

软件体系结构大作业

——Open CASCADE系统分析

|  |  |  |  |
| --- | --- | --- | --- |
| 组 号： | Open CASCADE - 1 | | |
| 学 号： | ZY2006109 | ZY2006321 | SY2006316 |
| 姓 名： | 姬轶 | 刘卓程 | 曲卓涵 |

**北京航空航天大学**

**2021 年 6 月 9 日**

目录

[一、 系统简介 1](#_Toc75188387)

[1.1 主要功能介绍 1](#_Toc75188388)

[1.2 代码行数 1](#_Toc75188389)

[1.3 包和类的数量 2](#_Toc75188390)

[二、 需求分析 3](#_Toc75188391)

[2.1 功能需求 3](#_Toc75188392)

[2.1.1 基本数据类型的创建及运算 3](#_Toc75188393)

[2.1.2 基本几何图元的创建 6](#_Toc75188394)

[2.1.3 基本几何操作 7](#_Toc75188395)

[2.1.4 空间变换 7](#_Toc75188396)

[2.1.5 解析几何运算 8](#_Toc75188397)

[2.1.6 解析几何建模 9](#_Toc75188398)

[2.1.7 网格表示及相关运算 10](#_Toc75188399)

[2.1.8 形状修复 11](#_Toc75188400)

[2.1.9 工业几何操作 12](#_Toc75188401)

[2.1.10 可视化 13](#_Toc75188402)

[2.1.11 数据交换 15](#_Toc75188403)

[2.1.12 提供应用框架 16](#_Toc75188404)

[2.2 主要质量属性（非功能需求） 18](#_Toc75188405)

[2.2.1 性能 18](#_Toc75188406)

[2.2.2 可用性和可靠性 19](#_Toc75188407)

[2.2.3 易用性 19](#_Toc75188408)

[2.2.4 可修改性 20](#_Toc75188409)

[2.2.5 可重用性 21](#_Toc75188410)

[2.2.6 可集成性 21](#_Toc75188411)

[2.2.7 可测试性 21](#_Toc75188412)

[2.3 商业质量属性 22](#_Toc75188413)

[2.3.1 开发成本与收益 22](#_Toc75188414)

[2.3.2 开发计划 23](#_Toc75188415)

[三、 软件体系结构设计与分析 24](#_Toc75188416)

[3.1 Open CASCADE体系结构建模 24](#_Toc75188417)

[3.1.1 Open CASCADE模块分解与对应 24](#_Toc75188418)

[3.1.2 Open CASCADE模块功能描述 24](#_Toc75188419)

[3.2 Open CASCADE体系结构分析 36](#_Toc75188420)

[3.2.1 功能需求与软件体系结构的对应关系 36](#_Toc75188421)

[3.2.2 主要质量属性（非功能需求）与软件体系结构的对应关系 40](#_Toc75188422)

[3.2.3 软件体系结构分析 44](#_Toc75188423)

[3.2.4 软件体系结构模型中的异味 45](#_Toc75188424)

[四、 设计特色分析 46](#_Toc75188425)

[4.1 值得借鉴之处 46](#_Toc75188426)

[4.2 可改进之处 47](#_Toc75188427)

[五、 组内分工情况 48](#_Toc75188428)

[参考文献 49](#_Toc75188429)

[附件 50](#_Toc75188430)

1. 系统简介
   1. 主要功能介绍

Open CASCADE（简称OCC）平台是由法国Matra Datavision公司开发的CAD/CAE/CAM软件平台，可以说是世界上最重要的几何造型基础软件平台之一。开源OCC对象库是一个面向对象C++类库，用于快速开发设计领域的专业应用程序。OCC主要用于开发二维和三维几何建模应用程序，包括通用的或专业的计算机辅助设计CAD系统、制造或分析领域的应用程序、仿真应用程序或图形演示工具。OCC通过有机组织的C++库文件提供了六个模块。可视化模块作为OCC的核心部分，是可视化技术的具体体现[1]。

Open CASCADE，是一个开放源码CAD内核，可以定制和扩展（添加新的功能组件，类的进一步继承），面向CAD/CAM，对主流CAD数据格式提供支持（STEP/STL/IGES等，可自行开发转换程序提供特定数据格式的支持），提供高级建模函数（拟合，有理样条曲线，拉伸、旋转、扫出、层叠拉伸、圆角、倒角、薄壳、修剪、偏移等），参数化模型，提供几何模型的特征提取，对Visual C++/MFC 有很好的支持。它提供了点、线、面、体和复杂形体的显示和交互操作，经过深度开发后可实现纹理、光照、图元填充、渲染等图形操作和放大、缩小、旋转、漫游、模拟飞行、模拟穿越等动态操作。

Open CASCADE提供二维和三维几何体的生成、显示和分析。主要功能有：

1. 创建锥、柱、环等基本几何体；
2. 对几何体进行布尔操作（相加，相减，相交运算）；
3. 倒角，斜切，镂空，偏移，扫视；
4. 几何空间关系计算（法线，点积，叉积，投影，拟合等）；
5. 几何体分析（质心，体积，曲率等）；
6. 空间变换（平移，缩放，旋转）。
   1. 代码行数

该项目共包含6470个文件，共1277161行代码，主要使用了C++语言和C语言，实现了绝大部分的系统功能。除此之外，项目也使用了toolkit语言来生成图形用户界面，以及HTML语言对项目进行描述，用于生成说明文档和访问在线平台[2]。

该项目代码行数如下图一所示。

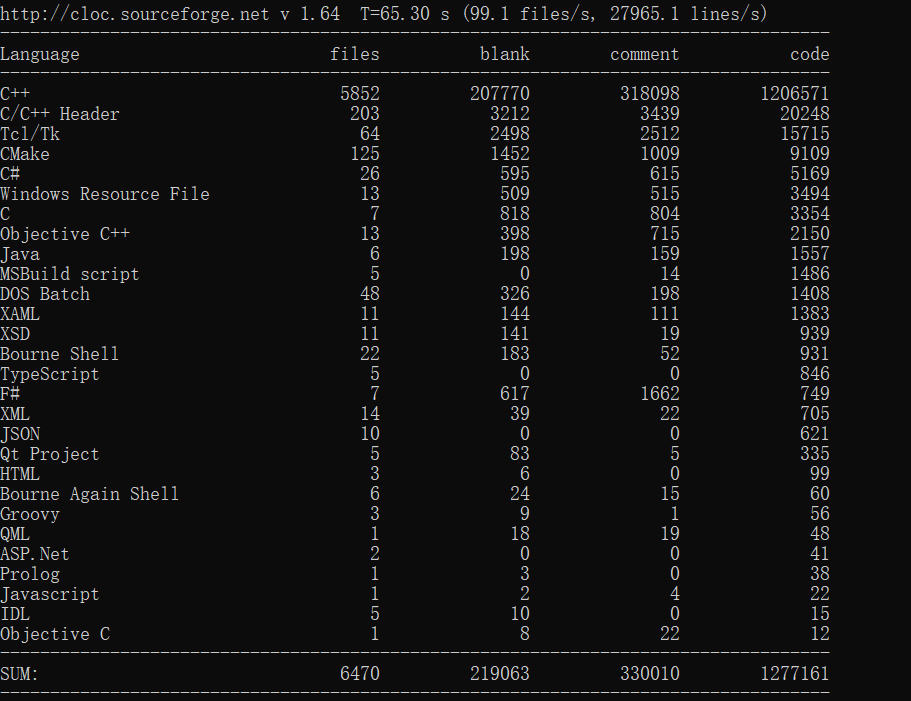


图1-1 Open CASCADE代码行数

* 1. 包和类的数量

项目共有440个包和412个类。

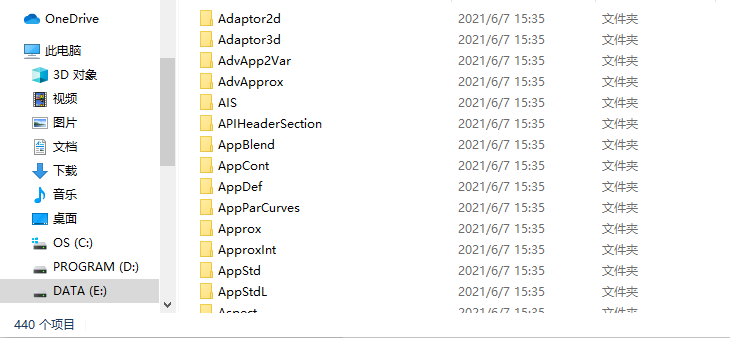


图1-2 Open CASCADE包数

1. 需求分析

OpenCaseCade作为一个开源的几何造型基础软件平台，主要用于开发二维和三维几何建模应用程序。本章将针对Open CASCADE的功能需求、主要质量属性（非功能需求）以及商业质量属性进行分析[3]。

* + 1. 功能需求

Open CASCADE旨在实现真正的模块化和可扩展性，为基本的几何建模提供数据结构、算法、可视化、数据交换等功能。其基本用例图如图2-1所示，图片中仅显示一级用例，具体用例将在2.1.1至2.1.12节中描述。



图2-1 Open CASCADE用例图

### 2.1.1 基本数据类型的创建及运算

基本数据类型的创建及运算功能描述如表2-1所示。

表2-1 基本数据类型的创建及运算功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的几何建模提供基本的数据类型以及运算。其中，基本的数据类型包括布尔型、字符型、整型、集合、向量以及矩阵；基本的运算包括线性方程求解算法、多变量函数极值算法、非线性方程求根算法、矩阵特征值和特征向量算法。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 运算符合数学定理。 |
| 输入 | | 数据类型或运算所需参数。 |
| 过程 | 主过程描述 | 1. 用户准备好所需的初始化值以及参数； 2. 用户通过C++类基本语法调用Open CASCADE； 3. Open CASCADE创建相应的数据对象/进行数据运算； 4. Open CASCADE返回相应对象/计算结果。 |
| 分支过程描述 | 无。 |
| 异常过程描述 | 使用Open CASCADE自定义数据异常对用户进行提示。 |
| 输出 | | 数据对象/计算结果 |
| 后置条件 | | 无。 |
| 业务规则 | | 无。 |

基本数据类型的创建及运算用例图如图2-2所示。



图2-2 基本数据类型的创建及运算用例图

基本数据类型的创建及运算活动图如图2-3所示。



图2-3 基本数据类型的创建及运算活动图

### 2.1.2 基本几何图元的创建

基本几何图元的创建功能需求描述如表2-2所示。

表2-2 基本几何图元的创建功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的几何建模提供基本的图元数据类型，包括点、线、圆和圆锥、平面和基本曲面。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 几何场景已初始化。 |
| 输入 | | 基本几何图元的参数。 |
| 过程 | 主过程描述 | 1. 用户提供基本几何图元的参数； 2. Open CASCADE为相应的几何图元创建对象； |
| 分支过程描述 | 无 |
| 异常过程描述 | 使用Open CASCADE自定义数据异常对用户进行提示。 |
| 输出 | | 几何图元对象 |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

基本几何图元的创建用例图如图2-4所示。



图2-4 基本几何图元的创建用例图

基本几何图元的创建活动图形如图2-3，此处不再赘述。

### 2.1.3 基本几何操作

基本几何操作功能需求描述如表2-3所示。

表2-3 基本几何操作功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的几何建模提供基本的操作，包括相加、相减、相交运算。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 几何场景已初始化。 |
| 输入 | | 操作所需几何图形。 |
| 过程 | 主过程描述 | 1. 用户调用相关的图形操作方法； 2. 用户提供图形运算需要的参数； 3. Open CASCADE针对图形进行对应的计算； 4. Open CASCADE返回图形运算结果。 |
| 分支过程描述 | 无。 |
| 异常过程描述 | Open CASCADE使用内置的图形运算异常对用户进行提示。 |
| 输出 | | 图形运算结果。 |
| 后置条件 | | 无。 |
| 业务规则 | | 无。 |

