

联宏电子期刊

第三百六十八期—201228



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 20201228

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 Sigani 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



目录

NX

因属性映射问题导致模型上传失败的解决方法	6
自定义属性模板的方法	12
NX 后处理打开报错处理方法.....	16
Win10 教育版后处理构建器双击无反应的解决方法	19
NX 二次开发-如何往表格中插入自定义符号	21
NX 二次开发-制图模块获得指定点所在投影面	23

TC

EWI 中查看动画	25
SE 集成连接 TC 时加载策略文件失败的解决办法	28
BMIDE 配置对象版本“操作”	31
TC 本地化数据库导出	35
TC 审计功能不同版本切换	38
如何查询处于流程中的零组件	41

TECNO

PDPS 部署问题-Oracle 字符集不匹配.....	44
关于联宏.....	47

因属性映射问题导致模型上传失败解决方法

作者: 王刚 审校: 陈昂

适用版本: TC(10.0)&NX(10.0)集成

对于很多使用 TC&NX 集成环境进行产品设计的企业来说,从线下导入 NX 模型数据时,有时会报下图 1 所示的错误。出现这种情况一般都是 NX 与 TC 属性映射出了问题,一般我们需要从以下几点去排除:

- 1、确认 Business Modeler IDE(以下简称 BMIDE)里是否增加了报错信息里提到的属性;
- 2、BMIDE 修改后否进行模板部署(同步到 TC);
- 3、是否将属性映射表(txt 文本格式)上传到 TC。以上几点确认并修改后,一般情况都能解决此问题。

