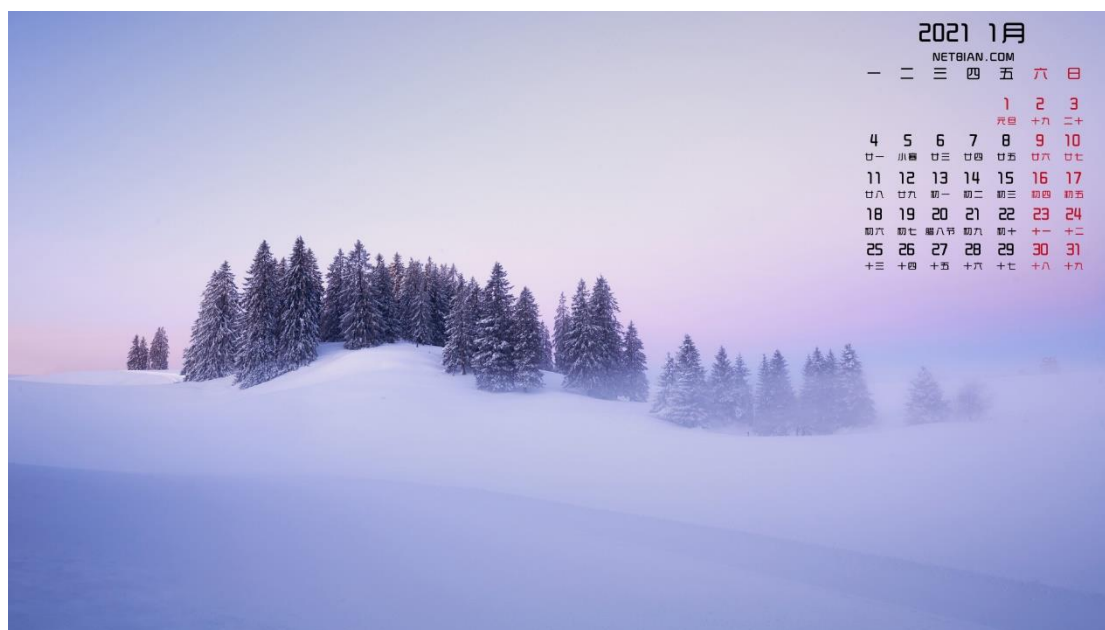


联宏电子期刊

第三百七十期—210111



软件升级版本

尊敬的联宏/优宏用户：

您好！

现为您提供最新的软件产品版本号。如您目前所使用的需要更新至最新版本，请与我公司技术总监冒小萍联系，邮箱：
kelly.mao@ugitc.com 祝您工作顺利！

NX1953

NX1926/NX1942

NX1899/NX1915

NX1872/NX1882

NX1847/NX1851

NX12.0.2MP14

NX11.0.2MP11

NX10.0.3MP19

NX9.0.3MP15

SE ST2 MP12

SE ST3 MP12

SE ST4 MP12

SE ST5 MP11

SE ST6 MP14

SE ST7 MP11

SE ST8 MP11

SE ST9 MP08

SE ST10 MP10

SE SE2019 MP4

SE SE2020

Process Simulate_11.1TR3

Process Simulate_12.1.3

Process Simulate_13.1.2

Process Simulate_14.0.2

Plant Simulation_12.2

Plant Simulation_13.2

Plant Simulation_14.1

Teamcenter visualization 8.1.2.2

Teamcenter visualization 8.3.3.10

Teamcenter visualization 9.1.2.6

Teamcenter visualization 10.1

Teamcenter visualization 11.1

I-deas 12 M4

I-deas 5 M3

I-deas 6 M2

I-deas 6.1M2

I-deas 6.2

I-deas 6.4

Teamcenter 2007.2.2

Teamcenter 8.3.3

Teamcenter 9.1.2

Teamcenter 10.1.1

有奖问答 20210111

小明在使用 Process Simulate 的 OPCUA 外部链接通道时遇到问题，他需要读取输送带的启动信号，在 PLC 中这个信号的信息是：变量名 Conveyor_Start, 地址 Q10.0, 外部链接通道名称: OPCUA_Test。

请问小明该如何在 Siganl Viewer 定义该信号？

A.创建输出变量: Conveyor_Start, 地址输入: 10.0, 外部链接选择: OPCUA_Test

B.创建输入变量: Conveyor_Start, 地址输入: 10.0, 外部链接选择: OPCUA_Test

C.创建输出变量: “Conveyor_Start”, 地址不输入, 外部链接选择: OPCUA_Test

D.创建输入变量: “Conveyor_Start”, 地址不输入, 外部链接选择: OPCUA_Test

答案: C



目录

NX

3D PDF 表格模板定制	6
拉伸产生收敛体的解决方法	9
NX 部件族中公差关联表达式.....	12
STAR-CCM+流场中运动轨迹几何设定方法.....	17
NX 二次开发-NX7.5 属性常见问题	21
NX 二次开发-装配环境移动组件面	27

TC

切换 visualization 中模型视角解决方案.....	30
安装 MRL 资源列表.....	33
解决产品配置器选配无法正常显示	42
如何设定数据集的历史记录版本数量	45
如何通过时间表任务属性查询项目	47

TECNO

Process Simulate on Teamcenter 机器人运动数据配置	51
关于联宏.....	54

3D PDF 表格模板定制

作者：赵冠兄

审校：陈昂

适用版本：NX1847 以上

为了更好地让 NX 导出的 3D PDF 符合企业规范与标准，我们可以定制专用的 3D PDF 的模板。一般在 3D PDF 模板中，通常会使用定制化表格，列出相关零组件信息，如图 1 所示。本篇技巧将为大家介绍如何定制个性化表格模板，以在 PDF 中呈现零组件的信息。

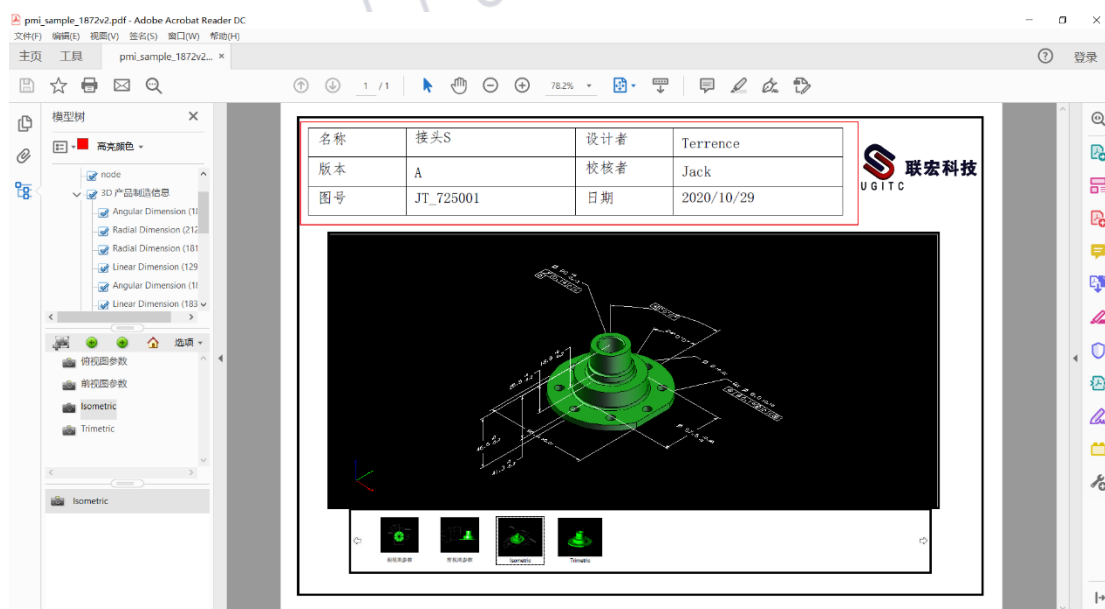


图 1

首先，此模板定制的前提是企业拥有模型的设计规范，确保设计模型拥有所需显示信息对应的属性，如图 2 所示。

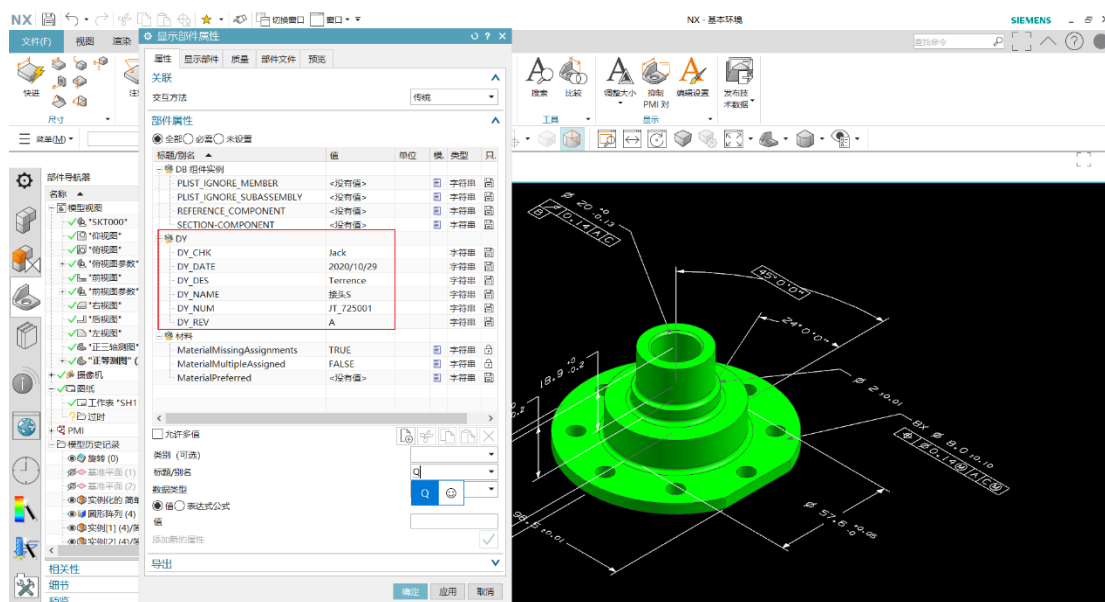


图 2

然后，如图 3 所示，通过“PMI”模块下的“新建模板”功能，建立所需模板，并如图 4 所示，在模板中建立对应的属性，这些属性名称应与设计模型属性的名称对应。



图 3

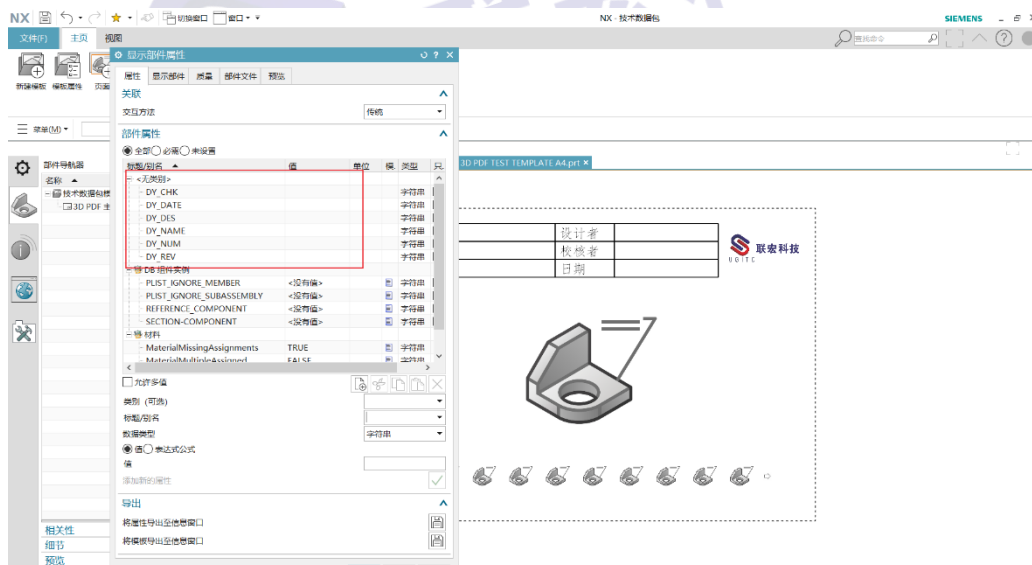


图 4

然后在模板中建立表格，在对应的单元格中，导入所需显示的属性名，如图 5 所示。

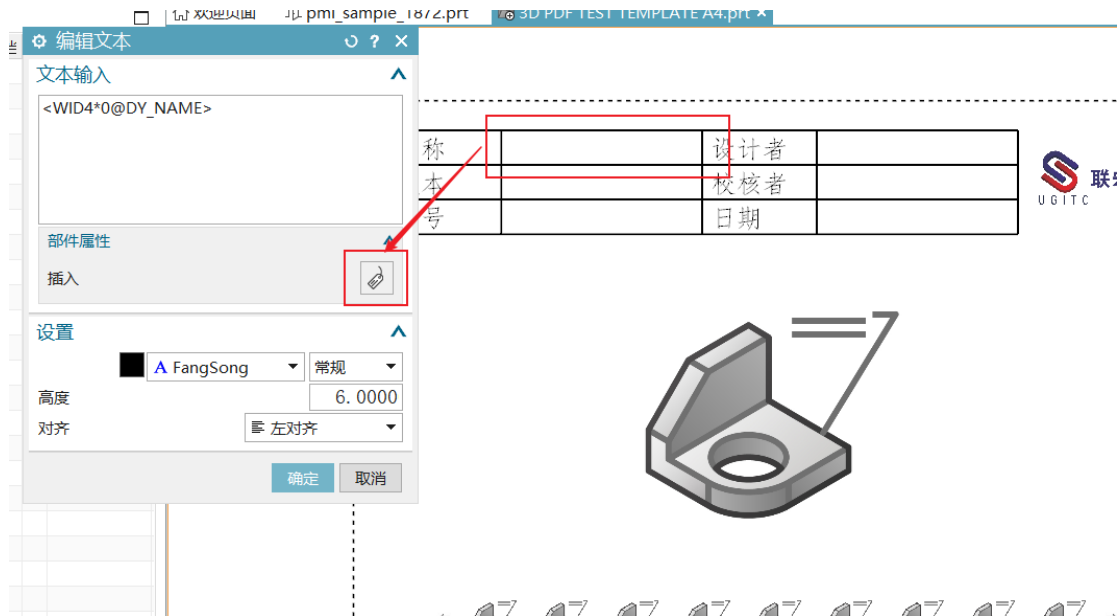


图 5

所有需要导入属性的单元格制作完成，完成 3D PDF 模板定制后，将模板文件放置于 UG 安装目录\TDP\templates 下即可使用。

拉伸产生收敛体的解决方法

作者：陈林生

审校：黎芳勇

适用版本：NX

使用建模命令拉伸体时，如果拉伸对象是多义线，产生的拉伸体可能是收敛体。拉伸后是否为收敛体，可用命令“移除参数”来判断，如下图 1，将拉伸后的体执行“移除参数”命令后，如果其效果如图 2 左所示，说明几何体为收敛体，如果其效果如图 2 右所示，说明几何体为参数体。

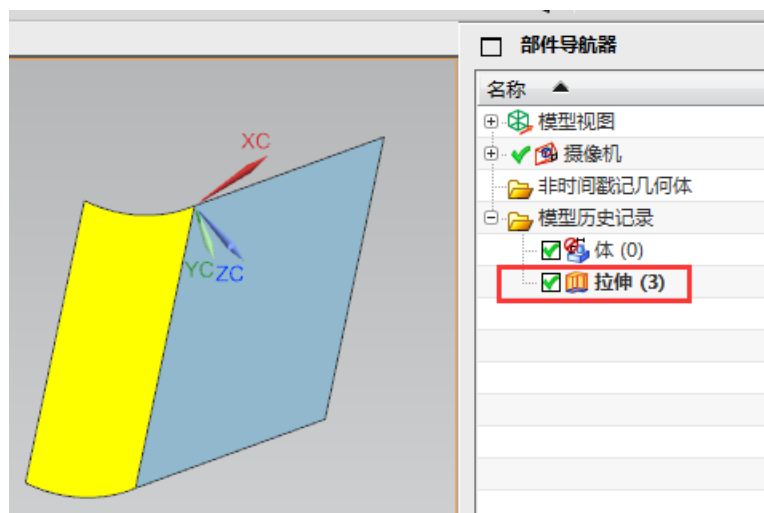


图 1



图 2

收敛体产生后，很多参数指令不能使用，比如图 3 所示的边倒圆。在 NX 加工模块中，收敛体也不能被用于曲面驱动刀路。有没有一种方法不产生收敛体呢？



图 3

解决方法如下：

一、执行菜单命令如下，首选项→建模→收敛，此处“将 1 次样条作为多段线处理”默认是勾选的，勾选状态下将会产生收敛体，将此勾去掉，再产生的几何体就不会是收敛体了。



