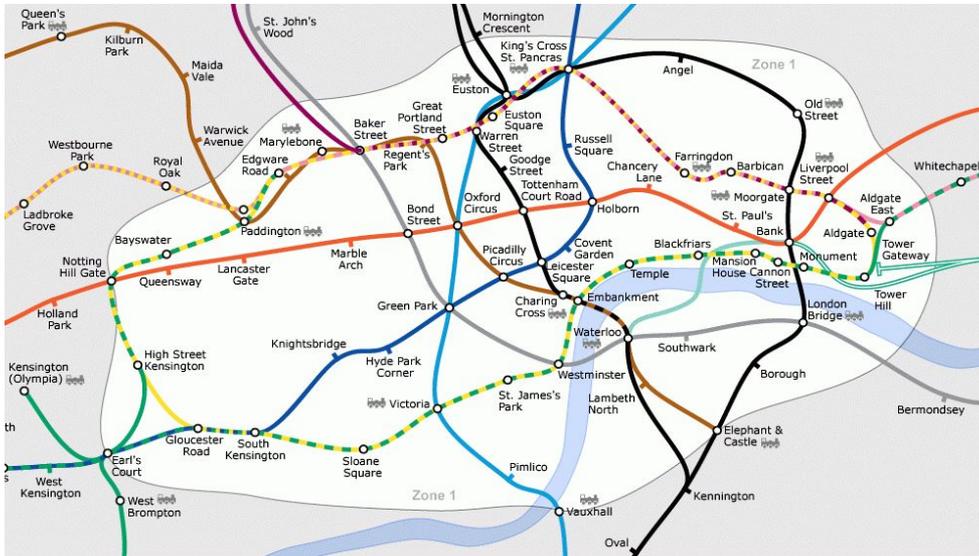
- Без понимания разницы между геометрией и топологией работа со структурами данных OpenCascade – это работа вслепую.
- Первые геометрические ядра не имели этого разделения.
- Сейчас это разделение – часть стандарта (ISO 10303).