基本几何操作用例图如图2-5所示。



图2-5 基本几何操作用例图

基本几何操作活动图形如图2-3，此处不再赘述。

### 2.1.4 空间变换

空间变换功能需求描述如表2-4所示。

表2-4 空间变换功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的几何建模提供基本的空间变换操作，包括平移、旋转、对称变换、缩放变换、组合变换。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 几何场景已初始化。 |
| 输入 | | 操作所需的参数，如平移距离、旋转角度等。 |
| 过程 | 主过程描述 | 1. 用户调用相关的图形变换方法； 2. 用户提供图形变换需要的参数； 3. Open CASCADE针对场景进行对应的变换。 |
| 分支过程描述 | 无。 |
| 异常过程描述 | 无。 |
| 输出 | | 无。 |
| 后置条件 | | 无。 |
| 业务规则 | | 无。 |

空间变换用例图如图2-6所示。



图2-6 空间变换用例图

空间变换活动图形如图2-3，此处不再赘述。

### 2.1.5 解析几何运算

解析几何运算功能需求描述如表2-5所示。

表2-5 解析几何运算功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的图形数据提供解析几何运算，包括插值、B样条曲线/面转换、2D/3D曲线上点的坐标计算、图形解析极值计算、质心计算、体积计算、曲率计算。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 1. 几何场景已初始化。 2. 运算符合解析几何相关定理； 3. 场景中存在对应图形。 |
| 输入 | | 解析几何运算所需参数，如插值端点坐标等。 |
| 过程 | 主过程描述 | 1. 用户调用相关的解析几何运算方法； 2. 用户提供解析几何运算参数； 3. Open CASCADE进行解析几何运算； 4. Open CASCADE返回运算结果。 |
| 分支过程描述 | 无 |
| 异常过程描述 | 使用Open CASCADE自定义数据运算异常对用户进行提示。 |
| 输出 | | 运算结果 |
| 后置条件 | | 根据具体运算判定是否更新几何数据。例如，插值方法执行后，几何数据就需要更新。 |
| 业务规则 | | 无 |

解析几何计算用例图如图2-7所示。



图2-7 解析几何计算用例图

解析几何计算活动图形如图2-3，此处不再赘述。

### 2.1.6 解析几何建模

解析几何建模功能需求描述如表2-6所示。

表2-6 解析几何建模功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的几何建模提供基本的解析几何建模算法，包括投影图形的建模、约束线条构造圆/曲线/曲面的建模、插值构建曲线/曲面的建模。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 1. 几何场景已初始化； 2. 约束线条/插值点已存在。 |
| 输入 | | 建模所需参数。 |
| 过程 | 主过程描述 | 1. 用户提供解析建模所需的参数； 2. Open CASCADE计算出模型的具体参数值； 3. Open CASCADE根据参数完成建模； 4. Open CASCADE返回建立好的模型。 |
| 分支过程描述 | 无 |
| 异常过程描述 | 使用Open CASCADE自定义数据异常对用户进行提示。 |
| 输出 | | 构建出的模型 |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

解析几何建模用例图如图2-8所示。



图2-8 解析几何建模用例图

解析几何建模活动图形如图2-3，此处不再赘述。

### 2.1.7 网格表示及相关运算

网格表示及相关运算功能需求描述如表2-7所示。

表2-7 网格表示及相关运算功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | 网格作为表示3D模型的常用数据结构，其易用性无疑是极高的。Open CASCADE将提供网格数据结构用以表征空间中的3D模型，同时将提供网格相关的基本算法，包括网格划分、网格转换。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 1. 几何场景已初始化； 2. 场景中存在3D模型。 |
| 输入 | | 几何图形数据 |
| 过程 | 主过程描述 | 1. 用户调用网格创建方法或网格算法； 2. 用户提供网格创建/算法所需参数； 3. Open CASCADE完成网格的创建或运算； 4. Open CASCADE返回/存储运算结果。 |
| 分支过程描述 | 无 |
| 异常过程描述 | 使用Open CASCADE自定义数据异常对用户进行提示。 |
| 输出 | | 网格对象/计算结果 |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

网格表示及相关运算用例图如图2-9所示。



图2-9 网格表示及相关运算用例图

网格表示及相关计算活动图形如图2-3，此处不再赘述。

### 2.1.8 形状修复

形状修复功能需求描述如表2-8所示。

表2-8 形状修复功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对工业设计领域基本的几何图形错误进行修复，包括接缝边缘面、线路方向、自相交线等相关的错误。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 1. 几何场景已初始化； 2. 场景中存在对应的几何图形； 3. 场景中的几何图形是可被修复的。 |
| 输入 | | 运算所需参数。 |
| 过程 | 主过程描述 | 1. 用户调用相关的形状修复算法； 2. 用户提供形状修复所需的参数； 3. Open CASCADE进行形状修复； 4. Open CASCADE返回修复后的几何图形。 |
| 分支过程描述 | 无 |
| 异常过程描述 | 使用Open CASCADE自定义图形修复异常对用户进行提示。 |
| 输出 | | 修复后的几何图形 |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

形状修复用例图如图2-10所示。



图2-10 形状修复用例图

形状修复活动图形如图2-3，此处不再赘述。

### 2.1.9 工业几何操作

工业几何操作功能需求描述如表2-9所示。

表2-9 工业几何操作功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的几何建模提供常用的工业变换，包括倒角，斜切，镂空，偏移，扫视。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 1. 几何场景已初始化； 2. 几何图形满足变换条件。 |
| 输入 | | 几何操作所需参数。 |
| 过程 | 主过程描述 | 1. 用户调用相应的工业几何操作方法； 2. 用户提供几何操作所需参数； 3. Open CASCADE针对图形进行相应操作； 4. Open CASCADE返回操作结果。 |
| 分支过程描述 | 无 |
| 异常过程描述 | 使用Open CASCADE自定义操作异常对用户进行提示。 |
| 输出 | | 图形变换结果。 |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

工业几何操作用例图如图2-11所示。



图2-11 工业几何操作用例图

工业几何操作活动图形如图2-3，此处不再赘述。

### 2.1.10 可视化

可视化功能需求描述如表2-10所示。

表2-10 可视化功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE将针对二维与三维的几何模型提供可视化的界面。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 1. 几何场景已初始化； 2. 软件运行在具备显示设备的计算机上。 |
| 输入 | | 几何场景。 |
| 过程 | 主过程描述 | 1. Open CASCADE读取存储在计算机中的几何场景数据。 2. Open CASCADE计算几何场景在二维显示器中的投影。 3. Open CASCADE将几何场景数据发送到显示设备。 |
| 分支过程描述 | 无。 |
| 异常过程描述 | 无。 |
| 输出 | | 二维几何可视化图像。 |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

可视化用例图如图2-12所示。



图2-12 可视化用例图

可视化活动图如图2-13所示。



图2-13 可视化活动图

### 2.1.11 数据交换

数据交换功能需求描述如表2-11所示。

表2-11 数据交换功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE作为几何软件平台中的一种，它将支持绝大多数常见的几何数据定义格式的读写，包括IGES、STEP。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 无 |
| 输入 | | IGES/STEP文件（读操作）  几何场景数据（写操作） |
| 过程 | 主过程描述 | 1. 用户准备好相应的IGES/STEP文件（几何场景数据）； 2. Open CASCADE读取文件/将数据写入到IGES/STEP文件。 |
| 分支过程描述 | 无 |
| 异常过程描述 | 使用Open CASCADE自定义数据异常对用户进行提示。 |
| 输出 | | IGES/STEP文件（写操作）  几何场景数据（读操作） |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

数据交换用例图如图2-14所示。



图2-14 数据交换用例图

数据交换活动图形如图2-3，此处不再赘述。

### 2.1.12 提供应用框架

提供应用框架功能需求描述如表2-12所示。

表2-12 提供应用框架功能需求描述

|  |  |  |
| --- | --- | --- |
| 说明 | | Open CASCADE作为一个基础的几何软件平台，为便于进一步开发，它将提供基本的应用框架，包括撤消重做、文档视图管理、文件存储、数据组织、对话框。 |
| 角色 | | 使用Open CASCADE进行开发的用户。 |
| 前置条件 | | 无。 |
| 输入 | | 视开发需求而定。 |
| 过程 | 主过程描述 | 1. 用户调用Open CASCADE提供的应用框架服务； 2. Open CASCADE提供相应的操作。 |
| 分支过程描述 | 视开发需求而定。 |
| 异常过程描述 | 使用Open CASCADE自定义异常对用户进行提示。 |
| 输出 | | 视开发需求而定。 |
| 后置条件 | | 无 |
| 业务规则 | | 无 |

提供应用框架用例图如图2-15所示。



图2-15 提供应用框架用例图

提供应用框架活动图如图2-16所示。



图2-16 提供应用框架活动图

* + 1. 主要质量属性（非功能需求）

Open CASCADE作为一个可定制、可扩展的几何造型基础软件平台，它在设计时需要关心的质量属性包括性能、可用性、可靠性、易用性、可修改性、可重用性、可集成性、可测试性。本节将针对Open CASCADE的主要质量属性即非功能需求进行具体描述。

### 2.2.1 性能

Open CASCADE性能质量属性分析如表2-13所示。

表2-13 Open CASCADE性能质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | 几何图形、基本数据类型 |
| 刺激 | 基本运算、几何图形运算、几何图形操作 |
| 制品 | 数据对象、运算结果 |
| 环境 | 在Open CASCADE运行过程中，正常运行或者高负载运行。 |
| 响应 | Open CASCADE使用硬件加速   * 使用多线程进行运算 * 调用Intel MKL库进行快速运算 |
| 响应度量 | * Open CASCADE等待程序运行的时间 * Open CASCADE处理的最大几何场景规模 * Open CASCADE极限运算提供的运算速度。 |

### 2.2.2 可用性和可靠性

Open CASCADE可用性和可靠性质量属性分析如表2-14所示。

表2-14 Open CASCADE可用性和可靠性质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | 系统内部（运算、图元），系统外部（框架使用者，待读取文件，操作系统） |
| 刺激 | 异常发生，数据错误，程序崩溃 |
| 制品 | 执行节点、数据、运行进程 |
| 环境 | 正常操作情况 |
| 响应 | Open CASCADE使用下列方式进行处理   * 使用合理的自定义的异常对用户进行提示 * 继续尝试在警告状态下运行 * 通过自己定义的方式，遏制错误的传递 |
| 响应度量 | * Open CASCADE计算结果与正确结果的误差 |

### 2.2.3 易用性

Open CASCADE易用性质量属性分析如表2-15所示。

表2-15 Open CASCADE易用性质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | Open CASCADE用户 |
| 刺激 | 用户学习系统的使用方式、特性、用户使用Open CASCADE实现新的应用、正确使用Open CASCADE，适配当前的操作环境和系统 |
| 制品 | Open CASCADE本身 |
| 环境 | Open CASCADE运行过程中、使用过程中、配置过程中。 |
| 响应 | Open CASCADE采取以下方式进行响应：   * Open CASCADE提供代码的注释说明。 * Open CASCADE提供API接口文档、特性说明文档、基础使用文档。 * Open CASCADE提供简单的实例，让用户了解系统的运行过程。 * Open CASCADE提供默认选项和用户配置选项，帮助用户配置Open CASCADE适应环境。并且提供简单详细的命令。 * Open CASCADE应该提供可视化的界面，供用户操作和了解Open CASCADE的运行过程。 * Open CASCADE提供定制化功能，满足用户对产品本身的修改、编译、运行。 |
| 响应度量 | * 用户满意度。 * 用户学习使用的时间。 * 用户部署修改所用的时间。 * 用户使用过程中出现错误的频率。 |