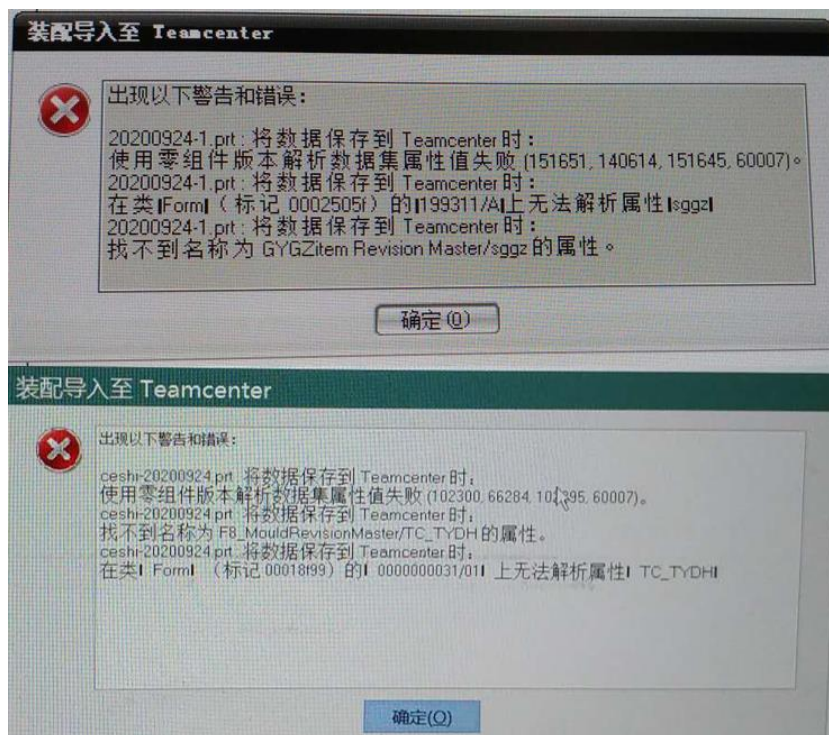


图 1

接下来为大家讲解 TC 里创建属性并与 NX 进行属性映射的流程。

具体操作方法如下：

一、在“BMIDE”里增加属性

1、在服务器端双击打开“Business Modeler IDE”，在左侧业务对象里找到要添加属性的零组件类型，随后双击进入右侧业务对象窗口，如下图 2 所示。

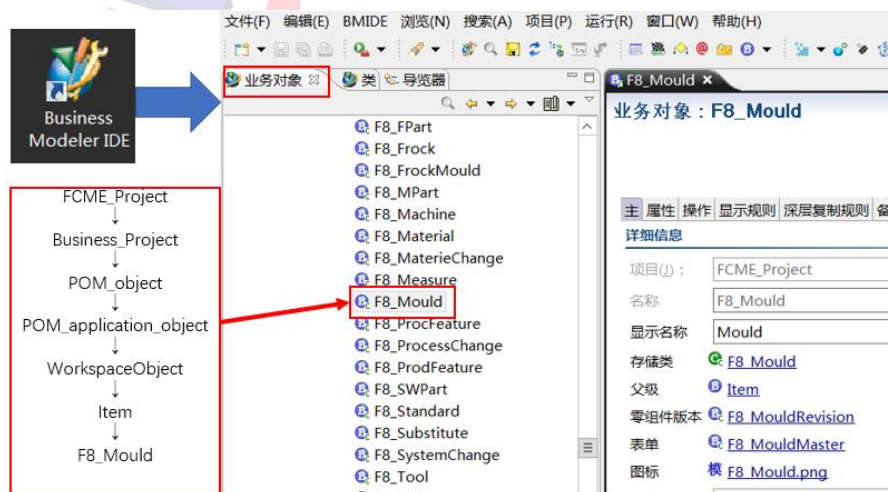


图 2

2、在右侧“F8_Mould”窗口选中“F8_MouldRevisionMaster”表单，随后点击下方窗口的属性，如图 3 所示。

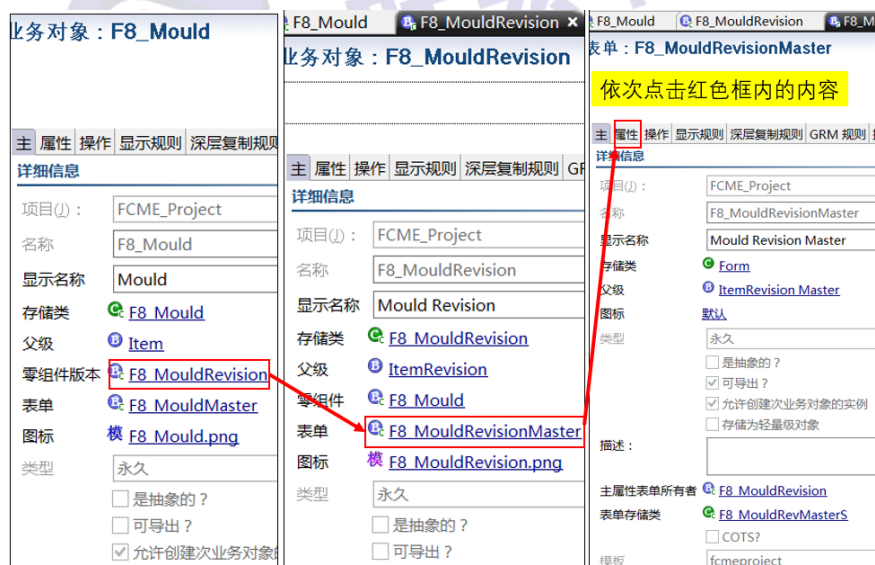


图 3

3、在“F8_MouldRevisionMaster”属性对话框，点击“添加”按钮进行添加属性，具体如下图 4 所示。

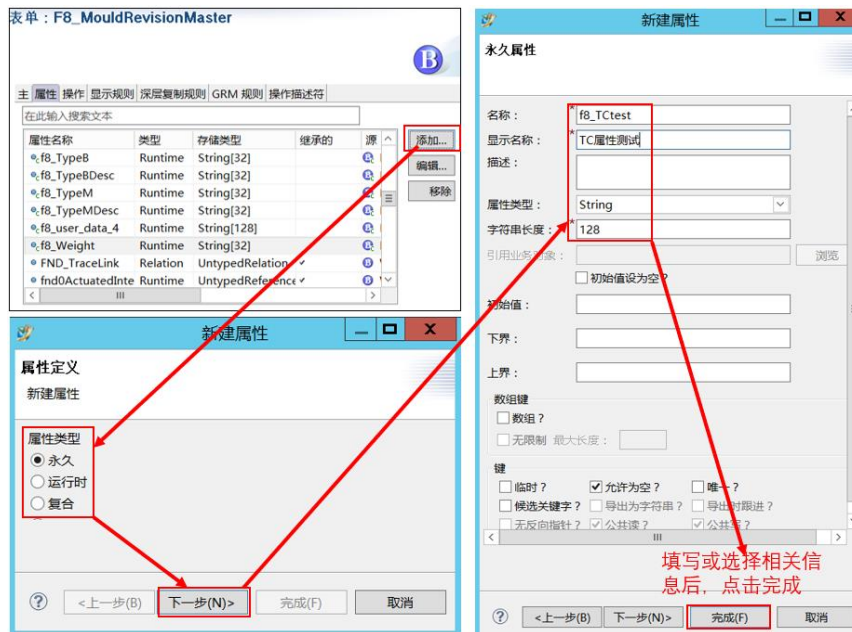


图 4

4、随后选中刚添加的属性，进行本地化设置，具体如下图 5 所示。

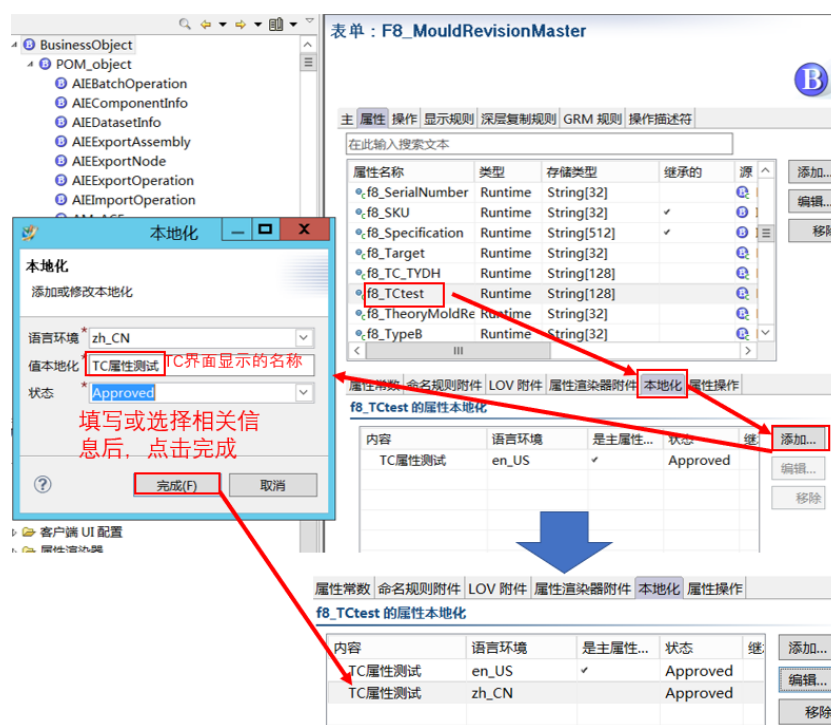


图 5

5、在“BMIDE”里进行模板部署，具体如下图 6 所示。

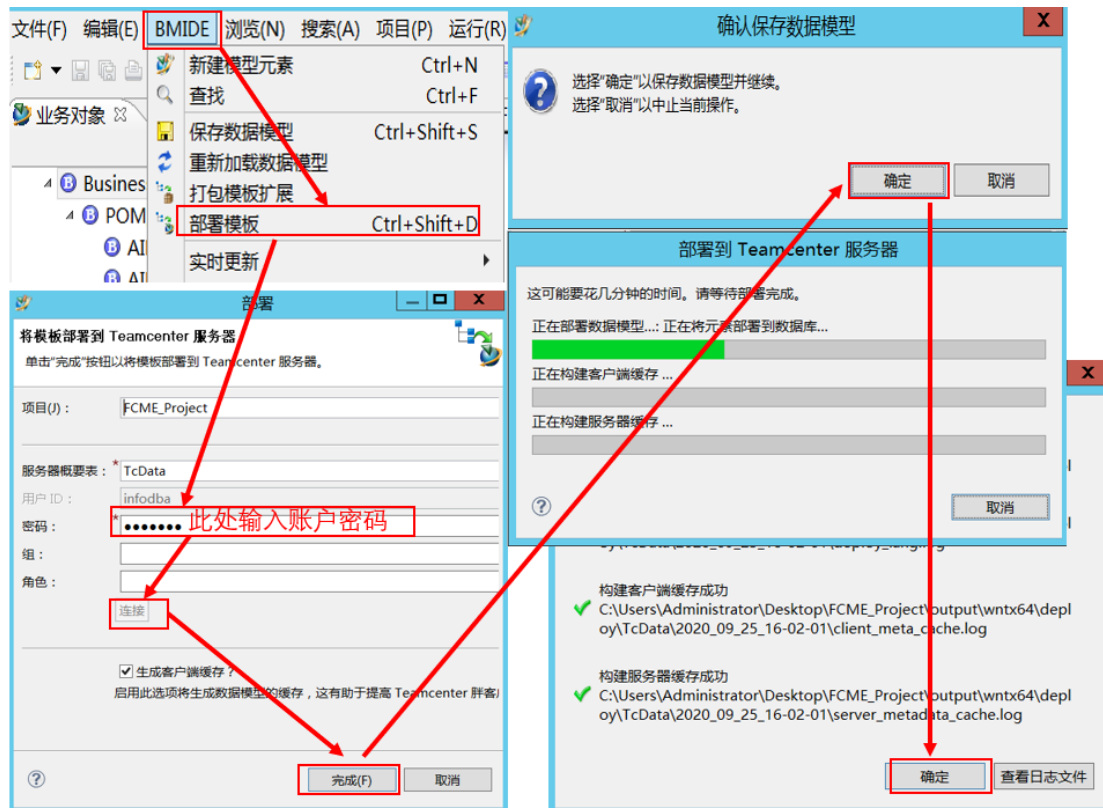


图 6

二、进行属性映射

1、打开 tc_config Command Prompt，输入（右键粘贴）命令
export_attr_mappings -u=infodba -p=infodba -g=dba -file=D:\mapping.txt,
此步骤为导出 NX 与 TC 属性映射文件，如下图 7 所示。



图 7

2、在 D 盘找到导出的“mapping”文件，打开并在“UGMASTER”数据集下 “F8_Mould”零组件类型下添加属性，具体如下图 8 所示。

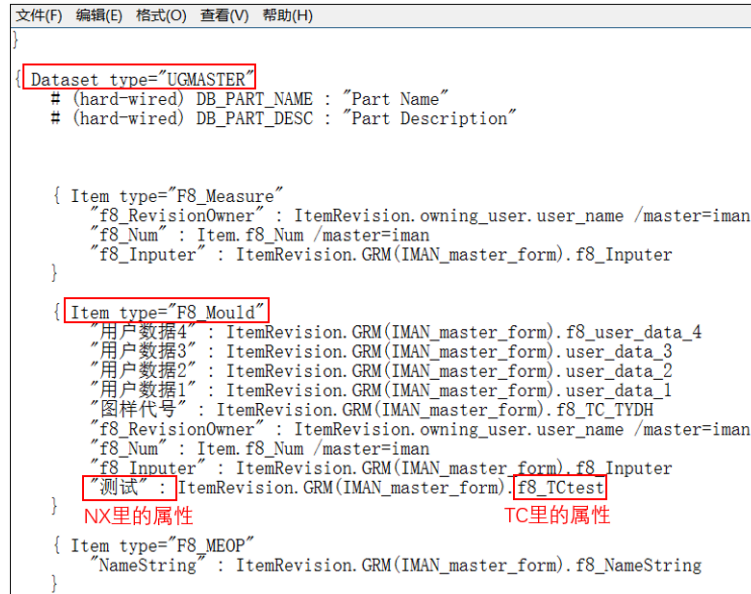


图 8

3、修改好映射文件后，在 tc_config Command Prompt 里输入命令 `import_attr_mappings -u=infodba -p=infodba -g=dba -file=D:\mapping.txt`，将修改好的映射文件导入 TC，如下图 9 所示。



图 9

三、模型导入测试

将带有“测试”属性名称和值的模型导入 TC 进行测试,测试结果如下图 10 所示,没有再出现报错的现象。

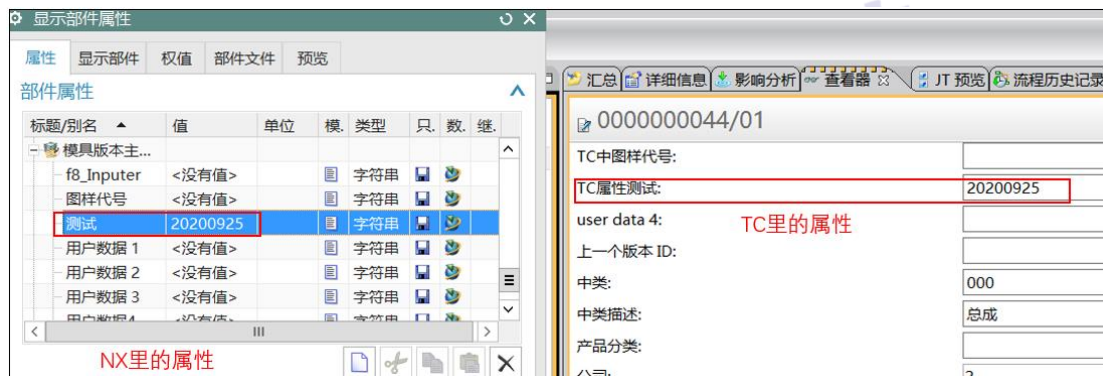


图 10

以上就是因属性映射问题导致模型上传失败的解决方法,希望对大家有所帮助。

自定义属性模板的方法

作者：李威 审校：陈杰

适用版本：NX10 及以上

在制作 NX 模板时，每一个模板我们都需要进行添加属性，NX 提供了一个属性模板的命令，我们只需要定义一次属性模板，以后我们在 NX 中新建任何模板文件，定义的属性都会自动添加进去；

1. 在 NX 中点击文件-实用工具-属性模板，将属性模板打开，如下图所示：

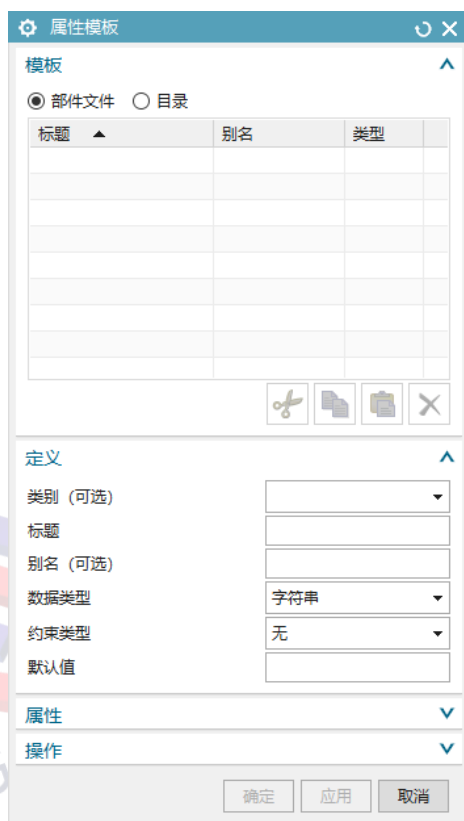


图 1

2. 在模板组下，将按钮切换到目录，在定义组下分别填写属性类别、属性标题，点击应用。创建属性完成之后，点击导出至目录按钮，如下图所示：



图 2

3. 将属性模板配置文件导出(建议配置文件的存放位置在 NX 根安装目录下), 取名 NX Attribute, 如下图 3 所示:

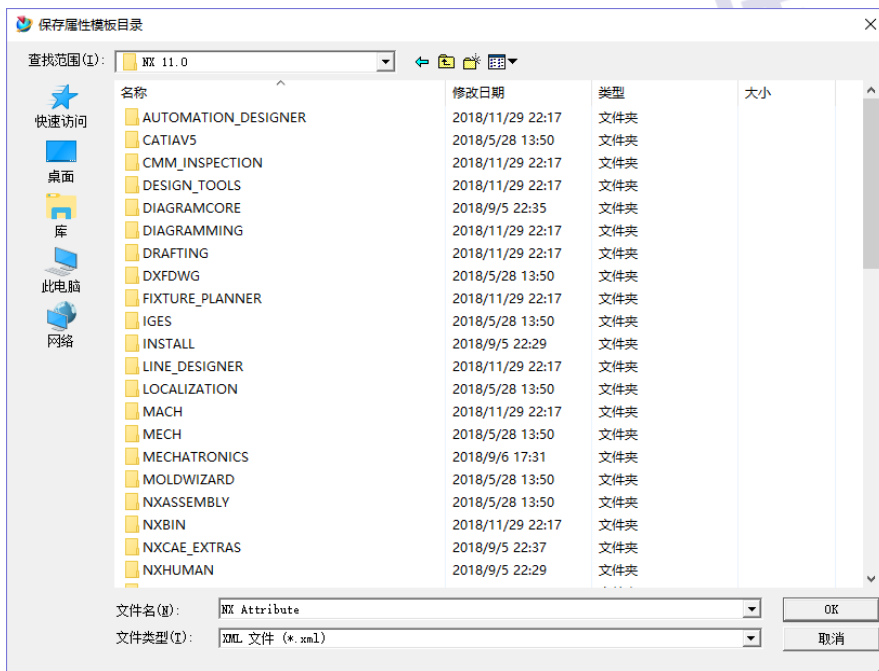


图 3

4. 最后，我们打开用户默认设置，选择基本环境，进入到用户属性界面，点击模板组下浏览按钮，如下图 4 所示：

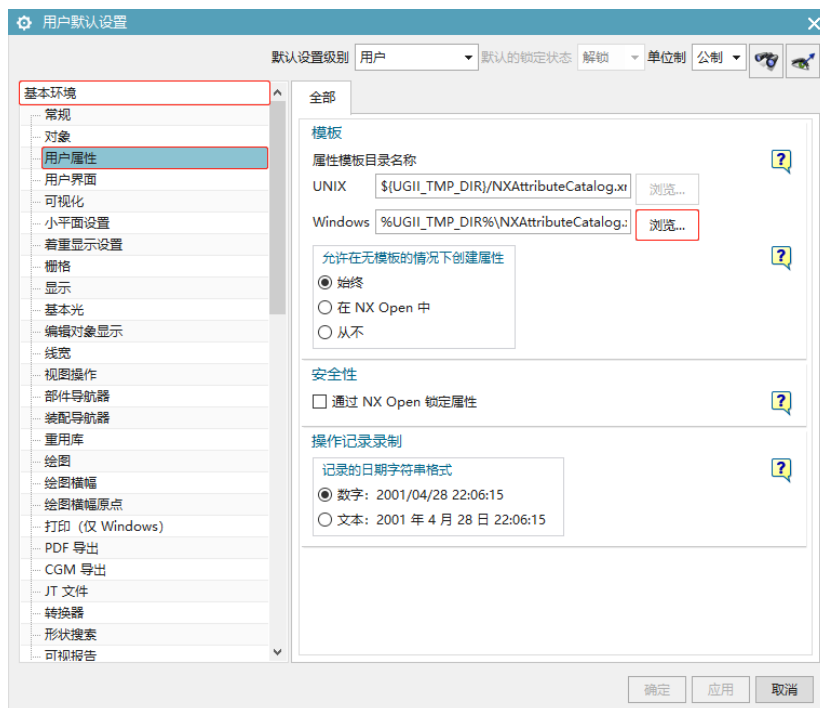


图 4

5. 将文件路径定位 NX 安装目录下，选择我们刚刚创建的属性模板配置文件，点击 OK，如下图 5 所示：

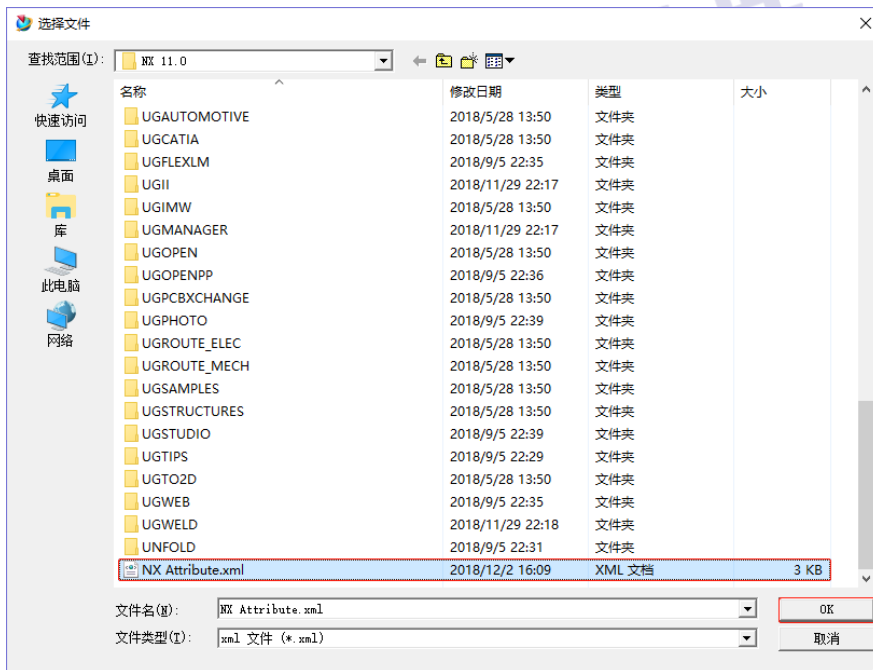


图 5

6.将用户默认设置进行确定保存，重启 NX，选择任意一个模板进入，查看属性，就会发现我们制作的属性就已经进入到模板中来了，如下图 6 所示：

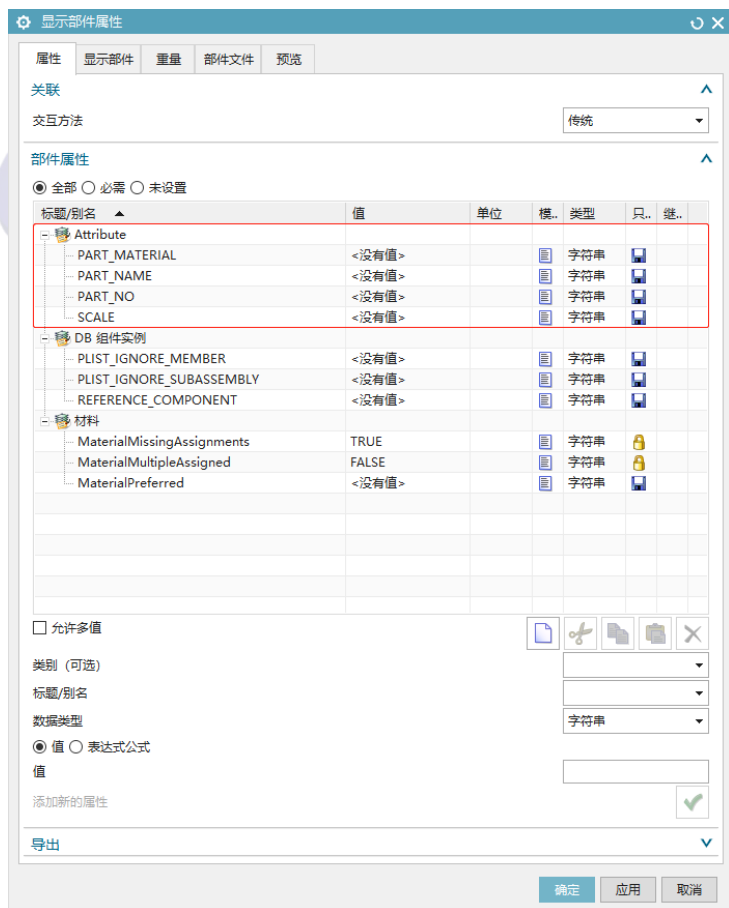


图 6

NX 后处理打开报错处理方法

作者：俞文清 审校：陈克荣

适用版本：NX6.0

由于 NX 后处理需满足各种不同类型的机床及不同的加工方式需求，所以工程师往往需要经常修改。但随着时间推移，人员更迭，修改的人员不了解当初编写时的环境，打开后处理时会出现报错。下面介绍几种打开后处理时比较常出现的报错及解决方法。

1.用较新版本的后处理构造器创建的后处理不能在该版本中打开。如下（图 1）。

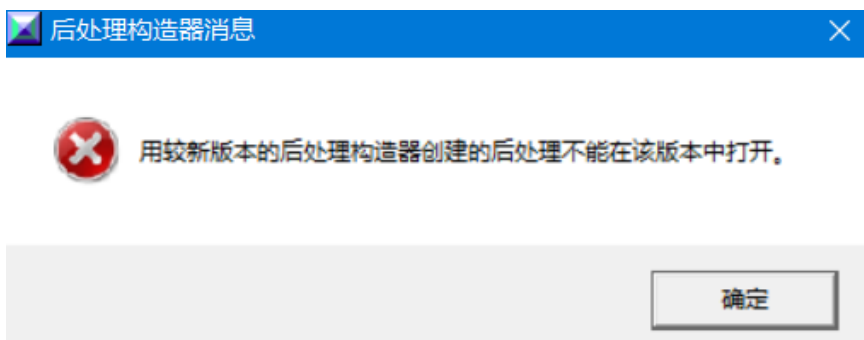


图 1

报错原因：这是由于后处理打开的版本低于编写时的版本导致。

查看后处理版本的方法：PUI 文件使用写字板方式打开如下（图

2)

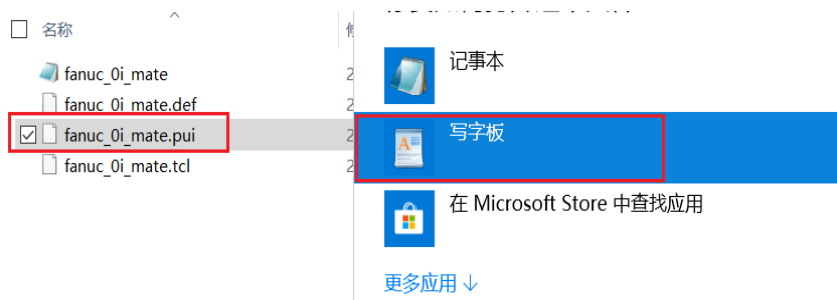


图 2

文中 with Post Builder version 8.0.0 为当前版本如下（图 3）

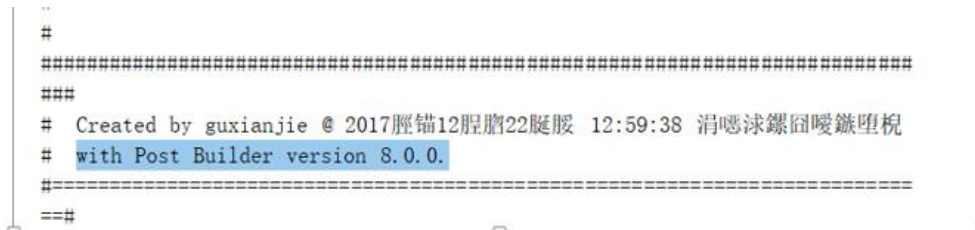


图 3

解决方法：使用高版本后处理打开。软件本身向下兼容如 NX10 后处理编辑器可以打开 NX8 生成的后处理，无法打开 NX12 生成的后处理。

2.缺少此后处理的部分或全部文件，您不能打开此后处理。

如下（图 4）。

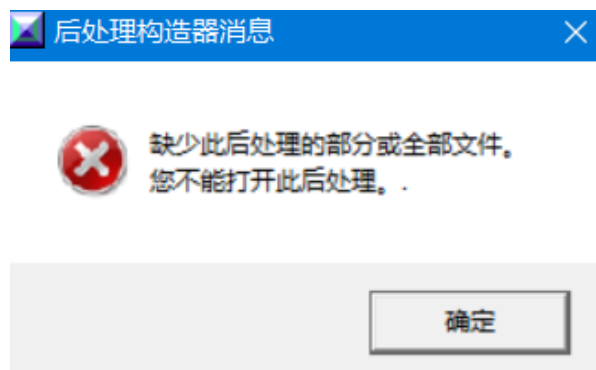


图 4

报错原因：后处理三个文件关联性不一致导致。

查看后处理文件关联性：PUI 文件使用写字板方式打开如下（图

5）

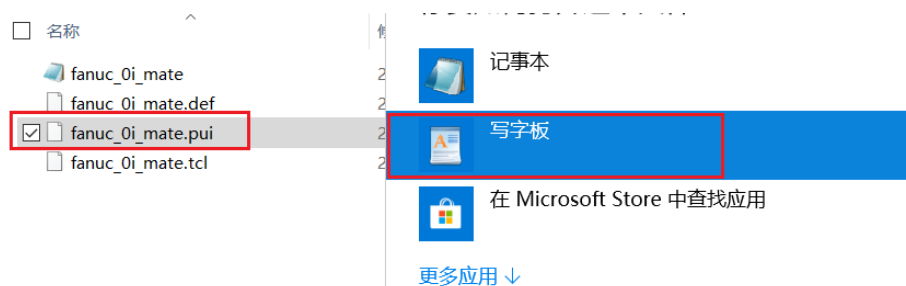


图 5

并且 def 和 tcl 文件名为：fanuc_Oi_mate 如下（图 6）

名称	修改日期	类型	大小
fanuc_Oi_mate	2017/12/22 12:59	CDL 文件	20 KB
fanuc_Oi_mate.def	2017/12/22 12:59	DEF 文件	22 KB
fanuc_Oi_mate.pui	2018/11/19 16:38	PUI 文件	55 KB
fanuc_Oi_mate.tcl	2017/12/22 12:59	TCL 文件	108 KB

图 6

文档中 def_file 文件对应的文件名为：fanuc_Oi_mate11.def

Tcl_file 文件对应文件名为：fanuc_Oi_mate11.tcl 如下（图 7）

```

## POST EVENT HANDLER START
def_file  fanuc_Oi_mate11.def
tcl_file  fanuc_Oi_mate11.tcl
## POST EVENT HANDLER END

## POST INFORMATION START
# DESCRIPTION START
This is a 3-Axis Milling Machine.
# DESCRIPTION END
# CONTROLLER START
Generic
# CONTROLLER END
# CONTROLLER TYPE START
# CONTROLLER TYPE END
# HISTORY START
  
