

イラスト作成支援ソフト ぷりスカ (R) ユーザーズマニュアル (ぷりスカ3Dモデル作成編)

® 商標登録 製作・著作 © 2021
オカダ・システムエンジニアリング研究所

1. 概要

本書では、イラスト作成支援ソフト ぷりスカ 用の「ぷりスカ3Dモデル」をユーザーが作成する方法を解説します。

2. フォーマット

三角錐型の非常に簡単なモデルのXMLファイルの記述内容を示します。
このファイルは、当研究所のホームページからダウンロードできます。

```
<?xml version="1.0" encoding="utf-8"?>  
<ぷりスカ保存用10 xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

```
<maker>作者</maker>
```

```
<name>テント</name> 登山用の一人用テントの形？
```

```
<content>評価用サンプル</content>
```

```
<view保存情報>-13.1648,0.1619,6.8127,0.0000,0.0000,0.0000,11.5098,1.3945,22.2083,-0.3983,25.0122,-  
1.3642,5.0000,0.0000,0.0000,0.0000,5.0000,0.0000,0.0000,0.0000,9.0000,-22.15,0.00,-  
9.23,72.33,0.0,80.0,0.63</view保存情報>
```

```
<線2D>
```

```
<L2D>
```

```
<tp1>1</tp1>
```

```
<tp2>0</tp2>
```

```
<qix>0</qix>
```

```
<cix>0</cix>
```

```
<zix>1</zix>
```

```
</L2D>
```

```
<L2D>
```

```
<tp1>1</tp1>
```

```
<tp2>0</tp2>
```

```
<qix>0</qix>
```

```
<cix>0</cix>
```

```
<zix>2</zix>
```

```
</L2D>
```

```
<L2D>
```

```
<tp1>0</tp1>
```

```
<tp2>1</tp2>
```

```
<qix>0</qix>
```

```
<cix>4</cix>
```

```
<zix>3</zix>
```

```
</L2D>
```

```
<L2D>
```

```
<tp1>1</tp1>
```

```
<tp2>0</tp2>
```

```
<qix>1</qix>
```

```
<cix>0</cix>
```

```
<zix>2</zix>
```

```
</L2D>
```

線に関する情報 (接続する点の情報)

1本目

tp1: 線のタイプ 1で直線 0でベジエ曲線

tp2: 線の目的 0で実線 1で補助線

起点のハンドルポイント(点のことです。)インデックス

ベジエ曲線制御点の点インデックス

終点の点インデックス

2本目 以下繰り返し

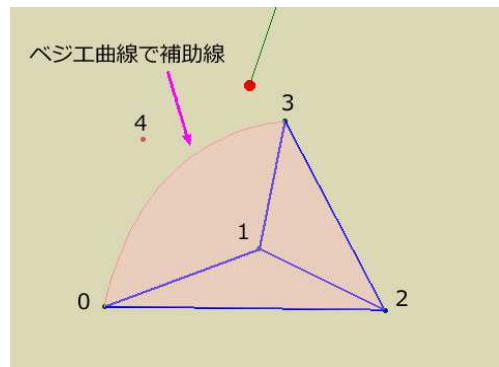
3本目

ベジエ曲線である

補助線である

制御点のインデックスは4

4本目



↑ モデルの形はこうなります。

<pre> <L2D> <tp1>1</tp1> <tp2>0</tp2> <qix>1</qix> <cix>0</cix> <zix>3</zix> </L2D> <L2D> <tp1>1</tp1> <tp2>0</tp2> <qix>2</qix> <cix>0</cix> <zix>3</zix> </L2D> </線2D> <po3D> <HP> <p3> <X>-2.86</X> <Y>-3.98</Y> <Z>-1.08</Z> </p3> <tp>0</tp> <ch>0</ch> </HP> <HP> <p3> <X>2.26</X> <Y>-3.85</Y> <Z>-0.70</Z> </p3> <tp>0</tp> <ch>0</ch> </HP> <HP> <p3> <X>0</X> <Y>-4</Y> <Z>3</Z> </p3> <tp>0</tp> <ch>0</ch> </HP> <HP> <p3> <X>0.22</X> <Y>-0.68</Y> <Z>0.70</Z> </p3> <tp>0</tp> <ch>0</ch> </HP> <HP> <p3> <X>-2.63</X> <Y>-1.01</Y> <Z>-0.63</Z> </p3> <tp>0</tp> <ch>0</ch> </HP> </po3D> </pre>	<p>5本目</p> <p>6本目</p> <p>点に関する情報 最初の点 インデックス0 ←0から始まる</p> <p>X座標 Y座標 Z座標</p> <p>ここには0を入力すること 系統 このモデルでは0のみを使用</p> <p>点 インデックス1</p> <p>点 インデックス2</p> <p>点 インデックス3</p> <p>点 インデックス4 ベジエ曲線の制御点として使用されている</p>
---	---

<除外多角形>
<string>0,2,3</string>
</除外多角形>
</ぷりスカ保存用10>

点0, 2, 3で構成される三角形は除外しています。

フォーマットの解説