图 4

二、首选项的设定只对当前文件有效，如果想一劳永逸不产生收敛体，可在用户默认设置里设定。执行菜单，文件→实用工具→用户默认设

置→建模→常规→收敛，如下图 5 所示操作，将默认的打勾去掉就行了。

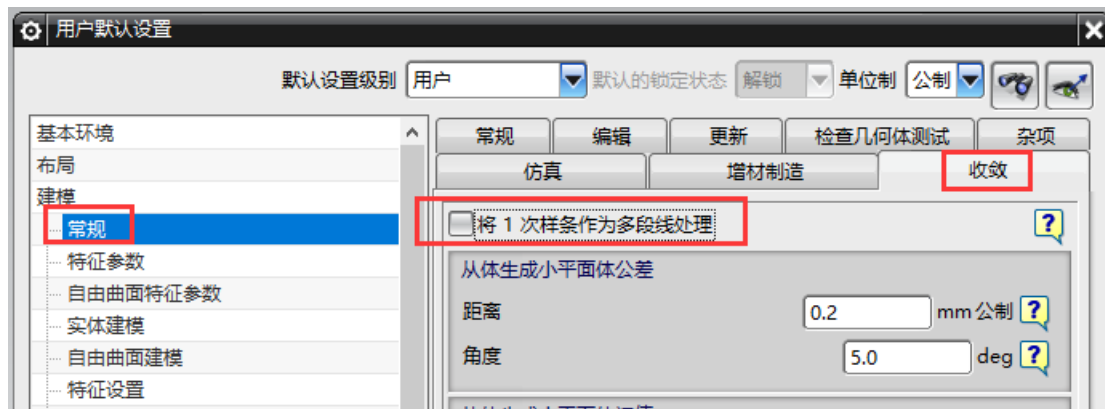


图 5

NX 部件族中公差关联表达式

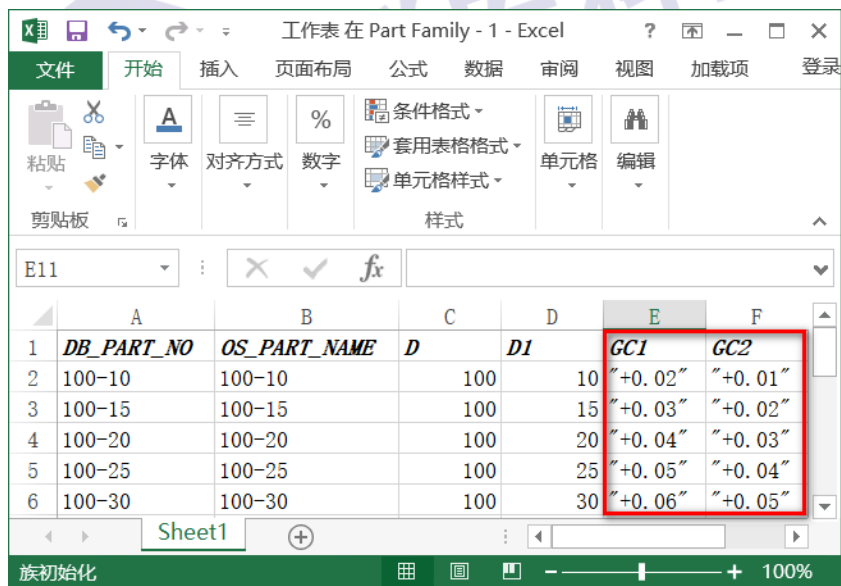
作者：孙立雪 审校：刘卫民

适用版本：NX7.5

在企业生产制造时经常会遇到形成企业标准的成系列的自制件，在生产时就必须根据表格图纸去查询尺寸信息，才能明确图纸中对应的字母具体的值，这时如标注尺寸相对较多的话，获取到图纸信息的时间就相对较慢，影响工作效率。遇到上述情况我们可以采用部件族的方式制作模型和图纸。

在 NX 工程图中公差的标注是必不可少的，但是相同位置的公差可能会随着尺寸变换，公差值也会发生变。此时就需要将公差与表达式相关联，才能保证公差与对应的主尺寸关系正确。

首先，新建字符串类型表达式 GC1 和 GC2，将公差输入对应的表达式中。将表达式 GC1 和 GC2 添加到部件族表格中，完善这个零件中对应尺寸的其它公差值，保存部件族表格。



	A	B	C	D	GC1	GC2
1	DB_PART_NO	OS_PART_NAME	D	DI	GC1	GC2
2	100-10	100-10	100	10	+0.02"	+0.01"
3	100-15	100-15	100	15	+0.03"	+0.02"
4	100-20	100-20	100	20	+0.04"	+0.03"
5	100-25	100-25	100	25	+0.05"	+0.04"
6	100-30	100-30	100	30	+0.06"	+0.05"

图 1

在工程图中将对应的表达式连接到公差的位置，具体方法如下。

1. 双击需要添加公差的尺寸，点击键盘上“→”按钮，再点击图示标识处文本按钮。

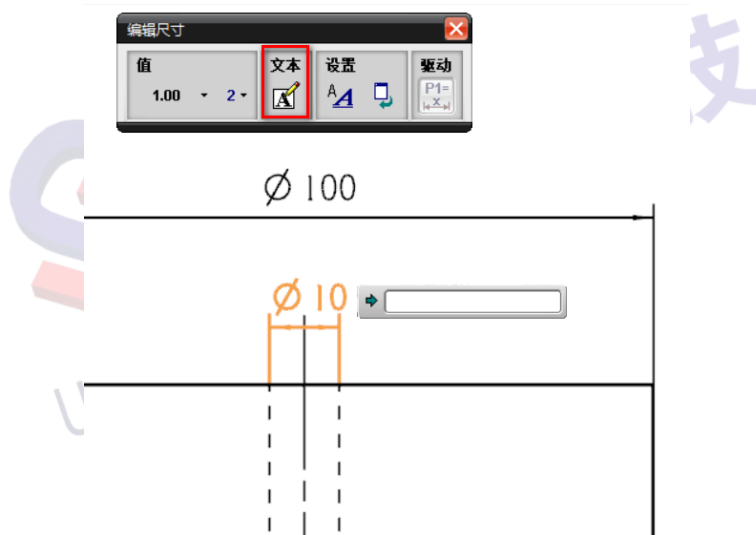


图 2

2. 选择关系按钮，再选择表达式。

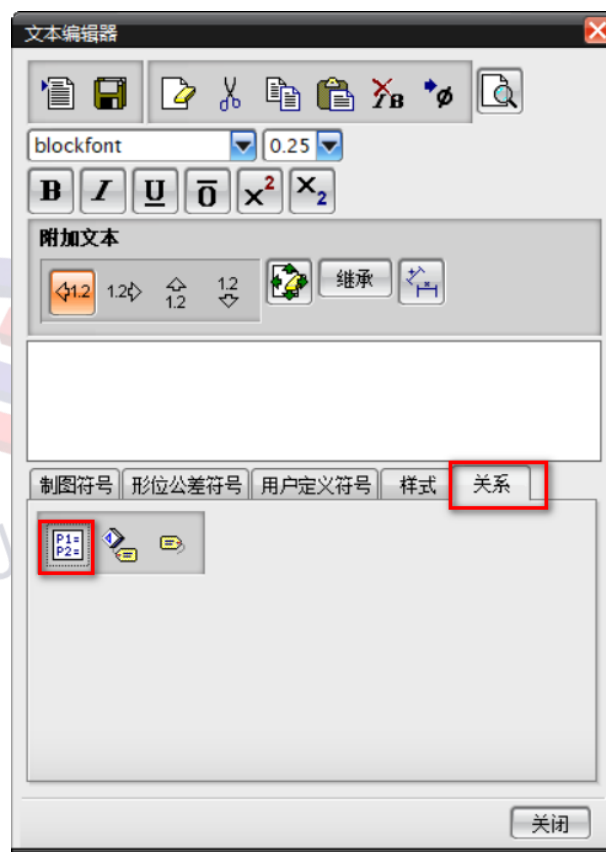


图 3

3.选择 GC1 代表公差的表情式,点击确认。重复操作在选择 GC2。

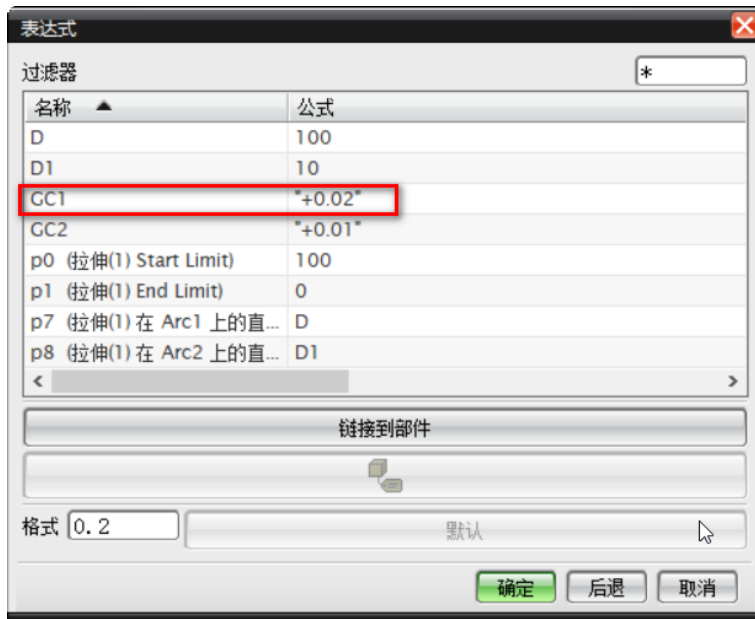


图 4

4.完成上步操作后界面如图 5 所示。



图 5

5.点击制图符号按钮,分别将两个表达式剪切到对应位置。点击

绿色标识按钮。

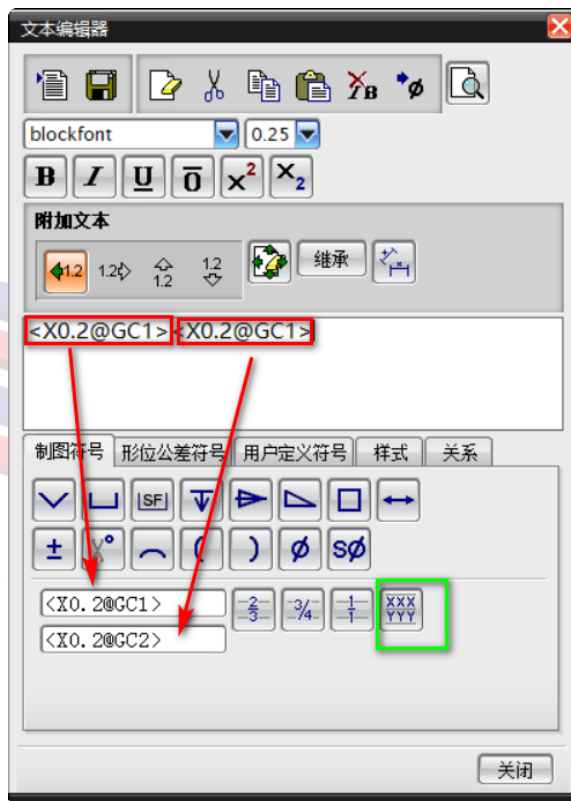


图 6

6.最终显示窗口如图 7 所示，点击确认，关闭窗口。

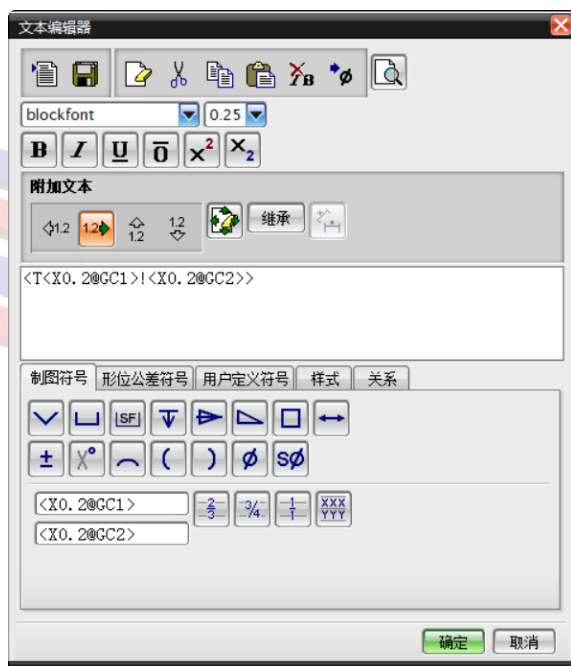


图 7

7.最终公差显示样式如图 8 所示。

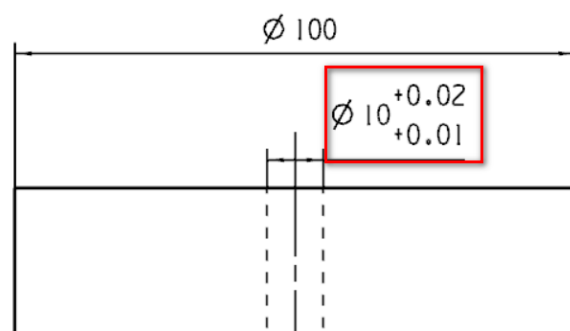


图 8

通过以上方法实现了表达式与公差关联，希望对大家有所帮助。

STAR-CCM+流场中运动轨迹几何设定方法

作者：张艳松

审校：陈志愿

适用版本：STAR-CCM+

1、概述

Siemens STAR-CCM+在模拟流场的过程中，常常在流体中有一些物体的运动，通常我们采用运动方程来描述其运动轨迹。对于一些复杂的运动，其实我们有时候很难用函数关系来描述其运动轨迹。这时，我们可以采用数表的形式，用离散的点来描述其运动轨迹。但是，对于复杂的运动，测量轨迹线上离散点的坐标，也是一项复杂而繁琐的工作。

针对上述问题，STAR-CCM+可以自动监测几何轨迹形貌，导出轨迹离散点的坐标值，从而大大简化描述几何运动轨迹的操作过程。如图 1 所示，一个小球沿着下面的轨迹运动，怎样得到图示的轨迹呢？下面以为例加以说明。

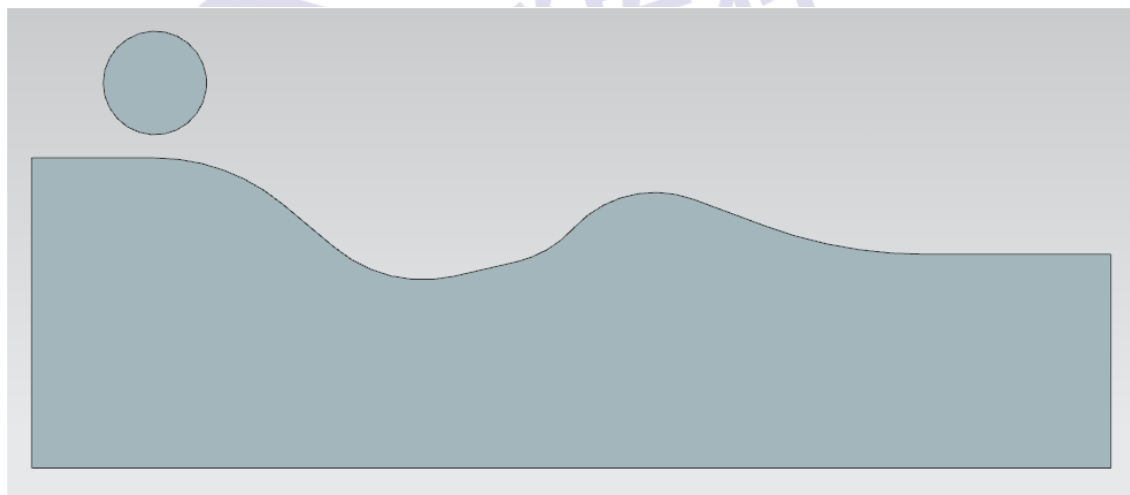


图 1 小球轨迹运动

把上述的几何模型导入到 STAR-CCM+中，表面离散化如下图 2

所示：

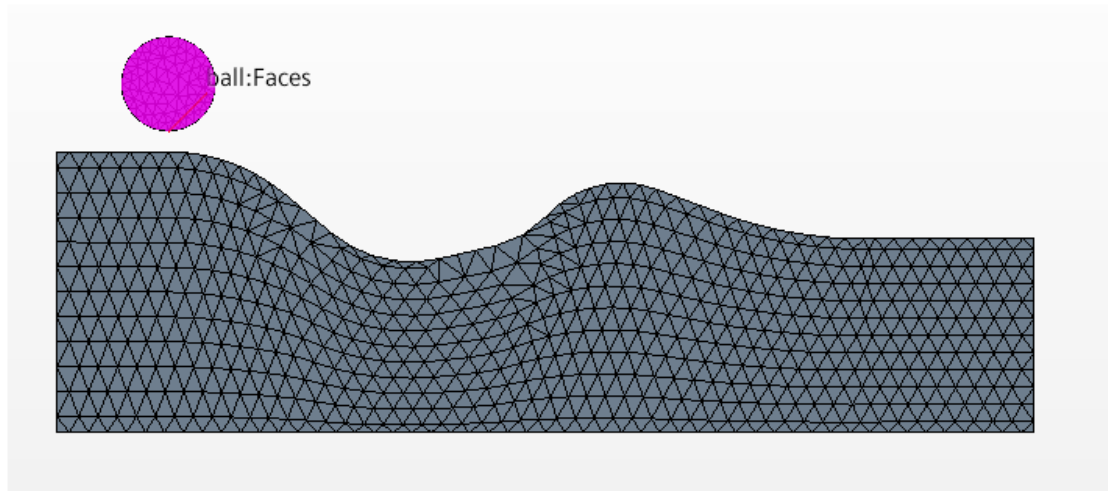


图 2 表面离散化

把轨迹表面单独显示出来，如下图 3 所示。并且创建一个平面来切割轨迹表面，如下图 4 所示。



图 3 轨迹表面

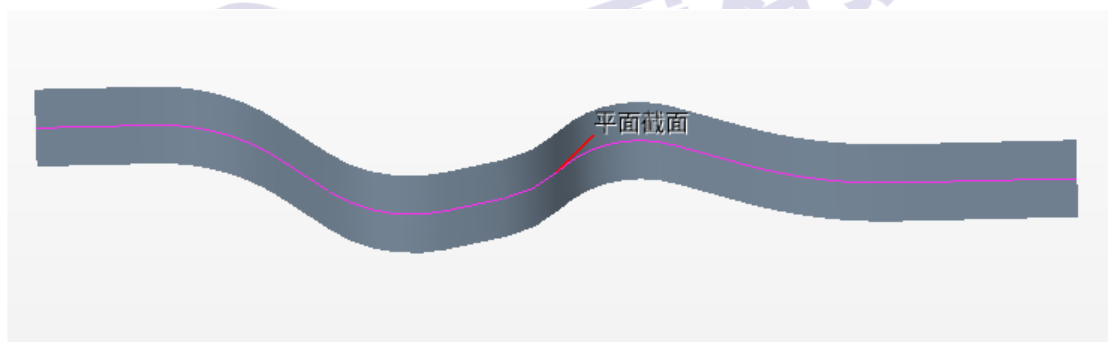


图 4 切割的轨迹截面

轨迹建好后，下面就是把轨迹坐标的数表到出。导出过程如下：
标量函数选择 position Z，零部件选择刚创建的平面截面，再选择 Latest Surface，如下图 5 所示。