### 2.2.4 可修改性

Open CASCADE可修改性质量属性分析如表2-16所示。

表2-16 Open CASCADE可修改性质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | Open CASCADE代码框架的使用者、底层框架、运行操作系统 |
| 刺激 | 希望增加、删除、修改某些功能、质量属性。或者底层的结构发生改变，需要适应新的应用场景。 |
| 制品 | Open CASCADE的编程接口、底层封装模块、功能模块 |
| 环境 | 在Open CASCADE编写构建过程中，使用Open CASCADE的开发过程中，或Open CASCADE的运行过程中 |
| 响应 | Open CASCADE采取以下方式进行响应：   * 能够扩展新的功能。 * 能够进行调试。 * 能够修改底层的封装过程与对外部显示的接口。 |
| 响应度量 | * Open CASCADE扩展所需要的时间。 * Open CASCADE修改需要的资金成本 * 修改Open CASCADE对其他部分的影响程度。 * Open CASCADE修改影响的代码范围。 |

### 2.2.5 可重用性

Open CASCADE可重用性质量属性分析如表2-17所示。

表2-17 Open CASCADE可重用性质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | Open CASCADE的上层应用 |
| 刺激 | Open CASCADE的功能进行扩展 |
| 制品 | Open CASCADE的编程接口、功能模块、设计书、代码、应用程序 |
| 环境 | 在Open CASCADE进行设计时、开发时、编译时、应用时 |
| 响应 | Open CASCADE应该实现以下策略：   * 将代码模块化，实现功能的分离独立。 * 代码具有良好的可读性。 * 代码能够实现抽象继承。 * 能够向前兼容应用程序。 |
| 响应度量 | * Open CASCADE代码重用的比例。 |

### 2.2.6 可集成性

Open CASCADE可集成性质量属性分析如表2-18所示。

表2-18 Open CASCADE可集成性质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | Open CASCADE上层应用 |
| 刺激 | Open CASCADE进行编译过程中 |
| 制品 | Open CASCADE的编程接口、配置文件 |
| 环境 | 编译时、应用时、修改后 |
| 响应 | Open CASCADE应该实现以下几种响应。   * Open CASCADE提供运行环境说明和相关配置方法。 * Open CASCADE向上层应用提供接口API。 * Open CASCADE能够适用不同版本的操作系统和依赖。 |
| 响应度量 | * Open CASCADE等待程序运行的时间 * Open CASCADE处理的最大几何场景规模 * Open CASCADE极限运算提供的运算速度。 |

### 2.2.7 可测试性

Open CASCADE可测试性质量属性分析如表2-19所示。

表2-19 Open CASCADE可测试性质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | 模块开发人员、模块测试人员、Open CASCADE用户、Open CASCADE的上层应用 |
| 刺激 | 已完成需求分析、已完成体系结构设计、已经完成新的类和子系统设计、Open CASCADE的上层应用搭建完成 |
| 制品 | Open CASCADE的编程接口、功能模块、设计书、代码、应用程序 |
| 环境 | 在Open CASCADE进行设计时、开发时、编译时、应用时 |
| 响应 | Open CASCADE提供对相关文档、相关代码的访问权限。   * Open CASCADE能够提供设计说明文档 * Open CASCADE提供对运行状态的描述和运行时值的访问 |
| 响应度量 | * 可测试语句的百分比。 * 黑盒测试与白盒测试的故障率。 * Open CASCADE在使用过程中报错的比例。 * 对Open CASCADE进行测试的时间。 |

* + 1. 商业质量属性

Open CASCADE的商业质量属性包括开发成本与收益以及开发计划。本节将对上述两者分别描述如下。

### 2.3.1 开发成本与收益

表2-20 Open CASCADE开发成本与收益质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | 模块开发人员、用户、工程的持有者。 |
| 刺激 | 项目需求分析完成、系统设计完成、系统交付使用、系统维护、系统出现故障。 |
| 制品 | Open CASCADE、Open CASCADE上层应用。 |
| 环境 | 在Open CASCADE进行设计时、开发时、编译时、应用时 |
| 响应 | Open CASCADE应该通过以下方式平衡开发成本与收益。   * 削减不必要的功能与设计 * 增加开发投入。 |
| 响应度量 | * 开发成本。 * 项目的直接收益 * 项目的间接收益 |

### 2.3.2 开发计划

表2-21 Open CASCADE开发计划质量属性分析

|  |  |
| --- | --- |
| 场景 | 可能的值 |
| 源 | 模块开发人员、用户。 |
| 刺激 | 项目截止时间、临时增加需求 |
| 制品 | Open CASCADE |
| 环境 | 在Open CASCADE进行设计时、开发时、编译时、应用时 |
| 响应 | Open CASCADE应该通过以下方式调整开发计划。   * 增加开发人员，加快开发进度。 * 去除多余的功能与模块。 * 重新进行设计与开发。 |
| 响应度量 | * Open CASCADE开发周期 * Open CASCADE开发人员与规模 * Open CASCADE开发进度 |

1. 软件体系结构设计与分析
   * + 1. Open CASCADE体系结构建模

### 3.1.1 Open CASCADE模块分解与对应

首先对模块进行分解，建立软件体系结构模块与代码的包类对应关系。对Open CASCADE的源码进行分析，将Open CASCADE分解为主要的六大模块[4]： Foundation Classes模块、Modeling Data模块、Modeling Algorithms模块、Visualization模块、Data Exchange & Application Framework模块和Draw模块。将各大模块分解为子模块。其中模块的分解及顶层模块间的关系如下图3-1所示。

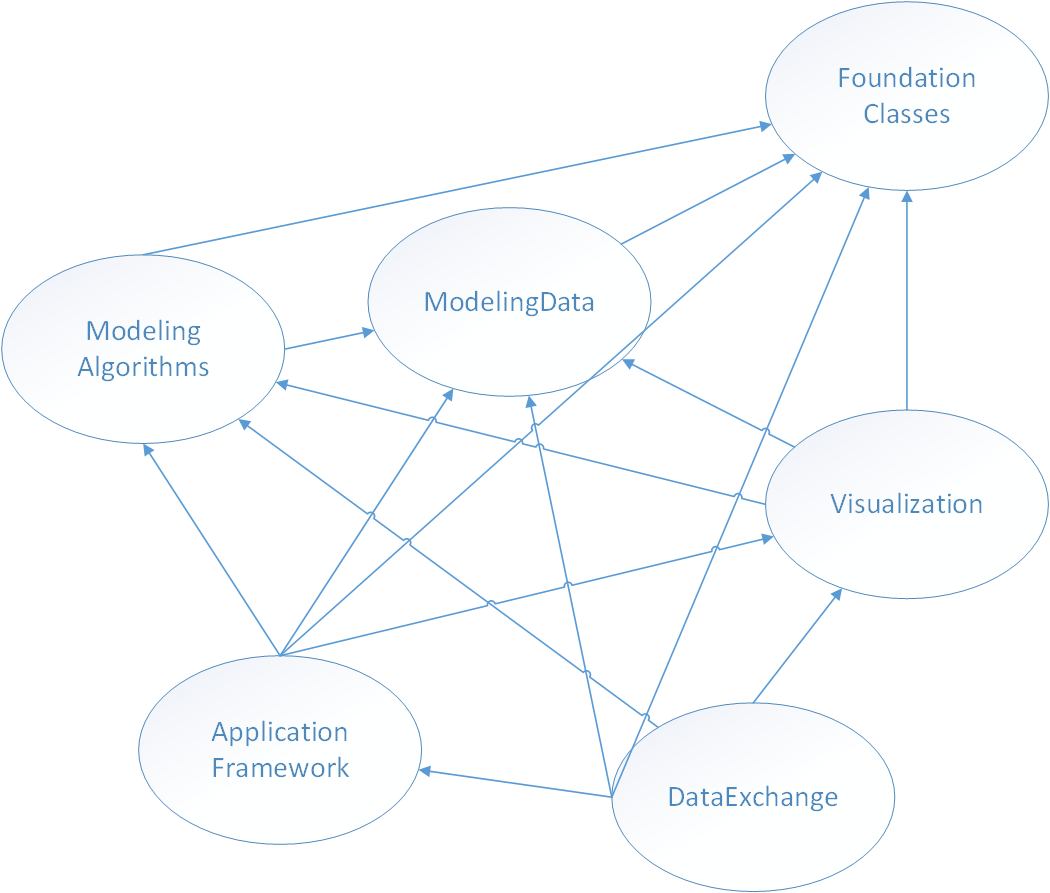


图3-1 Open CASCADE模块分解

### 3.1.2 Open CASCADE模块功能描述

#### 3.1.2.1 Foundation Classes模块

Foundation Classes基础类提供了一些通用的服务：基本数据类型，字符串等，堆内存的自动化管理异常处理，局部支持多线程，数据集合的处理，向量、矩阵及原始几何类型的数据计算工具，用ASCLL文件保存数据的基本服务[5]。