```

图 7

解决方法：将文档中的文件名与实际文件名对应一致，经测试无论修改写字板内的内容还是直接将 def 和 tcl 文件重命名都是有效的。

Win10 教育版后处理构建器双击无反应的解决方法

作者：柯成强

审校：陈克荣

适用版本：NX 所有版本

后处理在使用过程中难免会修改一下后处理，修改后处理有两种方式，一种是通过记事本打开 tcl 文件进行修改，还有一种方法就是通过后处理构建器进行修改，后一种修改方法比较简单，易调试，所以一般有 pui 的后处理都会选择使用后处理构建器修改。后处理构建器在 Window10 教育版的环境下会因为权限问题会导致双击后处理构建器的图标没有响应，程序无法运行的情况，这里介绍解决这个问题的两种方法。

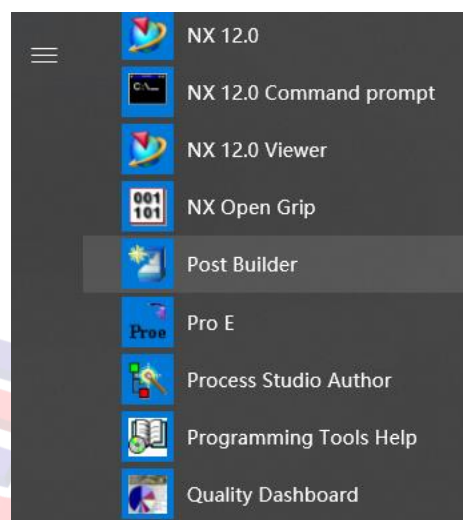


图 1

第一种方法，找到

x:\Program Files\Siemens\NX 12.0\POSTBUILD\post_builder.bat,

通过右键，用管理员权限运行这个批处理，大部份能解决双击无响应不运行程序的问题，如图 2。

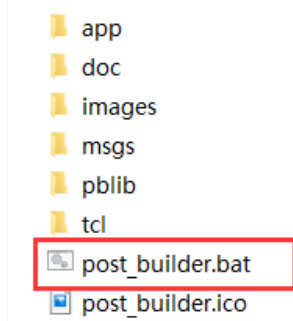


图 2

第二种方法是在任意位置新建一个批处理文件，比如：PB.bat 这个文件用记事本打开，写入下面的代码,NX 的安装路径查找本机的安装路径。

```
@echo off
```

```
Title Post Builder for NX
```

```
"x:\Program Files\Siemens\NX 10.0\POSTBUILD\post_builder.bat"
```

```
"x:\Program Files\Siemens\NX 10.0\"
```

通过右键，管理员权限运行上面的批处理也可以解决双击无响应的问题。

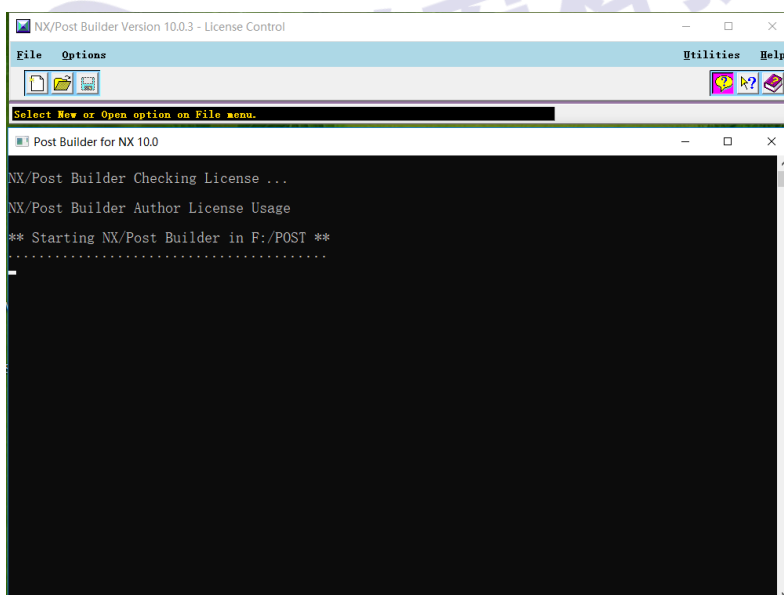


图 3

NX 二次开发-如何往表格中插入自定义符号

作者：黄盛益 审校：张季

适用版本：NX7.5 以上

一、概述

在工程图中，表格是十分常见的组成部分。表格主要用于制作标题栏、明细栏、参数系列统计表等，起着展示和归纳信息的作用。表格的绘制通常采用 NX 系统提供的表格绘制命令，通过对表格的放置方向、单元格大小和内容的定义，达到我们想要的效果；但是在有些场景中，在我们的表格内通常要插入一些自定义符号。

5	⊗			
4	⊗			
3	◇			
2	◇			
1	⊗			
序号 NO.	特殊特性符号 SPECIAL CHARACTERISTIC SYMBOL	特殊特性描述 SPECIAL CHARACTERISTIC DESCRIPTION	页码 PAGE NO.	分区号 ZONE

图 1

如果是手工操作是很容易的，编辑文本框插入定制符号就可以了。如果使用二次开发的话，则是使用“UF_TAG_ask_handle_of_tag”这个 UF 函数去获取自定义符号的 Handle 值，然后填入表格单元格中即可，但是这种做法存在一个问题，如果对单元格内文本进行再编辑后，图纸中的自定义符号会被删除掉，这明显不符合需求。本文便介绍一种规避这种错误的方法。

二、功能说明

首先，应保证制图中的自定义符号不被删除，而当我们制图中的自定义符号的 Handle 值填入表格中时，便不可避免的会发生这种情况。那么，不妨转换一种思路，既然获取制图中现有的自定义符号的 Handle 值填入表格单元，单元格再编辑时会删除掉现有的自定义符号，那可以引入一个中间自定义符号，通过获取图纸中自定义符号的类型，从重用库中导入该类型的自定义符号，然后直接获取该导入的自定义符号的 Handle 值填入表格单元格中即可。这样子避免了填入表格的 Handle 值与图纸中现有的自定义符号之间的关联性。这种做法会在图纸的原点位置生成一个自定义符号，由于这个自定义符号无法删除，故将其隐藏就可以了。下面以自定义符号中的重要特性符号和关键特性符号为例。具体代码如下：

```
// 获取图纸中自定义符号的Handle值
// </summary>
// <param name="customSymbolTagI">图纸中现有的自定义符号</param>
// <returns></returns>
public static string GetHandleString(CustomSymbol customSymbolTagI)
{
    string handleStr;
    string paraValue;
    GetConfigData("重用库符号位置", out paraValue); // 获取配置文件定义的重用库符号放置路径
    Part workPart = theSession.Parts.Work;
    CustomSymbol nullCustomSymbol = null;
    DraftingCustomSymbolBuilder draftingCustomSymbolBuilder1;
    draftingCustomSymbolBuilder1 = workPart.Annotations.CustomSymbols.CreateDraftingCustomSymbolBuilder(nullCustomSymbol);
    /*选择创建的自定义符号类型*/
    if (customSymbolTagI.SymbolName == "重要特性符号")
    {
        draftingCustomSymbolBuilder1.SelectSymbol(paraValue + @"\Important_character.sym.prt");
    }
    else if (customSymbolTagI.SymbolName == "关键特性符号")
    {
        draftingCustomSymbolBuilder1.SelectSymbol(paraValue + @"\Key_character.sym.prt");
    }
    // 默认设置
    draftingCustomSymbolBuilder1.GeometryPreference = NXOpen.Annotations.BaseCustomSymbolBuilder.GeometryPreferences.UseDefinition;
    draftingCustomSymbolBuilder1.AnnotationPreference = NXOpen.Annotations.BaseCustomSymbolBuilder.AnnotationPreferences.UseDefinition;
    NXOpen.NXObject nXObject1 = null;
    nXObject1 = draftingCustomSymbolBuilder1.Commit(); // 在原点出创建自定义符号
    draftingCustomSymbolBuilder1.Destroy();
    theUFSession.Tag.AskHandleFromTag(nXObject1.Tag, out handleStr); // 获取自定义符号的handle (句柄)
    handleStr = string.Format("<SYM={0}>", handleStr); // 合成表格接受的字符样式
    theUFSession.Obj.SetBlankStatus(nXObject1.Tag, UFConstants.UF_OBJ_BLANKED); // 隐藏创建的自定义符号对象
    return handleStr; // 返回
}
```

图 2

三、结论

当开发找不到思路时，不妨转换一下思路，有时候走直路不一定比绕路效率高、行得通。

NX 二次开发-制图模块获得指定点所在投影面

作者: 徐涛 审校: 倪海

适用版本: NX8.5 以上版本

一、概述

NX 的制图模块中, 有时候创建标注需要标注到指定组件的指定投影面上。但工程模块的是投影出来的面, 不是三维模型中本身的面, 是无法直接获取的。可以根据用户点选的位置, 获取到点所在的投影面, 识别出来创建出标注, 并指向对应投影面。

二、功能说明

通过指定点控件指定点的坐标位置, 转化到投影视图中, 根据视图的投影矩阵抓取投影矢量, 然后将转化点沿着这个矢量方向投影, 使用 UF 射线函数 UF_MODL_trace_a_ray, 获取第一个击中的面, 即为所求面, 如果没有找到, 再反方向投影, 找到最外面的那个面。

```
var bdytags = mListAllBodyTags.ToArray();
UFModl.RayHitPointInfo[] curInfo;
SessionHelper.GetUFSession().Mtx4.Identity(mtx);
if (bdytags.Length == 0)
{
    throw new NullReferenceException("未找到投影的体");
}
//先往外找, 如果数量>0, 则获取最后一个, 否则再往里找, 找到第一个就返回
SessionHelper.GetUFSession().Modl.TraceARay(bdytags.Length, bdytags, pnt3dI, curViewI.mZdir, mtx, 0, out cnt, out curInfo);

Point3d tmp3dPnt = new Point3d();
var datas = curInfo.ToList();
theSession.LogFile.WriteLine($"射线原点: X{pnt3dI[0]};Y{pnt3dI[1]};Z{pnt3dI[2]}");
if (datas.Count > 0)
{
    //日志信息
    theSession.LogFile.WriteLine("正面");
    theSession.LogFile.WriteLine($"射线方向: X{curViewI.mZdir[0]};Y{curViewI.mZdir[1]};Z{curViewI.mZdir[2]}");
    int idx = -1;
    double tmpDist = -999;
    for (int i = 0; i < cnt; i++)
    {
        var face =
            SessionHelper.GetSession().GetObjectManager().GetTaggedObject(curInfo[i].hit_face) as Face;
        var name = face.OwningPart.Name + "&" + face.OwningPart.Leaf;
        theSession.LogFile.WriteLine(name);
        double dist = LibPoint.GetDistance(LibPoint.DoubleToPoint3D(pnt3dI),
            LibPoint.DoubleToPoint3D(curInfo[i].hit_point));
        theSession.LogFile.WriteLine
            ($"射线射中的点: X{curInfo[i].hit_point[0]};Y{curInfo[i].hit_point[1]};Z{curInfo[i].hit_point[2]}");
        theSession.LogFile.WriteLine(dist.ToString("f2"));
        //后进算法, 获取距离最近的点
        if (dist > tmpDist)
        {
            tmpDist = dist;
            idx = i;
        }
    }
}
```