已提取	[Position[Z], X, Y, Z]
标量	[Position[Z]]
零部件	[平面截面]
坐标系	Laboratory
节点数据	☐
表示	Latest Surface
标签	

图 5 创建坐标数表

坐标数表创建完成后，把数表导出，如下图 6 所示。

Position[Z]	X (m)	Y (m)	Z (m)
0.01	0.127713	-0.01389	0.01
0.01	0.126525	-0.01389	0.01
0.01	0.122959	-0.01389	0.01
0.01	0.124148	-0.01389	0.01
0.01	0.122959	-0.01389	0.01
0.01	0.121488	-0.01389	0.01
0.01	0.117074	-0.01389	0.01
0.01	0.118545	-0.01389	0.01
0.01	0.117074	-0.01389	0.01

图 6 导出的数表

下一步就是在 EXCEL 表格中对数表进行排序：

- 冻结首行
- 对 X 坐标进行升序排列

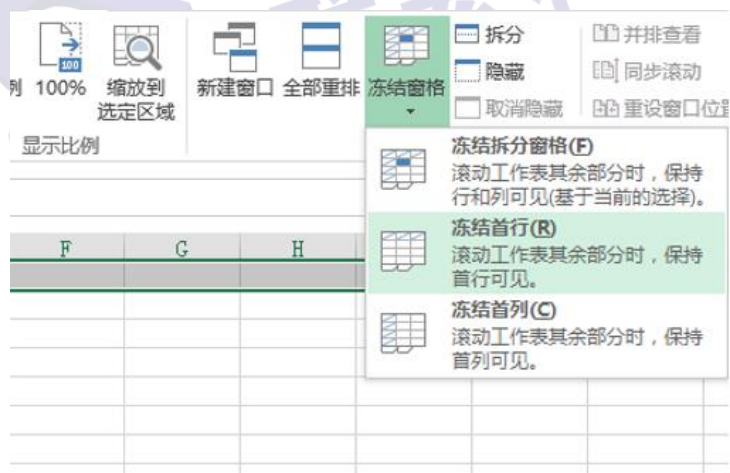


图 7 冻结首行

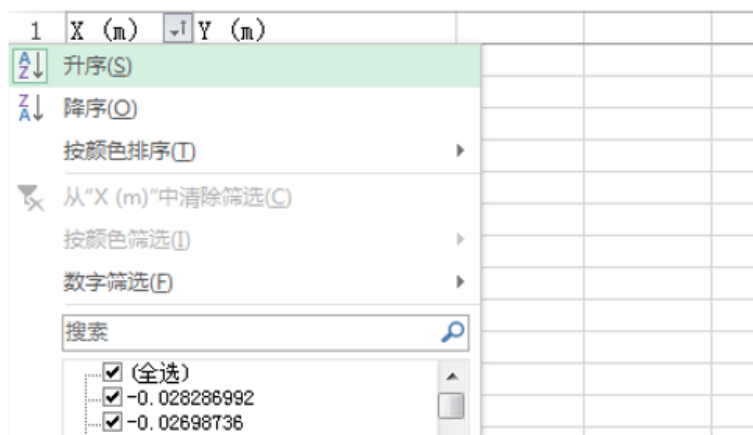


图 8 升序排列

完成后，为了检查创建的数表坐标值的正确性，可以把坐标通过绘制曲线图画出来，与实际的轨迹进行对比，图形一致说明创建的轨迹数表坐标值是正确的。如下图 9 所示。

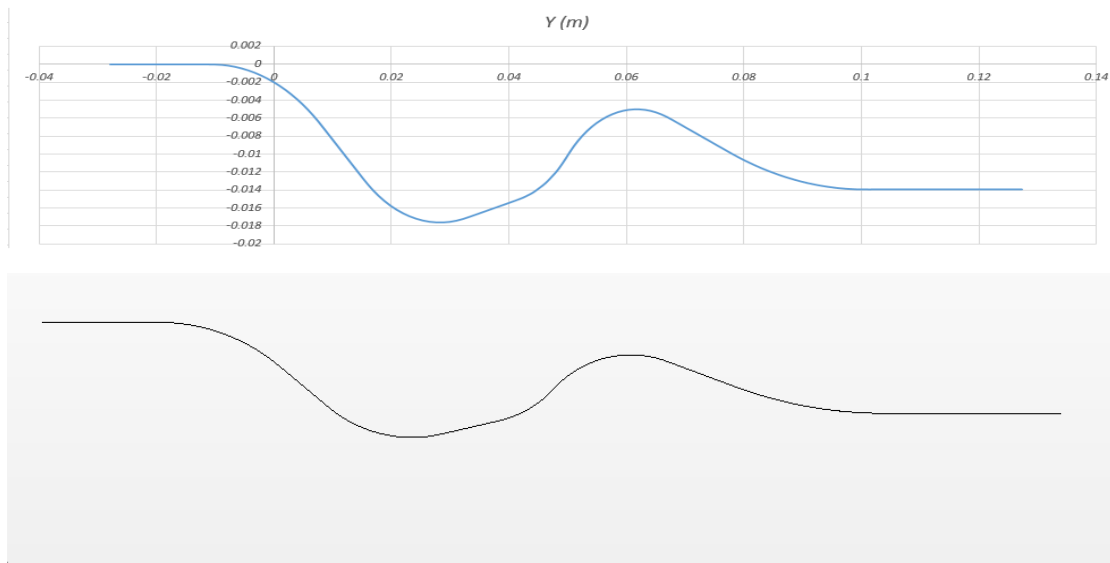


图 9 对比轨迹坐标值

通过上述操作，基本完成了轨迹坐标值的创建，从而可以把数表中的坐标值应用在运动物体的轨迹定义上，即简单又快捷。

NX 二次开发-NX7.5 属性常见问题

作者：纪新杭 审校：金雷

适用版本：NX7.5 版本

一、概述

在做 NX 二次开发时，开发人员会使用属性对特殊对象做标记处理或记录信息，而在 NX7.5 中，属性功能仍然只是“半成品”，还有很大的完善空间，下面我将介绍几个开发人员经常会遇到的问题以及相应的解决方法。

二、功能说明

1) NX7.5 版本的部件属性只支持字符串类型

用过高版本的人都知道，部件属性是支持所有属性类型，如图 1 所示。但是在 NX7.5 版本，部件属性只能输入字符串型，如图 2 所示。

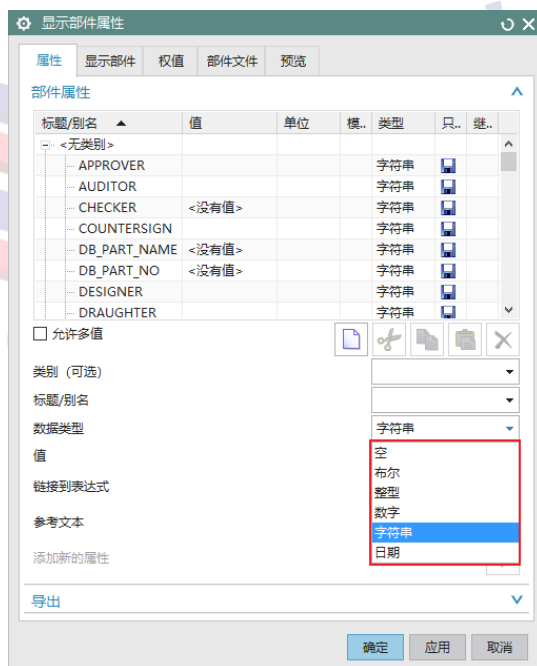


图 1

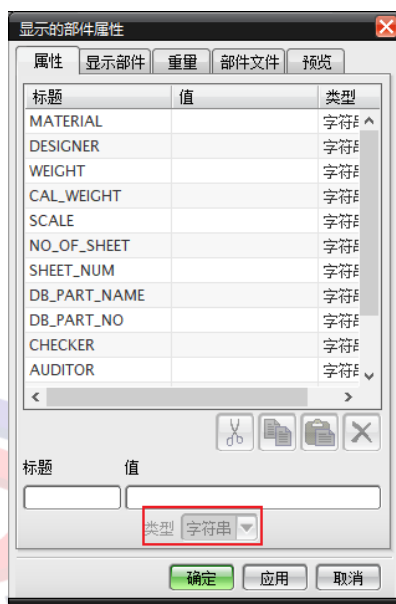


图 2

解决方法：尽可能只传入字符串型数据；若一定要其他类型数据，可以先转换成字符串型，后续读取时再反向转换读取即可；也可以考虑储存到其他对象属性上。

2) NX7.5 版本的属性不支持布尔类型和多值类型

用过高版本的人都知道，属性中一般含有布尔类型可以选择，且在任意类型下都有“允许多值”选项可以选择，如图 3 所示。但是在 NX7.5 版本的属性中，这两个功能都不支持，如图 4 所示。

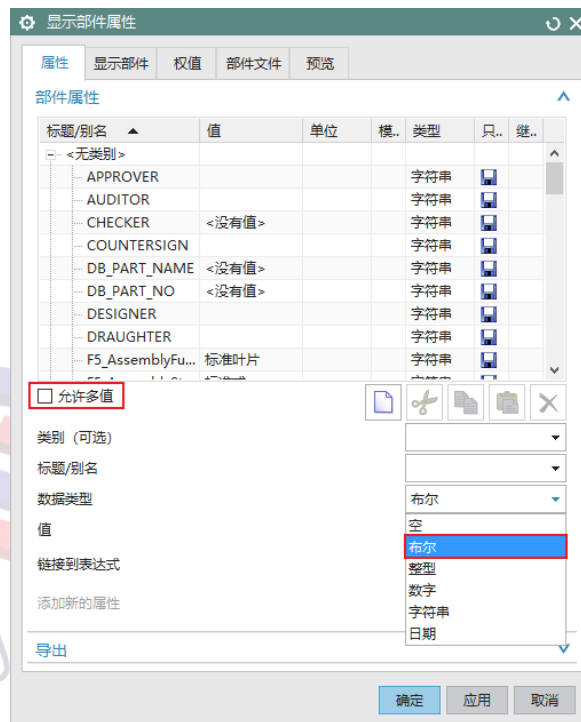


图 3

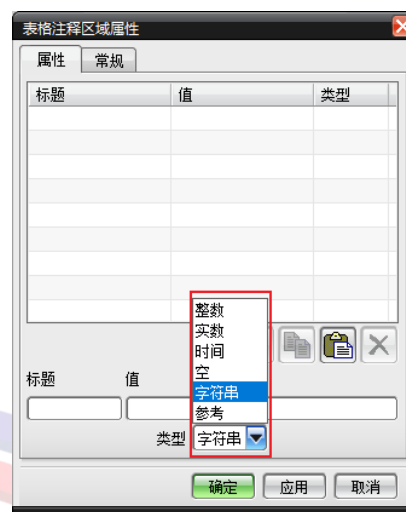


图 4

解决方法：尽可能规避布尔值以及多值情况；布尔值可以通过字符串值为“TRUE”或“FALSE”来替代；多值可以用给属性名称添加流水码后缀的方法，来区分每一级，这个方法需要封装相应的写入和读取功能，如图 5 所示。

```

    /// <summary>
    /// 获取对象多值字符串属性
    /// </summary>
    /// <param name="nXObjectI">对象</param>
    /// <param name="nameI">指定属性名</param>
    /// <returns>返回判断</returns>
    1 个引用
    public static List<string> CheckAndGetArrayAttribute(this NXObject nXObjectI, string nameI)
    {
        int NumIndex = 0;
        List<string> mFixture = new List<string>();
        while (true)
        {
            string oneValue = nXObjectI.CheckAndGetStringAttribute(nameI + "_" + NumIndex);
            if (oneValue == null)
            {
                break;
            }
            mFixture.Add(oneValue);
            NumIndex++;
        }
        return mFixture;
    }
    /// <summary>
    /// 设置多值字符串
    /// </summary>
    /// <param name="nXObjectI"></param>
    /// <param name="nameI"></param>
    /// <param name="valueI"></param>
    /// <param name="optionI"></param>
    1 个引用
    public static void SetAttribute(this NXObject nXObjectI, string nameI, List<string> valueI, Update.Option optionI)
    {
        for (int NumIndex = 0; NumIndex < valueI.Count; NumIndex++)
        {
            nXObjectI.SetAttribute(nameI + "_" + NumIndex, valueI[NumIndex], optionI);
        }
    }

```

图 5

3) NX7.5 版本的部分属性没有手动编辑界面

用过高版本的人都知道，属性一般都可以直接从对象的右键菜单中找到并编辑。但是在 NX7.5 版本的部分对象的属性，是没法在界面中直接编辑的，如图 6 所示。只能通过对对象的信息按钮查看属性信息，如图 7 所示。目前已知的这类对象有图纸页、投影视图和模型视图。

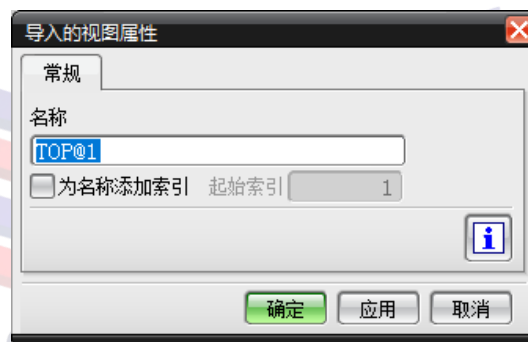


图 6

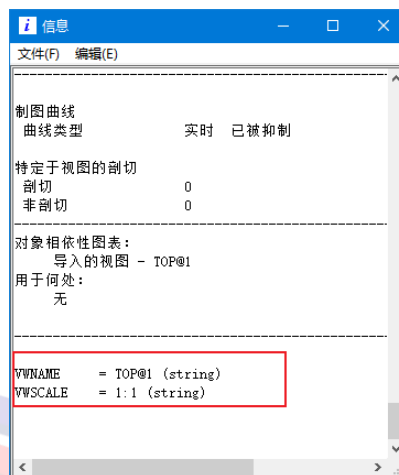


图 7

解决方法：尽可能规避在这类对象上储存或标记信息；尽量减少手动查看或编辑属性的需求；自行开发编辑界面，方便手动查看或编辑属性。

4) NX7.5 版本的属性值的字节长度有限制

对于储存信息或标记对象的属性来说，我们下意识认为其储存的内容应该是可以很多的才行。但是经过测试，在 NX7.5 版本的属性中，属性值只能储存 132 个字节的内容。超过该大小时，写入会报错，如图 8 所示。

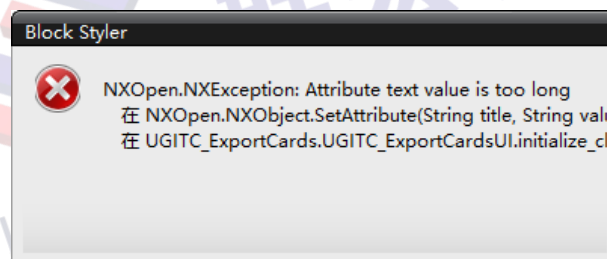


图 8

解决方法：尽可能规避在储存超过 132 个字节的信息；超出范围时，可以采用和上述问题 2 中多值问题类似方法，将属性拆分成多个值，并用流水码的方式区分；在代码上使用

“Encoding.UTF8.GetByteCount(“”)”的方法可以获取输入内容的字节数量。

三、总结

在无法规避开发版本的前提下，应对不同版本的一些开发问题，是需要经验慢慢累积。通过本文分享，让更多人了解 NX7.5 的一些开发小技巧。

NX 二次开发-装配环境移动组件面

作者：陈栩磊 审校：张季

适用版本：NX6 以上

一、概述

在 NX 二次开发中，需要对组件进行旋转及移动以达到所需装配效果，我们手动可以通过旋转组件方位然后拖动组件，随后进行组件约束达到我们预期的效果，但是对于组件尺寸不对从而需要临时修改的组件，我们只能进入工作部件去更改其参数或移动面来实现，相对于修改参数，移动面更直观和方便。