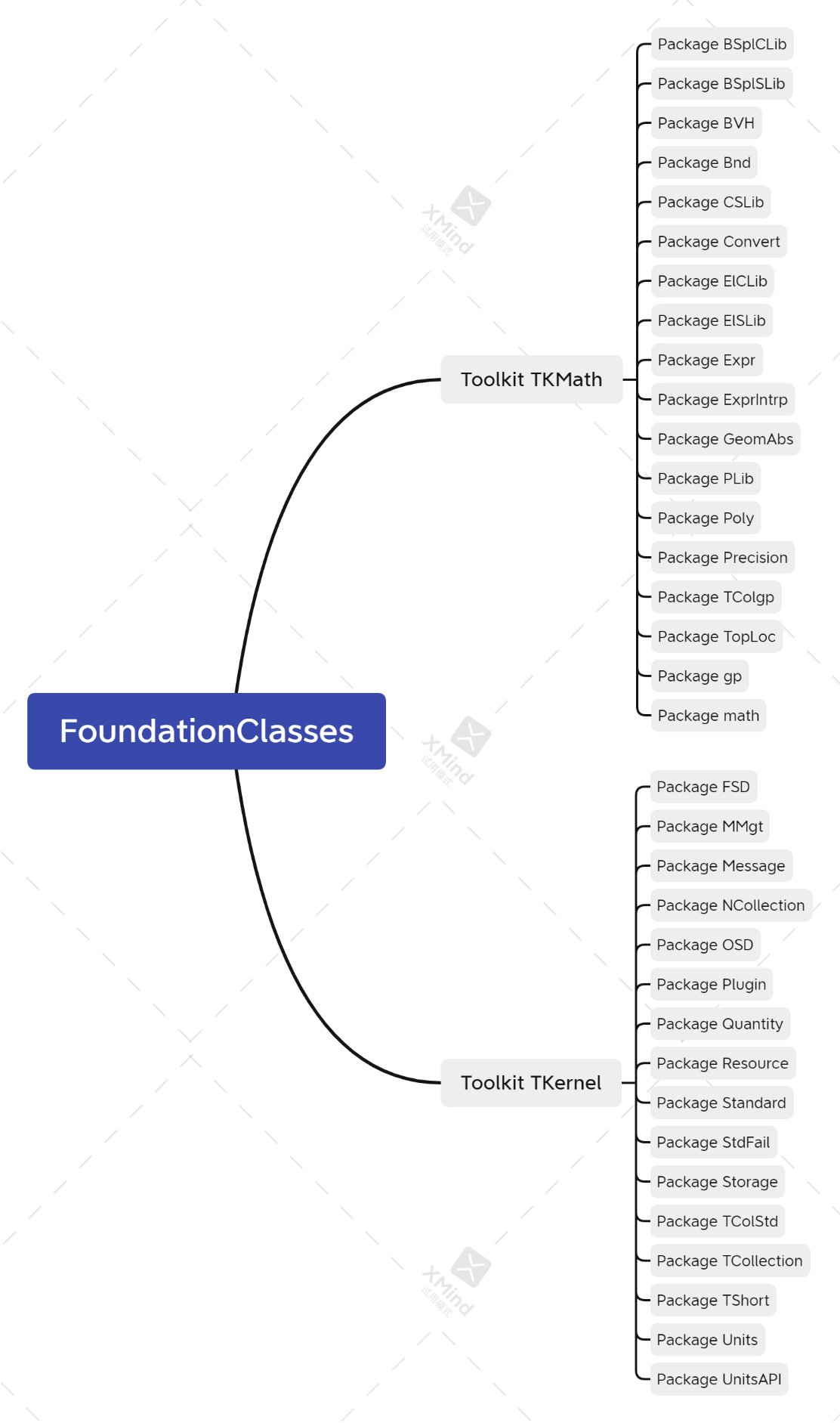


图3-2 Foundation Classes模块功能分解

表3-1 Foundation Classes模块功能概述

|  |  |  |  |
| --- | --- | --- | --- |
|  | 模块名称 | | 模块功能概述 |
| Toolkit TKMath | | P\_Bnd | 提供二维和三维空间中几何元素包围盒的计算功能 |
| P\_ElCLib | ElCLib代表：Elementary Curves Library. 提供初等曲线曲面的基本几何计算功能； |
| P\_ElSLib | ElSLib代表：Elementary Surfaces Library. 提供初等曲面的基本几何计算。 |
| P\_Precision | 由于浮点数在计算机内实际上是一个近似表示，在手工计算看来为正确的结果，在计算机中运算未必得出正确的结果。所以，我们得到一个重要的经验：使用浮点数进行相等（==）和不等（！=）比较的操作通常是有问题的。浮点数的相等比较，一般总是使用两者相减的值是否落在0的邻域中来判断。这就是邻域比较技术。在OpenCASCADE中专门提供包Precision来处理两个数值的比较问题。 |
| P\_TColgp | 提供类如：XY, XYZ, Pnt, Pnt2d, Vec, Vec2d, Lin, Lin2D, Circ, Circ2d的TCollection的实例。 |
| P\_TopLoc | OCC中使用的拓扑数据结构包；提供了处理三维局部坐标系的资源， |
| P\_gp | gp是几何处理程序包（Geometric Processor P\_），简称gp。包gp提供以下功能：  代数计算；如坐标计算、矩阵计算；  基本解析几何元素；如变换、点、矢量、线、面、轴、二次曲线和初等曲面； |
| Toolkit TKernel | | P\_Quantity | 数据及物理数量，是一个REAL 值；包括了所有您所需要的物理量。 |

#### 3.1.2.2 Modeling Data模块

Modeling Data模块提供了一些通用的服务：表现2D、3D几何模型的数据结构主要包含在四个类库中：2D几何（2D Geometry）、3D几何（3D Geometry）、几何工具（Geometry Utilities）、几何拓扑（Topology）。

2D Gemoetry类库提供2D几何数据结构及拓扑方向。Gemo2D几何包提供通过引用及遵循STEP标准处理2D几何数据结构处理，这些对象被参数化，面向对象的，包括Bezier贝塞尔曲线、B样条曲线、offset curves偏移曲线，提供从基本几何类型（非面向对象，非参数化）与Gemo 2D 对象的转换功能。Geom2dLProp几何属性包允许诸如此的计算：二维曲线基于一个参数点的Derivative向量、二维曲线（正常或曲率）上基于一个参数点的切向量

3D Gemoetry类库提供3D几何数据结构及拓扑方向。Gemo几何包提供通过引用及遵循STEP标准处理3D几何数据结构处理，这些对象被参数化，面向对象的，包括Bezier贝塞尔曲线、B样条曲面、offset curves偏移曲线及曲面，提供从基本几何类型（非面向对象，非参数化）与Gemo对象的转换功能。GeomLProp几何属性包允许诸如此的计算：二维曲线或曲面基于一个参数点的Derivative向量、二维曲线或曲面（正常或曲率）上基于一个参数点的切向量。

Geometry Utilities 该类库提供标准的在二维及三维几何上的高级功能：

1. 几何算法的直接构造；
2. 一组点插值形成一条曲线；
3. 初等几何转换B样条曲线及曲面；
4. 2D或3D曲线上点的计算；
5. 两种几何的极值计算；

具体如图3-3与表3-2所示。

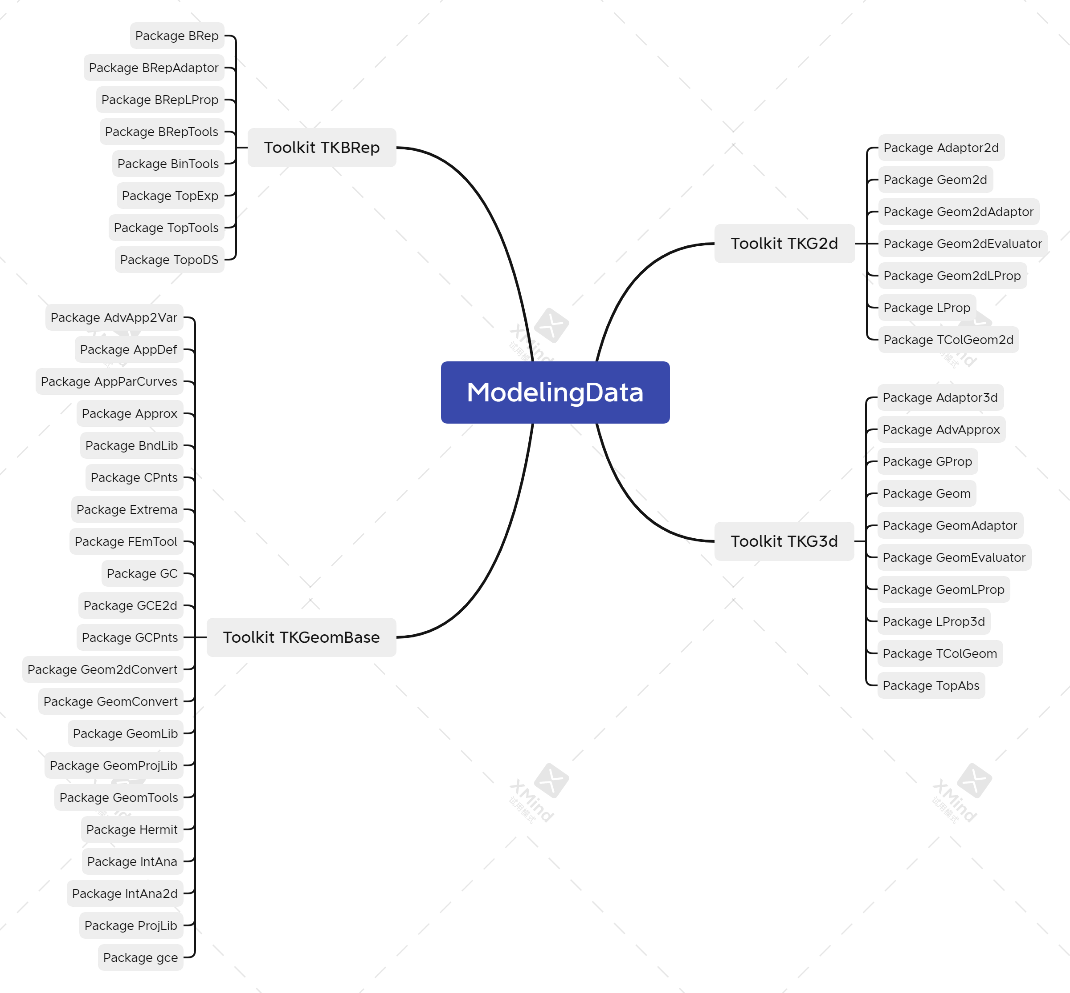


图3-3 Modeling Data模块功能分解

表3-2 Modeling Data模块功能概述

|  |  |  |
| --- | --- | --- |
|  | 模块名称 | 模块功能概述 |
| Toolkit TKBRep | P\_BRepTools | 访问与处理抽像拓扑数据结构的方法；explore, manipulate, 读、写BRep数据结构； |
| P\_TopExp | 访问与处理抽像拓扑数据结构的方法；explore and manipulate（操作）TopoDS中所描述的拓扑数据结构。 |
| P\_TopTools | 访问与处理抽像拓扑数据结构的方法；用在拓扑数据结构上的工具库。 |
| P\_TopoDS | OCC中使用的拓扑数据结构包；描述模型的类集合，建立纯拓扑的数据结构 |
| Toolkit TKG2d | P\_Geom2d | 描述简单几何对象的基本数据结构。用STEP实现二维几何对象；依据继承关系来组织类库；如：Geom2d\_Ellipse inherit Geom2d\_Conic  Geom2d\_BezierCurve inherit Geom2d\_BoundedCurve  Curve inherit Geom2d\_Curve  Curves, points and vectors inherit Geom2d\_Geometry |
| P\_Geom2dLProp | 二维曲线的本地属性； |
| P\_LProp | 提供一个枚举用于标记二维曲线上的特征点 |
| Toolkit TKG3d | P\_Geom | 描述简单几何对象的基本数据结构。用STEP实现三维几何对象；依据继承关系来组织类库；如：Geom2d\_Geometry inherit Geom\_ElementarySurface  Geom\_BezierSurface inherit Geom\_BoundedSurface  surfaces inherit Geom\_Surface  Curves, points and vectors inherit Geom\_Geometry |
| P\_GeomLProp | 三维曲线及曲面的本地属性 |
| Toolkit TKGeomBase | P\_GC | 通过Direct Construction组件构造诸如（gp\Geom2d\Geom）的对象类型。但是前者实现了几何构造算法，用于转换构造器参数到每个对象的特殊数据结构。  它们提供了一组构造算法类； |
| P\_GCE2d | gce\_MakeCire:定义了八个，在圆的几何构造中遇到的问题。实现了八个相关的构造算法； |
| P\_Geom2dConvert | Geom2d包相关；  transformation Bezier->BSpline；  transformation BSpline ->Bezier |
| P\_GeomConvert | Geom包相关；  BSpline、Bezier曲线，曲面的构造、转换等与之相关的算法 |

#### 3.1.2.3 Modeling Algorithms模块

Modeling Data模块在模型中使用拓扑算法形成建模算法模块组，通过这些，可以找到几何算法，服务封装在以下类库中，具体类库模块间关系如图3-4所示，Modeling Algorithms模块功能分解如图3-5所示。