3行目の `<maker>作者</maker>` は`<maker></maker>`で囲まれた部分に作成者を記述して下さい。

4行目の `<name></name>`、5行目の `<content></content>`にはモデルに関するコメントを記述することができます。

5行目の `<view保存情報>…</view保存情報>`の内容は非公開とします。内容を変えないで下さい。記述されている情報は視点の位置などです。

その下の `<線2D>…</線2D>`タグはハンドルポイントを連結する線の情報です。

線は1本ごとに `<L2D>…</L2D>`のタグで情報を記述します。

`<tp1>…</tp1>` は線のタイプで、0で直線 1でベジエ曲線(2次ベジエ曲線)となります。

`<tp2>…</tp2>` は線の使用目的で、0で実線 1で補助線となります。

`<qix>…</qix>` は起点のハンドルポイントインデックスです。(インデックスについては下記参照)

`<cix>…</cix>` はベジエ曲線の制御点のハンドルポイントインデックスです。

`<zix>…</zix>` は終点のハンドルポイントインデックスです。

* ハンドルポイントインデックスとは、後述する`<po3D>`タグでのハンドルポイントの情報の記述で、登録されたハンドルポイントの順番で0から始まる連続した整数です。

* 起点、終点の区別はありません。

* 直線を記述する場合、ベジエ曲線の制御点の指定は不要です。(ダミーで入れる数値は0を推奨します。)

次の `<po3D>…</po3D>`タグはハンドルポイントの情報です。

ハンドルポイントは1点ごとに `<HP>…</HP>`のタグで情報を記述します。

(1) ハンドルポイントの3D座標 (例)

```
<p3>
  <X>-2.86</X>
  <Y>45.7369327545166</Y>
  <Z>0.01</Z>
</p3>
```

上の様に計5行を記述して下さい。数値部分はx,y,zの座標です。精度は小数点以下13桁まで入力できます。座標の絶対値は50以内に抑えて下さい。大き過ぎるとズームアウトしても全体が表示されなくなります。

(2) ハンドルポイントのタイプ

`<tp>0</tp>` ←この様に記述して下さい。数値は0に固定して下さい。

(3) 系統

`<ch>0</ch>` ←この様に記述して下さい。数値が系統になります。系統は0から整数で連続的に入力して下さい。

系統についてはユーザーズマニュアルの5-7章を参照して下さい。

* `<HP>…</HP>`のタグが記述された順番(上から下に向けて0から始まる連続した整数)がハンドルポイントのインデックスになり、線情報で参照されます。

`<除外多角形>…</除外多角形>`タグには、面として扱わない箇所をハンドルポイントのインデックスで記述します。

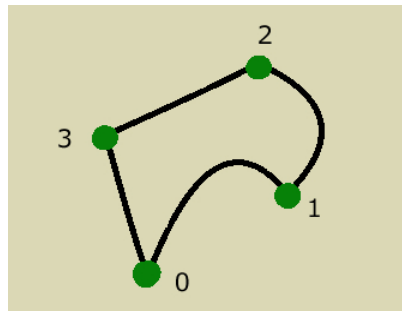
`<string>0,2,3</string>` ←面1枚ごとに、この様に記述します。

上の例では0,2,3と3つの整数がコンマ区切りで記述されていますので、ハンドルポイントインデックス0,2,3を接続する三角形が対象となります。四角形ならば整数の要素は4つ、Lips型ならば2つとなります。

* ハンドルポイントインデックスの記述順番は、三角形とLips型は制限がありませんが、四角形の場合は、4頂点を1周するように記述して下さい。周回方向はどちらでも構いません。

下図の四角形では、「0,1,2,3」「2,1,0,3」は可能ですが、「0,2,3,1」は入力できません。

Lips型では、「0,5」の様な記述になりますが、ハンドルポイントインデックス0と5を連結する線が2本あり、一方または2本がベジエ曲線でなければなりません。0と5を連結する線が3本以上ある場合は、Lips型の面は2枚形成されますが、仕様上1枚しか形成されません。



最終行は、</ぷりスカ保存用10> でタグを閉じます。

* モデル作成時は、ぷりスカのモデルロード時に実行される面の自動検出が有効に機能するよう、多角形の頂点数を考慮して設計する必要があります。

5角形以上の多角形は面として検出されませんので、四角形+三角形などとなるように線で分割して下さい。

3. プログラムによるモデルの作成

ここからは応用編として、マイクロソフト社の「Visual Studio 2019 Community」を使ってVB.netによるWindowsアプリケーションからぷりすか3Dモデルを作成する方法を解説します。

コンパイルされるプロジェクトはWPFアプリケーション、.Net Framework 4.6によるものですが、VB.net+Windows Formでも一部を改変すれば動作すると思われます。

このソースリストは当研究所のホームページからダウンロードできます。

* このソースリストをコンパイルしてWindowsアプリケーションを実行すると、デスクトップに“test000.xml”というぷりすか3Dモデルを作成します。これをぷりすかで読み込むとテントの形のモデルが表示されます。ソースにはバージョン1.0用のクラスが定義されていますのでタグの記述などを考えなくても.Net Frameworkの機能によりXMLファイルとして書き込まれます。

■ ぷりすか(R) 3Dモデル(XMLファイル形式)の作成

製作・著作 (C)2021 オカダ・システムエンジニアリング研究所