本文主要介绍移动组件面的封装方法。

二、功能说明

通过 NXopen 封装的方法，如图 1 所示。

```

// <summary>
// 装配环境-移动组件面
// </summary>
// <param name="facesInComp">输入-装配中的面集合</param>
// <param name="distance">输入-移动距离</param>
// 2 个引用|0 项更改|0 名作者, 0 项更改
public static void MoveCompFaces(IEnumerable<Face> faces, double distance, Direction direction)
{
    Part workCompPart = theSession.Parts.Work;
    NXOpen.Features.AdmMoveFaceBuilder admMoveFaceBuilder = workCompPart.Features.CreateAdmMoveFaceBuilder(null);
    admMoveFaceBuilder.FaceToMove.RelationScope = 511;
    admMoveFaceBuilder.Motion.DistanceAngle.OrientXpress.AxisOption = NXOpen.GeometricUtilities.OrientXpressBuilder.Axis.Passive;
    admMoveFaceBuilder.Motion.DistanceAngle.OrientXpress.PlaneOption = NXOpen.GeometricUtilities.OrientXpressBuilder.Plane.Passive;
    admMoveFaceBuilder.Motion.AlongCurveAngle.AlongCurve.IsPercentUsed = true;
    admMoveFaceBuilder.Motion.OrientXpress.AxisOption = NXOpen.GeometricUtilities.OrientXpressBuilder.Axis.Passive;
    admMoveFaceBuilder.Motion.OrientXpress.PlaneOption = NXOpen.GeometricUtilities.OrientXpressBuilder.Plane.Passive;
    admMoveFaceBuilder.FaceToMove.CoplanarEnabled = false;
    admMoveFaceBuilder.FaceToMove.CoplanarAxesEnabled = false;
    admMoveFaceBuilder.FaceToMove.CoaxialEnabled = false;
    admMoveFaceBuilder.FaceToMove.EqualDiameterEnabled = false;
    admMoveFaceBuilder.FaceToMove.TangentEnabled = false;
    admMoveFaceBuilder.FaceToMove.SymmetricEnabled = false;
    admMoveFaceBuilder.FaceToMove.OffsetEnabled = false;
    admMoveFaceBuilder.FaceToMove.UseFaceBrowse = true;
    admMoveFaceBuilder.Motion.Option = NXOpen.GeometricUtilities.ModlMotion.Options.Distance;
    admMoveFaceBuilder.Motion.DeltaEnum = NXOpen.GeometricUtilities.ModlMotion.Delta.ReferenceAcsWorkPart;
    admMoveFaceBuilder.Motion.DeltaXc.RightHandSide = "0";
    ScCollector scCollector1 = (ScCollector)workCompPart.FindObject("ENTITY 113 2");
    SelectionIntentRule[] rules1 = new SelectionIntentRule[0];
    scCollector1.ReplaceRules(rules1, false);
    /*将组件中的面转换成部件中的真实面*/
    //Face[] facesInPart = facesInComp.Select(q => q.Prototype as Face).ToArray();
    Face[] facesInPart = faces.Select(q => q.Prototype as Face).ToArray();
    FaceDumbRule faceDumbRule1;
    faceDumbRule1 = workCompPart.ScRuleFactory.CreateRuleFaceDumb(facesInPart);
    SelectionIntentRule[] rules2 = new SelectionIntentRule[1];
    rules2[0] = faceDumbRule1;
    admMoveFaceBuilder.FaceToMove.FaceCollector.ReplaceRules(rules2, false);
    Scalar scalar1 = workCompPart.Scalars.CreateScalar(0.5, NXOpen.Scalar.DimensionalityType.None, NXOpen.SmartObject.UpdateOption.WithinModeling);
    Scalar scalar2 = workCompPart.Scalars.CreateScalar(0.5, NXOpen.Scalar.DimensionalityType.None, NXOpen.SmartObject.UpdateOption.WithinModeling);
    Point point = workCompPart.Points.CreatePoint(facesInPart[0], scalar1, scalar2, NXOpen.SmartObject.UpdateOption.WithinModeling);
    admMoveFaceBuilder.Motion.DistanceVector = direction;
    admMoveFaceBuilder.Motion.DistanceValue.RightHandSide = distance.ToString();
    NXObject nxObj = admMoveFaceBuilder.Commit();
    admMoveFaceBuilder.Destroy();
}

```

图 1

其中需要作为输入参数的包含三个关键参数，第一个参数为我们需要移动的面集合，第二个参数为需要移动的距离，第三个参数为移动方向。

最后可以实现如下效果，如图 2、3 所示。

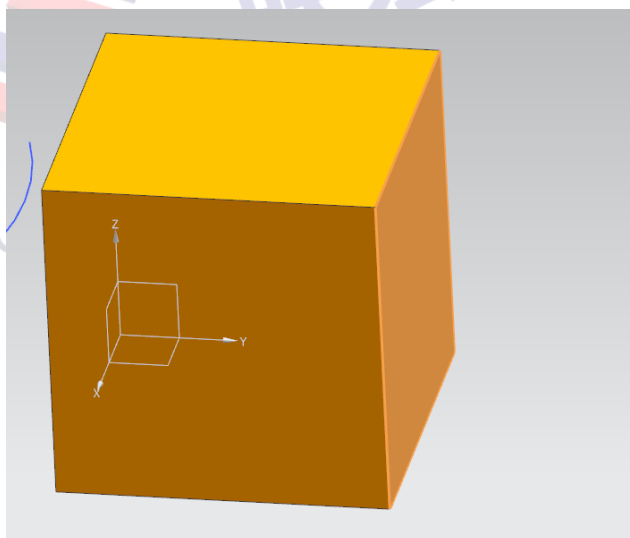


图 2

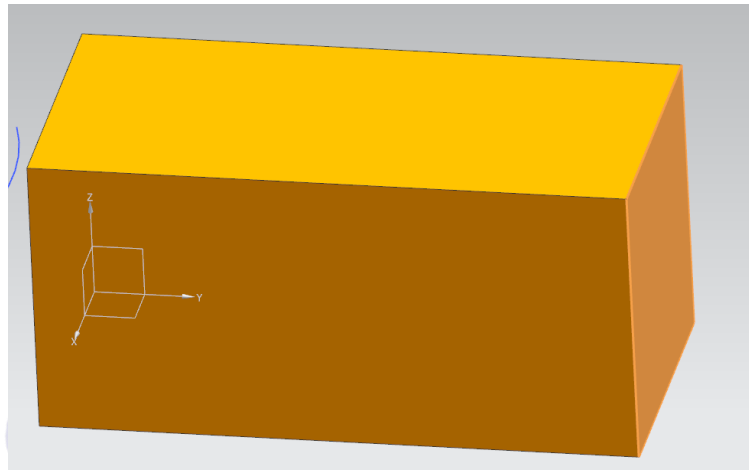


图 3

三、总结

通过以上方法的封装和调用，可以快速移动组件或面集合，达到我们所需的装配效果。同时通过对 NX 的录制代码分析，可以帮助我们更好的了解 NXOpen 开发方式。

切换 visualization 中模型视角解决方案

作者：黄鹏

审校：李达

适用版本：TC11

用户可以在 visualization 中随意的改变模型的角度，但是如果要从 X、Y、Z 三个正负方向来看，如果手动去调整，可能会造成误差，所以需要用到 visualization 的标准视图。

1. 如图 1，我们可以在 TC 中选择要加载的轻量化模型，然后点击标识的图标。

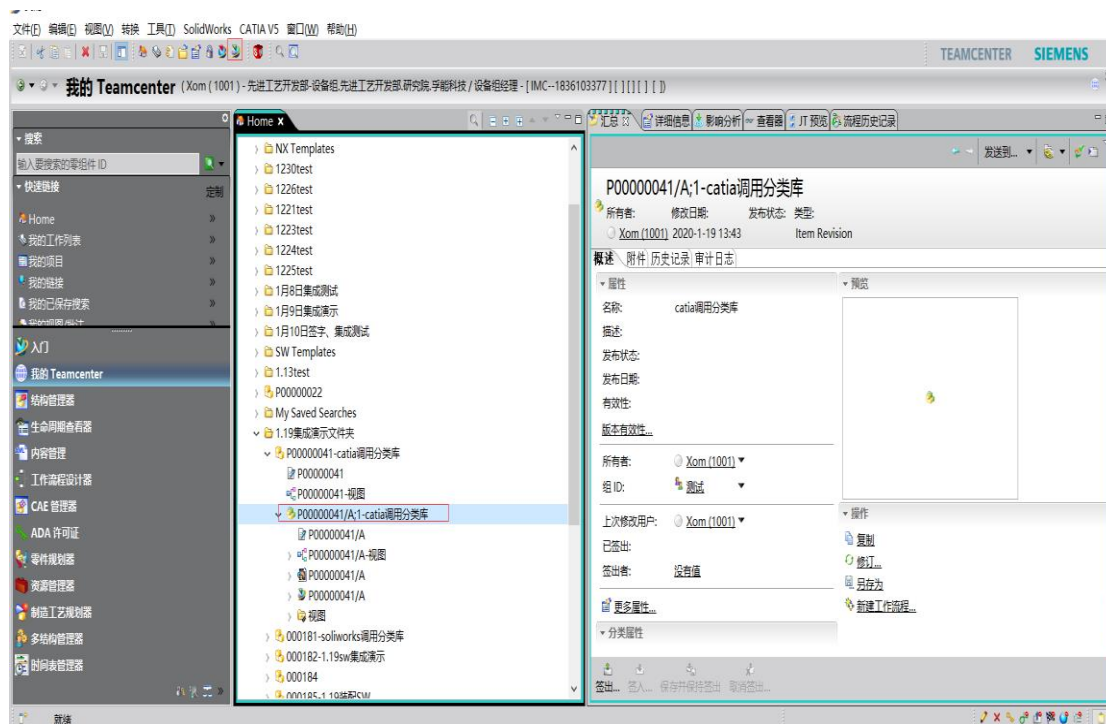


图 1

2. 如图 2，勾选数据流水左侧的方框，显示出数模。

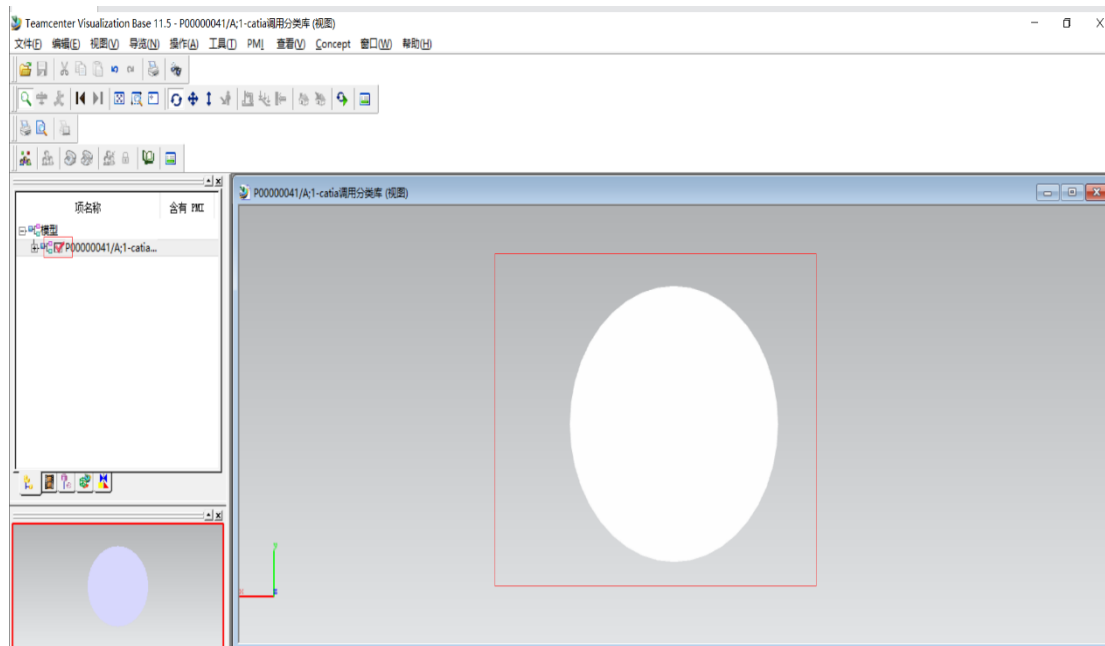


图 2

3. 如图 3，选择视图-标准视图-+X，效果如图 4。

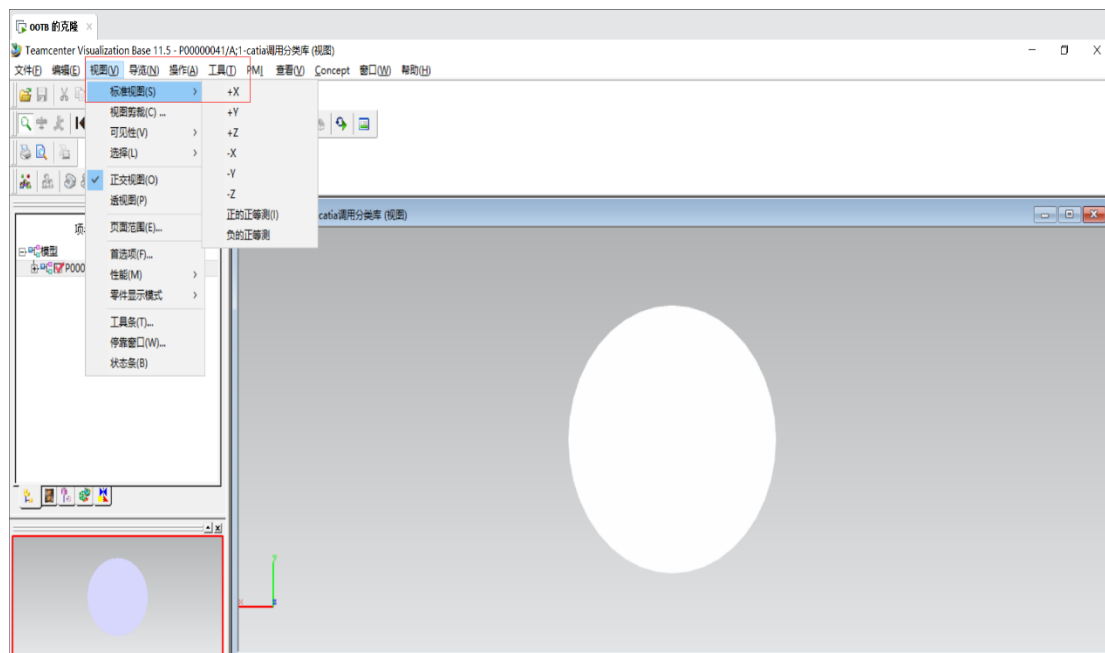


图 3

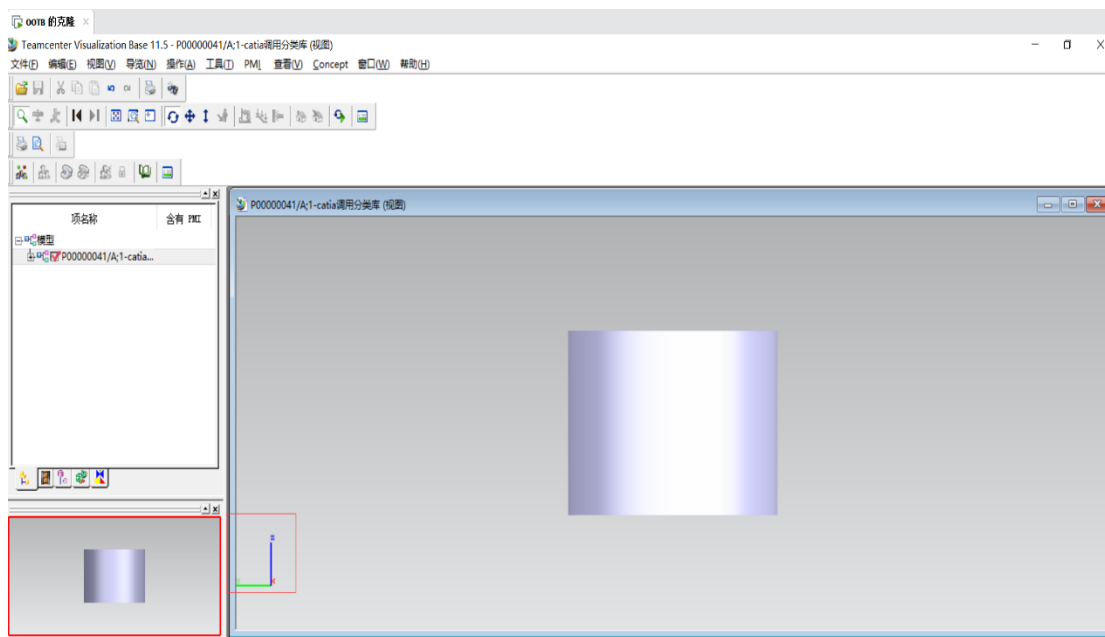


图 4

4. 如图 5，选择+Z 的效果。

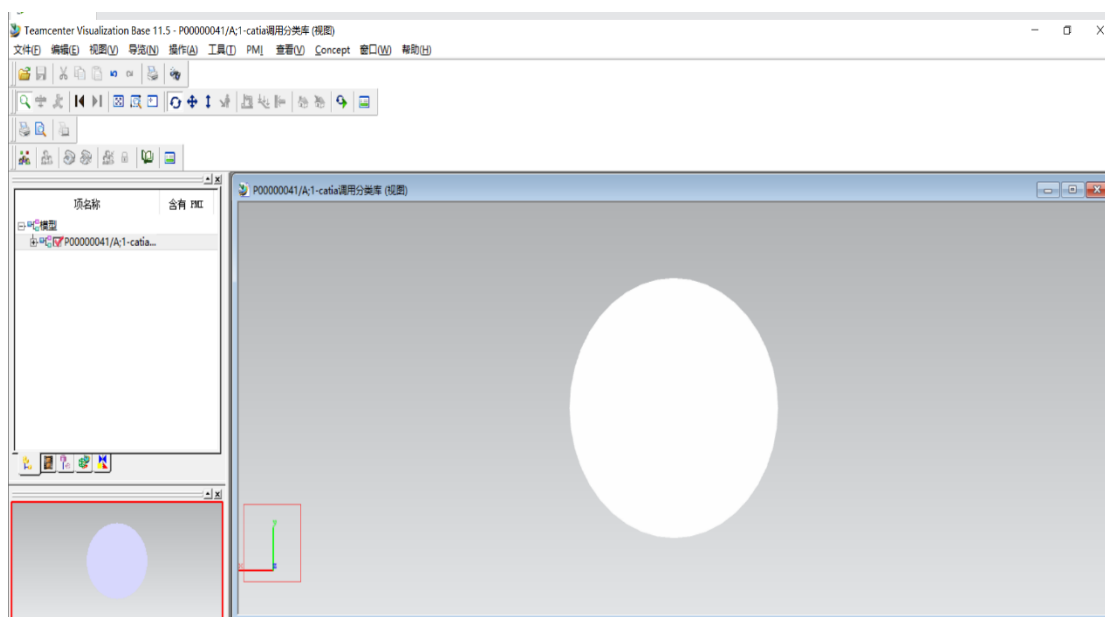


图 5

5. 熟练运用此功能，便于用户查看数模，节省时间，提升效率。

安装 MRL 资源列表

作者：柳汉杰 审校：王明皓

测试版本： TC11.4

在企业中，基于 Teamcenter 与 NX 集成环境的工作环境，工艺规划员（主要在 Teamcenter 中工作）或 NC 编程人员（主要在 NX 中工作），可采用不同方式指派工具、机床或固定设备。NC 编程人员使用 NX 库机制在 NX CAM 中检索或加载工具。通常，这些资源存储在 Teamcenter 数据库中。另外，NC 编程人员通过在工具创建对话框中指定工具参数，可以在 NX 加工模块中创建非库工具。当 CAM 零件保存在 NX 中时，MRL 资源列表就可供 Teamcenter 中对应的 MENCMMachining 规划工序调用。下面就介绍如何安装 MRL 资源列表：