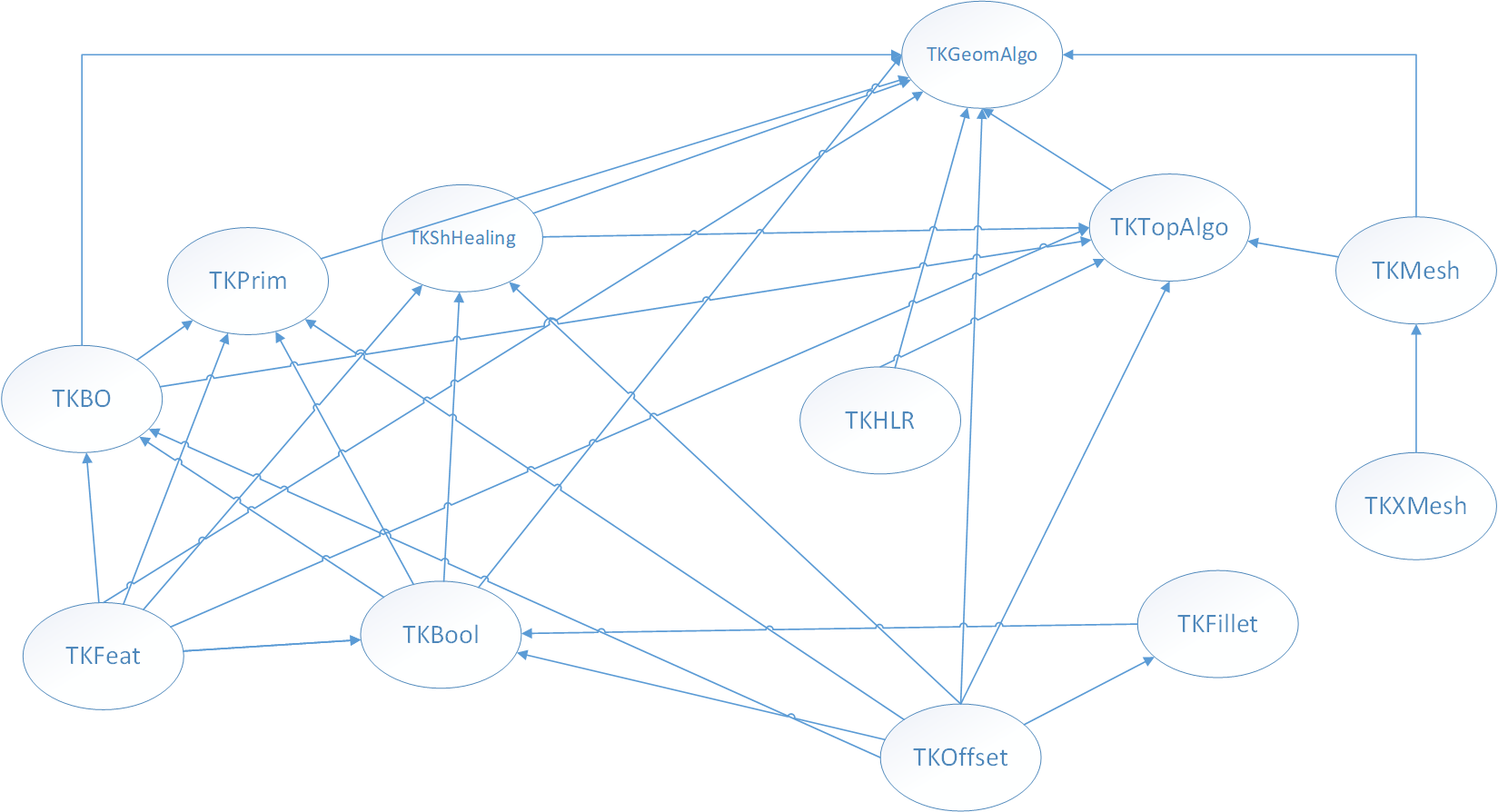


图3-4 Modeling Algorithms模块间关系

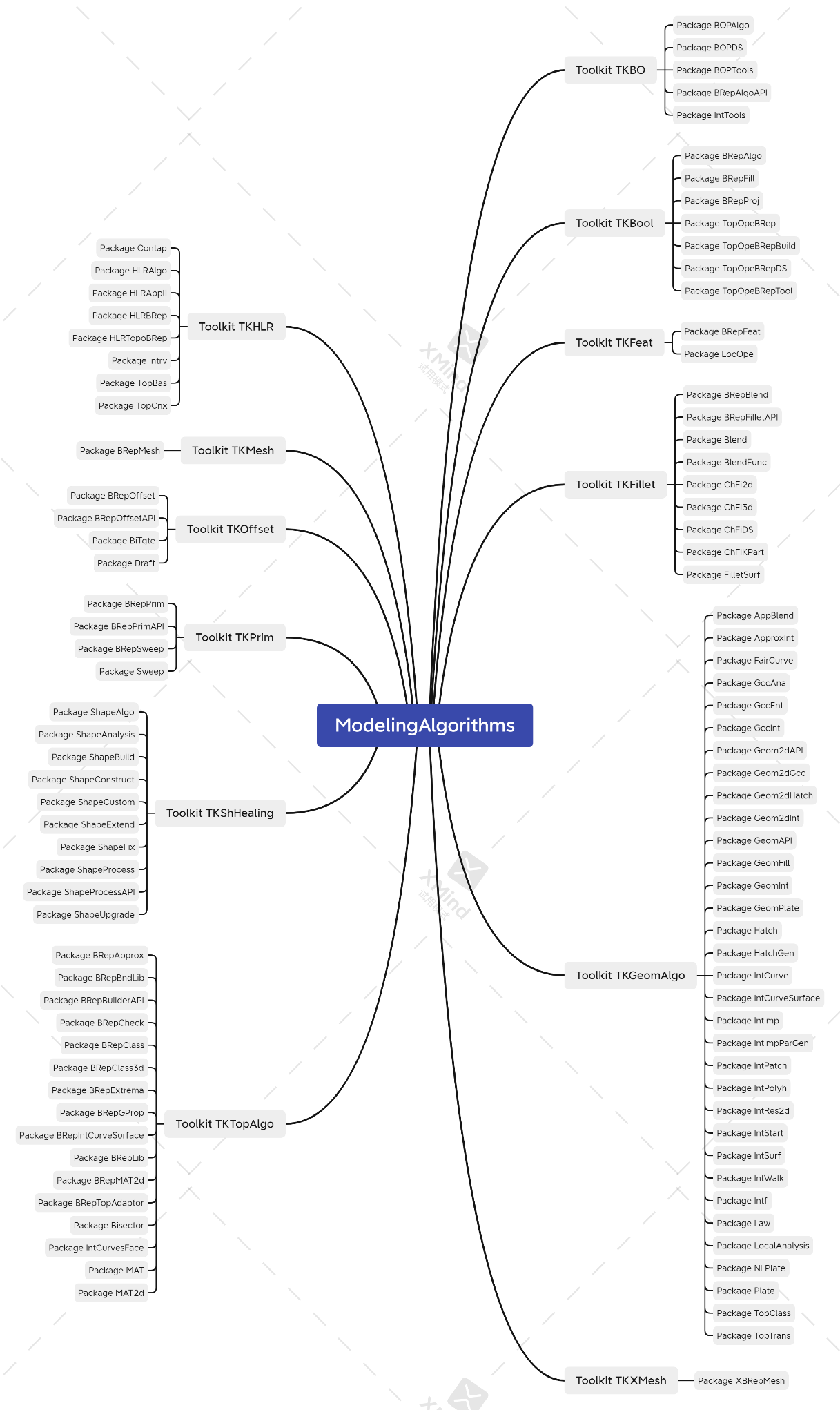


图3-5 Modeling Algorithms模块功能分解

#### 3.1.2.4 Visualization模块

Visualization模块提供了通用的二维与三维数据结构并分别通过不同的Viewer去处理。3D Graphics运行在OPENGL图像平台的基础上；支持对图元（折线、带孔或无孔的平面多边形、文本以及标志控制属性：颜色、透明度、反射、线型、线宽、字体等）进行三维操作，这些图元可以在三维视图中进行显示、绽放、旋转等操作；2D Graphics支持对图元（直线、线段、折线、文本以及标志控制属性：颜色、线型、线宽、字体等）进行二维操作，这些图元可以在二维视图中进行显示、绽放、旋转等操作；以PostScript, HPGL2, and CGM格式输出，同时也支持xwd,gif,bmp格式的图片。在二维，OCC提供了完整的国际化，中文与日文的字符文本均可以在绘图仪上输入输出打印。

