OpenNURBS to OpenCASCADE

eryar@163.com

Abstract. The OpenNURBS initiative provides CAD/CAM/CAE and computer graphics software developers the tools to accurately transfer 3D geometry between applications. The OpenNURBS C++ source code is clean and fairly simple. The OpenNURBS Toolkit is intended for use by C++ and .NET programmers. The toolkit includes complete source code to create a library that will read and write 3dm files. OpenCASCADE providing services for 3D surface and solid modeling, CAD data exchange, and visulization. Most of OCCT functionality is available in the form of C++ libraries. If can convert OpenNURBS curve/surface to OpenCASCADE, then it will fill the gap of data exchange between OpenNURBS and OpenCASCADE.

Key Words. OpenCASCADE, OpenNURBS, NURBS, 3DM, DataExchange

1. Introduction

OpenNURBS 旨在为 CAD、CAM、CAE 与计算机图形软件开发人员提供一个在不同的软件间精确转换 3D 几何的工具。OpenNURBS 所提供的功能包括：

- 可以读、写 3DM 文件格式的 C ++原代码库，支持目前的 Microsoft、Apple 与 Gnu for Windows、Windows X64、Mac 与 Linux 编绎器...
- 可以读、写 3DM 文件格式的.NET 源代码库。
- 品质保证与版本控制。
- 技术支持。

与其他开源产品的开发平台不同之处：

- 鼓励使用在商业用途。
- 提供免费的开发工具与技术支持。
- 无任何约束，不受版权与公共版权（copyleft ）条款约束。
- 鼓励但不强迫用户分享自己的研发成果。

NURBS 是当前流行几何造型系统中通用的曲线曲面表达形式，OpenNURBS 开源库提供了 NURBS 曲线曲面的数据表达，但缺少一些造型算法，而 OpenCASCADE 中提供了大量的造型算法。通过将 OpenNURBS 中的曲线曲面转换到 OpenCASCADE 中，就可以将 OpenNURBS 和 OpenCASCADE 结合起来使用了。

OpenNURBS 中的代码简洁，易懂；OpenCASCADE 的造型功能强大。所以先将 OpenNURBS 中的曲线曲面转换到 OpenCASCADE 中，为 OpenCASCADE 的 3DM 格式的数据交换打下基础。

2. Convert OpenNURBS Curve

OpenNURBS 中统一使用 NURBS 方式来表达曲线和曲面。OpenCASCADE 中对应 NURBS 曲线的类为 Geom_BSplineCurve。因此，可以将 OpenNURBS 的曲线转换到 OpenCASCADE 中。转换时需要注意节点矢量 Knot Vector 的方式有很大的不同。OpenNURBS 中的节点矢量是所有的节点值，包括了重节点，而且在节点矢量的首尾还少了两个节点。OpenCASCADE 中的节点矢量由不包含重复节点的节点矢量及其重数来构造。其它数据基本相同，将转换曲线的示例代码列出如下：

```
/**
* @breif Convert OpenNURBS NURBS curve to OpenCASCADE Geom_BSplineCurve.
* @param [in] theCurve opennurbs nurbs curve;
* @param [in] theBRepFile the curve is in brep file of opencascade;
* @note pay attention to the knots of opennurbs nurbs curve/surface.
*/
void ConvertCurve(const ON_NurbsCurve& theCurve, const std::string& theBRepFile)
{
    TColgp_Array1OfPnt aPoles(1, theCurve.CVCount());
    TColStd_Array1OfReal aWeights(1, theCurve.CVCount());

    TColStd_Array1OfReal aKnotSequence(1, theCurve.KnotCount() + 2);

    bool IsRational = theCurve.IsRational();
    bool IsPeriodic = (theCurve.IsPeriodic()) ? true: false;

    // Control point and its weight.
    for (int i = 0; i < theCurve.CVCount(); ++i)
    {
        if (IsRational)
        {
            ON_4dPoint aPole;

            theCurve.GetCV(i, aPole);

            aPoles.SetValue(i + 1, gp_Pnt(aPole.x / aPole.w, aPole.y / aPole.w,
aPole.z / aPole.w));
            aWeights.SetValue(i + 1, aPole.w);
        }
        else
        {
            ON_3dPoint aPole;

            theCurve.GetCV(i, aPole);

            aPoles.SetValue(i + 1, gp_Pnt(aPole.x, aPole.y, aPole.z));
        }
    }

    // Knot vector and its multiplicity.
    for (int i = 0; i < theCurve.KnotCount(); ++i)
    {
```

```
        aKnotSequence.SetValue(i + 2, theCurve.Knot(i));
    }

    aKnotSequence.SetValue(aKnotSequence.Lower(), theCurve.Knot(0));
    aKnotSequence.SetValue(aKnotSequence.Upper(),
theCurve.Knot(theCurve.KnotCount() - 1));

    TColStd_Array1OfReal     aKnots(1,     BSplCLib::KnotsLength(aKnotSequence,
IsPeriodic));
    TColStd_Array1OfInteger aMultiplicities(1, aKnots.Upper());

    BSplCLib::Knots(aKnotSequence, aKnots, aMultiplicities);

    Handle_Geom_BSplineCurve aBSplineCurve = new Geom_BSplineCurve(
        aPoles, aWeights, aKnots, aMultiplicities,
        theCurve.Degree(), theCurve.IsPeriodic());

    GeomTools_CurveSet::PrintCurve(aBSplineCurve, std::cout);

    TopoDS_Edge anEdge = BRepBuilderAPI_MakeEdge(aBSplineCurve);

    BRepTools::Write(anEdge, theBRepFile.c_str());
}
```