1. MRL 的介质在 TC 的安装包中自带，找到名为 advanced_installations 压缩包并进行解压，然后右键选择 setup.exe 程序管理员身份运行（如图 1）：

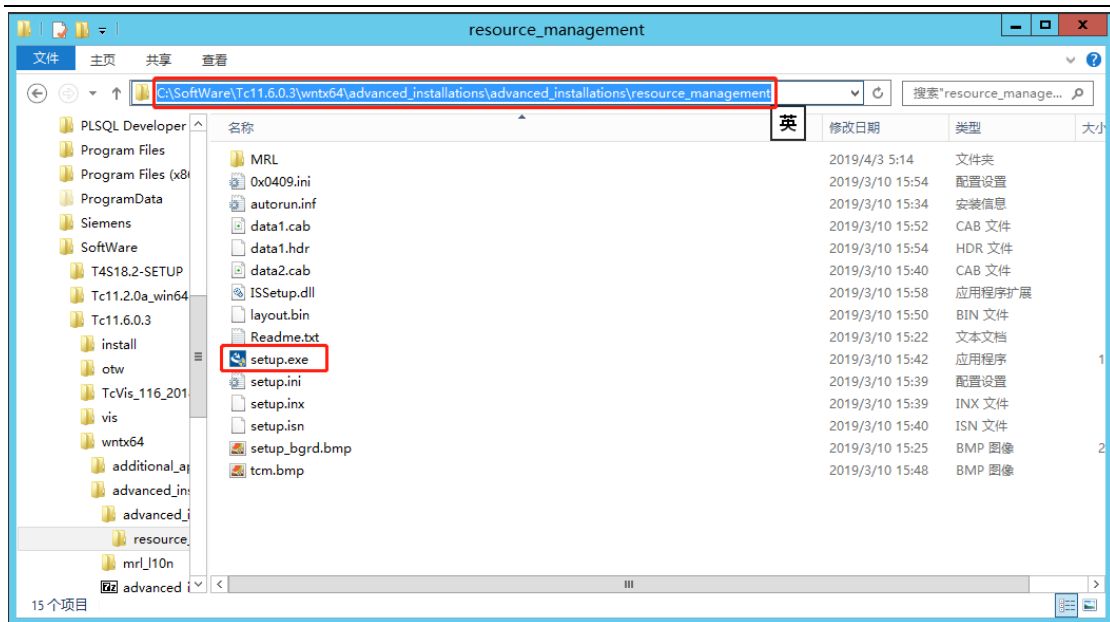


图 1

2. 点击 Next，进入下一步（如图 2）：

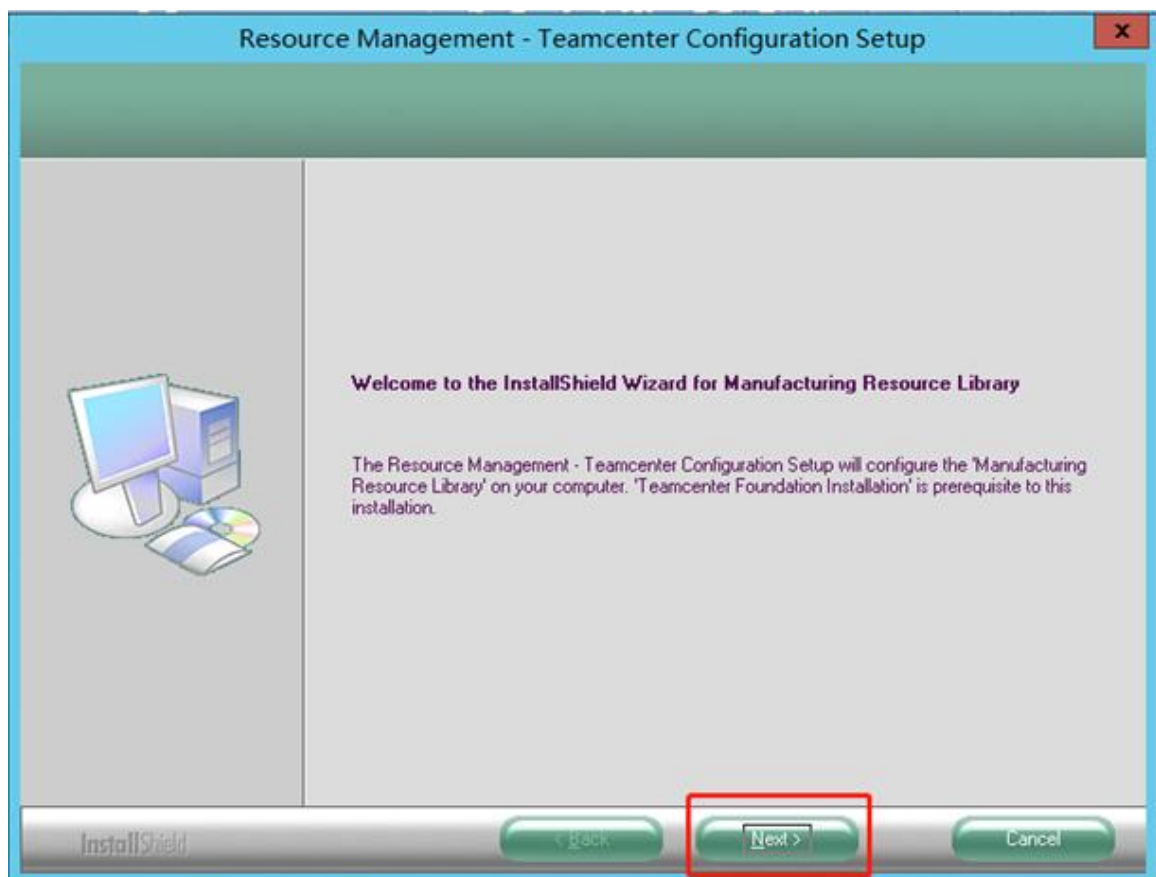


图 2

3. 在安装 MRL 资源列表前，一般我们会先创建 MRL 资源表的

用户组织，如下选择 Manufacturing resources-Configure MRL User，点击 Next 进入下一步安装，此处不展开（如图 3）：

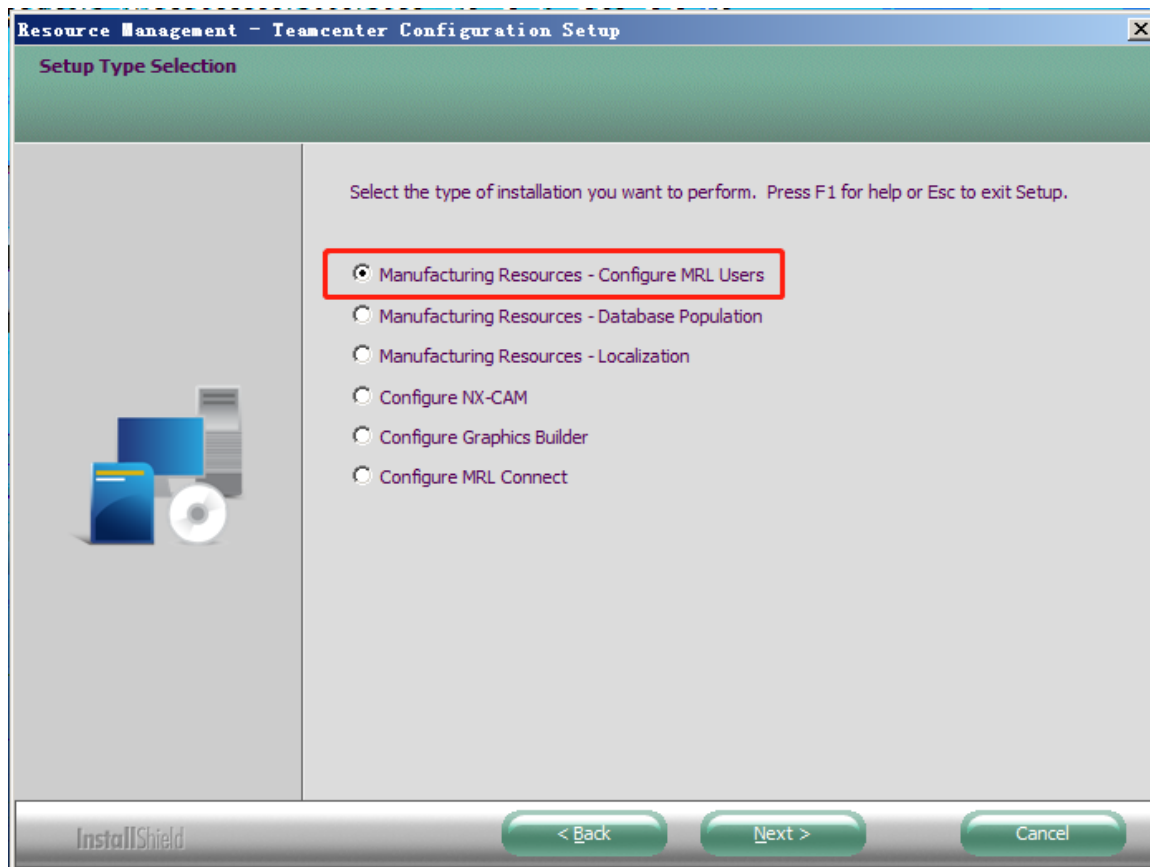


图 3

4. 完成创建 MRL 资源表的用户组织后，再回到此界面，选择 Manufacturing Resources-Database Population，开始安装分类层次结构（如图 4）：