模块功能分解如图3-6所示。

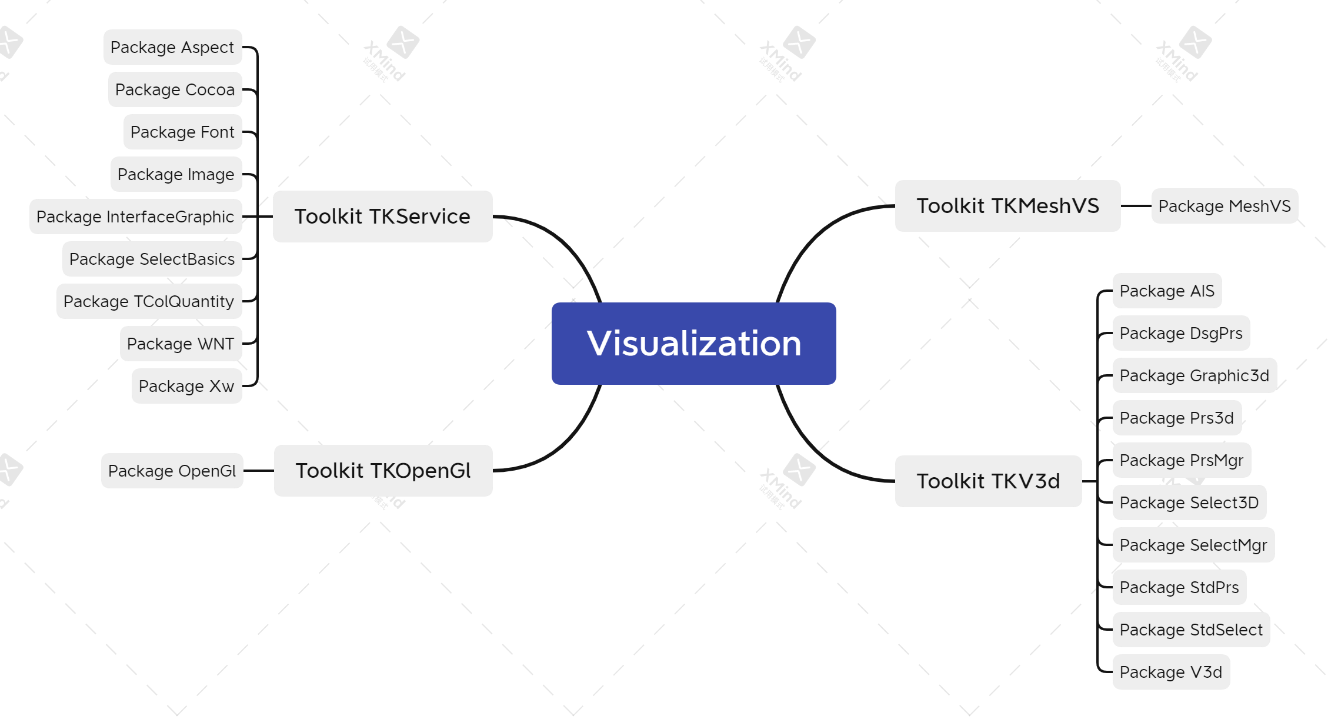


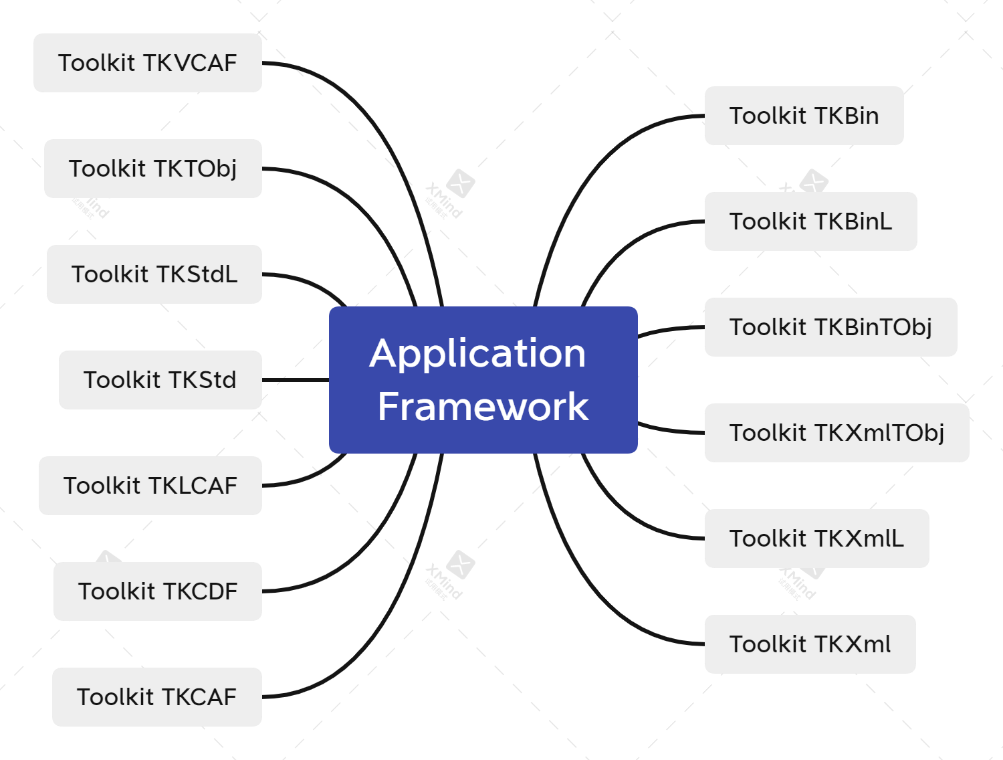
图3-6 Visualization模块功能分解

表3-3 Visualization模块功能概述

|  |  |  |
| --- | --- | --- |
|  | 模块名称 | 模块功能概述 |
| Toolkit TKMeshVS | P\_TKMeshVS | 网格显示服务，对OCC三维显示能力的扩展 |
| Toolkit TKService | P\_Image | 提供伪彩色与彩色图像的定义，图像的一组关键功能； |
| P\_SelectBasics | sensitive primitive原型的定义；在视窗中一个可选择的实体；  sensitive primitive原型所有者的定义；这个实体所原始实体与在视窗中可选择的应用程序实体相关系 |
| P\_WNT | 提供通用的WINDOWS NT的图像接口 |
| Toolkit TKV3d | P\_AIS | 应用交互服务（Application Interactive   Services (AIS)），提供方法创建应用程序GUI VIEWER之间的连接，用于管理选择与显示。定位于建模环境、VIEW以及选择。必须有几种定义的不同的可选择、可显示的对象。必须也是可交互的，也就是说，连接基本参考几何与图像显示；这些实体被称为交互对象，被分成以下四种类型：the Datum\the Relation\ the Object\ None。   * the Datum:元素的构造如：线，圆、点、trihedra, plane trihedra、平面、轴线等（tri---三，hedra—多面体，异面体）。 * the Relation：由一个或者多个交互形状与相应的几何关系的约束构成； * the Object：包括拓扑形状以及几何之间的连接 * None：为了不排除对象，告诉应用程序，更深入分析，直到在其产生的对象（in its generation）中找到一个可接受的对象定义 |
| P\_DsgPrs | 尺寸、关系及三面体的显示工具。 |
| P\_Graphic3d | 在视窗中创建三维图像对象；这些对象称为结构，由一组原件及属性组成。这个组是一个最小的可编辑的结构元素。这个结构可被显示、擦出、高亮。可以对它应用变换。结构通过变换来构成，并连接形成一个继承体系。VIEWER可执行全局的结构操作处理。 |
| P\_Prs3d | 提供通用几何及拓扑的标准显示工具；  显示对象；属性管理员控制如何显示对象，如颜色、宽度、线的类型（generic objects）；通用算法提供默认对象的基本设置如：点，曲线、曲面、形状等，一个根对象提供抽象的框架，对于DsgPrs包中定义的工作； |
| P\_PrsMgr | 提供一个低级别的服务，仅当您不使用AIS提供的服务时，才使用它。  支持一个显示对象的图像结构；当需要时进行重新计算（移动、改变颜色等）；  定义显示对象的显示模式；就AIS\_Shape而言，确定是否对象以（线框0、阴影1）来显示。  每一个新的交互对象必须由它定义自己的显示模式。 |
| P\_Select3D | 定义标准的三维原件如：点，曲线，面  在二维图像选择空间中the bounding boxes的复原。  三维-二维的投影器 |
| P\_SelectMgr | 提供一个低级服务，仅当您不使用AIS提供的服务时，才使用（SelectMgr\_SelectionManager and SelectMgr\_ViewerSelector）。  它通过以下服务来管理进程的动态选择：  激活与关闭交互对象选择模式；  添加与移除视察选择器；  定义抽象的过滤类； |
| P\_StdPrs | 提供对特殊几何及拓扑的标准显示工具；这些类是特殊几何及拓扑，在务种显示模式下显示的定义，如：线框、阴影、去隐藏线等； |
| P\_StdSelect | 定义拓扑几何的选择方式；定义几种过滤类；二维与三维的视窗选择器（viewer   selectors）。 |
| P\_V3d | 包含了一组三维显示（3D Viewer）的命令与服务。控制场景及显示模式。这个包与Visual3D是互补的。 |

#### 3.1.2.5 DataExchange & ApplicationFramework模块

DataExchange & ApplicationFramework类提供了一些程序框架中的基本服务：数据框架的使用，数据分配的说明等，以及导入导出的各种类型转换，具体模块功能分解如图3-7所示。



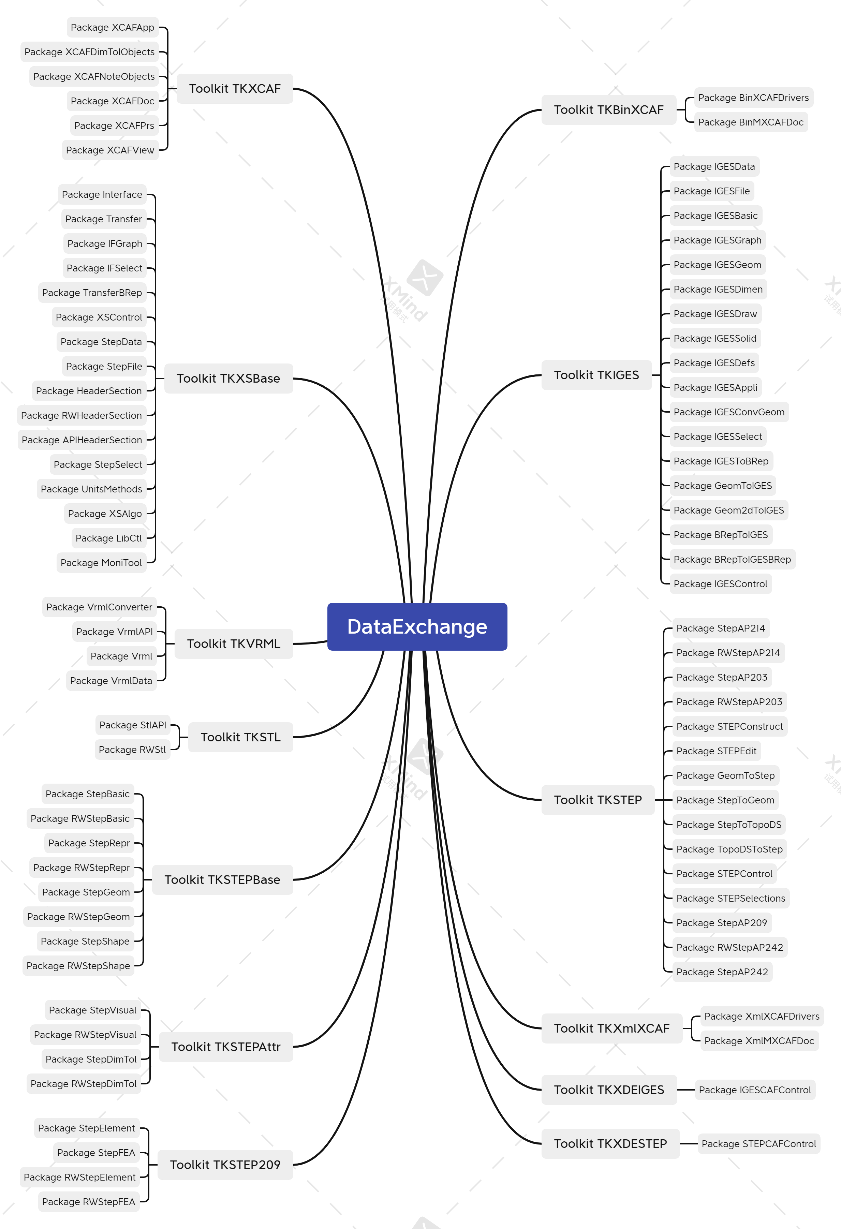


图3-7 DataExchange & ApplicationFramework模块功能分解

#### 3.1.2.6 Draw模块

Draw模块为Visualization、Model Algorithms、DataExchange & ApplicationFramework提供测试，负责在用户画2D，3D草图时对界面进行更新，草图的展示，可视化操作和模型的存取对象，在数据变化时通过控制器更新视图。具体模块间关系与模块功能分解如图3-8所示。