# Geometry vs Topology



# Geometry vs Topology

Geometry = real locations



Topology = connections

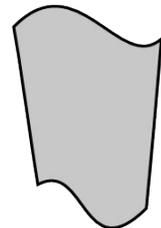


# Geometry vs Topology

## Geometrical Entity

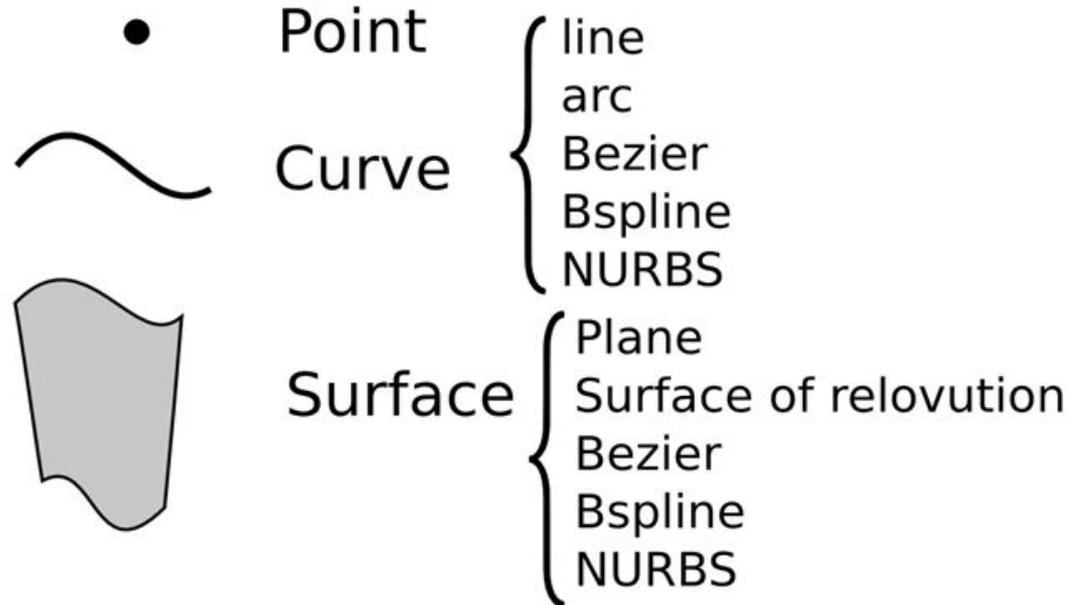
• Point

 Curve

