

# Open Cascade DataExchange IGES

[eryar@163.com](mailto:eryar@163.com)

**摘要 Abstract:** 本文结合 OpenCascade 和 Initial Graphics Exchange Specification 来对 IGES 格式进行详细说明，理解 IGES 格式的结构，进而方便对 OpenCascade 中 IGES 格式文件读写的实现进行理解。

**关键字 Key Words:** IGES、Open Cascade、Data Exchange

## 一、引言 Introduction

目前市面上的 CAD 系统都有自己的数据文件，各个系统之间的数据结构和格式各不相同，这样极大影响了设计和制造部门之间或企业之间的数据传输和程序衔接的自动化，同样给 CMM 和 CAD/CAM 的数据通信带来困难。因此，迫切需要数据交换文件格式的标准化。

产品数据的正确交换可以使双方不必重建数据，提高效率。产品数据交换的通常做法是使用数据交换接口，即需要交换的每一系统与标准数据格式（中性格式）之间开发双向转换接口，两系统通过中性格式进行交换。

20 世纪 70 年代末 80 年代初以来，国际上已做了大量数据交换标准的研究、制定工作。自 1980 年最早出现 IGES1.0 后，又产生了许多其他规范如 XBF、SET、VDA-18、PDDI、ESP、CAD\*1、PDES 等。这些规范各有不同的目的和使用范围，同时也存在着各种问题。在这些规范中，应用最成熟、最广泛的是 IGES (Initial Graphics Exchange Specification)。ISO 在上述规范的基础上，制定了新的数据交换标准 STEP (Standard for the Exchange of Product Model data)，其目的是覆盖过去已经存在的规范的功能，并解决实际应用中的问题。

IGES 是在美国国家标准局的倡导下，由美国国家标准协会 (ANSI) 组织波音公司、通过公司等共同商议制订的，1980 年公布第一版，以后不断完善和扩充，版本不断升级，从 IGES1.0 仅包含二维图形到 IGES3.0 支持曲面和三维线框几何，IGES4.0 版支持构造实体几何 (CSG) 造型技术，IGES5.0 支持边界表示 (BRep) 造型技术。目前几乎所有有影响的 CAD/CAM 系统都配有 IGES 接口，如 Pro/E、CATIA、OpenCascade 等。它由一系列产品的几何、绘图、结构和其他信息组成，可以处理 CAD/CAM 系统中的大部分信息，是用来定义产品几何形状的现代交互图形标准。

本文结合 OpenCascade 和 Initial Graphics Exchange Specification 来对 IGES 格式进行详细说明，理解 IGES 格式的结构，进而方便对 OpenCascade 中 IGES 格式文件读写实现的方式进行理解。

## 二、IGES 规范 the Initial Graphics Exchange Specification

IGES 是一种按特定的文件结构安排的数据格式，用来描述产品的设计和生 产信息，可用它来交换 CAD/CAM 系统中以计算机可读的形式产生和存储的各种信息。标准的 IGES 文件包含固定长 ASCII 码（Fixed Format）、压缩的 ASCII 码（Compressed Format）和二进制（Binary Format）三种格式。

固定长 ASCII 码的 IGES 文件每行为 80 个字符，整个文件分为五段（Section）：开始段（Start）、全局段（Global）、元素索引段（Directory Entry）、参数数据段（Parameter Data）和结束段（Terminate）。如图 2.1 所示：

Section name	Col. 73 Letter Code
Start	S
Global	G
Directory Entry	D
Parameter Data	P
Terminate	T

Figure 2.1 Fixed File Formats of ASCII IGES File

段标识符位于每行的第 73 列，74~80 列指定为用于段每行的序号。序号都以 1 开始，且连续不间断，其值对应应该段的列数。段标识符是这样规定的，字符“B”和“C”表示标记段；“S”表示开始段；“G”表示全局段；“D”表示元素索引段；“P”表示参数记录参数记录段；“T”表示结束段。固定长的 ASCII 格式的 IGES 格式的大概结构如下图 2.2 所示：