图 1

三、总结

在工程图模块的投影视图其实也相当于一个 3D 世界，可以右键视图，点击展开，通过拖拽鼠标，展现效果和 3D 建模环境相同。所以这里也可以通过点投影的方法，找到指定点在投影视图中的所在投影面，但这些数据必须都属于这个投影视图的 3D 坐标系。

注意点，这里用的投影函数，输入的体也是二维组件所属的体，获得的击中面也是二维组件所属面，非零件本身对象。

EWI 中查看动画

作者：朱鑫 审核：陈泓希

适用版本：Teamcenter11

随着 TC 版本的不断更新，使用 AWC 客户端逐渐成为一种主流方式。在 AWC 客户端如果我们需要对 BOP 进行查看，这里就需要用到 EWI，在 EWI 中我们可以查看 BOP 中的 3PR,同时也可以直观的查看 BOP 中的作业指导书及模拟动画。

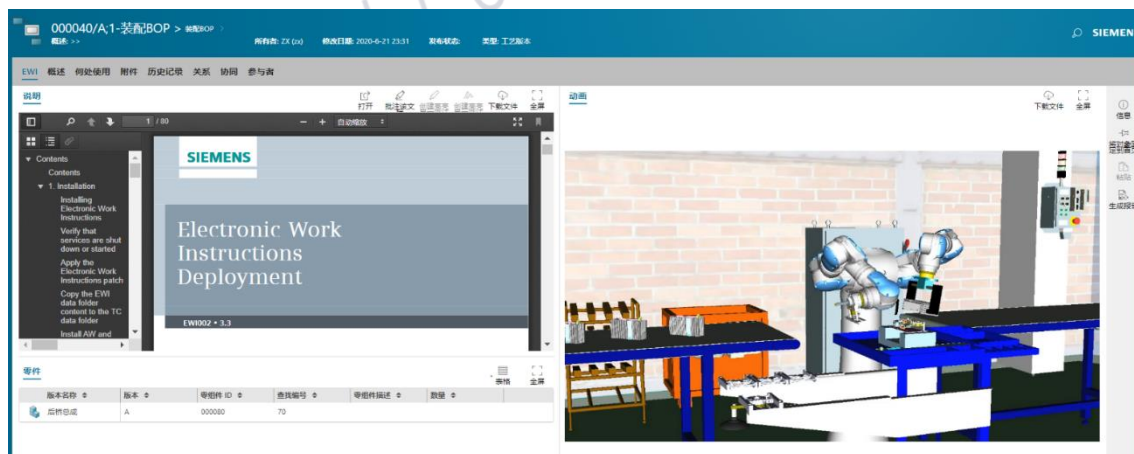


图 1

在查看动画时，我们需要做一些系统配置，操作过程如下：

1.新建 MP4 数据集，EWI 现支持 MP4 与 WEBM 格式的动画进行直接播放，其他格式需要调用外部工具

数据集 : Y5_mp4

数据集属性	属性	操作	显示规则	深层复制规则	GRM 规则	操作描述符	引用	工具操作
详细信息								
项目(I):	y5project							
名称	Y5_mp4							
显示名称	mp4							
存储类	Dataset							
父级	Dataset							
图标	默认							
类型	永久							
	☐ 是抽象的 ?							
	☒ 可导出 ?							
	☒ 允许创建业务对象的实例							
	☐ 存储为轻量级对象							
Teamcenter 组件:	Y5_project							浏览(W)...
描述:								
编辑工具:								
	工具	COTS	模板					添加...
	Image Viewer		y5project					移除
查看工具:								
	工具	COTS	模板					添加...
	Image Viewer		y5project					移除

图 2

2.在首选项 EWI_ThreePanelLayout 中增加值 Animations

选项

按搜索的首选项

定义 | 实例 | 类别 | 导入 | 导出

单击“编辑”按钮可修改定义并更新任意字段，以便启用“保存”按钮。请注意，“描述”字段不可留空。单击“保存”按钮可保存现有首选项的定义。

名称	位置	保护范围
EWI_ThreePanelLayout	站点	角色

类别	环境变量	类型	多个值
Electronic Work Instructions	已禁用	字符串型	多个

描述

Defines the tabs that will be seen in each panel (top left, bottom left and right panel) of Electronic Work Instructions (EWI) solution in the Active Workspace client main panel when ThreePanelLayout is defined as the value in EWI_Layout preference.

值

topLeftPanelTabs:WorkInstructions
bottomLeftPanelTabs:Parts,Tools,Activities
rightPanelTabs:Images,Documents,References **Animations**

找到的首选项数: 2

取消 保存

选项 | 过滤器 | 搜索 | 组织

图 3

3.在工艺对象版本下新建 MP4 数据集并导入动画

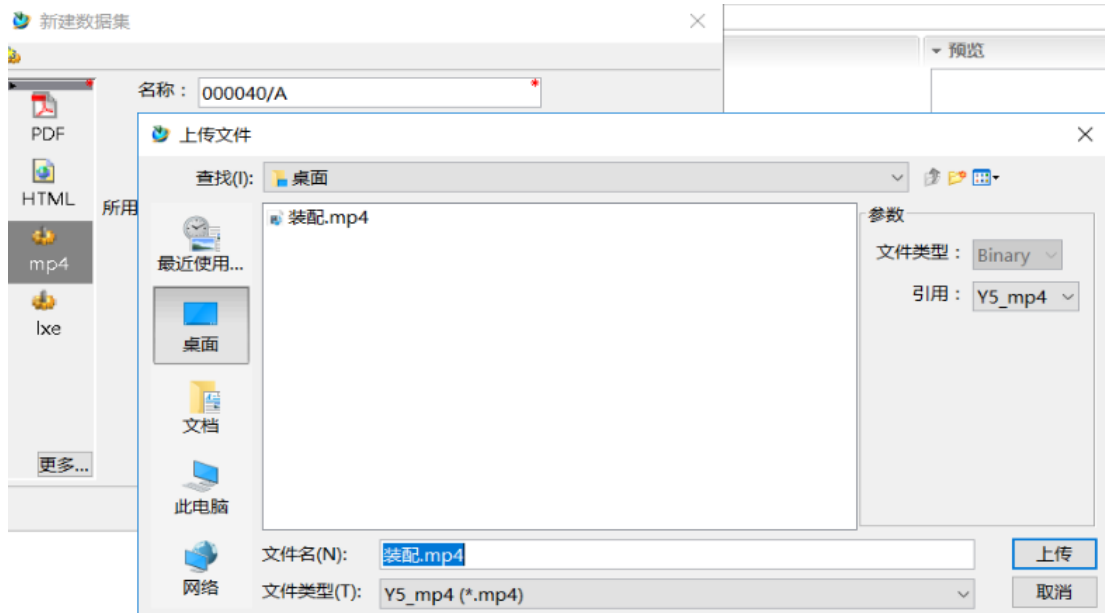


图 4

4.重启 AWC 客户端，进行动画播放

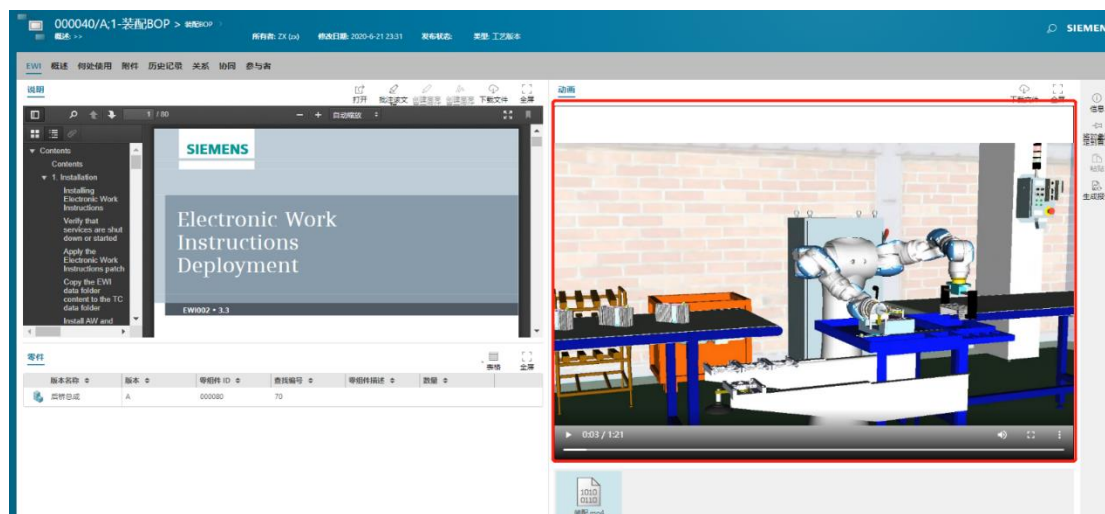


图 5

SE 集成连接 TC 时加载策略文件失败的解决办法

作者：陈涛

审校：张梦萍

测试版本：ST5 和 Teamcenter9

用户在安装完 SolidEdge ST5 后，打算进行与 Teamcenter 进行连接测试，系统提示“加载 SolidEdge 策略文件失败”，如图 1 所示。这种问题该如何解决？

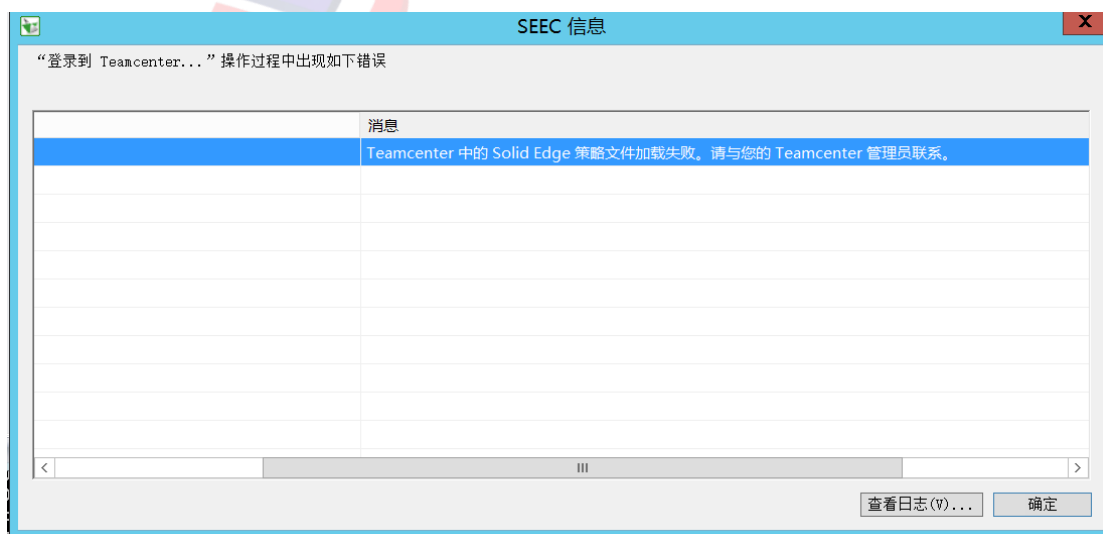


图 1

出现 SolidEdge 加载策略文件失败，需要查看池中的服务器数量是否满足要求，一般导入大型数据集时，会出现此错误（添加到 Teamcenter），如何增加池中服务器数，启动环境管理器（Environment Manager），基于 J2EE 的服务器管理器（修改设置），如图 2 所示。