 Surface

# Geometry vs Topology

## Geometrical Entity



# Geometry vs Topology

## Topological Entity

● Vertex

~ Edge

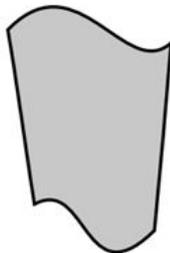
▭ Face

# Geometry vs Topology

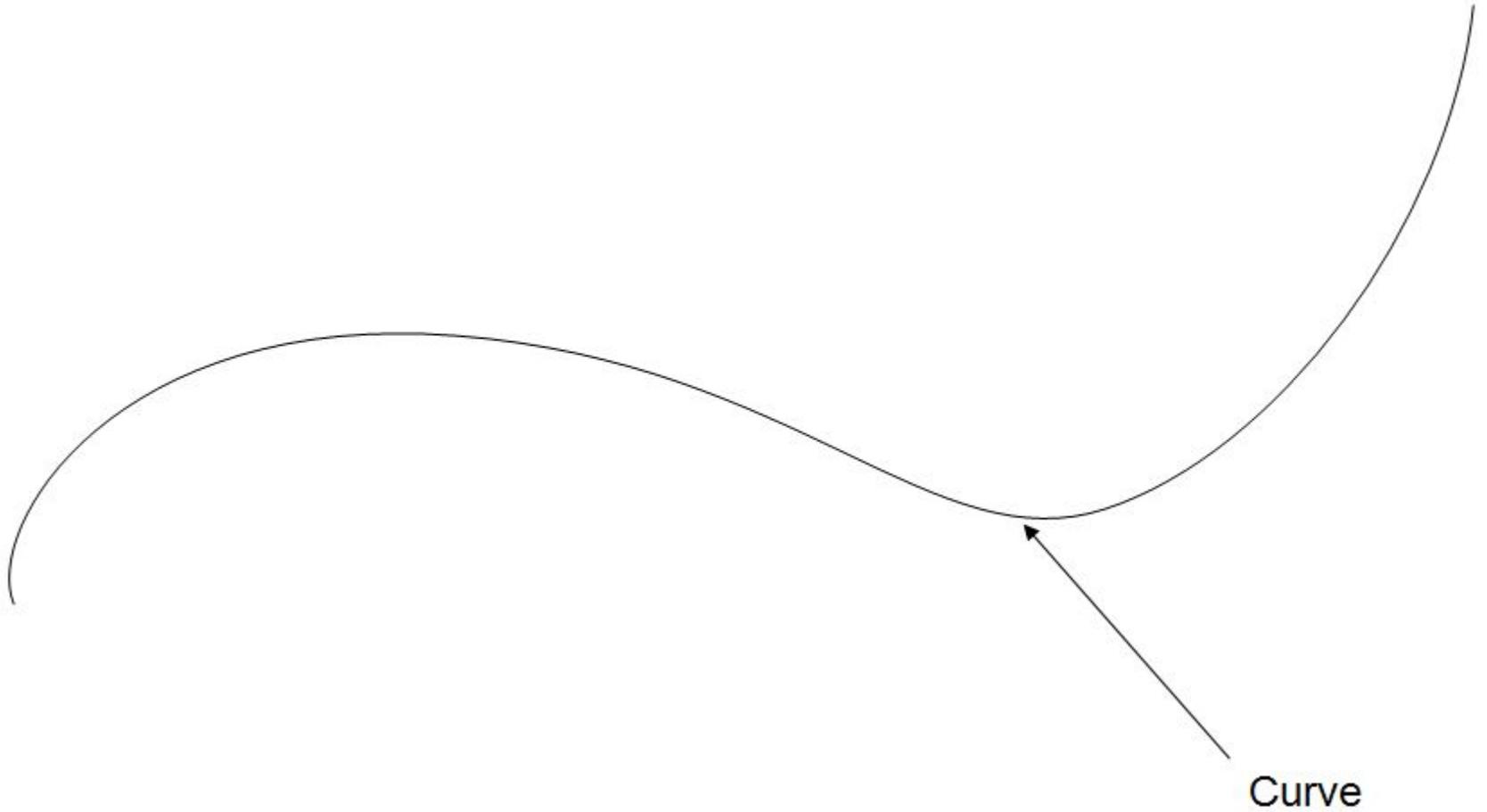