1	8 9	16 17	24 25	32 33	40 41	48 49	56 57	64 65	72	73	80
<b>Start Section</b> – a human readable prologue to the file.										S0000001	
It contains one or more lines										S0000002	
⋮										⋮	
using ASCII characters in columns 1–72.										S000000N	
<b>Global Section</b> – sending system and file information.										G0000001	
It contains the number of lines needed to hold the parameter fields, separated by										G0000002	
⋮										⋮	
parameter delimiters, and terminated by one record delimiter, in columns 1–72.										G000000N	
<b>Directory Entry Section</b> - contains one pair of lines for each entity										D0000001	
Directory entry fields 1-9 in nine 8-column-wide fields										D0000002	
Directory entry fields 10-18 in nine 8-column-wide fields											
<b>Parameter Data Section</b> – values and parameter delimiters terminated by one record delimiter, in columns 1-64; column 65 is unused									DE back Pointer	P0000001	
S0000020 G0000003D0000500 P0000261									<b>Terminate Section</b> – record counts for preceding sections; columns 33–72 unused	T0000001	

Figure 2.2 General file structure of the Fixed Format

下面结合具体的 IGES 文件来对每个段进行说明，仅包含一个点的示例文件如下所示：

	1	2	3	4	5	6	7	8
1	.....	.....	.....	.....	.....	.....	.....	S0000001
2	,,31HOpen	CASCADE	IGES	processor	6.6,13H	Filename.iges,	.....	G0000001
3	16HOpen	CASCADE	6.6,31HOpen	CASCADE	IGES	processor	6.6,32,308,15,308,15,	G0000002
4	,1.,2,2HMM,	1,0.01,15H20140104.192326,	5E-008,3.,5Heryar,,11,0,	.....	.....	.....	.....	G0000003
5	15H20140104.192326,;	.....	.....	.....	.....	.....	.....	G0000004
6	.....116	.....1	.....0	.....0	.....0	.....0	.....0	00000000D0000001
7	.....116	.....0	.....0	.....1	.....0	.....0	.....0	0D0000002
8	116,1.,2.,3.,0;	.....	.....	.....	.....	.....	.....	0000001P0000001
9	S	.....1G	.....4D	.....2P	.....1	.....	.....	T0000001
10								

Figure 2.3 IGES demo file Generated by OpenCascade

**开始段 Start Section:** 该段是为提供一个可读文件的序言，主要记录图形文件的最初来源及生成该 IGES 文件的相同名称。IGES 文件至少有一个开始记录。

**全局段 Global Section:** 主要包含前处理器的描述信息及为处理该文件的后处理器所需要的信息。参数以自由的格式输入，用逗号分隔参数，用分号结束一个参数。主要参数有文件名、前处理器版本、单位、文件生成日期、作者姓名及单位、IGES 的版本、绘图标准代码等。如图 2.3 中的 2~5 行。

**元素索引段 Directory Entry Section:** 该段主要为文件提供一个索引，并含有每个实体的属性信息，文件中每个实体都有一个目录条目，大小一样，由 8 个字符组成一个域，共 20 个域，每个条目占用两行。如图 2.4 所示：

1	8	9	16	17	24	25	32	33	40	41	48	49	56	57	64	65	72	73	80
(1) Entity Type Number #	(2) Para- meter Data ⇒	(3) Structure #,⇒	(4) Line Font Pattern #,⇒	(5) Level #,⇒	(6) View 0,⇒	(7) Transfor- mation Matrix 0,⇒	(8) Label Display Assoc. 0,⇒	(9) Status Number #	(10) Sequence Number D #	(11) Entity Type Number #	(12) Line Weight Number #	(13) Color Number #,⇒	(14) Para- meter Line Count #	(15) Form Number #	(16) Reserved	(17) Reserved	(18) Entity Label	(19) Entity Subscript Number #	(20) Sequence Number D # + 1

**Nomenclature:**  
 (n) - Field number n  
 # - Integer  
 ⇒ - Pointer  
 #,⇒ - Integer or pointer (pointer is negated)  
 0,⇒ - Zero or pointer

Figure 2.4 Format of the Directory Entry(DE) Section in the Fixed Format

每一项内容如下：

- 元素类型号 (Entity Type Number): 说明元素的类型；
- 参数指针 (Parameter Data): 说明该元素的参数在参数数据段的开始序号；
- 图层: 存放图层名和它的指针；
- 参数记录数 (Parameter Line Count): 数字代表该元素的参数在参数数据段的行数；