程序先转换控制顶点，若是有理曲线，还要设置控制顶点的权值 w。在转换节点矢量时，使用了 BSplCLib::Knots()函数将 OpenNURBS 的节点矢量转换为 OpenCASCADE 的形式，并补齐了缺少的端部的两个节点。最后为了在 OpenCASCADE 中显示转换结果，将曲线生成 TopoDS_Edge 并保存为 BRep 格式，方便在 Draw Test Harness 中查看结果如下图所示：

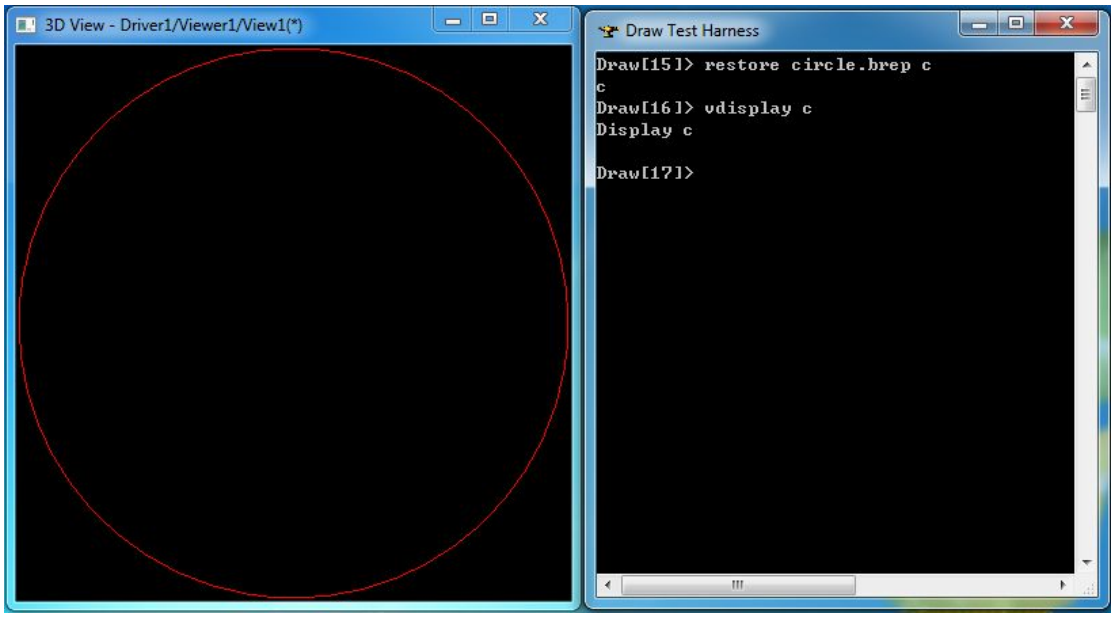

Figure 2.1 OpenNURBS nurbs circle in OpenCASCADE Draw Test Harness

3. Convert OpenNURBS Surface

OpenNURBS 曲面的转换同样也需要注意 u, v 方向上的节点矢量的问题，和曲线的转换类似。除此之外就是把相同的属性分别赋值即可，列出代码如下所示：

```
/**
* @breif Convert OpenNURBS NURBS surface to OpenCASCADE Geom_BSplineSurface.
* @param [in] theSurface opennurbs nurbs surface;
* @param [in] theBRepFile the surface is in the brep file of opencascade;
* @note pay attention to the knots of opennurbs nurbs curve/surface.
*/
void ConvertSurface(const ON_NurbsSurface& theSurface, const std::string&
theBRepFile)
{
    TColgp_Array2OfPnt aPoles(1, theSurface.CVCount(0), 1,
theSurface.CVCount(1));
    TColStd_Array2OfReal aWeights(1, theSurface.CVCount(0), 1,
theSurface.CVCount(1));

    TColStd_Array1OfReal aUKnotSequence(1, theSurface.KnotCount(0) + 2);
    TColStd_Array1OfReal aVKnotSequence(1, theSurface.KnotCount(1) + 2);

    bool IsRational = theSurface.IsRational();
    bool IsUPeriodic = (theSurface.IsPeriodic(0)) ? true: false;
    bool IsVPeriodic = (theSurface.IsPeriodic(1)) ? true: false;

    // control point and its weight.
    for (int i = 0; i < theSurface.CVCount(0); ++i)
    {
        for (int j = 0; j < theSurface.CVCount(1); ++j)
        {
            if (IsRational)
            {
                ON_4dPoint aPole;

                theSurface.GetCV(i, j, aPole);

                aPoles.SetValue(i + 1, j + 1, gp_Pnt(aPole.x / aPole.w, aPole.y /
aPole.w, aPole.z / aPole.w));
                aWeights.SetValue(i + 1, j + 1, aPole.w);
            }
            else
            {
                ON_3dPoint aPole;