## Topological Entity

• Vertex → Point

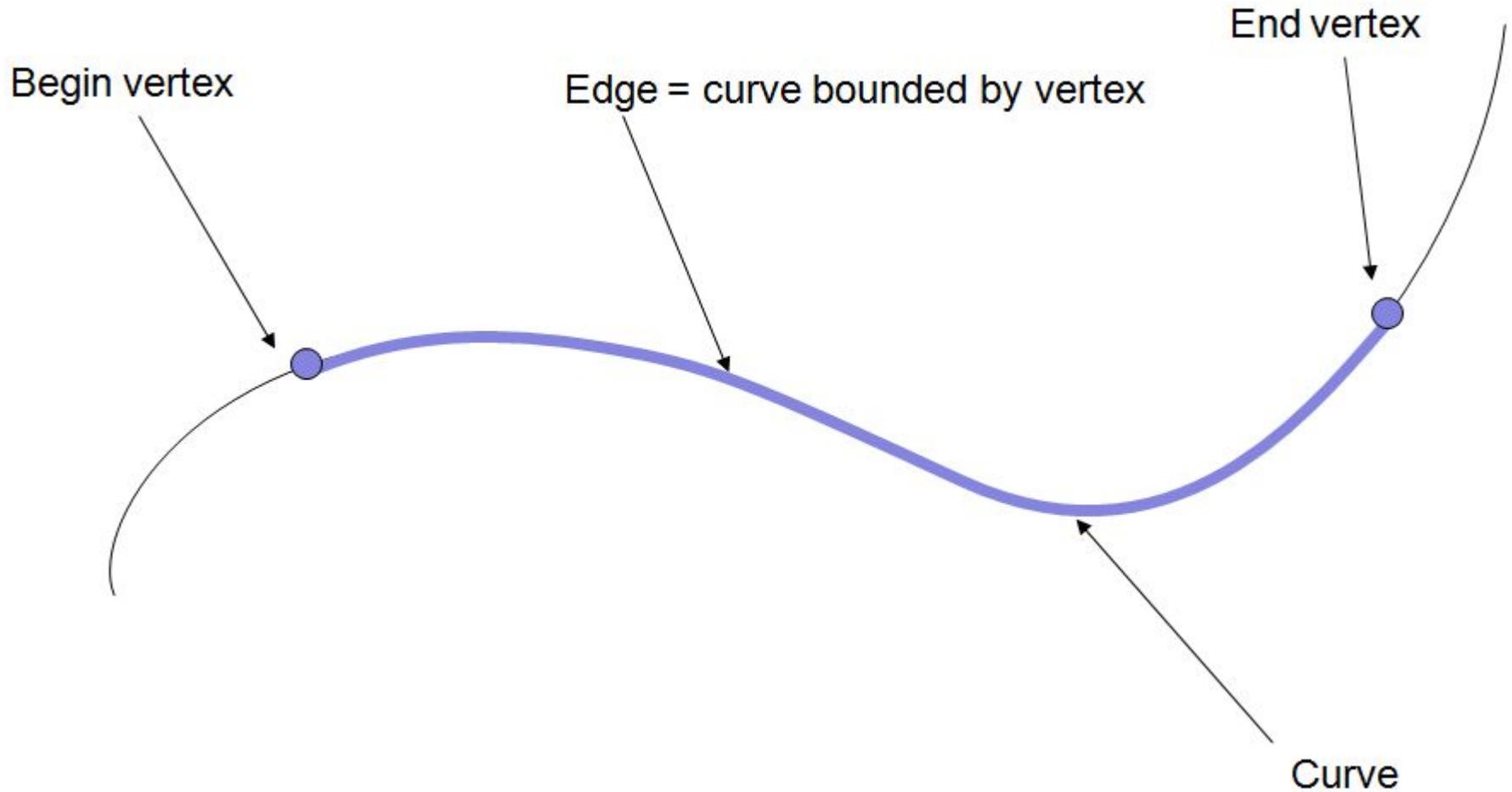
 Edge → Curve, Bound by Vertex

 Face → Surface, Bound by Edge

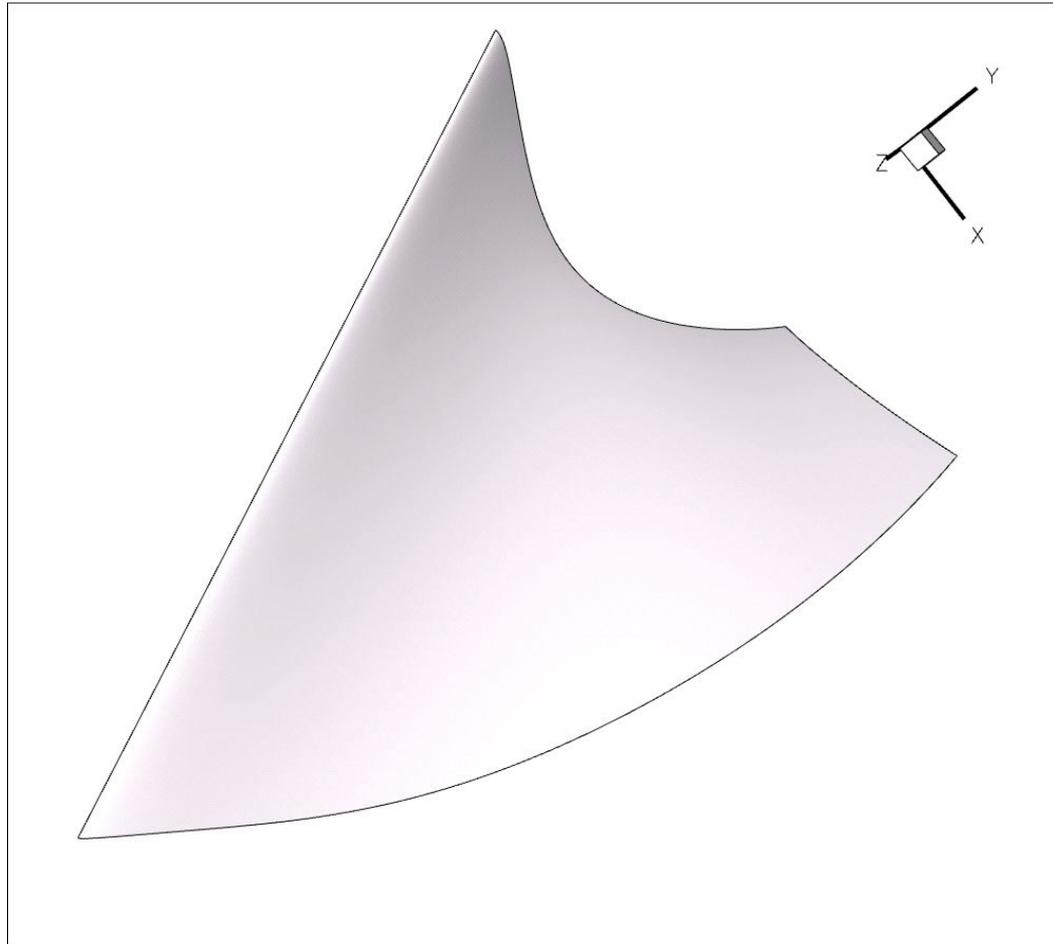
# Geometry vs Topology



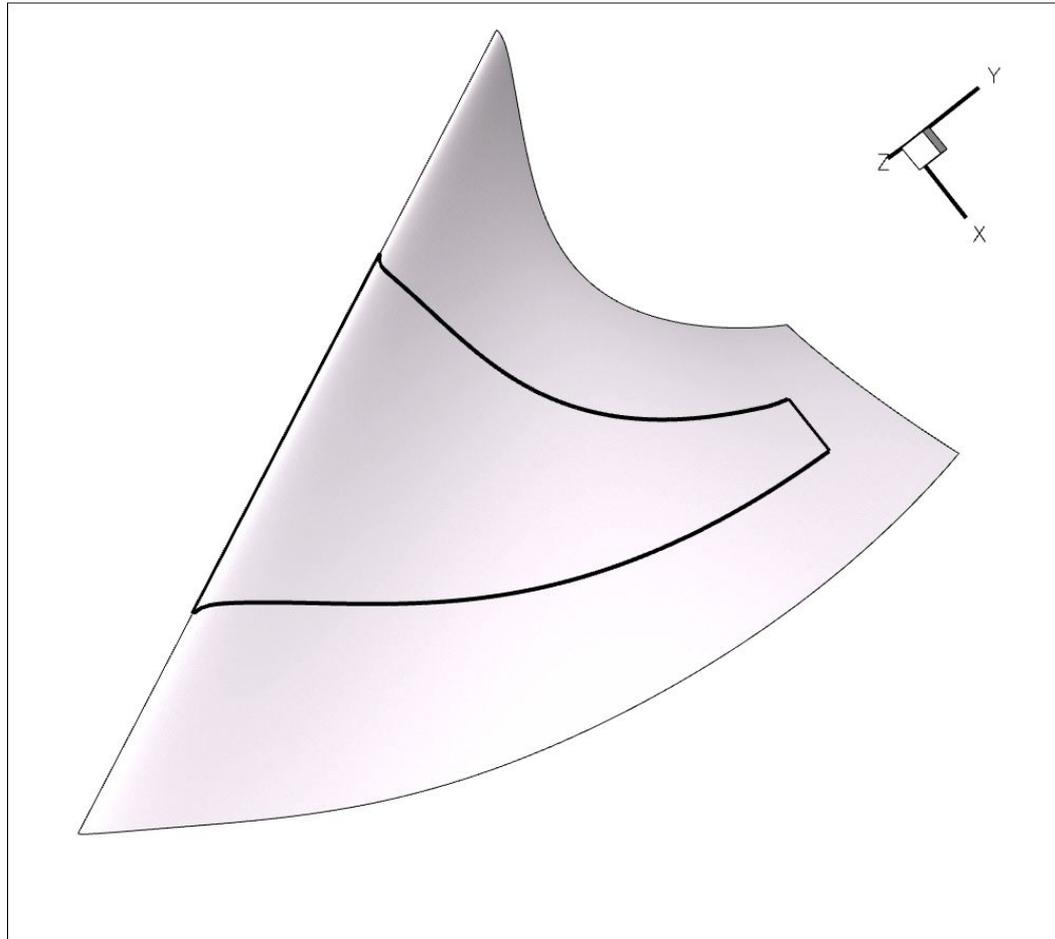