**参数数据段 Parameter Data Section:** 该段记录了每个元素的几何数据。根据每个图形元素参数数的多少，决定它将有几行。如图 2.3 所示，116 为点，它有三个数据 X, Y 和 Z, 分别为 1, 2, 3。

**结束段 Terminate Section:** 只占一行，在前 32 个字符里，分别用 8 个字符记录了开始段、全局段、索引段和参数数据段的段码和每段的总行数。第 33~72 个字符没有用到。最后 8 个字符为结束段的段码和行数。

### 三、OpenCascade 中 IGES 读写 OpenCascade IGES Reader/Writer

OpenCascade 中提供 IGES 的程序接口有如下功能：

- 加载 IGES 文件到内存；loading IGES files into memory;
- 检查 IGES 文件的一致性；checking IGES files consistency;
- 转换 IGES 文件到 OCCT 的形状；translating IGES files into OCCT shapes;
- 转换 OCCT 的形状到 IGES 文件；translating OCCT shapes into IGES files;
- 访问 IGES 文件模型；accessing the IGES model;
- 选择 IGES 文件模型中的实体；selecting entities from the IGES model;
- 访问 IGES 文件模型中的每个实体；accessing each entity in the IGES model;

实现上述功能的包是 IGESControl，可处理的 IGES 文件格式的版本为 5.3。读取 IGES 文件实体到 OCCT 形状的方法如下所示：

```
Standard_Integer ReadIGES(const Standard_CString& aFileName,
                          Handle(TopTools_HSequenceOfShape)& aHSequenceOfShape)
{
    IGESControl_Reader Reader;

    Standard_Integer status = Reader.ReadFile(aFileName);

    if (status != IFSelect_RetDone)
    {
        return status;
    }

    Reader.TransferRoots();

    TopoDS_Shape aShape = Reader.OneShape();
    aHSequenceOfShape->Append(aShape);

    return status;
}
```

将 OCCT 形状转换为 IGES 文件方法如下所示：

```
Standard_Boolean SaveIGES(const Standard_CString& aFileName,
                           const Handle(TopTools_HSequenceOfShape)&
aHSequenceOfShape)
{
    IGESControl_Controller::Init();
    IGESControl_Writer ICW (Interface_Static::CVal("XSTEP.iges.unit"),
                             Interface_Static::IVal("XSTEP.iges.writebrep.mode"));

    for (Standard_Integer i=1;i<=aHSequenceOfShape->Length();i++)
    {
        ICW.AddShape (aHSequenceOfShape->Value(i));
    }

    ICW.ComputeModel();
    Standard_Boolean result = ICW.Write(aFileName);
    return result;
}
```

#### 四、程序示例 Example Code

通过将 OpenCascade 中的形状数据导出到 IGES 文件，结合 IGES 规格书，可以方便对 IGES 格式的理解。如下程序所示为将简单的点、线和圆导出到 IGES 文件：

```
/*
 *   Copyright (c) 2014 eryar All Rights Reserved.
 *
 *   File : Main.cpp
 *   Author : eryar@163.com
 *   Date : 2014-01-04 20:00
 *   Version : 1.0v
 *
 *   Description : Export OpenCascade shape to IGES entities.
 *
 *   Key Words : OpenCascade, IGES, DataExchange
 */

#define WNT
#include <gp_Pnt.hxx>
#include <gp_Circ.hxx>

#include <TopoDS_Vertex.hxx>
#include <TopoDS_Edge.hxx>

#include <TopTools_ListIteratorOfListOfShape.hxx>

#include <BRepBuilderAPI_MakeVertex.hxx>
#include <BRepBuilderAPI_MakeEdge.hxx>

#include <IGESControl_Controller.hxx>
#include <IGESControl_Writer.hxx>

#pragma comment(lib, "TKernel.lib")
#pragma comment(lib, "TKMath.lib")
#pragma comment(lib, "TKBRep.lib")
#pragma comment(lib, "TKTopAlgo.lib")
#pragma comment(lib, "TKIGES.lib")

void SaveIges(const std::string& igesFile, const TopTools_ListOfShape& shapes)
{
    IGESControl_Controller::Init();
    IGESControl_Writer igesWriter;

    for (TopTools_ListIteratorOfListOfShape si(shapes); si.More(); si.Next())
    {
        igesWriter.AddShape(si.Value());
    }

    igesWriter.ComputeModel();
    igesWriter.Write(igesFile.c_str());
}

int main(void)
```

```

{
  // vertices.
  TopTools_ListOfShape vertices;
  TopoDS_Vertex theVertex = BRepBuilderAPI_MakeVertex(gp_Pnt(1.0, 2.0, 3.0));

  vertices.Append(theVertex);

  SaveIges("vertex.iges", vertices);

  // edges.
  TopTools_ListOfShape edges;
  TopoDS_Edge theLine = BRepBuilderAPI_MakeEdge(gp_Pnt(3.0, 4.0, 5.0), gp_Pnt(6.0, 7.0,
8.0));
  TopoDS_Edge theCircle = BRepBuilderAPI_MakeEdge(gp_Circ(gp::XOY(), 9.0));

  edges.Append(theLine);
  edges.Append(theCircle);

  SaveIges("edge.iges", edges);

  // put them together.
  TopTools_ListOfShape shapes;
  shapes.Append(theVertex);
  shapes.Append(theLine);
  shapes.Append(theCircle);

  SaveIges("shapes.iges", shapes);

  return 0;
}