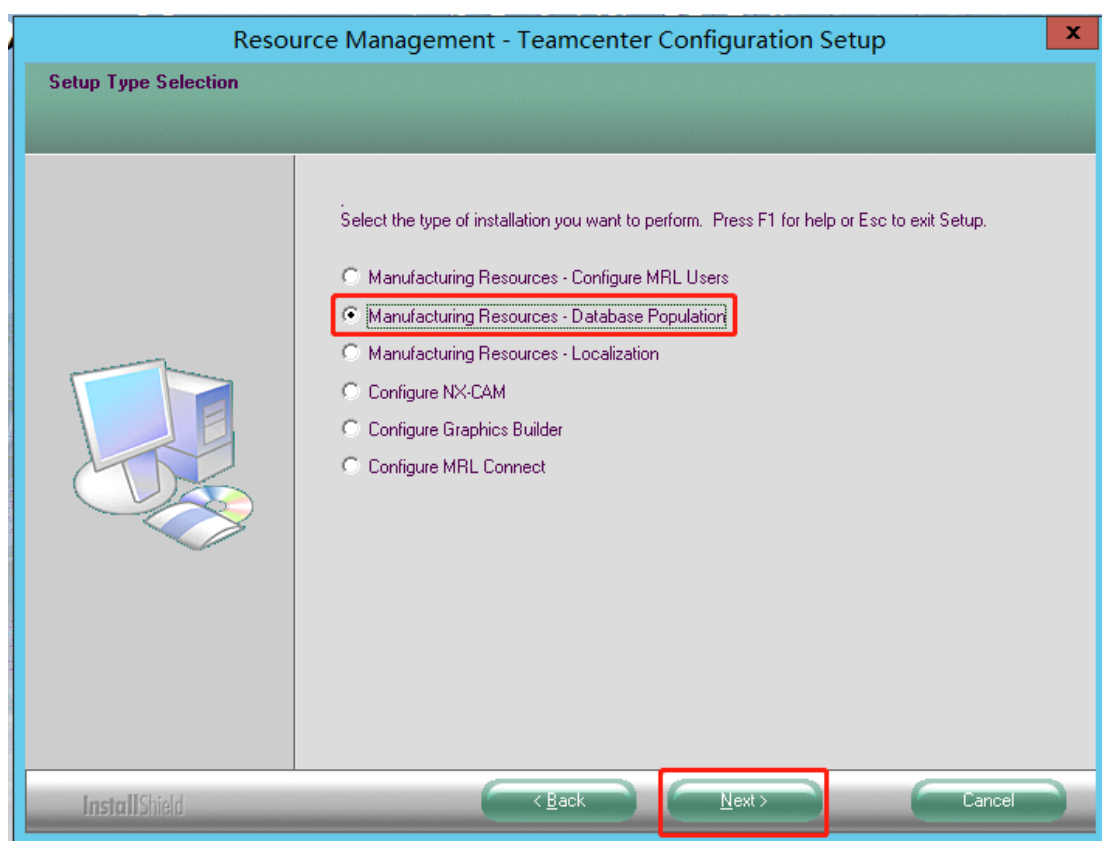


图 4

5. 选择 Database Population-Import Class Hierarchy，点击 Next，导入类层次结构（如图 5）：



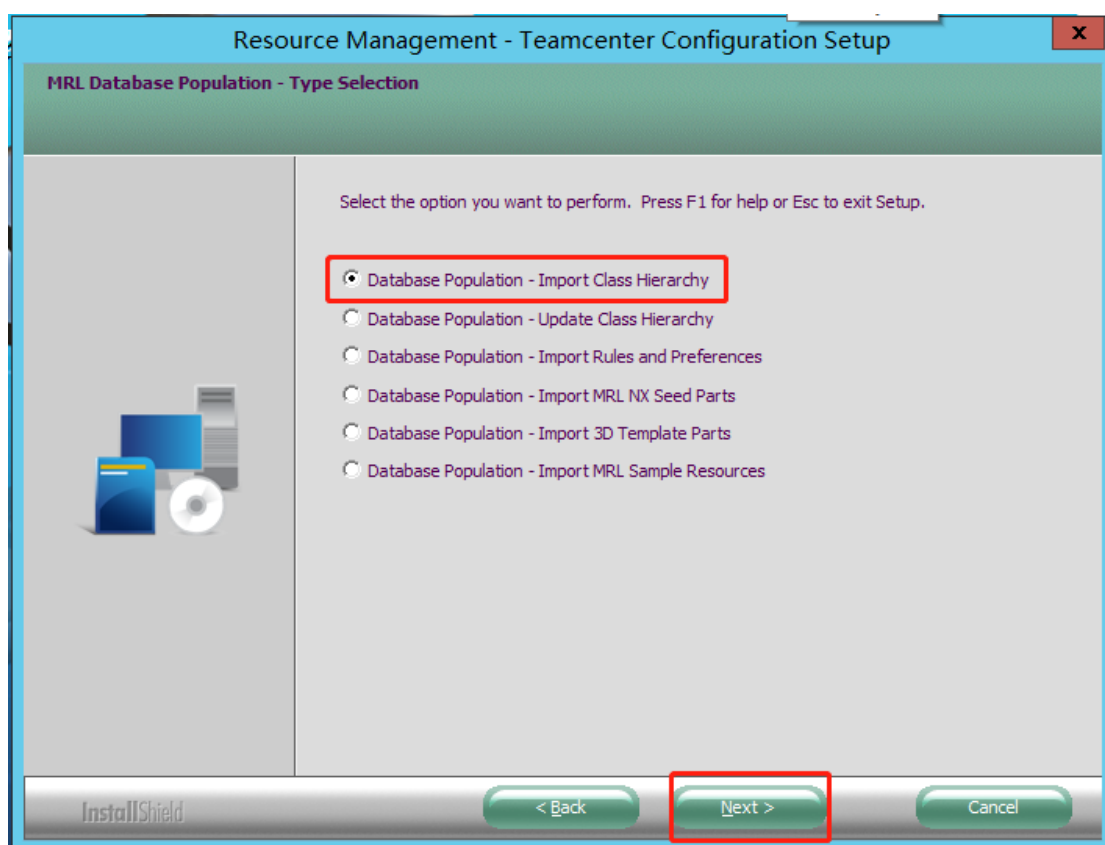


图 5

6. 选择 TC_ROOT 安装目录，点击 Next 下一步（如图 6）：



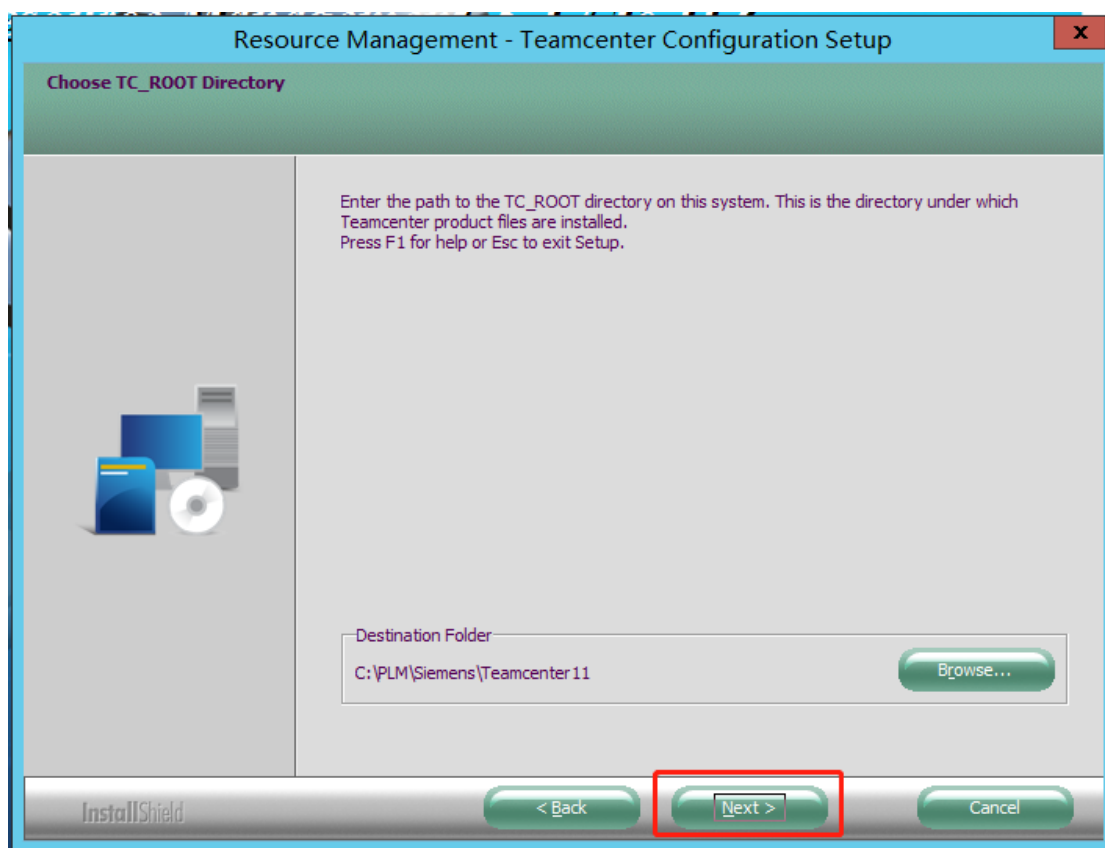


图 6

7. 选择 TC_DATA 目录，点击 Next 下一步（如图 7）：



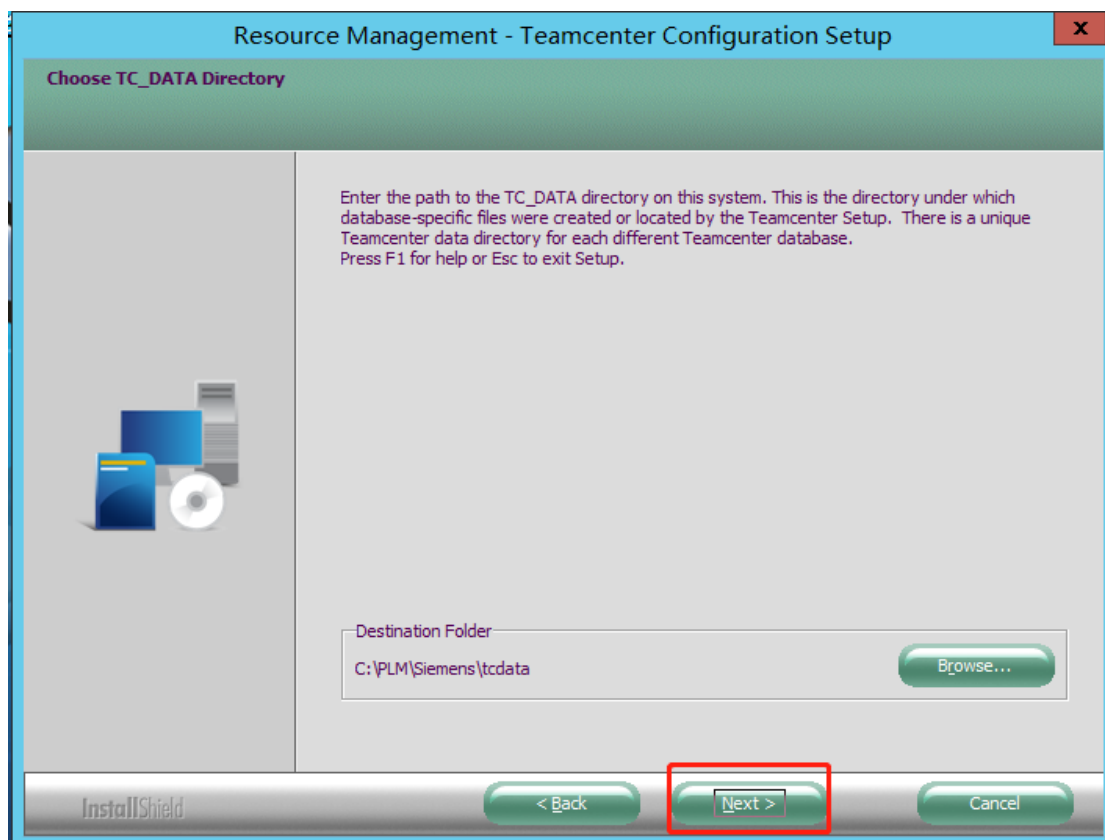


图 7

8. 输入登陆账户密码，点击 Next（如图 8）：

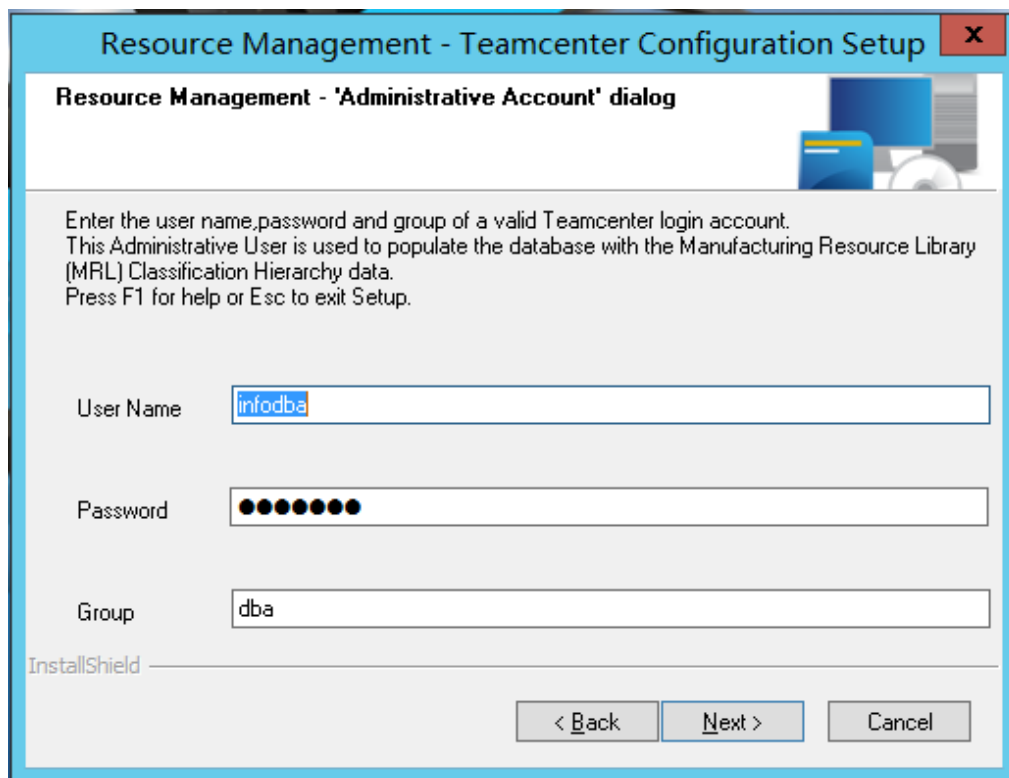


图 8

9. 按配置需求勾选即可，点击 Next 进入下一步（如图 9）：

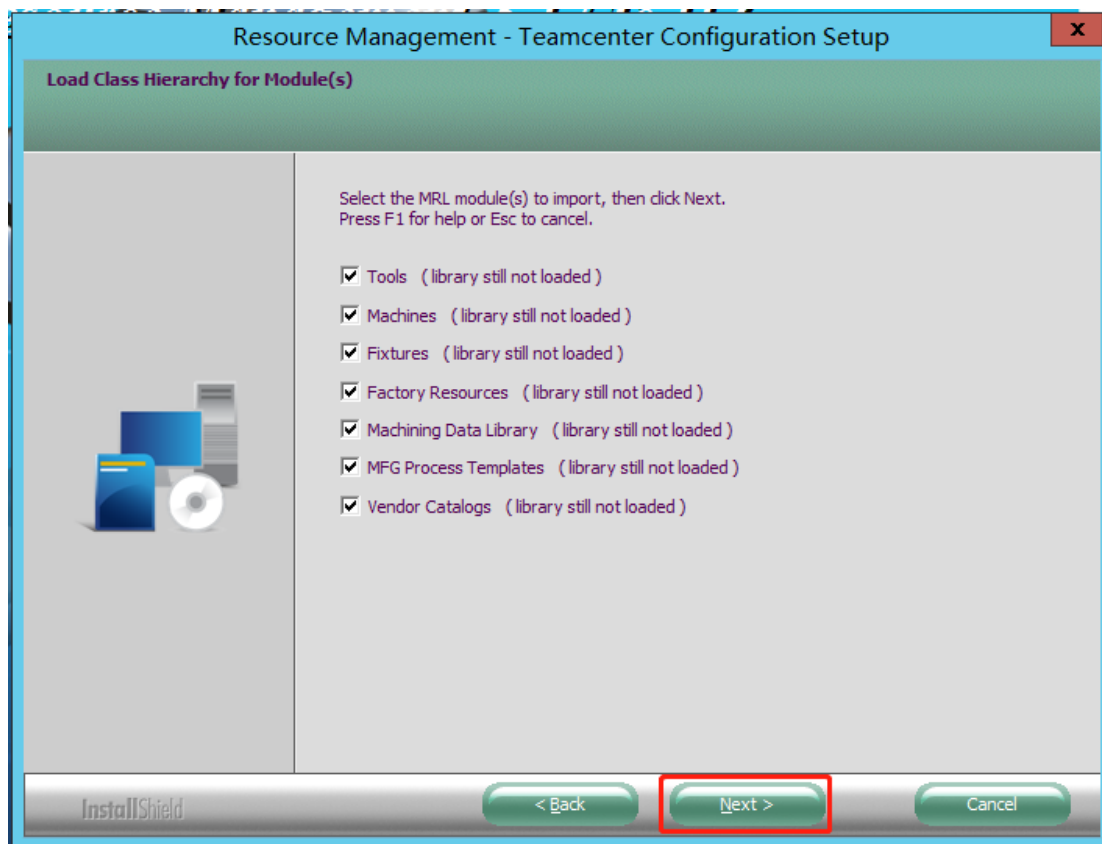


图 9

10. 点击 Next，进入安装 MRL 分类库（如图 10）：

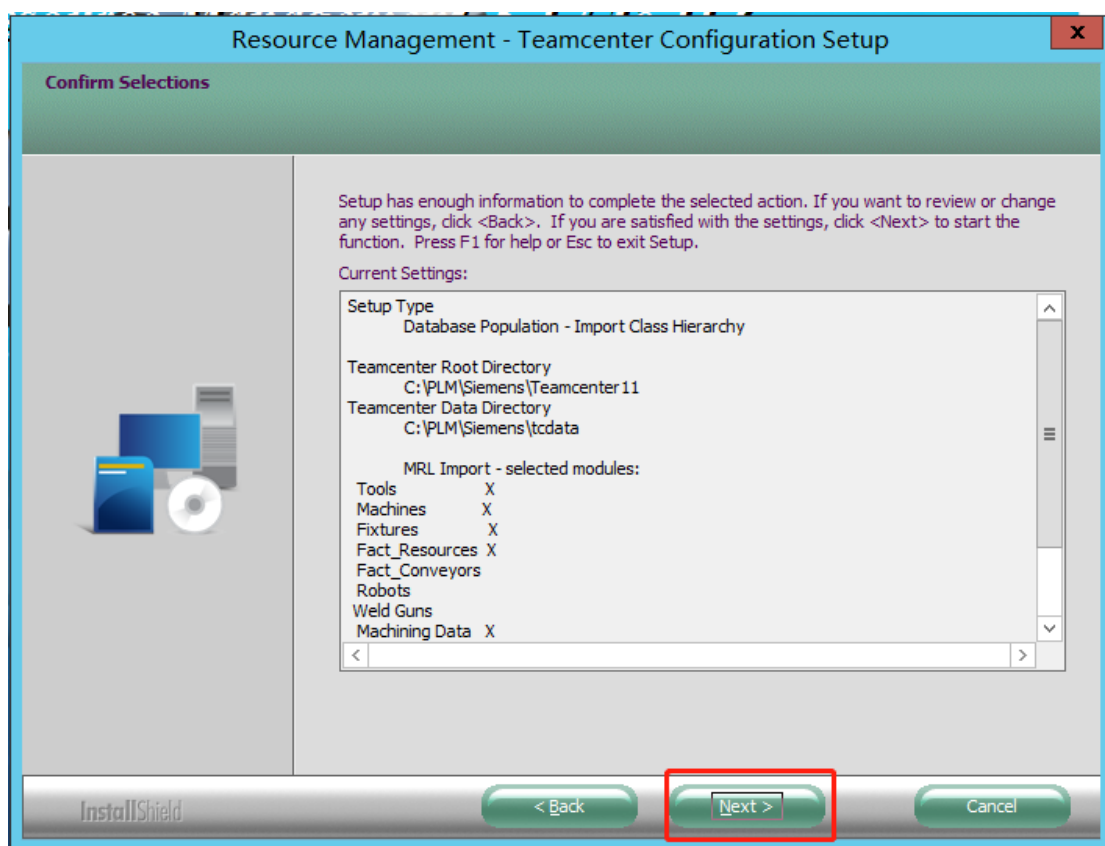


图 10

11. 安装完成后，在弹框点击“是”可回到安装模块选择界面，来安装 MRL 其他模块或导入资源数据（如图 11）：

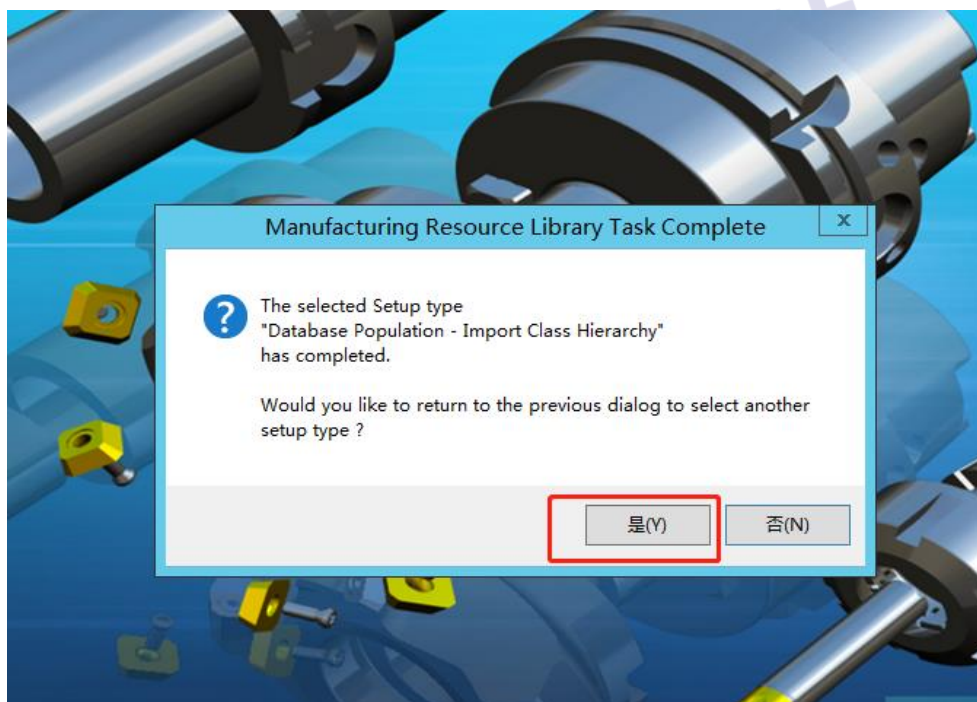


图 11

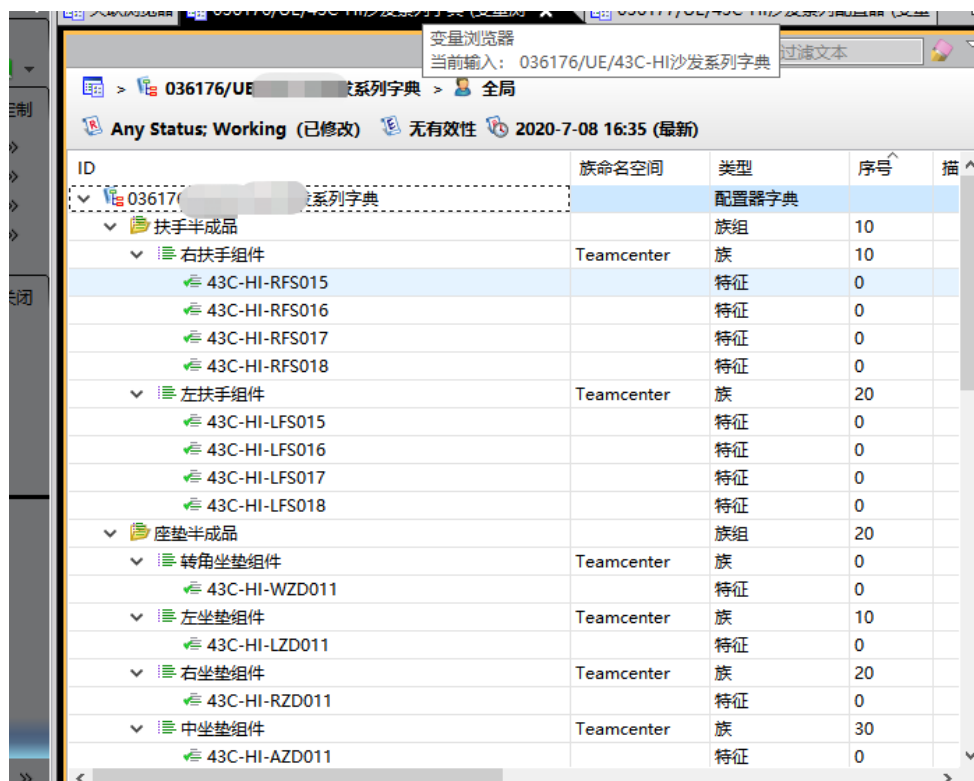
解决产品配置器选配无法正常显示

作者: 郑煜鑫 审校: 杨崇华

适用版本: TC11

在 TC11 以上的版本, 客户根据需求可以在产品配置器选择不同功能模块的不同功能部件, 来自动生成个性化 BOM, 从而达到减轻工程师 BOM 配置工作量, 快速生成满足客户需求的产品 BOM, 提高产品交付速度的目的。

前期工作需要新建字典与对应的配置器来配置选配条件和选配关系, 如下图 1, 2 所示;