                theSurface.GetCV(i, j, aPole);

                aPoles.SetValue(i + 1, j + 1, gp_Pnt(aPole.x, aPole.y, aPole.z));
            }

        }
```

```
    }

    // Knot vector and its multiplicity.
    for (int i = 0; i < theSurface.KnotCount(0); ++i)
    {
        aUKnotSequence.SetValue(i + 2, theSurface.Knot(0, i));
    }

    aUKnotSequence.SetValue(aUKnotSequence.Lower(), theSurface.Knot(0, 0));
    aUKnotSequence.SetValue(aUKnotSequence.Upper(),          theSurface.Knot(0,
theSurface.KnotCount(0) - 1));

    TColStd_Array1OfReal    aUKnots(1,    BSplCLib::KnotsLength(aUKnotSequence,
IsUPeriodic));
    TColStd_Array1OfInteger aUMults(1, aUKnots.Upper());

    BSplCLib::Knots(aUKnotSequence, aUKnots, aUMults);

    for (int j = 0; j < theSurface.KnotCount(1); ++j)
    {
        aVKnotSequence.SetValue(j + 2, theSurface.Knot(1, j));
    }

    aVKnotSequence.SetValue(aVKnotSequence.Lower(), theSurface.Knot(1, 0));
    aVKnotSequence.SetValue(aVKnotSequence.Upper(),          theSurface.Knot(1,
theSurface.KnotCount(1) - 1));

    TColStd_Array1OfReal    aVKnots(1,    BSplCLib::KnotsLength(aVKnotSequence,
IsVPeriodic));
    TColStd_Array1OfInteger aVMults(1, aVKnots.Upper());

    BSplCLib::Knots(aVKnotSequence, aVKnots, aVMults);

    Handle_Geom_BSplineSurface aBSplineSurface = new Geom_BSplineSurface(aPoles,
        aWeights, aUKnots, aVKnots, aUMults, aVMults,
        theSurface.Degree(0), theSurface.Degree(1),
        IsUPeriodic, IsVPeriodic);

    GeomTools_SurfaceSet::PrintSurface(aBSplineSurface, std::cout);

    TopoDS_Face      aFace      =      BRepBuilderAPI_MakeFace(aBSplineSurface,
Precision::Confusion());

    BRepTools::Write(aFace, theBRepFile.c_str());
}
```

转换过程与曲线的类似，只是多了 v 方向上的处理。以上两个函数分别实现曲线曲面的转换，完整程序可以在本文后面的链接中下载。以圆柱面和球面为例，测试了一下转换效果，在 Draw Test Harness 中显示结果如下图所示：