# Geometry vs Topology



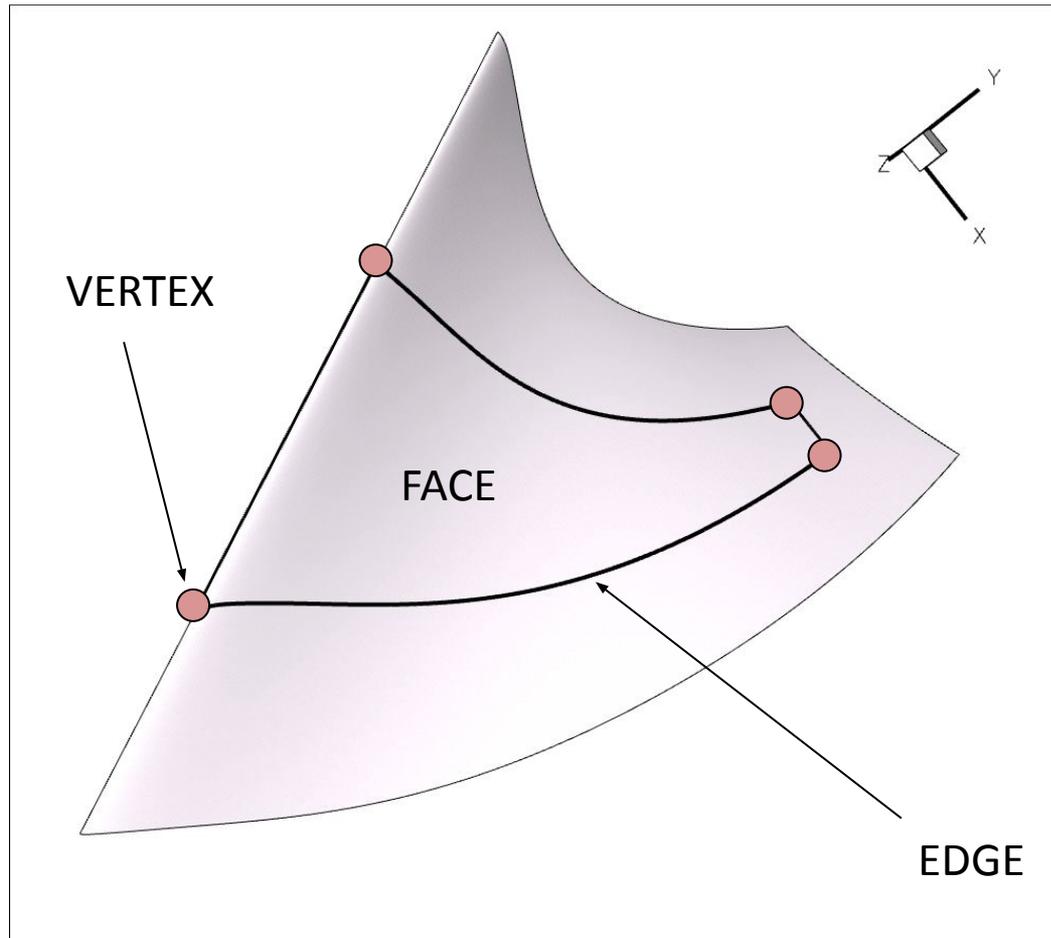
# Geometry vs Topology



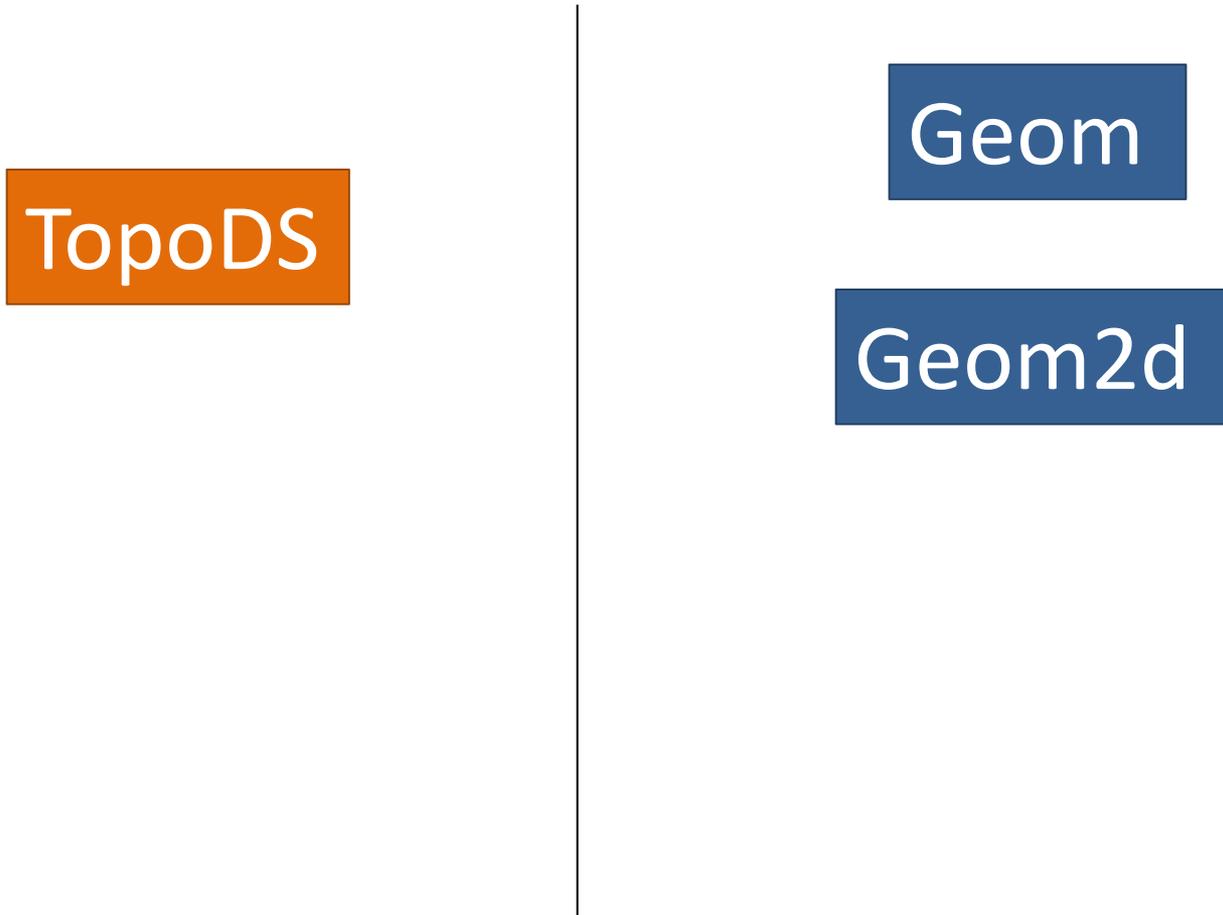
# Geometry vs Topology



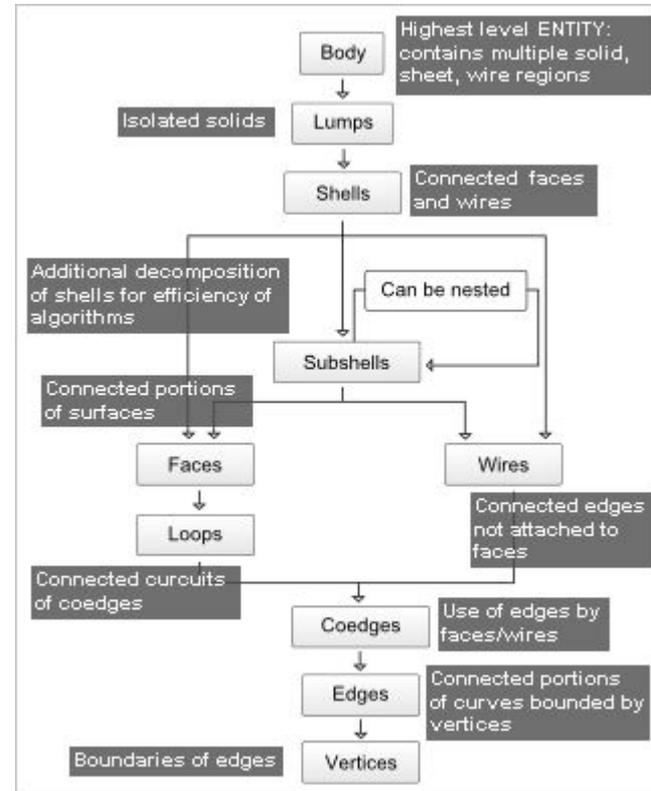
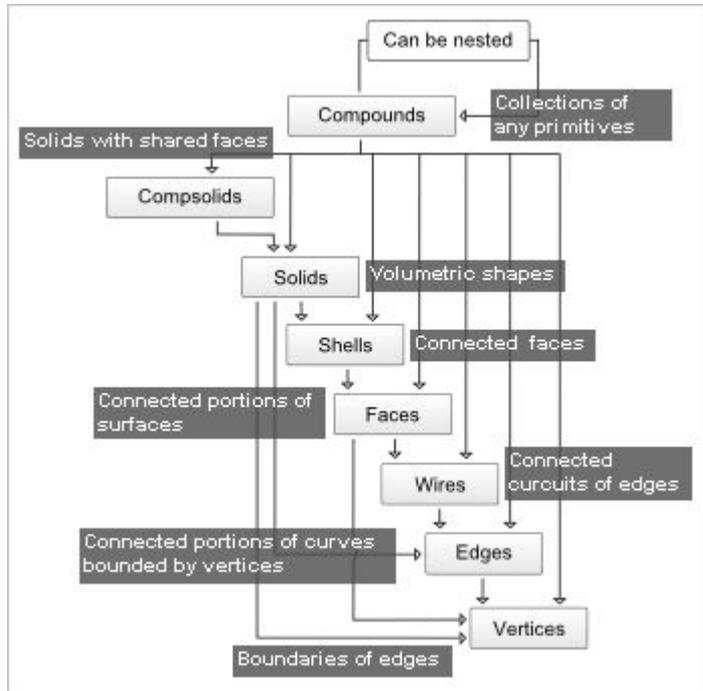
# Geometry vs Topology