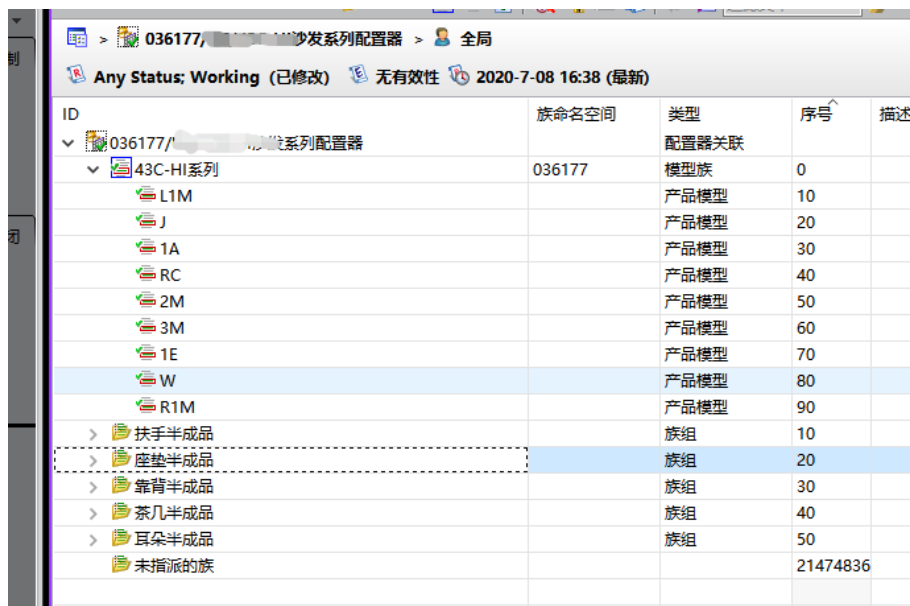
变量浏览器
当前输入: 036176/UE/43C-HI沙发系列字典

036176/UE 系列字典 > 全局

Any Status; Working (已修改) 无有效性 2020-7-08 16:35 (最新)

ID	族命名空间	类型	序号	描
036176/UE/43C-HI沙发系列字典		配置器字典		
扶手半成品		族组	10	
右扶手组件	Teamcenter	族	10	
43C-HI-RFS015		特征	0	
43C-HI-RFS016		特征	0	
43C-HI-RFS017		特征	0	
43C-HI-RFS018		特征	0	
左扶手组件	Teamcenter	族	20	
43C-HI-LFS015		特征	0	
43C-HI-LFS016		特征	0	
43C-HI-LFS017		特征	0	
43C-HI-LFS018		特征	0	
座垫半成品		族组	20	
转角坐垫组件	Teamcenter	族	0	
43C-HI-WZD011		特征	0	
左坐垫组件	Teamcenter	族	10	
43C-HI-LZD011		特征	0	
右坐垫组件	Teamcenter	族	20	
43C-HI-RZD011		特征	0	
中坐垫组件	Teamcenter	族	30	
43C-HI-AZD011		特征	0	

图 1



ID	族命名空间	类型	序号	描述
036177/...		配置器关联		
43C-HI系列	036177	模型族	0	
L1M		产品模型	10	
J		产品模型	20	
1A		产品模型	30	
RC		产品模型	40	
2M		产品模型	50	
3M		产品模型	60	
1E		产品模型	70	
W		产品模型	80	
R1M		产品模型	90	
扶手半成品		族组	10	
座垫半成品		族组	20	
靠背半成品		族组	30	
茶几半成品		族组	40	
耳朵半成品		族组	50	
未指派的族			21474836	

图 2

在结构管理器中，根据需求产品筛选条件，如下图 3，发现结构管理器中的结构并没有变化，此时需要在视图将显示未配置变量取消，如下图 4；

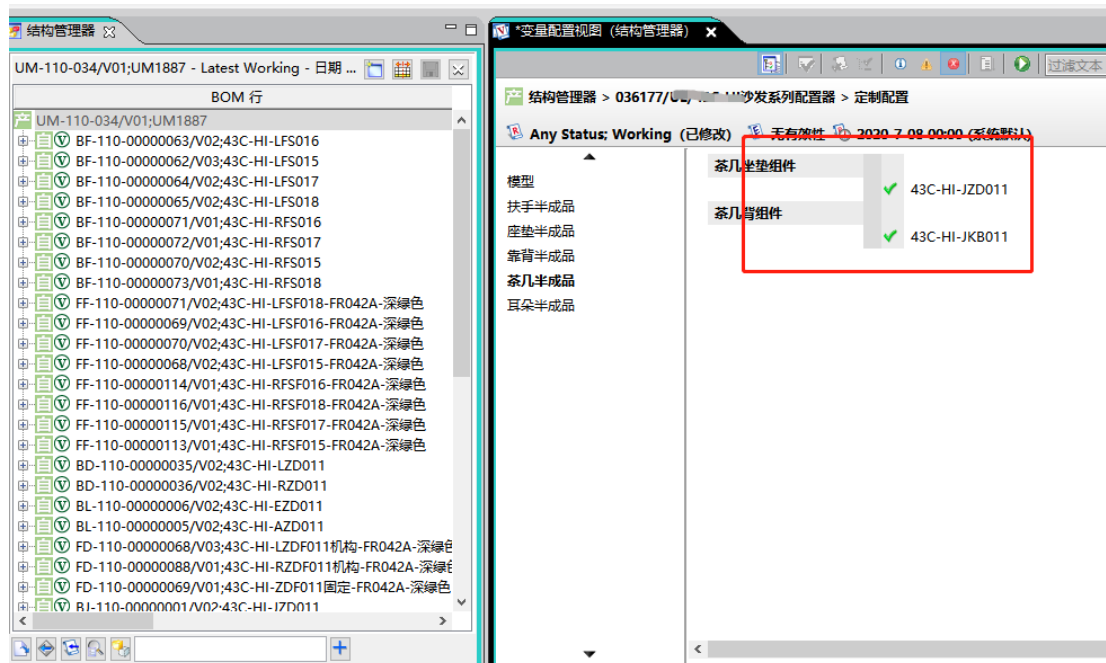


图 3

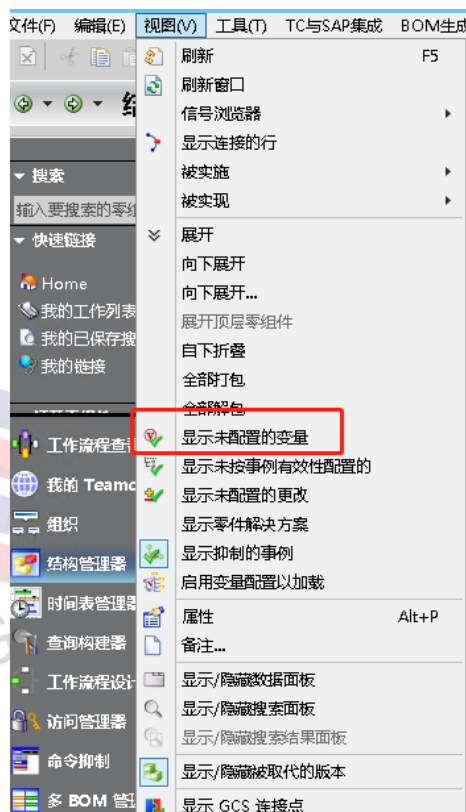


图 4

修改后，效果如下，根据选配条件过滤，如下图 5。

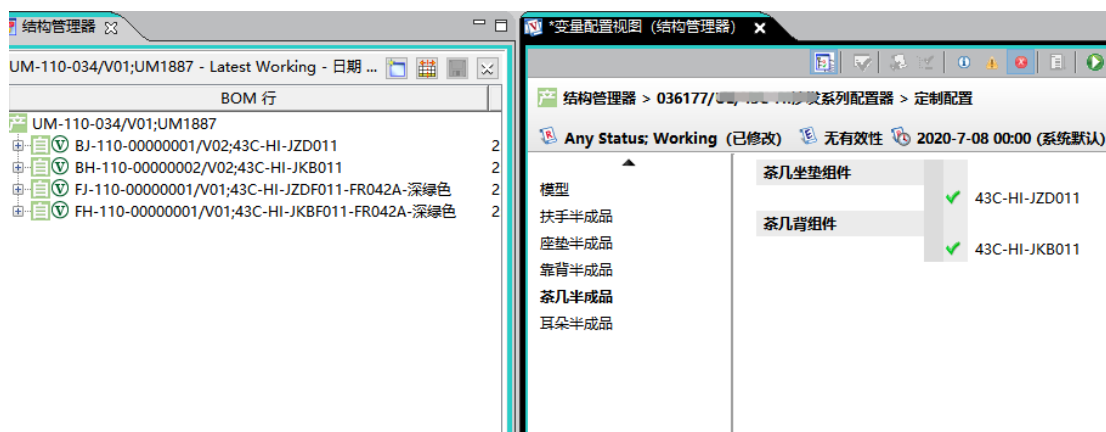


图 5

如何设定数据集的历史记录版本数量

作者：杨崇华 审校：郑煜鑫

适用版本：TC11

在 TC 中每次签出修改一次数据集都会生成一个新的版次，用户通过针对性的搜索可以查找到 TC 中储存的数据集历史版本，默认情况下 TC 中会储存数据集的三个历史版本，如果想要让系统多储存一些历史版本，则可以通过设置首选项去进行实现。

具体功能如下：

1、找到一个数据集，可以在其“版本限制”属性中看到最多储存的版本量，一般为默认的 3 个，如图 1 所示：



图 1

2、想要更改限制量，需要使用管理员账号找到首选项 AE_dataset_default_keep_limit，将其的值由默认的 3 改为想要的值(改动只对改动后创建的数据集有作用)，如示例改为 5，然后保存。如图

2 所示:

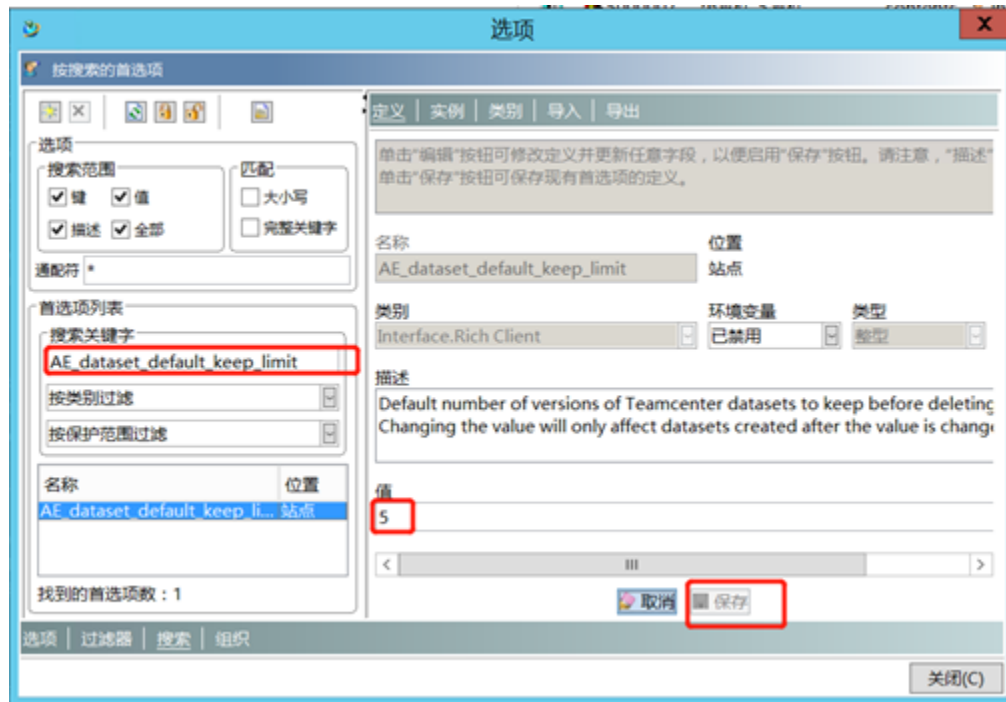


图 2

3、这时我们再去重新创建一个数据集，便可以看到数据集“版本限制”属性值由 3 变为 5 了，如图 3 所示:



图 3

4、这样我们再去修改数据集，就会在 TC 中最多储存 5 个历史版本，即可实现我们想要增加储存的历史版本的效果，这样在数据集的变化过程的追踪上就更加方便了。

如何通过时间表任务属性查询项目

作者：唐栋 审校：陆云龙

适用版本：Teamcenter11.5.0

很多时候我们会给时间表任务添加定制属性，如图 1。

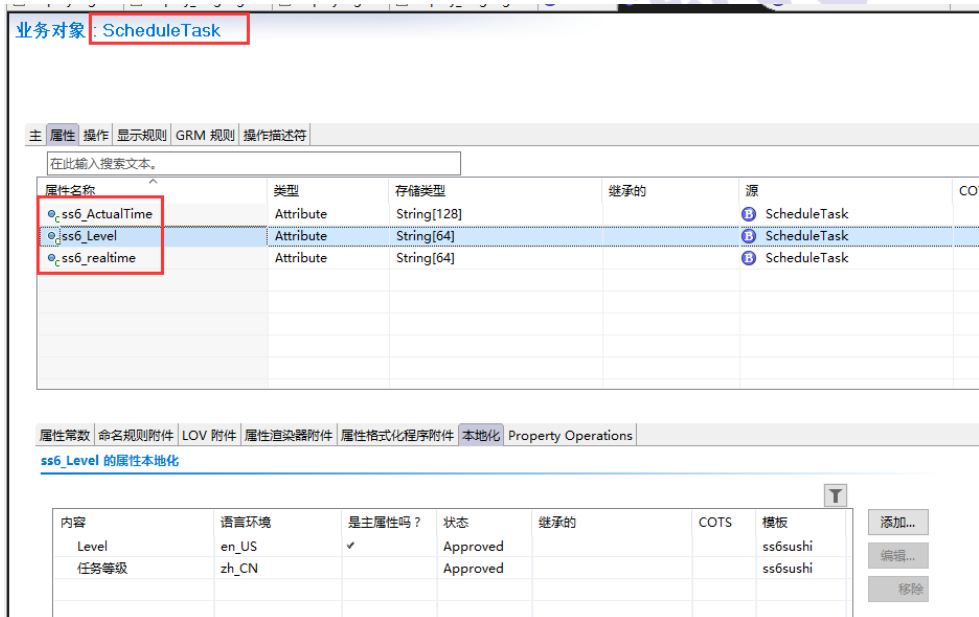


图 1

此时我们可能需要通过这些定制属性来查询这些时间表任务所在的项目，需要在查询构建器中进行配置，首先搜索对象选择 TC_Project，如图 2。

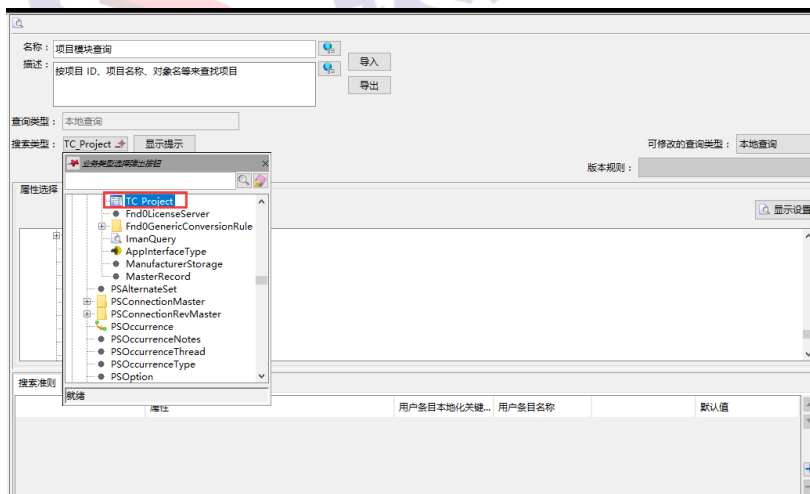


图 2

显示设置改为所有属性和真实名称，如图 3。



图 3

首先下拉至底部，选择 reference_by，双击，输入 Schedule，关系需要一层层遍历，所以先到时间表，如图 4。

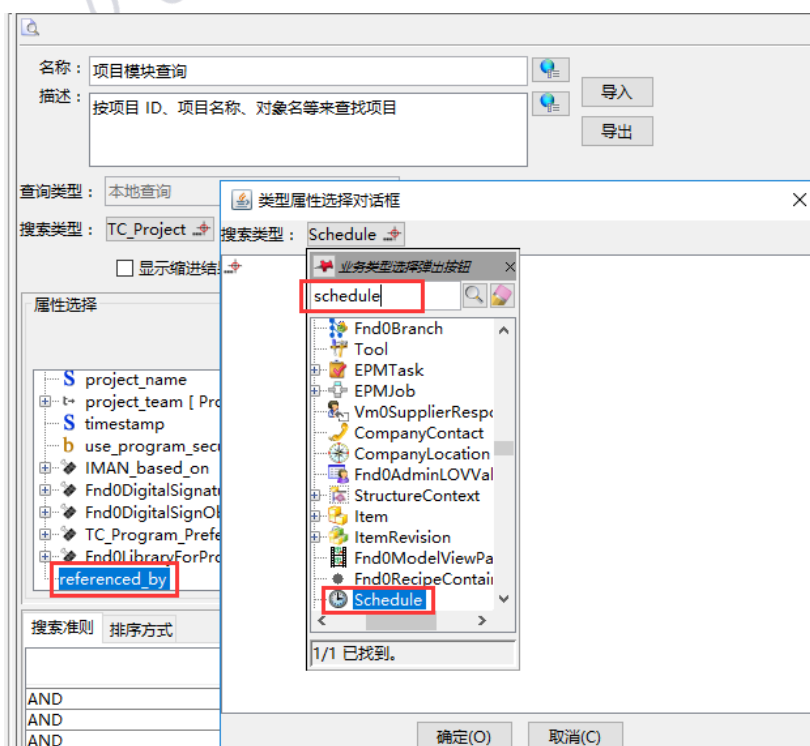


图 4

双击 Schedule，以为项目和时间表此时之间的关系式 project_list，所以选择 project_list，点击确定，如图 5。

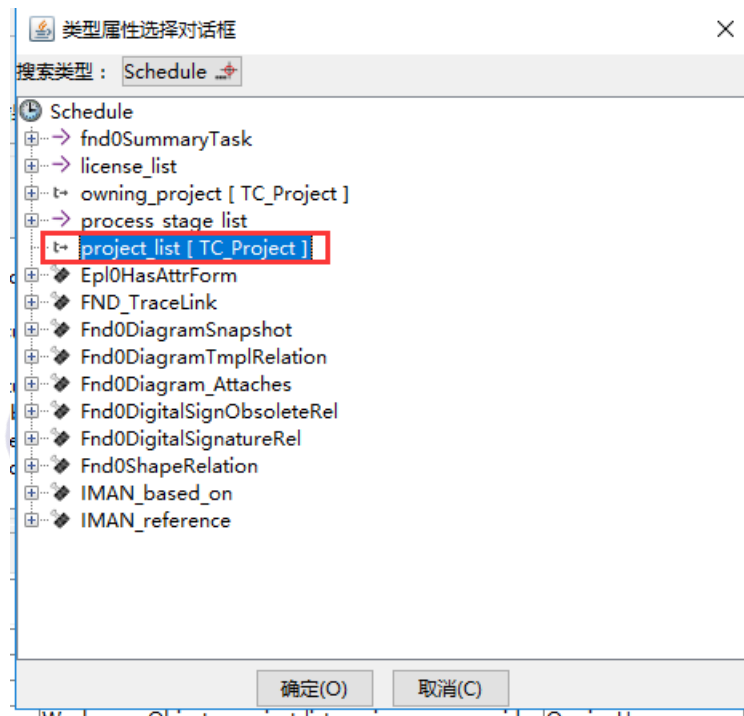


图 5

此时再下拉到底部，找到时间表的 `reference_by`，此时我们通过时间表的游泳者来搜索时间表任务，输入 `ScheduleTask`，选中双击，如图 6。