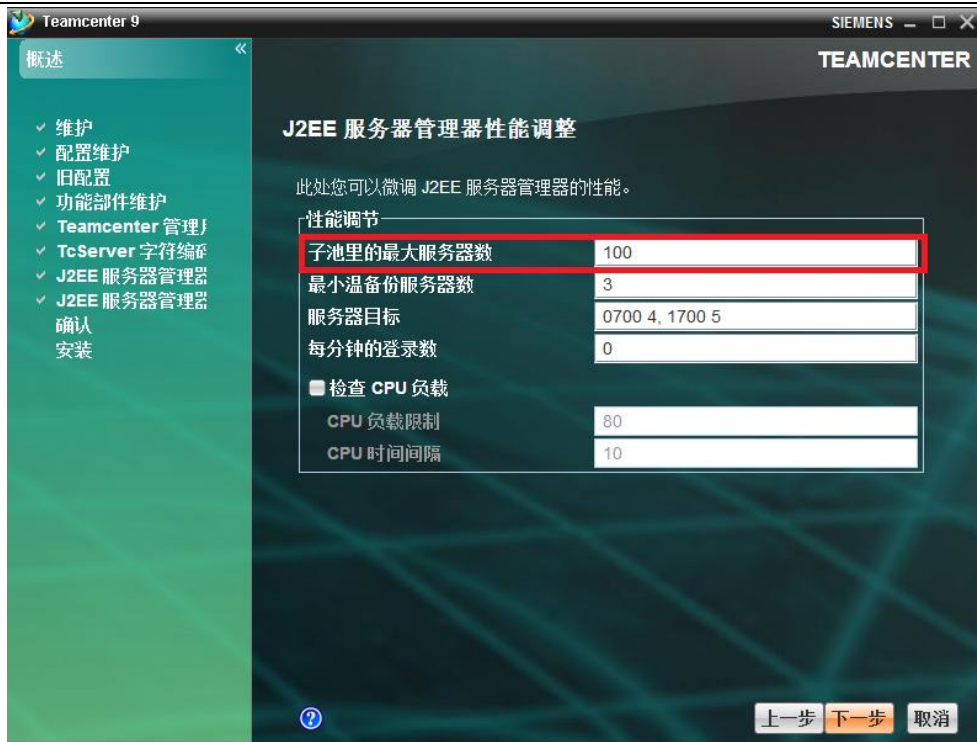


图 2

如果还出现加载策略文件失败，还需要确认，端口号是否正确以及 SEEC 管理器是否成功安装，如图 3 所示。



图 3

如果端口号，SEEC 管理器都加载正确，即为策略文件未能加载到 TC_DATA，可以按照以下顺序重新安装集成并更新数据库（完整模型）即可，如图 4 所示。

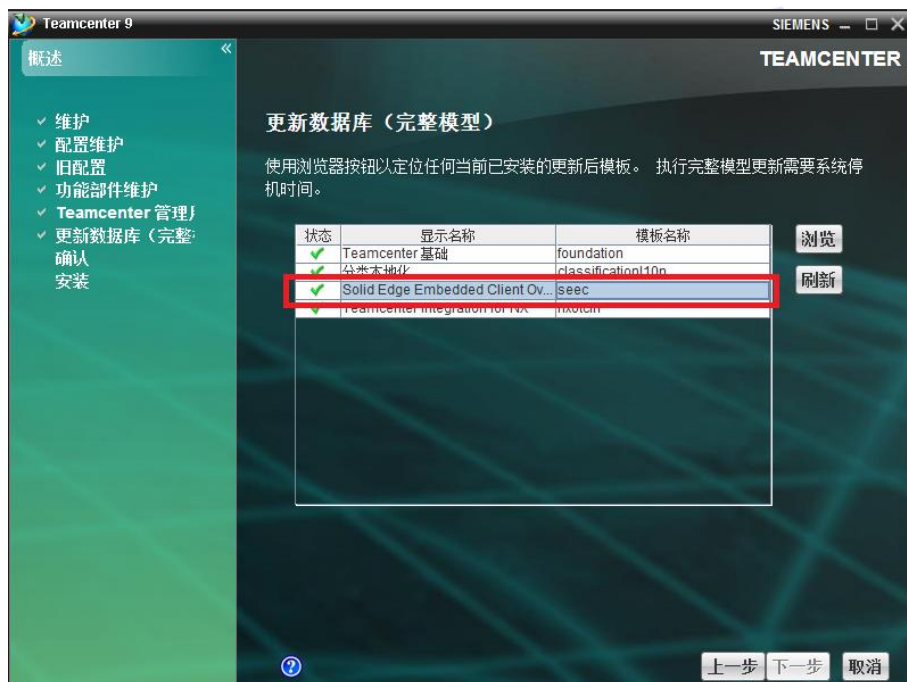


图 4

按照顺序重新安装集成并部署环境后，最后可以通过诊断应用程序测试是否成功连接，如图 5 所示。



图 5

BMIDE 配置对象版本“操作”

作者：李晓东 审校：游振君

测试版本：Teamcenter11.6

有些项目会存在部分特别需求，如需要将某个对象的版本表单在创建该对象的时候自动带出多个定制表单，通常一般情况下创建某个对象时，该对象版本下的会默认带出一个版本表单，如何做才能自动带出多个表单呢？本文讲述的是用 BMIDE 对象版本下“操作”方式设定好创建默认多表单，并可展示出来。首先，请创建好必须的自定义表单和关系。

具体操作方式如下：

1、打开 BMIDE 模板，选中对象版本，打开“操作”选项卡如图

1:

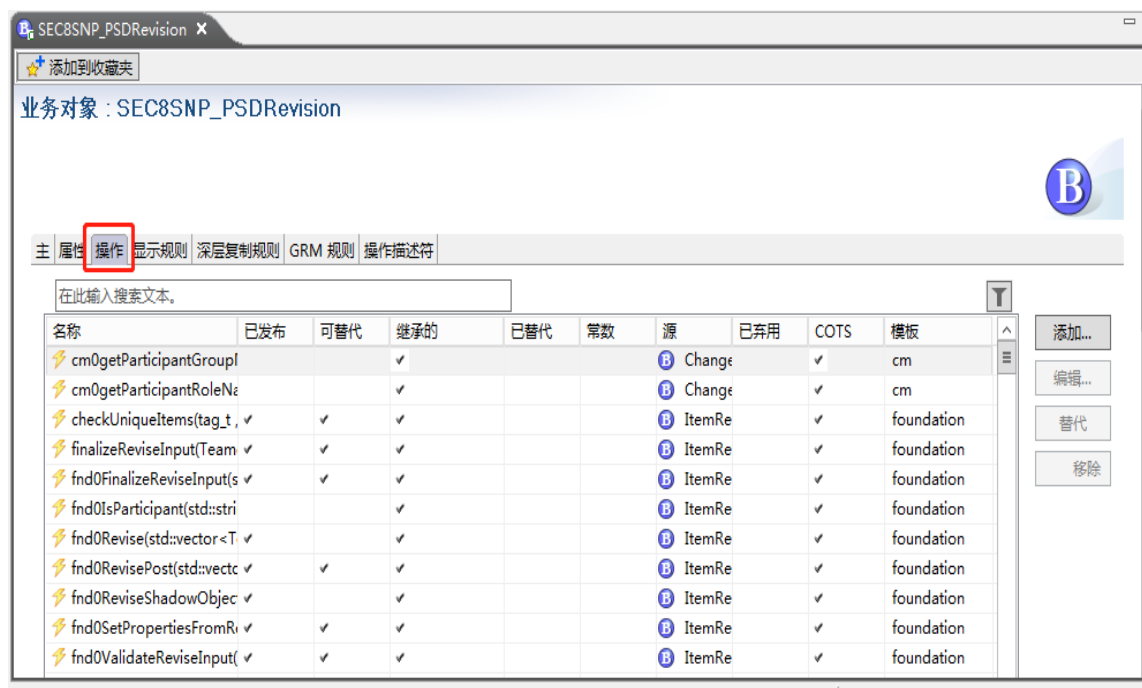


图 1

2、在搜索框中搜索“item_create_rev”如图 2:

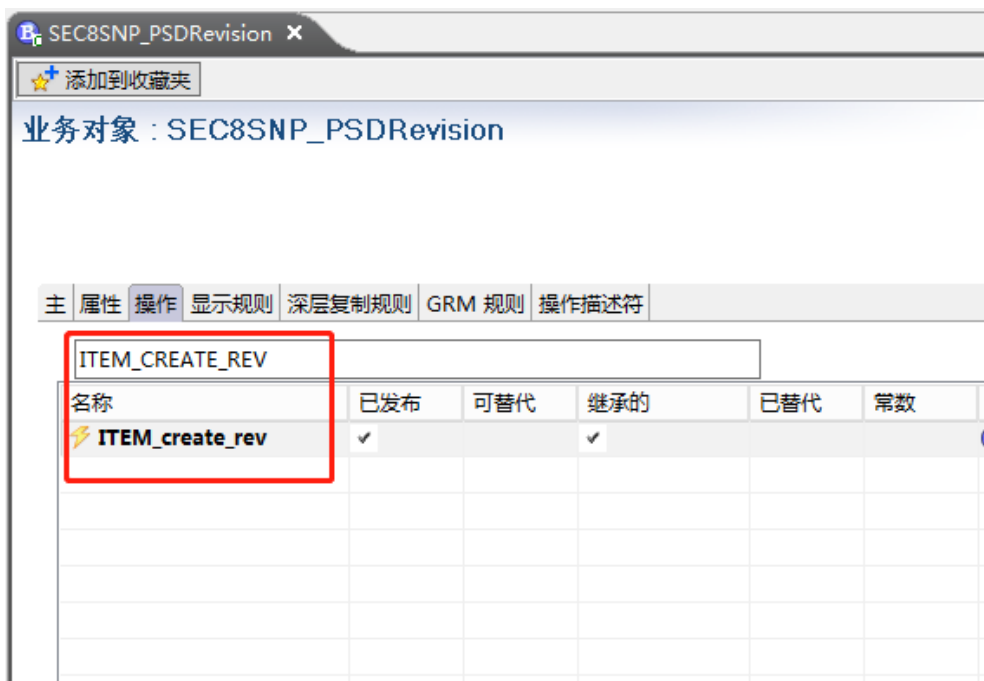


图 2

3、选中搜索结果，展开下方“扩展附件”点击“添加”项，如图

3:

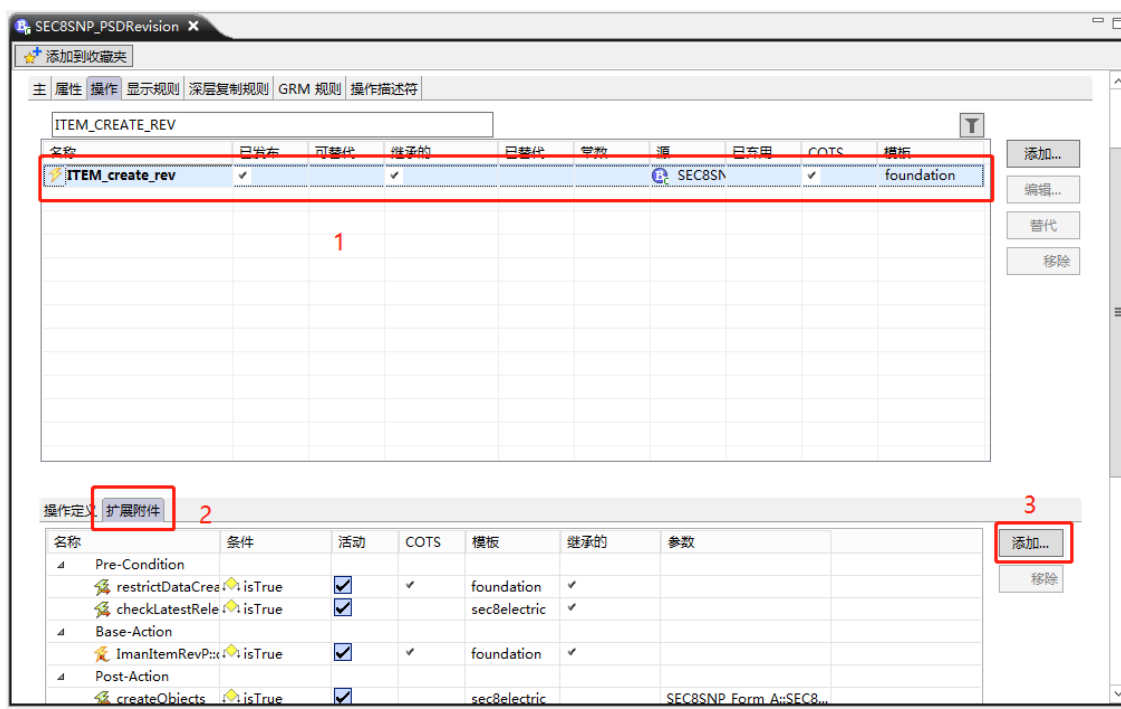


图 3

4、在弹出的添加扩展规则框中，扩展点选择“PostAction”，扩展一栏点击“浏览”如图 4、所示：

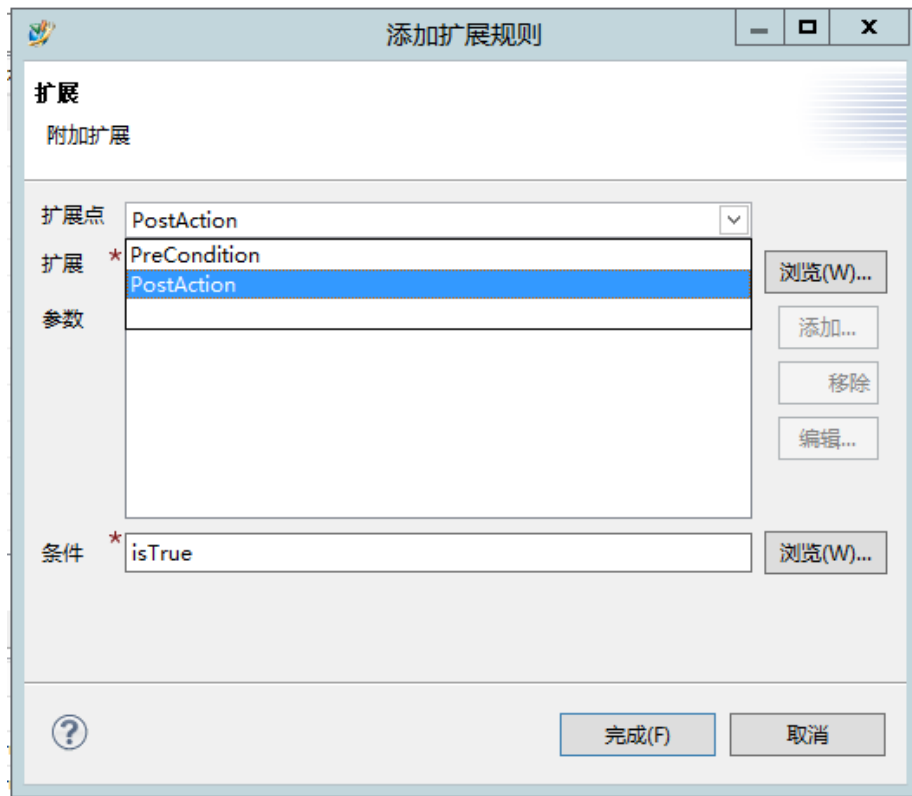


图 4

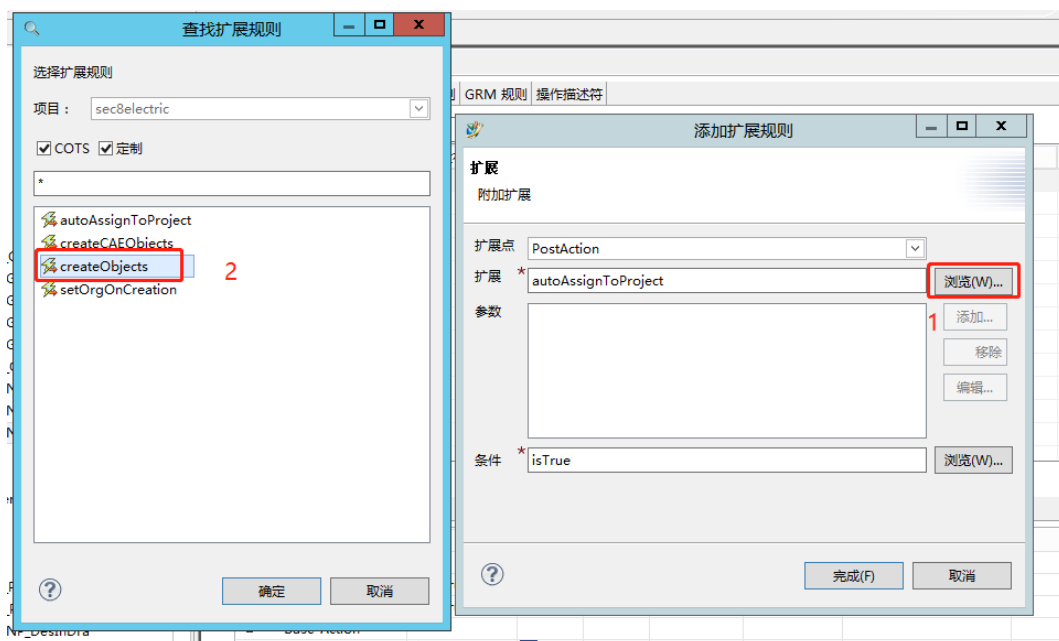


图 5

5、添加完成后，在参数中选择添加。对象类型和关系类型框中分别选择事先准备好的自定义的表单和表单关系，如图 6：

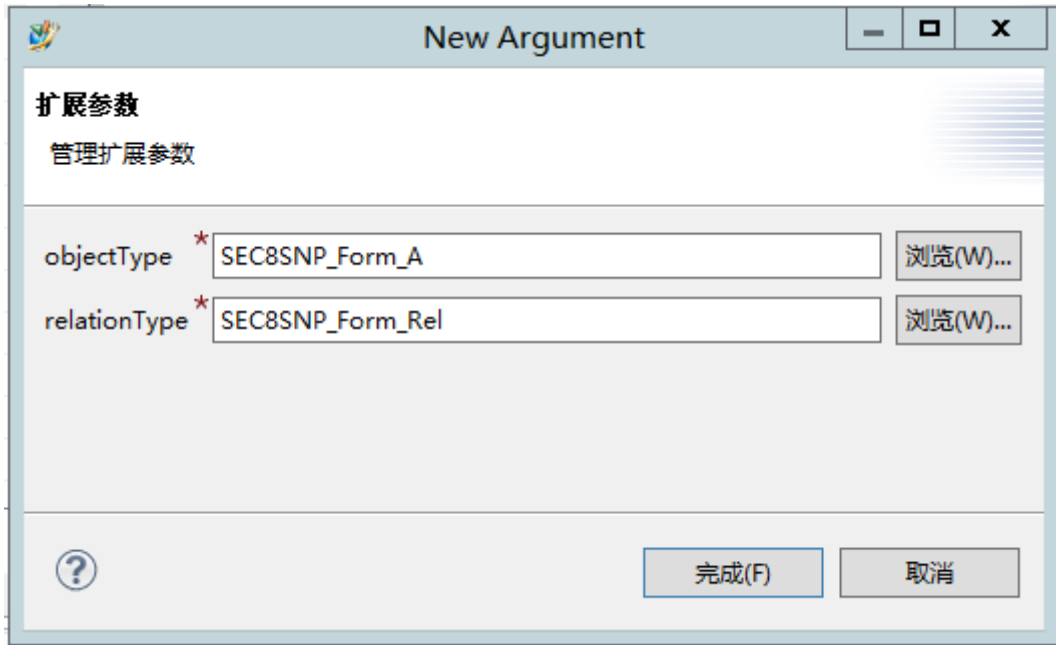


图 6

6、完成后可在扩展附件中查看添加结果。如图 7：

操作定义 扩展附件							
名称	条件	活动	COTS	模板	继承的	参数	
Pre-Condition							
restrictDataCrea	isTrue	☒	✓	foundation	✓		添加...
checkLatestRele	isTrue	☒		sec8electric	✓		移除
Base-Action							
ImanItemRevPoc	isTrue	☒	✓	foundation	✓		
Post-Action							
createObjects	isTrue	☒		sec8electric		SEC8SNP_Form_A::SEC8...	
createObjects	isTrue	☒		sec8electric		SEC8SNP_Form_B::SEC8...	
createObjects	isTrue	☒		sec8electric		SEC8SNP_Form_C::SEC8...	

图 7

TC 本地化数据库导出

作者：马正兴 审校：史道胜

测试版本：TC11

TC 在进行 BMIDE 配置的时候，需要配置本地化，但有的时候，部署完 BMIDE 之后，可能因为某些原因，导致 BMIDE 工程包丢失，从而可能导致重新整理的 BMIDE 包对应不上数据库内设置的版本，就会存在 TC 内的属性显示有本地化，而 BMIDE 里面的属性没有本地化，这个时候，就可以通过命令，将数据库内的本地化导出，与 BMIDE 进行对比，修改。