# Реализация в OpenCascade

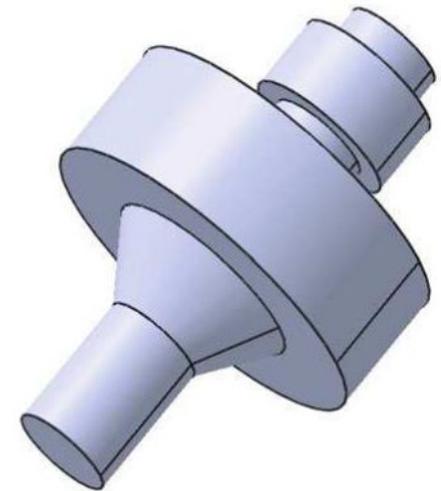
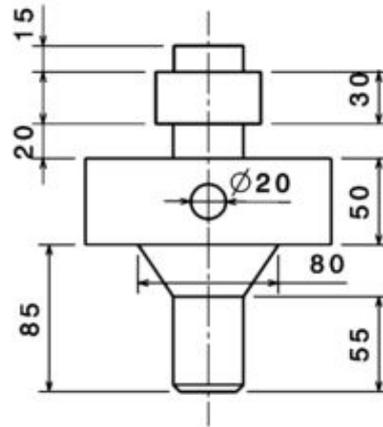
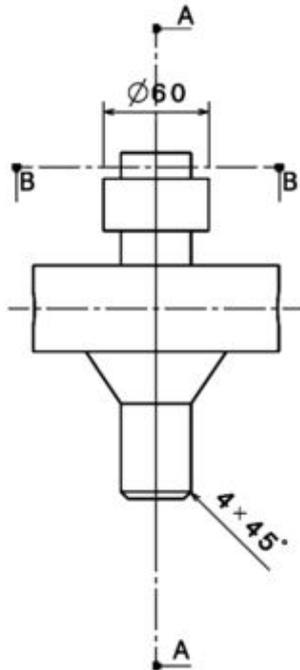
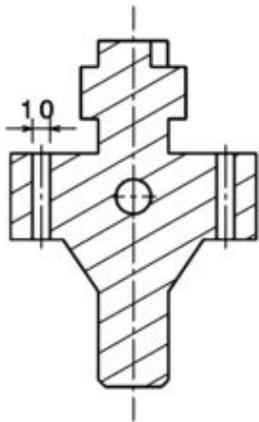
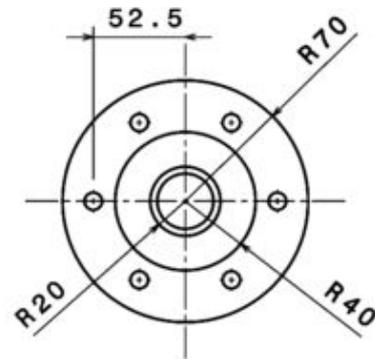
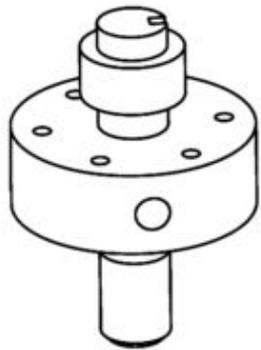


# OpenCascade vs ACIS

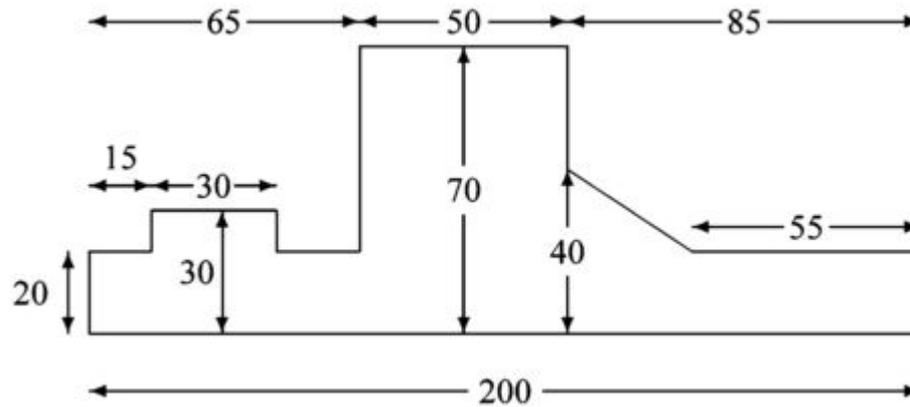


# Урок 3: Твердотельное Моделирование

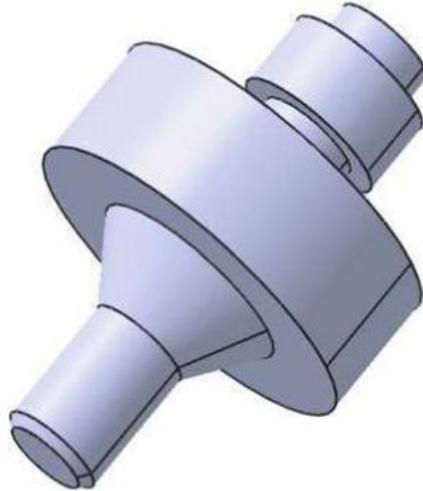
# Задача 1 (I. Stroud, p.17)