```

生成的 IGES 文件内容如下所示:

```

1 .....S0000001
2 ,,31HOpen CASCADE IGES processor 6.6,13HFilename.iges, .....G0000001
3 16HOpen CASCADE 6.6,31HOpen CASCADE IGES processor 6.6,32,308,15,308,15,G0000002
4 ,1.,2,2HMM,1,0.01,15H20140104.211131,8.33333E-008,9.74153,5Heryar,,11,0,G0000003
5 15H20140104.211131,; .....G0000004
6 .....116 .....1 .....0 .....0 .....0 .....0 .....0 .....0 .....00000000D0000001
7 .....116 .....0 .....0 .....1 .....0 .....0 .....0 .....0 .....0 .....0D0000002
8 .....110 .....2 .....0 .....0 .....0 .....0 .....0 .....0 .....0 .....0D0000003
9 .....110 .....0 .....0 .....0 .....1 .....0 .....0 .....0 .....0 .....0D0000004
10 .....100 .....3 .....0 .....0 .....0 .....0 .....0 .....0 .....0 .....0D0000005
11 .....100 .....0 .....0 .....1 .....0 .....0 .....0 .....0 .....0 .....0D0000006
12 116,1.,2.,3.,0; .....0000001P0000001
13 110,3.,4.,5.,6.,7.,8.; .....0000003P0000002
14 100,0.E+000,0.E+000,0.E+000,9.,0.E+000,9.,0.E+000; .....0000005P0000003
15 S .....1G .....4D .....6P .....3 ..... ..... .....T0000001
16

```

Figure 4.1 Simple Fixed Format IGES File

结合下图 4.2 中实体类型号与实体类型的对应表可知, 上图 4.1 中的元素索引 D 段可以看总共有三个实体, 一个是类型号为 116 的点; 一个是类型号为 110 的直线; 一个是类型号为 100 的圆。

从数据参数 P 段可以看出 116 对应的点的坐标为 (1, 2, 3); 110 对应的直线的两个端点坐标为 (3, 4, 5) 和 (6, 7, 8); 100 对应的圆的参数数据项为圆心在 (0, 0) 半径为 9 的位于 XY 平面上的圆。圆的参数数据项每项数据说明见图 4.3。

将 IGES 在 OpenCascade 中显示如图 4.4 所示。

Entity Type Number	Entity Type
100	Circular Arc
102	Composite Curve
104	Conic Arc
106	Copious Data
106/11	2D Linear Path
106/12	3D Linear Path
106/63	Simple Closed Planar Curve
108	Plane
110	Line
112	Parametric Spline Curve
114	Parametric Spline Surface
116	Point
118	Ruled Surface
120	Surface of Revolution
122	Tabulated Cylinder
124	Transformation Matrix
125	Flash
126	Rational B-Spline Curve
128	Rational B-Spline Surface
130	Offset Curve
140	Offset Surface
141	Boundary
142	Curve on a Parametric Surface
143	Bounded Surface
144	Trimmed Parametric Surface
190	Plane Surface
192	Right circular cylindrical Surface
194	Right Circular Conical Surface
196	Spherical Surface
198	Toroidal Surface