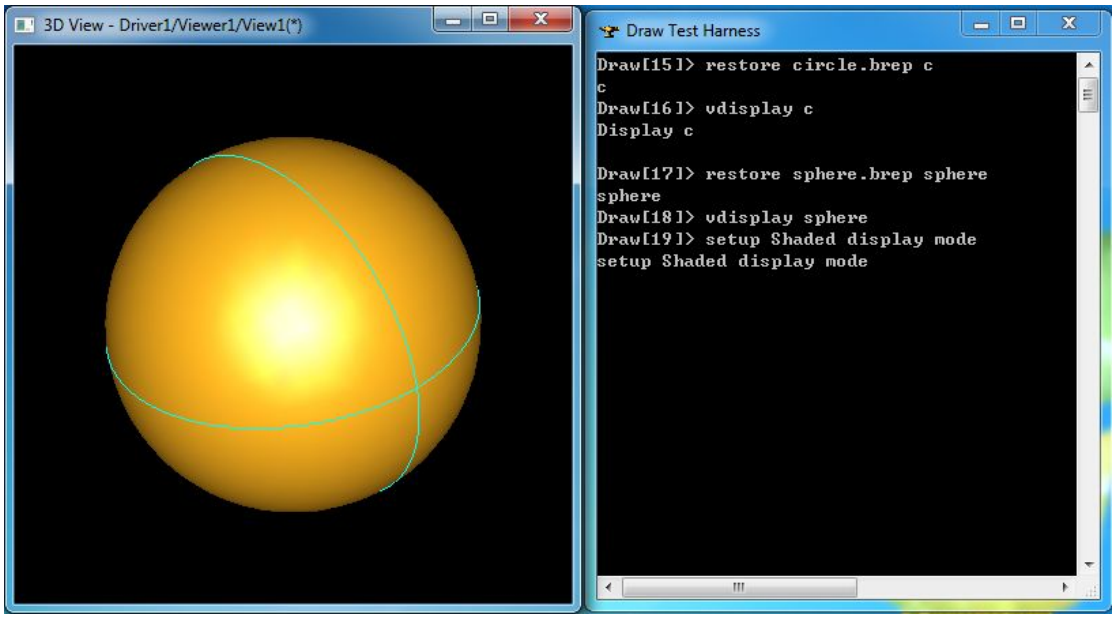

Figure 3.1 OpenNURBS nurbs sphere in OpenCASCADE Draw Test Harness

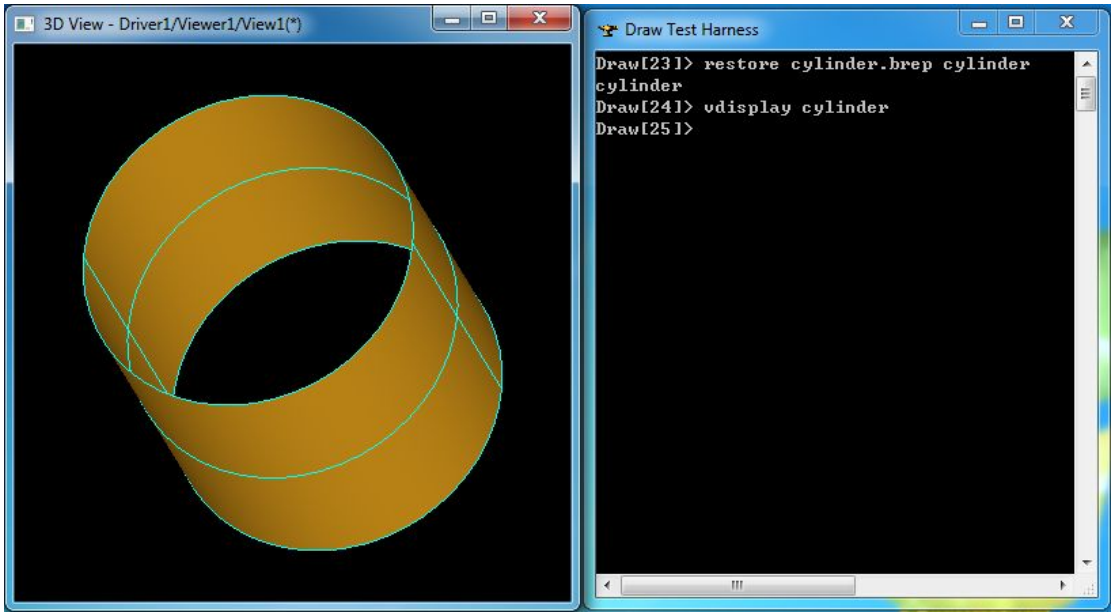

Figure 3.2 OpenNURBS nurbs cylinder in OpenCASCADE Draw Test Harness

由上图可知，OpenCASCADE 中的 Geom_BSplineCurve/Geom_BSplineSurface 可以完美表示 OpenNURBS 中的曲线曲面，不过有最高次数为 25 次的限制。

4. Conclusion

现可看出，OpenNURBS 中的曲线曲面可以转换成 OpenCASCADE 中。不过在转换时需要注意 OpenNURBS 的节点矢量数据与教科书中及 OpenCASCADE 中不同，需要做些处理。还有需要注意的事项就是 OpenNURBS 中数组是从 0 开始的，而在 OpenCASCADE 中可以指定为从 1 开始。

将 OpenNURBS 的曲线曲面转换成 OpenCASCADE 的曲线曲面后，下一步准备将 OpenNURBS 中的 BRep 体也转换到 OpenCASCADE 中，从而对 BRep 表示形式做进一步的学习。

5. References

1. OpenNURBS Web: http://www.rhino3d.com/cn/opennurbs
2. Les Piegl, Wayne Tiller. The NURBS Book. Springer-Verlag. 1997
3. 赵罡，穆国旺，王拉柱译. 非均匀有理 B 样条. 清华大学出版社. 2010