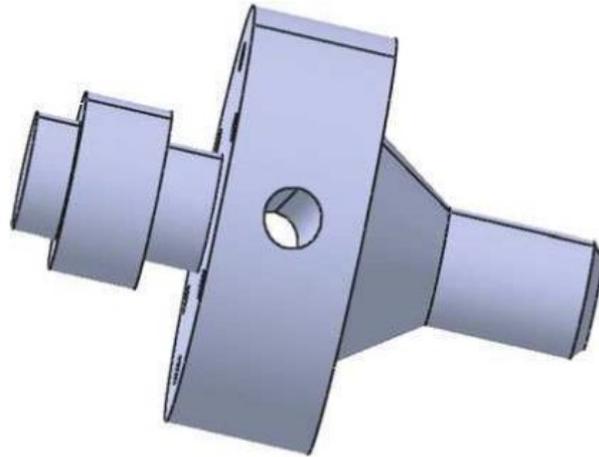
# Задача 1, Шаг 1: эскиз



# Задача 1, Шаг 2: фаски

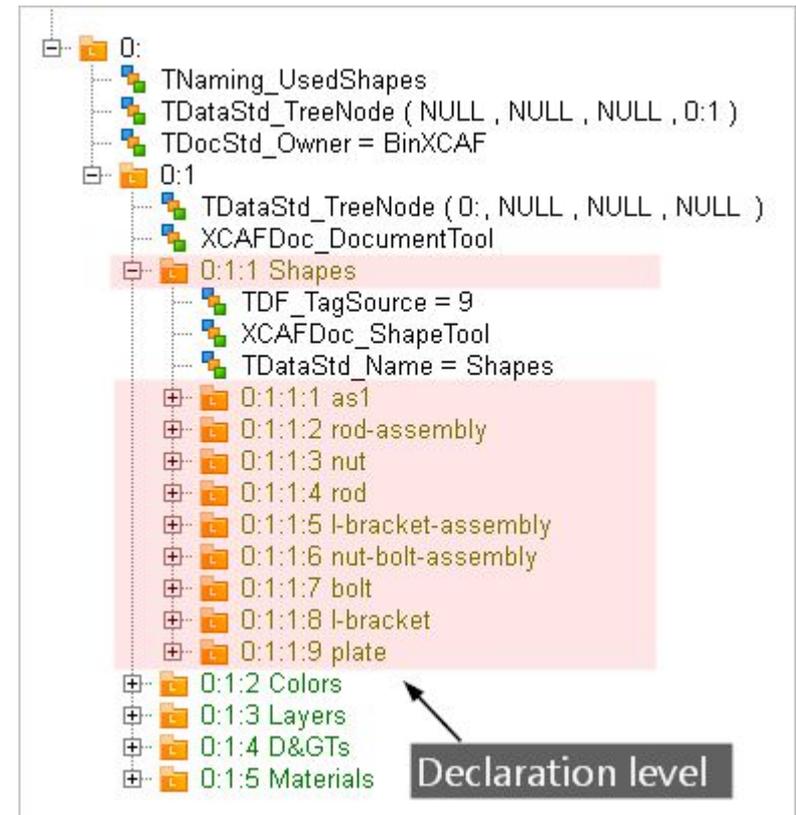
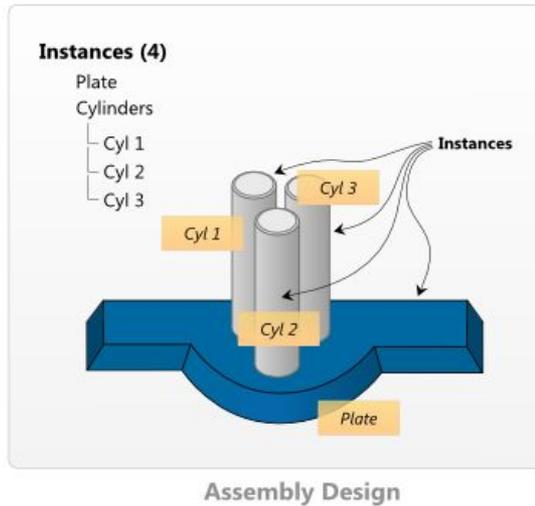
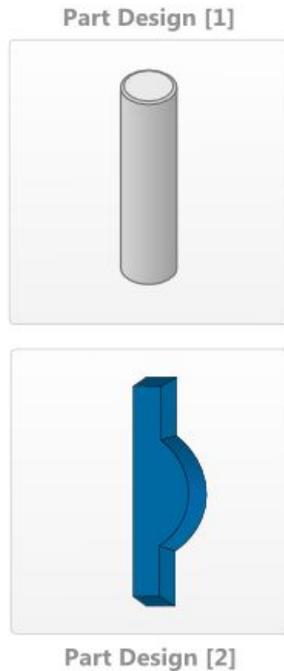


# Задача 1, Шаг 3: отверстие



# Задача 1, Заключительный шаг: обмен данными (1)

- Геометрическая модель
- Метагеометрическая модель: XDE



# Задача 1, Заключительный шаг: обмен данными (2)

- Задача:
  - Создать метагеометрическую модель из твердотельной.
    - Использовать команды Draw.
    - Использовать DF Browser для профилировки.
    - Назначить различные цвета граням детали.
  - Записать метагеометрическую модель в формате STEP.
  - Использовать стороннюю САПР для проверки.

# Резюмируем

- **API**
  - API моделирования – инструментальные классы (НЕ функции).
  - Точки входа: названия пакетов заканчиваются на «API».
  - История построения доступна в API-классах.
  - Диагностика через методы `IsDone()`, коды ошибок.
- Если API недостаточен, можно изучать исходные коды и пользоваться средствами более низкого уровня, вплоть до функций математического обеспечения библиотеки.
- Если есть вопросы:
  - Смотреть официальную документацию.
  - Спрашивать на форуме.

# Литература

1. Solid Modelling and CAD Systems. How to Survive a CAD System. I. Stroud. 2011.

# Урок 4: OpenCascade + VTK

## Приложение Analysis Situs

# Exe & SDK

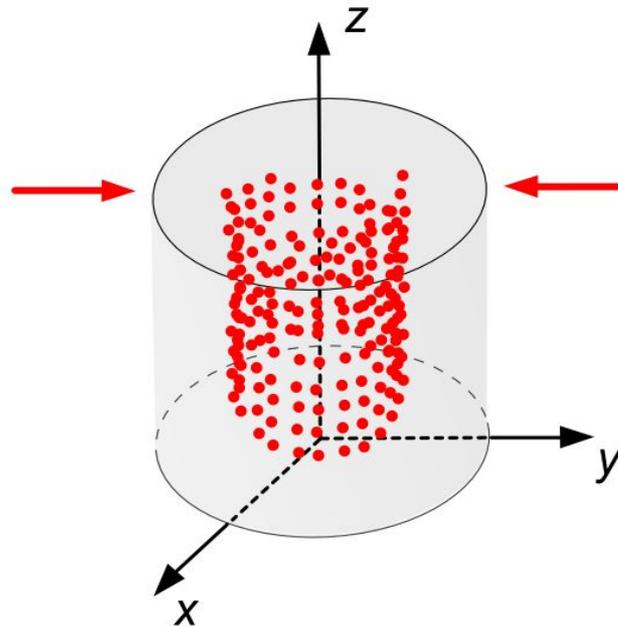
- Скачайте приложение на сайте практикума:
  - <https://sites.google.com/site/modelingpractice/>
    - Analysis Situs (minimal)
    - **Analysis Situs (SDK)**
- Сборка MSVC2013