```
Imports System.Windows.Media.Media3D
```

```
Class MainWindow
```

```
Dim rand1 As New Random
```

```
'プログラムはここから実行される
```

```
Private Sub MainWindow_Loaded(sender As Object, e As RoutedEventArgs) Handles Me.Loaded
```

```
Dim obj1 As New ぷりすか保存用10
```

```
Dim 系統 As Integer = 0 '0から始めること
```

```
'書き込むファイル名 デスクトップ上の“test000.xml”
```

```
Dim fileName = System.Environment.GetFolderPath(Environment.SpecialFolder.DesktopDirectory) & “¥test000.xml”
```

```
obj1.make = “maker”
```

```
obj1.name = “モデル名”
```

```
obj1.content = “モデルコメント”
```

```
'視点(カメラ位置)などの座標情報
```

```
obj1.view保存情報 = “-13.1648,0.1619,6.8127,0.0000,0.0000,0.0000,11.5098,1.3945,22.2083,-0.3983,25.0122,-1.3642,5.0000,0.0000,0.0000,0.0000,5.0000,0.0000,0.0000,0.0000,9.0000,-22.15,0.00,-9.23,72.33,0.0,80.0,0.63”
```

```
makeテント(obj1, 系統)
```

```
'ファイルセーブ
```

```
Dim serializer2 As New System.Xml.Serialization.XmlSerializer(GetType(ぷりすか保存用10))
```

```
Using sw1 As New IO.StreamWriter(fileName, False, New System.Text.UTF8Encoding(False))
```

```
serializer2.Serialize(sw1, obj1)
```

```
End Using
```

```
Me.Close()
```

```
End Sub
```

```
Private Sub makeテント(ByRef obj0 As ぷりすか保存用10, ByVal 系統0 As Integer)
```

```
Dim 登録点IX As New List(Of Integer)
```

```
'ハンドルポイント情報
```

```
登録点IX.Add(obj0.点add(New Point3D(-2.86, -3.98, -1.08), 系統0, True))
```

```
登録点IX.Add(obj0.点add(New Point3D(0, -4, 3), 系統0, True))
```

```
登録点IX.Add(obj0.点add(New Point3D(0.22, -0.68, 0.7), 系統0, True))
```

```
登録点IX.Add(obj0.点add(New Point3D(-2.63, -1.01, -0.63), 系統0, True))
```

```
'線連結情報
```

```
obj0.線2D.Add(New ぷりすか保存用10.L2D(登録点IX(0), 0, 登録点IX(1), 1, 0)) '直線 実線
```

```
obj0.線2D.Add(New ぷりすか保存用10.L2D(登録点IX(0), 0, 登録点IX(2), 1, 0))
```

```
obj0.線2D.Add(New ぷりすか保存用10.L2D(登録点IX(0), 登録点IX(4), 登録点IX(3), 0, 1)) 'ベジェ 補助線
```