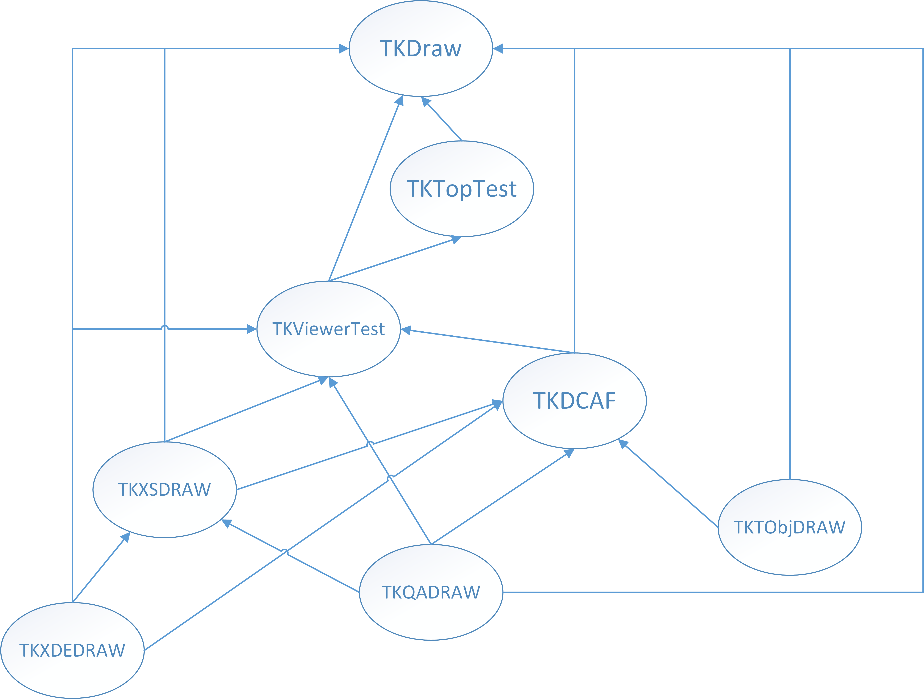


图3-10 Draw模块间关系

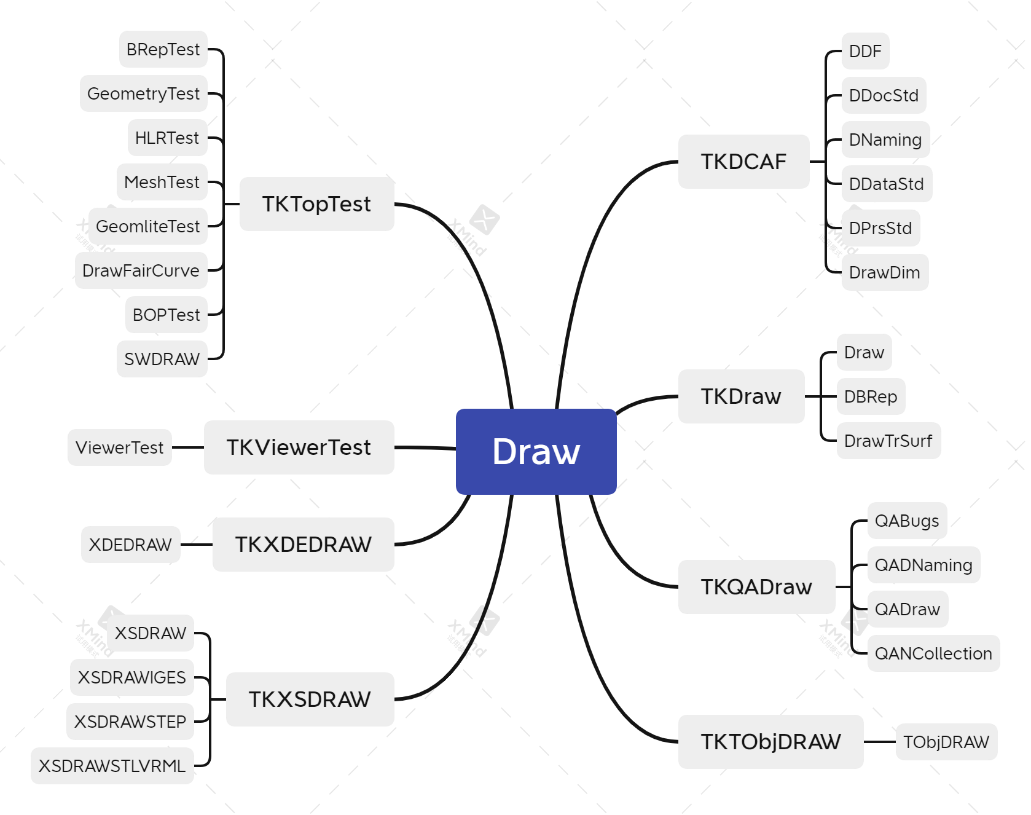


图3-9 Draw模块功能分解

* + - 1. Open CASCADE体系结构分析

### 3.2.1 功能需求与软件体系结构的对应关系

1. 基本数据类型的创建及运算

表格 3-4 基本数据类型的创建及运算需求对应表

|  |  |
| --- | --- |
| 软件需求 | 基本数据类型的创建及运算 |
| 对应模块 | M\_FoundationClasses |
| 对应的包和类 | P\_BSplCLib，P\_BSplSLib，P\_BVH  P\_Bnd，P\_CSLib，P\_Convert  P\_ElCLib，P\_ElSLib，P\_Expr  P\_ExprIntrp，P\_GeomAbs，P\_PLib  P\_Poly，P\_Precision，P\_TColgp  P\_TopLoc，P\_gp，P\_math |
| 模块与代码说明 | Foundation Classes基础类提供了一些通用的服务：基本数据类型，字符串等，堆内存的自动化管理异常处理，局部支持多线程，数据集合的处理，向量、矩阵及原始几何类型的数据计算工具，用ASCLL文件保存数据的基本服务。 |

1. 基本几何图元的创建功能

表格 3-5 基本几何图元的创建功能对应表

|  |  |
| --- | --- |
| 软件需求 | 基本几何图元的创建 |
| 对应模块 | M\_ModelingData |
| 对应的包和类 | P\_BRep，P\_BRepAdaptor，P\_BRepLProp  P\_BRepTools，P\_BinTools，P\_TopExp  P\_TopTools，P\_TopoDS，P\_Adaptor2d  P\_Geom2d，P\_Geom2dAdaptor  P\_Geom2dEvaluator，P\_Geom2dLProp  P\_LProp，P\_TColGeom2d，P\_Adaptor3d  P\_AdvApprox，P\_GProp，P\_Geom  P\_GeomAdaptor，P\_GeomEvaluator  P\_GeomLProp，P\_LProp3d，P\_TColGeom  P\_TopAbs，P\_AdvApp2Var，P\_AppCont  P\_AppDef，P\_AppParCurves，P\_Approx  P\_BndLib，P\_CPnts，P\_Extrema  P\_FEmTool，P\_GC，P\_GCE2d  P\_GCPnts，P\_Geom2dConvert，P\_GeomConvert  P\_GeomLib，P\_GeomProjLib，P\_GeomTools  P\_Hermit，P\_IntAna，P\_IntAna2d  P\_ProjLib，P\_gce |
| 模块与代码说明 | 表现2D、3D几何模型的数据结构主要包含在四个类库中：2D几何（2D Geometry）、3D几何（3D Geometry）、几何工具（Geometry Utilities）、几何拓扑（Topology）。 |

1. 基本几何操作、空间变换、解析几何运算、解析几何建模、网格表示及相关运算、形状修复、工业几何操作功能

表格 3-6 基本几何操作、空间变换、解析几何运算等功能对应表

|  |  |
| --- | --- |
| 软件需求 | 基本几何操作、空间变换、解析几何运算、解析几何建模、网格表示及相关运算、形状修复、工业几何操作 |
| 对应模块 | M\_Modeling Algorithms |
| 对应的包和类 | P\_BRepAlgo，P\_BRepFill，P\_BRepProj  P\_TopOpeBRep，P\_TopOpeBRepBuild  P\_TopOpeBRepDS，P\_TopOpeBRepTool  P\_AppBlend，P\_ApproxInt，P\_FairCurve  P\_GccAna，P\_GccEnt，P\_GccInt  P\_Geom2dAPI，P\_Geom2dGcc，P\_Geom2dHatch  P\_Geom2dInt，P\_GeomAPI，P\_GeomFill  P\_GeomInt，P\_GeomPlate，P\_Hatch  P\_HatchGen，P\_IntCurve，P\_IntCurveSurface  P\_IntImp，P\_IntImpParGen，P\_IntPatch  P\_IntPolyh，P\_IntRes2d，P\_IntStart  P\_IntSurf，P\_IntWalk，P\_Intf  P\_Law，P\_LocalAnalysis，P\_NLPlate  P\_Plate，P\_TopClass，P\_TopTrans  P\_Contap，P\_HLRAlgo，P\_HLRAppli  P\_HLRBRep，P\_HLRTopoBRep，P\_Intrv  P\_TopBas，P\_TopCnx，P\_ShapeAlgo  P\_ShapeAnalysis，P\_ShapeBuild  P\_ShapeConstruct，P\_ShapeCustom  P\_ShapeExtend，P\_ShapeFix，P\_ShapeProcess  P\_ShapeProcessAPI，P\_ShapeUpgrade |
| 模块与代码说明 | Modeling Data模块在模型中使用拓扑算法形成建模算法模块组，OCC也提供一项服务被叫做SSP（基于离散点的曲面重构） Surfaces from Scattered Points (SSP)..，算法首先构造一个初始的B样曲面，再查找其满足条件的变形（有限元法）；利用优化的算法，可以利用500000点来构造出一个曲面；NURBS是非均匀有理样条曲线：NURBS是一种非常优秀的建模方式，在高级三维软件当中都支持这种建模方式。NURBS能够比传统的网格建模方式更好地控制物体表面的曲线度，从而能够创建出更逼真、生动的造型。 |

1. 可视化功能

表格 3-7 可视化功能对应表

|  |  |
| --- | --- |
| 软件需求 | 可视化功能 |
| 对应模块 | M\_Visualization |
| 对应的包和类 | P\_OpenGl，P\_Aspect，P\_Cocoa  P\_Font，P\_Image，P\_InterfaceGraphic  P\_SelectBasics，P\_TColQuantity  P\_WNT，P\_Xw，P\_AIS  P\_DsgPrs，P\_Graphic3d，P\_Prs3d  P\_PrsMgr，P\_Select3D，P\_SelectMgr  P\_StdPrs，P\_StdSelect，P\_V3d |
| 模块与代码说明 | Visualization模块提供了通用的二维与三维数据结构并分别通过不同的Viewer去处理。3D Graphics运行在OPENGL图像平台的基础上；支持对图元（折线、带孔或无孔的平面多边形、文本以及标志控制属性：颜色、透明度、反射、线型、线宽、字体等）进行三维操作，这些图元可以在三维视图中进行显示、绽放、旋转等操作；2D Graphics支持对图元（直线、线段、折线、文本以及标志控制属性：颜色、线型、线宽、字体等）进行二维操作，这些图元可以在二维视图中进行显示、绽放、旋转等操作；以PostScript, HPGL2, and CGM格式输出，同时也支持xwd,gif,bmp格式的图片。在二维，OCC提供了完整的国际化，中文与日文的字符文本均可以在绘图仪上输入输出打印。 |

1. 数据交换功能

表格 3-8 数据交换功能对应表

|  |  |
| --- | --- |
| 软件需求 | 数据交换功能 |
| 对应模块 | M\_ DataExchange &ApplicationFramework |
| 对应的包和类 | P\_TKBinXCAF，P\_TKIGES，P\_TKSTEP  P\_TKSTEP209，P\_TKSTEPAttr  P\_TKSTEPBase，P\_TKSTL，P\_TKVRML  P\_TKXCAF，P\_TKXDEIGES，P\_TKXDESTEP  P\_TKXSBase，P\_TKXmlXCAF |
| 模块与代码说明 | DataExchange &ApplicationFramework模块中都是对外提供接口或是向外导出、向内导入文件的功能，属于一个开源项目在提供自身功能时应该开放的功能的集合。 |

1. 提供应用框架功能

表格 3-9 提供应用框架功能对应表

|  |  |
| --- | --- |
| 软件需求 | 对外提供应用框架 |
| 对应模块 | M\_ DataExchange &ApplicationFramework |
| 对应的包和类 | Toolkit TKBin，Toolkit TKBinL，Toolkit TKBinTObj  Toolkit TKCAF，Toolkit TKCDF，Toolkit TKLCAF  Toolkit TKStd，Toolkit TKStdL，Toolkit TKTObj  Toolkit TKVCAF，Toolkit TKXml，Toolkit TKXmlL  Toolkit TKXmlTObj |
| 模块与代码说明 | DataExchange &ApplicationFramework模块中都是对外提供接口或是向外导出、向内导入文件的功能，属于一个开源项目在提供自身功能时应该开放的功能的集合。 |

### 3.2.2 主要质量属性（非功能需求）与软件体系结构的对应关系

1. 性能

表格 3-10 性能与软件体系结构的对应关系

|  |  |
| --- | --- |
| 质量属性 | 性能 |
| 对应模块 | M\_FoundationClasses，M\_ModelingData，M\_Modeling Algorithms |
| 对应的包和类 | P\_BSplCLib，P\_BSplSLib，P\_BVH  P\_Bnd，P\_CSLib，P\_Convert  P\_ElCLib，P\_ElSLib，P\_Expr  P\_ExprIntrp，P\_GeomAbs，P\_PLib  P\_Poly，P\_Precision，P\_TColgp  P\_TopLoc，P\_gp，P\_math  P\_BRep，P\_BRepAdaptor，P\_BRepLProp  P\_BRepTools，P\_BinTools，P\_TopExp  P\_TopTools，P\_TopoDS，P\_Adaptor2d  P\_Geom2d，P\_Geom2dAdaptor  P\_Geom2dEvaluator，P\_Geom2dLProp  P\_LProp，P\_TColGeom2d，P\_Adaptor3d  P\_AdvApprox，P\_GProp，P\_Geom  P\_GeomAdaptor，P\_GeomEvaluator  P\_GeomLProp，P\_LProp3d，P\_TColGeom  P\_TopAbs，P\_AdvApp2Var，P\_AppCont  P\_AppDef，P\_AppParCurves，P\_Approx  P\_BndLib，P\_CPnts，P\_Extrema  P\_FEmTool，P\_GC，P\_GCE2d  P\_GCPnts，P\_Geom2dConvert，P\_GeomConvert  P\_GeomLib，P\_GeomProjLib，P\_GeomTools  P\_Hermit，P\_IntAna，P\_IntAna2d  P\_ProjLib，P\_gce  P\_BRepAlgo，P\_BRepFill，P\_BRepProj  P\_TopOpeBRep，P\_TopOpeBRepBuild  P\_TopOpeBRepDS，P\_TopOpeBRepTool  P\_AppBlend，P\_ApproxInt，P\_FairCurve  P\_GccAna，P\_GccEnt，P\_GccInt  P\_Geom2dAPI，P\_Geom2dGcc，P\_Geom2dHatch  P\_Geom2dInt，P\_GeomAPI，P\_GeomFill  P\_GeomInt，P\_GeomPlate，P\_Hatch  P\_HatchGen，P\_IntCurve，P\_IntCurveSurface  P\_IntImp，P\_IntImpParGen，P\_IntPatch  P\_IntPolyh，P\_IntRes2d，P\_IntStart  P\_IntSurf，P\_IntWalk，P\_Intf  P\_Law，P\_LocalAnalysis，P\_NLPlate  P\_Plate，P\_TopClass，P\_TopTrans  P\_Contap，P\_HLRAlgo，P\_HLRAppli  P\_HLRBRep，P\_HLRTopoBRep，P\_Intrv  P\_TopBas，P\_TopCnx，P\_ShapeAlgo  P\_ShapeAnalysis，P\_ShapeBuild  P\_ShapeConstruct，P\_ShapeCustom  P\_ShapeExtend，P\_ShapeFix，P\_ShapeProcess  P\_ShapeProcessAPI，P\_ShapeUpgrade |
| 模块与代码说明 | 基于C++编写了一套原生的数据处理方案，在代数运算上兼容了OpenBLAS加速，使得OpenCASCADE可以调用Intel MKL库进行高效运算；  在几何数据中，OpenCASCADE从底层进行了过程优化，降低了几何图元之间运算的复杂性。  综上，OpenCASCADE通过在基本代数几何运算与建模中充分利用硬件，实现了性能的保障。 |