Figure 4.2 Curve and Surface Entities

Parameter Data			
Index	Name	Type	Description
1	ZT	Real	Parallel ZT displacement of arc from X <sub>T</sub> , Y <sub>T</sub> plane
2	X1	Real	Arc center abscissa
3	Y1	Real	Arc center ordinate
4	X2	Real	Start point abscissa
5	Y2	Real	Start point ordinate
6	X3	Real	Terminate point abscissa
7	Y3	Real	Terminate point ordinate

Figure 4.3 Parameter Data of Circle Entity in IGES

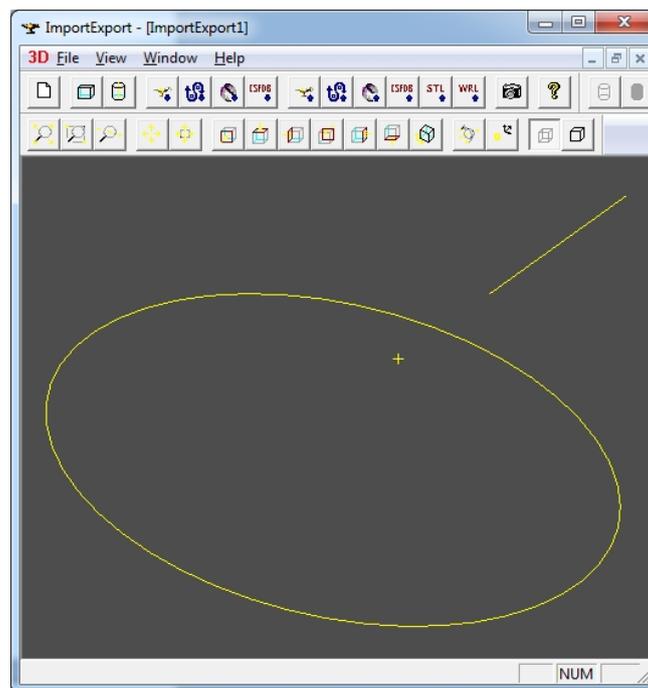


Figure 4.4 IGES Entities in OpenCascade

## 五、结论 Conclusion

结合 OpenCascade 生成的 IGES 文件和 Initial Graphics Exchange Specification 来对 IGES 格式进行学习。

虽然目前许多 CAD/CAM 系统主要应用是在不同系统间交换工程图纸、零件模型、运动仿真和动态实验所需要的几何数据，装配或销售产品所需要的图形文件，但是 IGES 标准在具体应用中经常出现以下问题：

- 交换过程中会出现错误或数据丢失现象。由于 IGES 所支持的实体十分庞大，在实际应用中，每一个 CAD/CAM 系统不可能实现 IGES 支持的所有实体，而只能实现它所涉及的实体的交换，即只能实现 IGES 实体集的一个子集，但不同系统所实现的子集不可能完全相同。系统之间只能交换所实现子集的交集部分。
- 发送系统前处理器生成的 IGES 文件的实体类型超出了接收系统后处理器可能接受的实体类型范围，以及发送系统与接收系统之间实体类型集虽相同，但有关属性如颜色、字体、线型等代码不同，也将使部分数据丢失。
- IGES 的目的只是传输工程图形及相应信息，它无法描述 CIMS 环境所需要的产品数据的全部信息。而且 IGES 对实体的定义是模糊的，这将造成各 CAX 系统对 IGES 中定义的实体有不同的理解，如相同的 3D 曲线功能会因不同软件的计算方式不同而使在使用 IGES 来执行图形交换时出现不同的曲线或丢失该曲线。
- IGES 文件本身规模太大，影响数据文件的处理速度，使数据传输效率不高。
- 在转换数据的过程中发生的错误很难确定，常要人工去处理 IGES 文件，如在转换数据的过程中经常发生某个或某几个小曲面丢失的情况，这是要利用原有曲面边界重新生成曲面；某些小曲面（Face）在转换过程中变成大曲面（Surface），此时要对曲面进行裁剪。

## 六、参考资料 References

1. 李原，张开富，余剑锋. 计算机辅助几何设计技术及应用. 西北工业大学出版社. 2007
2. US PRO. Initial Graphics Exchange Specification .1997
3. OpenCascade. Data Exchange IGES Format User's Guide. 2013