```
obj0.線2D.Add(New ぷりすか保存用10.L2D(登録点IX(1), 0, 登録点IX(2), 1, 0))
```

```
obj0.線2D.Add(New ぷりすか保存用10.L2D(登録点IX(1), 0, 登録点IX(3), 1, 0))
```

```
obj0.線2D.Add(New ぷりすか保存用10.L2D(登録点IX(2), 0, 登録点IX(3), 1, 0))
```

```
'除外多角形があれば記述する
```

```
obj0.除外多角形.Add(“0,1,3”)
```

```
End Sub
```

End Class

'ぶりスカ3Dモデルの保存用クラス

Public Class ぶりスカ保存用10 '10はバージョン1.0であることを示す

Public Property maker As String = ""
Public Property name As String = ""
Public Property content As String = ""

Public Property view保存情報 As String
Public Property 線2D As New List(Of L2D)
Public Property po3D As New List(Of HP)
Public Property 除外多角形 As New List(Of String)

'ハンドルポイントを追加

'既設の点と距離が近過ぎる場合(重複)は追加せず、既設の点のインデックスを返す

'Is重複統合 True:重複した場合は追加せずインデックスを返す False:常に追加

Public Function 点add(ByVal po3 As Point3D, ByVal 系統0 As Integer, ByVal Is重複統合 As Boolean)

Dim v3 As Vector3D
Dim i As Integer = 0

If Not Is重複統合 Then
i = po3D.Count
GoTo L100
End If

For Each _hp As HP In po3D '既設の点配列についてサーチ
v3 = New Vector3D(_hp.p3.X - po3.X, _hp.p3.Y - po3.Y, _hp.p3.Z - po3.Z)
If v3.Length < 0.01 And _hp.tp = 0 And _hp.ch = 系統0 Then
Return i '既設の点のインデックスを返す
End If

i += 1
Next

L100:

po3D.Add(New HP(po3, 系統0)) '新規に点を追加 新規のインデックスを返す
Return i
End Function

Public Class L2D '線情報

Public Property tp1 As Integer '0:ベジエ 1:直線
Public Property tp2 As Integer 'type2:線2DType2
Public Property qix As Integer '起点Index
Public Property cix As Integer '制御点Index
Public Property zix As Integer '終点Index

Public Sub New() 'これを消すと、~にはパラメータを持たないコンストラクタが含まれていないため、
'これをシリアル化することはできません。 のエラーで例外が発生する
End Sub

Public Sub New(ByVal 起点Index As Integer, ByVal 制御点Index As Integer, ByVal 終点Index As Integer,
ByVal _type1 As Integer, ByVal _type2 As Integer)

Me.qix = 起点Index
Me.cix = 制御点Index
Me.zix = 終点Index
Me.tp1 = _type1
Me.tp2 = _type2
End Sub

End Class

Public Class HP 'ハンドルポイント情報

Public Property p3 As Point3D '3D形式座標
Public Property tp As Integer 'ハンドルポイント属性 'ここは0を書き込むこと
Public Property ch As Integer '系統

Public Sub New() 'これを消すと、~にはパラメータを持たないコンストラクタが含まれていないため、
'これをシリアル化することはできません。 のエラーで例外が発生する
End Sub

Public Sub New(ByVal _p3 As Point3D, ByVal _ch As Integer)
Me.p3 = _p3
Me.tp = 0
Me.ch = _ch

End Sub

End Class

End Class

4. プログラムによるモデルの作成 ー応用編

当研究所のホームページからダウンロード可能な”桜の花びら.vb”(VB.netソースファイル)をコンパイルし実行すると、下図の様なモデルが作成されます。

これは桜の花びらの1枚の形状を登録したものを200枚作成し、1枚ごとにランダムに変形・回転・移動させ、陰線表示切り替えをONにして表示させたものです。

花びら1枚は4点のハンドルポイントを2本の直線と2本のベジエ曲線で連結したもので、簡単な1枚の”四角形”面オブジェクトとして扱われています。