1. 可用性和可靠性

表格 3-11 可用性和可靠性与软件体系结构的对应关系

|  |  |
| --- | --- |
| 质量属性 | 可用性和可靠性 |
| 对应模块 | 全局 |
| 对应的包和类 | 全局 |
| 模块与代码说明 | OpenCASCADE通过自定义异常，捕获程序运行时发生的数据异常或运算异常等，并采取保护措施，处理异常。 |

1. 易用性

表格 3-12 易用性与软件体系结构的对应关系

|  |  |
| --- | --- |
| 质量属性 | 易用性 |
| 对应模块 | 全局 |
| 对应的包和类 | 全局 |
| 模块与代码说明 | 1. 符合编程规范，代码中有大量注释。  3. 接口定义较为规范。  4. 不同版本能够实现向前兼容。 |

1. 可修改性

表格 3-13 可修改性与软件体系结构的对应关系

|  |  |
| --- | --- |
| 质量属性 | 可修改性 |
| 对应模块 | 全局 |
| 对应的包和类 | 全局 |
| 模块与代码说明 | OpenCASCADE提供应用框架模块，可以在此基础上进行方便的二次开发；  OpenCASCADE的基础模块高度松耦合，基于其开源特性，可以方便地修改其中某模块的实现。 |

1. 可重用性

表格 3-14 可重用性与软件体系结构的对应关系

|  |  |
| --- | --- |
| 质量属性 | 可重用性 |
| 对应模块 | 全局 |
| 对应的包和类 | 全局 |
| 模块与代码说明 | 1. 使用了接口和抽象类，能够通过继承的方式进行丰富的扩展和重用。  2. 代码不同版本之间，实现了向前兼容。 |

1. 可集成性

表格 3-15 可集成性与软件体系结构的对应关系

|  |  |
| --- | --- |
| 质量属性 | 可集成性 |
| 对应模块 | 全局 |
| 对应的包和类 | 全局 |
| 模块与代码说明 | 1. 通过丰富的配置文档，使得用户可以在不同的运行环境进行定制化，适用于多种平台。  2. 开源代码，用户可以根据自己的环境重新编译，更适应于当前的运行环境。 |

1. 可测试性

表格 3-16 可测试性与软件体系结构的对应关系

|  |  |
| --- | --- |
| 质量属性 | 可测试性 |
| 对应模块 | 全局 |
| 对应的包和类 | 全局 |
| 模块与代码说明 | OpenCASCADE提供扩展的OCCT自动测试系统，它围绕DRAW Test Harness组织，测试分为3个级别：  1. Group：一组相关的测试网格，通常测试特定的OCCT功能（例如blend）；  2. Mesh：一个Group内的一组测试用例，通常旨在测试相关功能的某些特定方面或执行模式（例如 buildevol）；  3. 测试用例：实现单个测试的脚本（例如 K4）。 |

### 3.2.3 软件体系结构分析

Open CASCADE体系结构框架图[5]如图3-10所示，根相关的类由包来分组，避免了类名的冲突。C++类名前加上包名作为前缀。如包Geom中所有的3D几何体，实现Bezier曲面的类为BezierSurface，其全称为Geom\_BezierSurface。包组成库，即提供给程序链接使用。库分组成六个模块，最小的模块Foundation Classes包含两个库，最大的模块Modeling Algorithms，包含八个库。

Open CASCADE下模块采用了多种设计模式，M\_ModelingData和M\_DataExchange采用了适配器模式，旨在让不兼容的接口类之间可以互相工作，M\_ApplicationFramework则采用了抽象工厂模式，用以提高代码的可复用性。

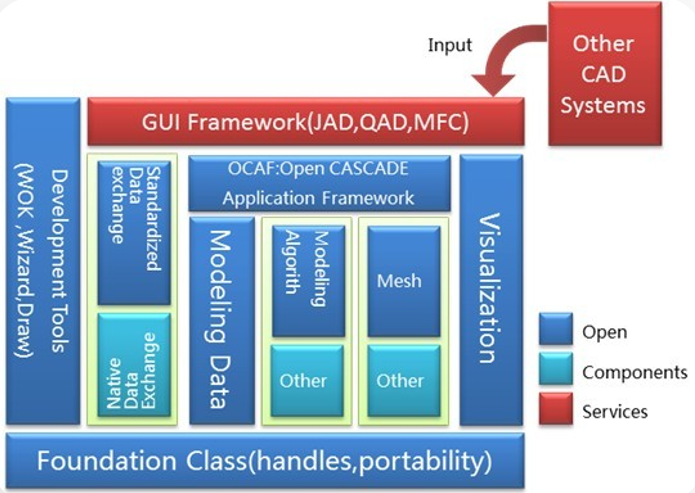


图3-10 Open CASCADE的体系结构架构图

### 3.2.4 软件体系结构模型中的异味

1）模块间存在相互继承、有多版本的情况，导致代码冗余的问题；

2）在包的划分上，存在大量同名的XX包、XX2d包与XX3d包，使得部分模块中的逻辑较为混乱。

除此外，该项目无明显异味。

1. 设计特色分析
   * + - 1. 值得借鉴之处

Open CASCADE包括3D曲面和实体建模组件、可视化组件、数据交换组件和适合快速开发的应用程序开发框架，提供二维和三维几何体的生成、显示和分析，是一款可以用来进行计算机可视化的几何内核[6]，并且是开放源代码的。Open CASCADE提供了点、线、面、体和复杂形体的显示和交互操作，经过深度开发后可实现纹理、光照、图元填充、渲染等图形操作和放大、缩小、旋转、漫游、模拟飞行、模拟穿越等动态操作[7]。

1) 开源：当今的 CAD 软件都是建立在几何内核的基础上开发的，比如 AutoCAD、金银花的内核是 ACIS，Pro/E、UG、Solid Works 和 Solid Edge 的内核是 parasolid，还有北航海尔的 CAXA 的内核也是 parasolid。所谓几何内核实际上就是一个类库，里面定义了图形数据的存储格式以及大量的图形算法，而 Open CASCADE 是目前唯一可用的开源几何建模内核[8]。库的数据结构与ISO 10303-42兼容(STEP)。给目前的方法带来了一定程度的普遍性。已知类似的软件也存在于其他几何内核中。然而，其他内核是封闭源代码的，不容易用于无限制的学术和商业用途。

2) 丰富的平台：Open CASCADE 定义了图形数据的存储格式以及大量的图形算法，可以帮助开发人员快速地进行面向对象程序的开发。目前支持3种操作系统：Linux，Windows NT和Sun Solaris。

3) 快速开发：Open CASCADE 为用户提供了 OCAF（Open CASCADE AppliCation Framework），它是一种基于 Open CASCADE的RAD（Rapid Applieation Development，快速开发工具）框架[9]，是Open CASCADE类库的辅助工具，在几何建模过程中，不仅能处理用到的类库里的任何数据和算法，还可以用于组织、保存应用数据，开发人员可以使用这一工具进行应用程序的快速开发[10]。

4） 类库的丰富性与较流行：Open CASCADE 类库的二维和三维建模、布尔操作、可视化的特性。任何基于Open CASCADE的算法都可以传输到另一个库。从这个角度来看，基于Open CASCADE的算法可以被认为是学术界和产业界交流的原型。

5） 原型框架：Open CASCADE库提供了自己的原型框架，称为Draw Test Harness(或简称Draw)。

6） 非流形模型：Open CASCADE和ACIS利用能够表示非流形模型的Object/CO-Object(例如，EDGE/CO-EDGE)数据结构。

7） 内存管理器：Open CASCADE的内存管理器使用小块和大块的循环使用，当程序需要连续分配和释放大小差不多的内存块时，可以加速程序的执行。实际应用中提升幅度可以高达50%[3]。

* + - * 1. 可改进之处

1) 内存管理器：程序运行时，被循环使用的内存块不能返回到操作系统中，这会导致内存的严重消耗，甚至会被操作系统误认为时内存泄露。为了减少这种影响，在频繁对内存进行操作后，Open CASCADE 将调用Standard::Purge()方法。

2） 布尔操作的局限：Open CASCADE 的布尔操作(Boolean Operations：BOPs)非常的慢。构造函数只是简单的为内部类型分配空间。每一个IntPolyh\_MaillagaAffinage的构造函数为每一个面都生成了10000个点、20000个三角形、30000条边的数组。但通常它们只填充使用了不超过100个元素[11]。

1. 组内分工情况

组内成员贡献度相当，均为33%。

姬轶：搜寻资料，完成软件体系结构设计与分析、设计特色分析、整合文档。

曲卓涵：搜寻资料，完成需求分析，对系统源码进行阅读分析。

刘卓程：搜寻资料，完成软件系统的综述，分析软件整体结构并完成目录树的整理与相关图的绘制。

# 参考文献

[1]OpenCASCADE 简介: https://blog.csdn.net/qq\_22642239/article/details/88342794

[2]OpenCASCADE 官网：[www.openCASCADE.com/](http://www.opencascade.com/)

[3]OpenCASCADE中文帮助文档：https://wenku.baidu.com/view/c11d144b2e3f5727a5e96237.html

[4] Object libraries application framework uses guide [EB/OL]. www.Open CASCADE.org. 2020.

[5]OpenCASCADE学习笔记解析: https://max.book118.com/html/2017/0110/82015109.shtm

[6]Slyadnev S, Malyshev A, Turlapov V. CAD model inspection utility and prototyping framework based on OpenCASCADE[C]//Conference Paper: GraphiCon. 2017.

[7]丁伟.基于Open CASCADE下的点云数据的研究与实现[J].制 造业自动化，2010，32（11）：216⁃218

[8]Yuan G, Zhang Y. Development of 3D modeling platform based on Open CASCADE [J][J]. Journal of Engineering Graphics, 2008, 4: 146-9.

[9]Yuan Y, Wang Y H, Jiang L, et al. Research on modeling technology based on Qt and OpenCASCADE[J]. Modern Electronics Technique, 2013, 10: 74-77.

[10] 陈月凤.基于虚拟环境的工程图学三维建模系统的开发[D]. 济南：山东大学，2010

[11]Open\_CASCADE学习笔记-为什么布尔操作如此之慢：https://wenku.baidu.com/view/14003809f68a6529647d27284b73f242336c31ac.html

# 附件

**附件1:** **体系结构的分解过程**

用文件目录与思维导图表示体系结构的分解过程，其中模块、包和类的详细对应关系