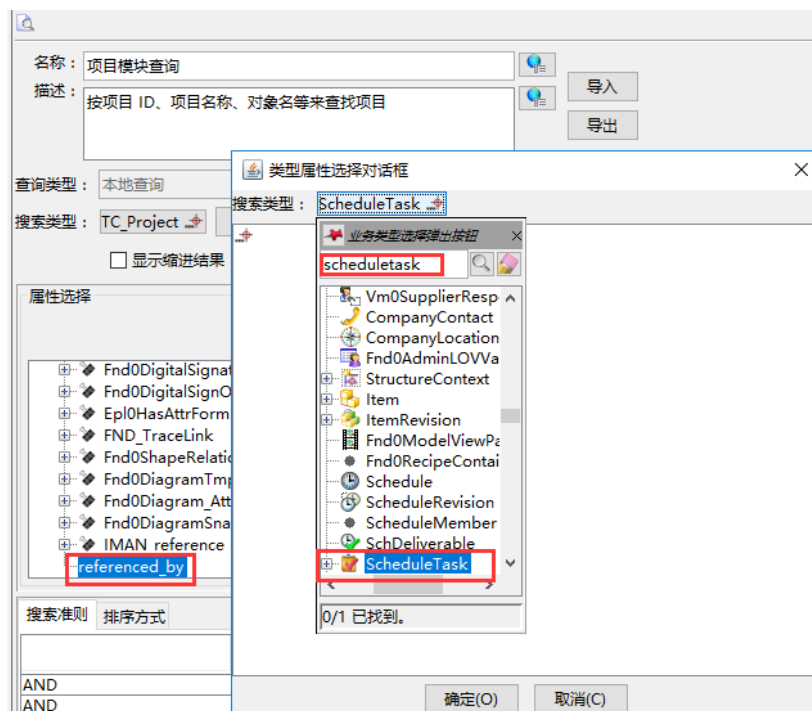


图 6

时间表和任务之间靠关系 Schedule_tag 挂靠，所以选择 Schedule_tag,点击确定 ,如图 7。

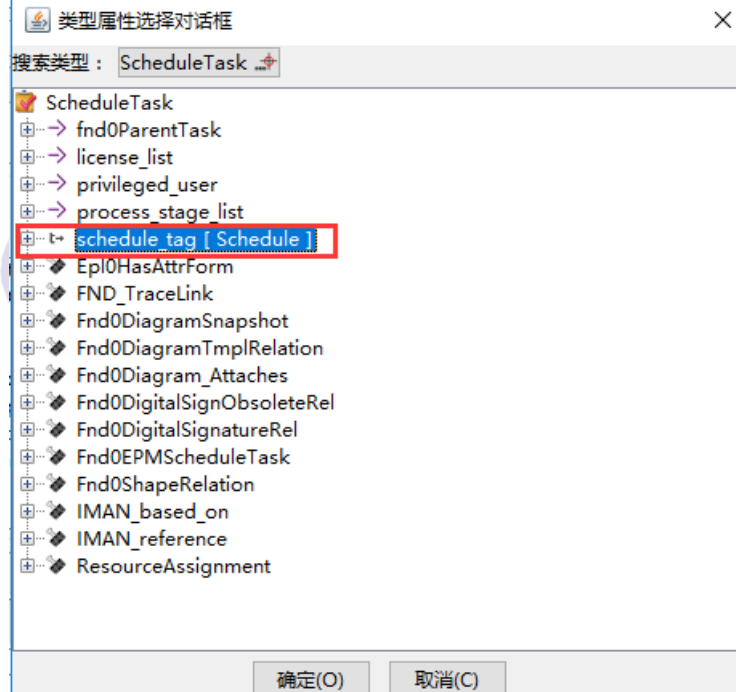


图 7

此时往下拉，则会显示我们定制的属性，选择对应属性即可，如图 8。此时就创建成功了。

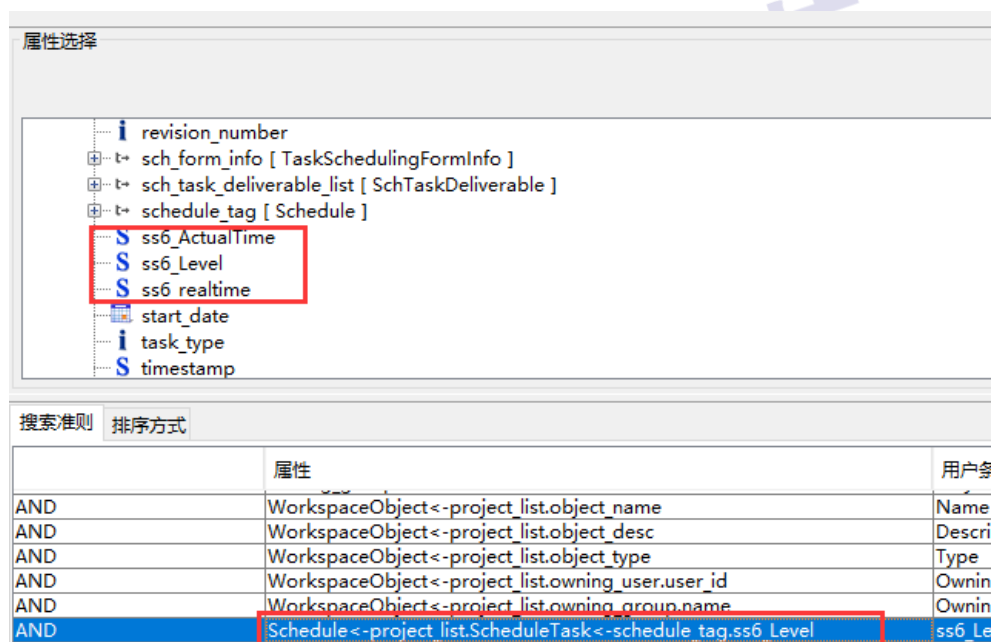


图 8

Process Simulate on Teamcenter 机器人运动数据配置

作者：钟耀洪

审校：张崇宏

适用版本：Process Simulate V15 版本 Teamcenter V11

Plant Simulation 与 Teamcenter 是具有标准集成接口的，接下来本期的技术期刊我们介绍的就是 Plant Simulation 与 Teamcenter 的集成方法。

1、通过“机器人配置”选项，您可以查看和教导在机器人操作中到达选定位置的反向解决方案。当将机器人分配给操作时，系统会针对每个位置计算并显示解决方案。Process Simulate 可以读取机器人实例下的 motionparameters.e 运动参数数据集。

2、如果在 Process Simulate on Teamcenter 中机器人实例下没有含有 motionparameters.e 文件的数据集，机器人将无法做机器人轨迹点的示教与进行自动路径规划，机器人离线编程也无法完成全部设置。

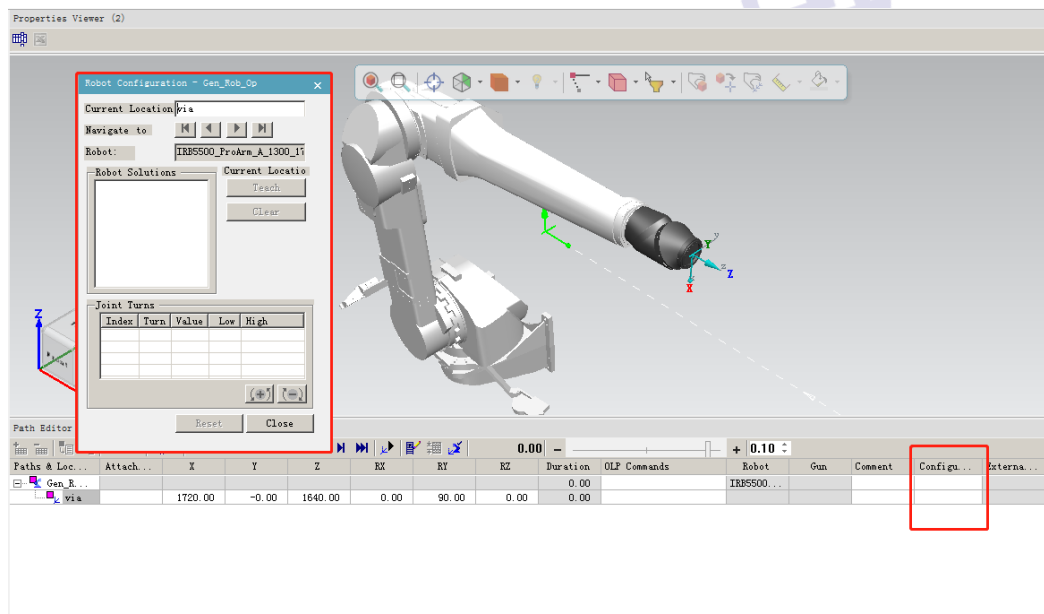


图 1

3、在机器人实例下添加 ProcessSimulateMotionParams 数据集，并将机器人对应的 motionparameters.e 文件上传至 TC。



图 2

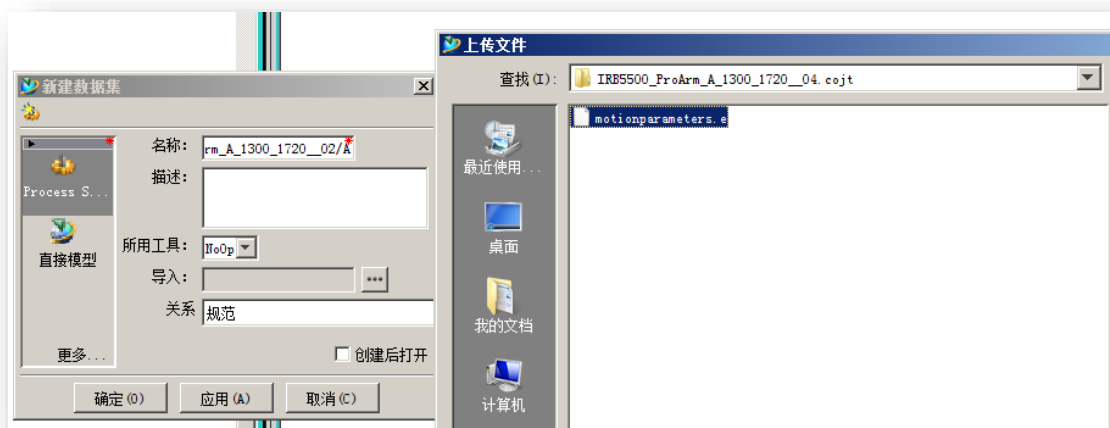


图 3

4、机器人运动参数文件 motionparameters.e 上传完成后，Process Simulation 中的机器人就可以进行相应的轨迹点示教。

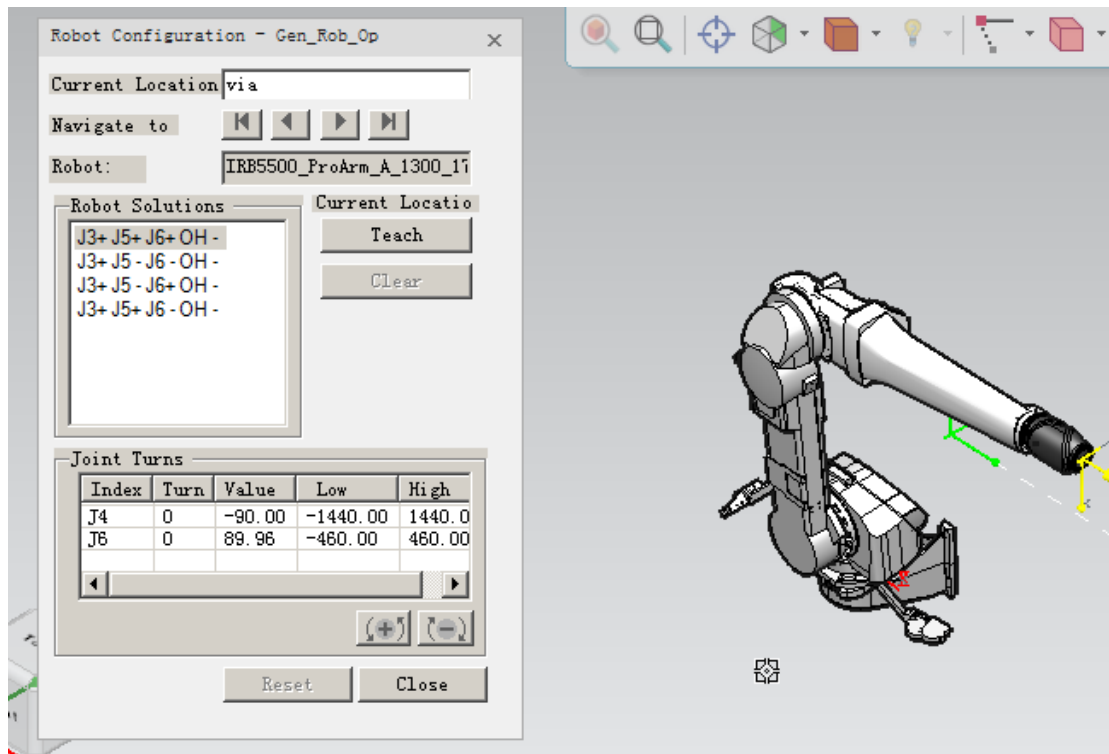


图 4

关于联宏

作为一家高科技咨询服务类企业，上海联宏创能信息科技有限公司是在优宏信息技术有限公司的基础上成立的专注于 Siemens PLM Software 工业设计软件全线产品的专业企业，并兼营当今世界著名的电气设计软件 EPLAN、仿形切割 CAD/CAM 系统 SigmaNEST 等。现已成为 Siemens PLM Software 最高级别的专业授权代理商，拥有铂金代理资质；同时，是 EPLAN、SigmaNEST 在中国的重要代理商。专注于为汽车、通用机械、消费电子、航空航天、船舶等机械制造业领域内的广大用户提供完整的数字化产品工程解决方案及全方位资深咨询服务。

我们的员工队伍由从事多年工业设计软件业务的专业人士组成，拥有丰富的行业经验和为客户服务的赤诚之心。共同的信念和目标使我们共聚并组成一支精良的销售、技术支持与应用服务团队。我们恪守“诚信、专业、奉献”的企业信念，以优良品质和快速响应为目标，致力于数字化产品工程解决方案的推广应用，并已为企业用户提供了相应的咨询服务、技术培训、软件安装、售后支持、业务外包、系统集成等综合服务。公司在全国多个重点城市建立分支机构，更快捷的服务客户。

关注客户利益，服务客户所需，实现与客户的共赢互利是我们最大的愿望。

请相信，联宏，是您值得信赖的合作伙伴！

联系我们
Contact us



上海联宏创能信息科技有限公司

Shanghai United Grand Info-tech Co.,Ltd.

➤ **上海总公司地址:**

上海市浦东新区耀元路 58 号环球都会广场 3#楼 15 层, 200125

15th Floor, 3 # Building, International Metropolitan Plaza, 58 Yaoyuan Road,
Pudong New Area, Shanghai

电话 Tel: (021)5103 5212

➤ **天津分公司地址:**

天津市河西区郁江道 21 号 一号楼 305 室, 300220

Add:Unit 305 No.1 BuildingNo.21YuJiangRd.,HeXiDistrict,Tianjin,300220,PRC

电话 Tel: (022)2816 2058 传真 Fax: (022)2816 2098

➤ **重庆分公司地址:**

重庆市北部新区金童路 251 号(奥林匹克花园十期)19 幢 6-2 室邮编: 401147

Add:Room 602 Unit 19,No.251 JinTong Road, North New District ,Chongqing
401147,PRC

电话 Tel: (023)6308 7957 传真 Fax: (023) 6308 7957

➤ **北京分公司地址:**

北京市海淀区西北旺东路 10 号院 5 号楼中关村互联网创新中心, 100193

Add: Zhongguancun Internet Innovation Center, Building 5, No.10 Xibeiwang
East Road, Haidian District, Beijing 100193, China

电话 Tel:(010)5874 1907 传真 Fax:(010) 3133 8568



关注我-就扫扫我