# Урок 5: Задачи реконструкции

## Приложение OpenCascade к задачам реинжиниринга

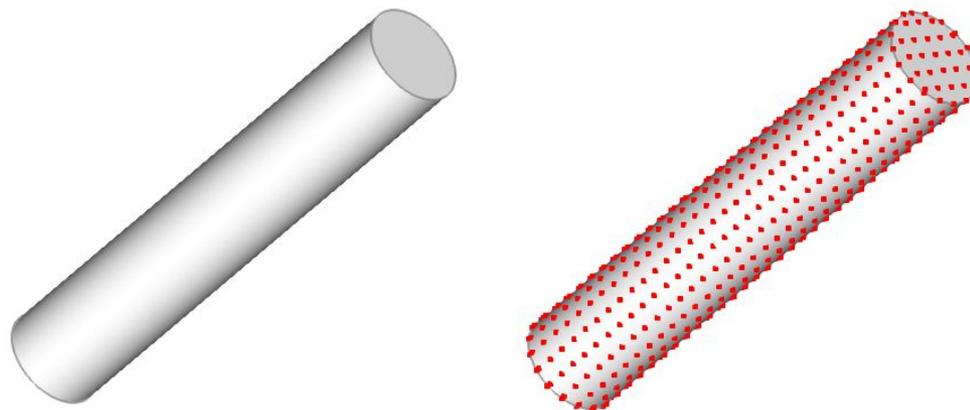
# Задача 2 (подгонка цилиндра с фиксированной осью)

- План работ:
  - Создание тестового облака;
  - Создание грубого приближения вручную;
  - Оптимизация радиуса грубого приближения.



# Задача 2, Шаг 1 (1)

Равномерное сэмплирование поверхности цилиндра.



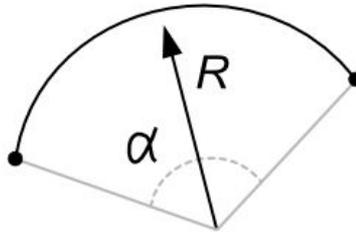
Пользователь работает в пространстве моделирования (длина шага должна задаваться по поверхности).

# Задача 2, Шаг 1 (2)

Трудности:

- Параметризация неравномерна. Как обеспечить равномерность шага

$$l = R * \alpha$$

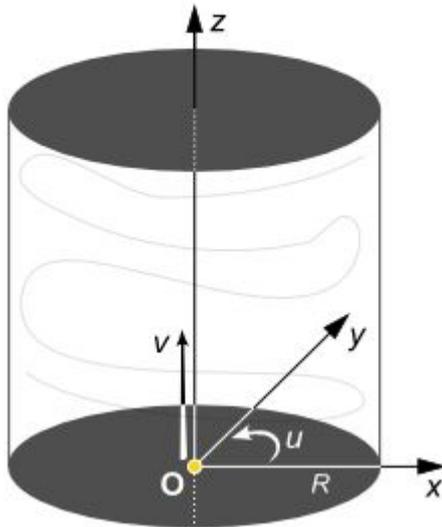


- Сэмплирование обеспечивается путем вычисления точек на несущих поверхностях граней. Что если грань содержит внутренние контуры?

# Задача 2, Шаг 2

В данной задаче предполагается, что облако точек и грубое приближение соосны. В этом случае единственным неизвестным параметром остается радиус цилиндра.

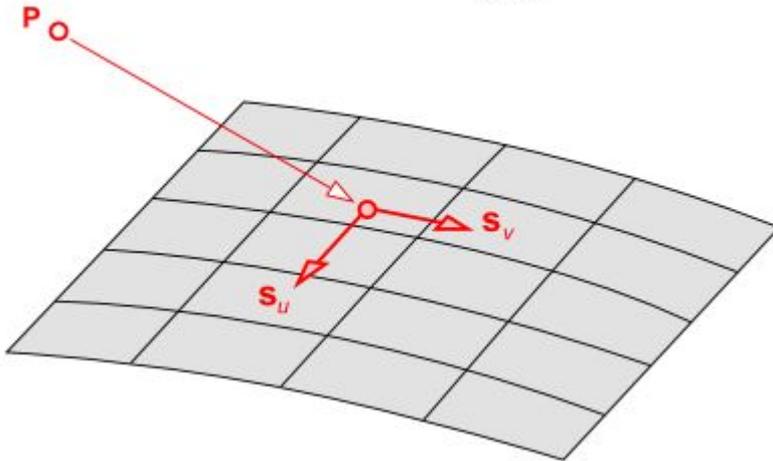
$$\mathbf{P}(u, v) = \mathbf{O} + xR \cos u + yR \sin u + zv.$$



# Задача 2, Шаг 3

Требуется минимизировать среднее расстояние от точек  $R_i$  до цилиндра  $s$ . Если количество точек  $(M + 1)$ , то целевая функция имеет вид:

$$\rho[\mathbf{s}, R] = \frac{1}{M + 1} \sum_{l=0}^M \rho[\mathbf{s}, \mathbf{R}_l].$$



Расстояние от точки до поверхности находится путем *инверсии точки* (*point inversion*).

Какой метод оптимизации выбрать?

Дополнительный урок (\*): Разработка  
алгоритмов геометрического  
моделирования

# Эйлеровы операторы

- Готовых нет
- Могут быть построены на
  - BRepTools\_ReShape (изменение топологии)
  - BRepTools\_Modifier (рехост геометрии)