具体使用方式如下：

1、打开 TC 的黑窗口，将导出命令输入

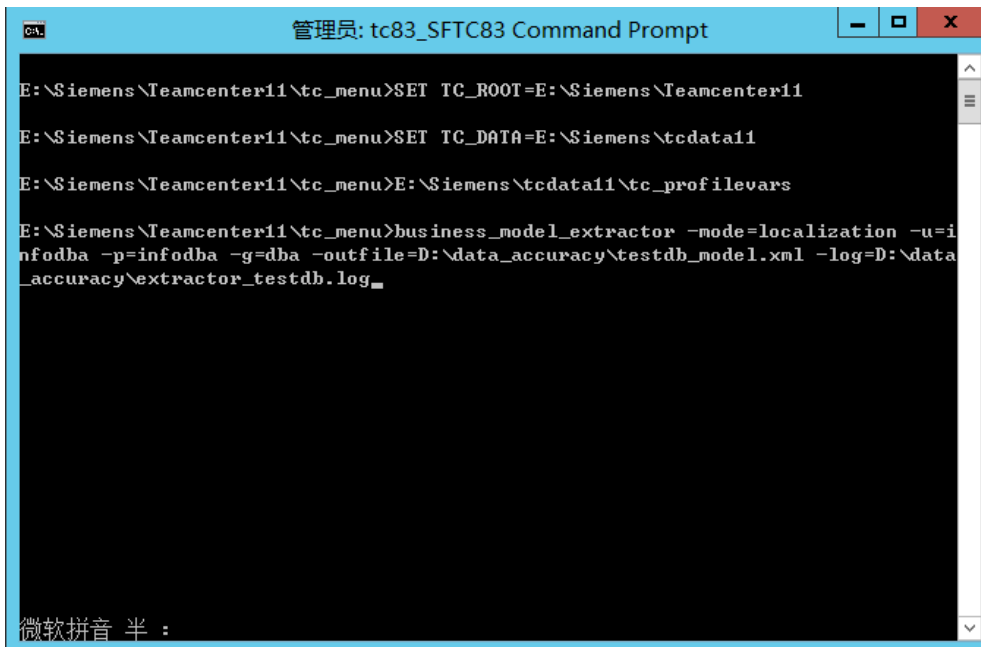


图 1

2、命令对应的意思：

business_model_extractor：表示的是命令类型

-mode: 值有 all、shame、localization 三种

选择 all 表示输出全部的对象模型

选择 shame 表示输出类和属性

选择 localization 表示输出本地化

-u: 账户

-p: 密码

-g: 组织

-outfile: 输出文件存放地址

-log: 输出日志存放地址

3、最后输出文件可以在对应的地址查看，下图为日志：

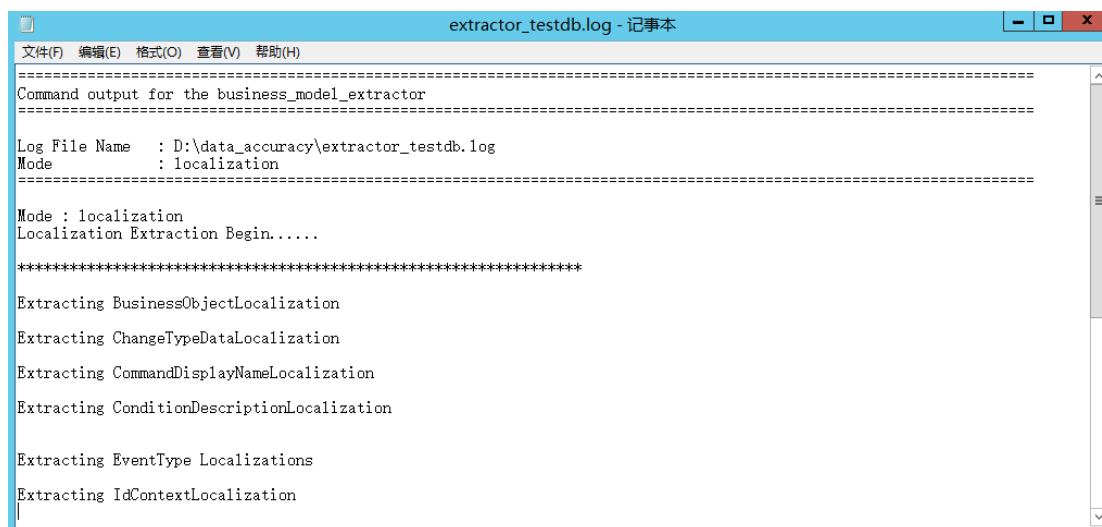
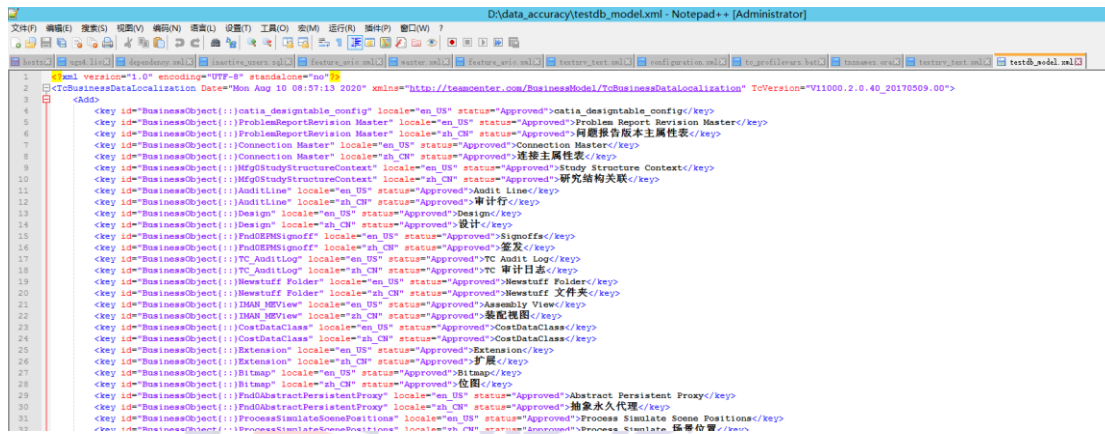


图 2

4、下图为输出的本地化文件，可以在此文件中查看数据库内所有的本地化：



```

1  <?xml version="1.0" encoding="UTF-8" standalone="no" ?>
2  <TCBusinessDataLocalization Date="Mon Aug 10 08:57:13 2020" xmlns="http://testcenter.com/BusinessModel/TCBusinessDataLocalization" TCVersion="V11000.2.0.40_20170509.00">
3
4      <Add>
5          <key id="BusinessObject::JotatisDesignableConfig" locale="en_US" status="Approved">otatis_designable_config</key>
6          <key id="BusinessObject::ProblemReportRevisionMaster" locale="en_US" status="Approved">Problem Report Revision Master</key>
7          <key id="BusinessObject::ProblemReportRevisionMaster" locale="zh_CN" status="Approved">问题报告版本主属性表</key>
8          <key id="BusinessObject::ConnectionMaster" locale="en_US" status="Approved">Connection Master</key>
9          <key id="BusinessObject::ConnectionMaster" locale="zh_CN" status="Approved">连接主属性表</key>
10         <key id="BusinessObject::MFgStudyStructureContext" locale="en_US" status="Approved">Study Structure Context</key>
11         <key id="BusinessObject::MFgStudyStructureContext" locale="zh_CN" status="Approved">研究结构关联</key>
12         <key id="BusinessObject::AuditLine" locale="en_US" status="Approved">Audit Line</key>
13         <key id="BusinessObject::AuditLine" locale="zh_CN" status="Approved">审计行</key>
14         <key id="BusinessObject::Design" locale="en_US" status="Approved">Design</key>
15         <key id="BusinessObject::Design" locale="zh_CN" status="Approved">设计</key>
16         <key id="BusinessObject::PdDERMSignoff" locale="en_US" status="Approved">Signoffs</key>
17         <key id="BusinessObject::PdDERMSignoff" locale="zh_CN" status="Approved">签发</key>
18         <key id="BusinessObject::TC_AuditLog" locale="en_US" status="Approved">TC Audit Log</key>
19         <key id="BusinessObject::TC_AuditLog" locale="zh_CN" status="Approved">TC 审计日志</key>
20         <key id="BusinessObject::Newstuff Folder" locale="en_US" status="Approved">Newstuff Folders</key>
21         <key id="BusinessObject::Newstuff Folder" locale="zh_CN" status="Approved">Newstuff 文件夹</key>
22         <key id="BusinessObject::ZMAN_MView" locale="en_US" status="Approved">Assembly View</key>
23         <key id="BusinessObject::ZMAN_MView" locale="zh_CN" status="Approved">装配视图</key>
24         <key id="BusinessObject::CostDataClass" locale="en_US" status="Approved">CostDataClass</key>
25         <key id="BusinessObject::CostDataClass" locale="zh_CN" status="Approved">CostDataClass</key>
26         <key id="BusinessObject::Extension" locale="en_US" status="Approved">Extension</key>
27         <key id="BusinessObject::Extension" locale="zh_CN" status="Approved">扩展</key>
28         <key id="BusinessObject::Bitmap" locale="en_US" status="Approved">Bitmap</key>
29         <key id="BusinessObject::Bitmap" locale="zh_CN" status="Approved">位图</key>
30         <key id="BusinessObject::PdAbstractPersistentProxy" locale="en_US" status="Approved">Abstract Persistent Proxy</key>
31         <key id="BusinessObject::PdAbstractPersistentProxy" locale="zh_CN" status="Approved">抽象永久代理</key>
32         <key id="BusinessObject::ProcessSimulateScenePositions" locale="en_US" status="Approved">Process Simulate Scene Positions</key>
33         <key id="BusinessObject::ProcessSimulateScenePositions" locale="zh_CN" status="Approved">Process Simulate 场景位置</key>
34     </Add>
35 </TCBusinessDataLocalization>

```

图 3

TC 审计功能不同版本切换

作者：史道胜 审校：马正兴

测试版本：TC11

TC 内的审计功能，在不同的版本内是不一样的，在 TC8.3 的时候，审计功能有一个单独的功能，然后再 TC11 内，审计功能就换成了新的，同时也可以切换为旧的模式，两者之间审计内容存放位置是有区别的。

具体使用方式如下：

1、在 TC11 内，比如说工作流程历史记录，可以在视图内调出来

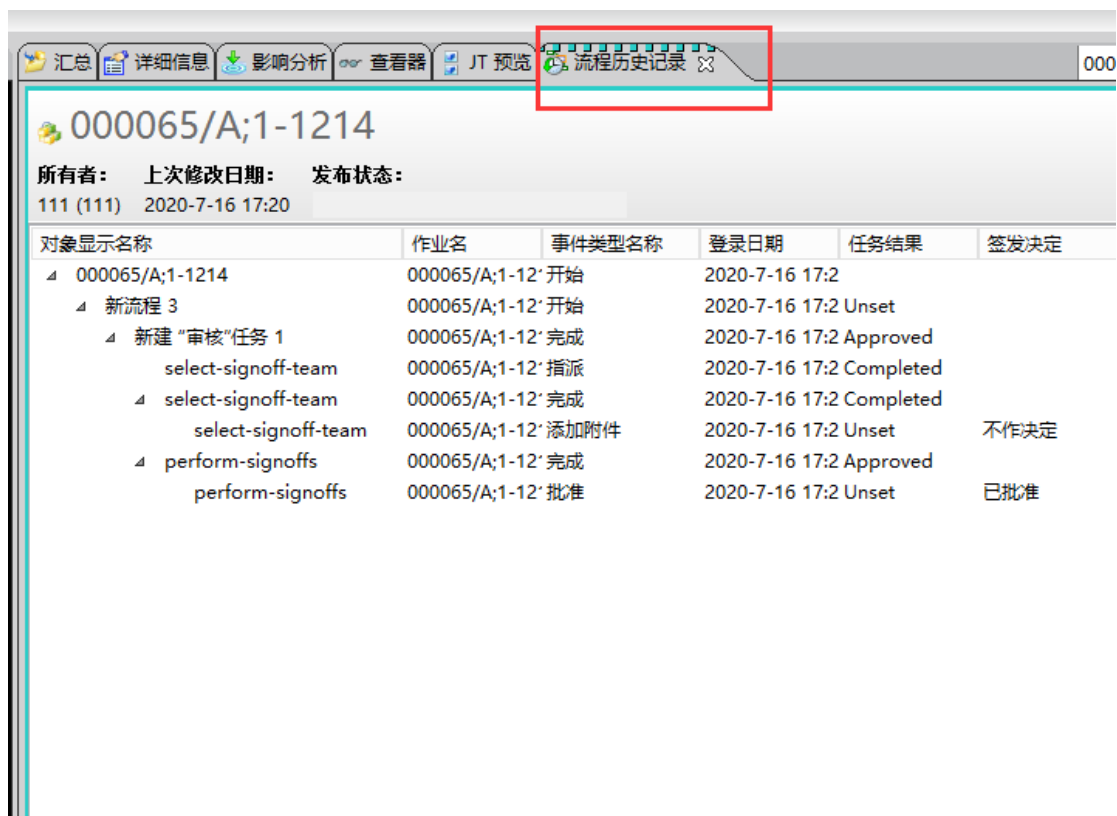


图 1

2、然后我们打开编辑→选项，找到首选项

“TC_audit_manager_version”

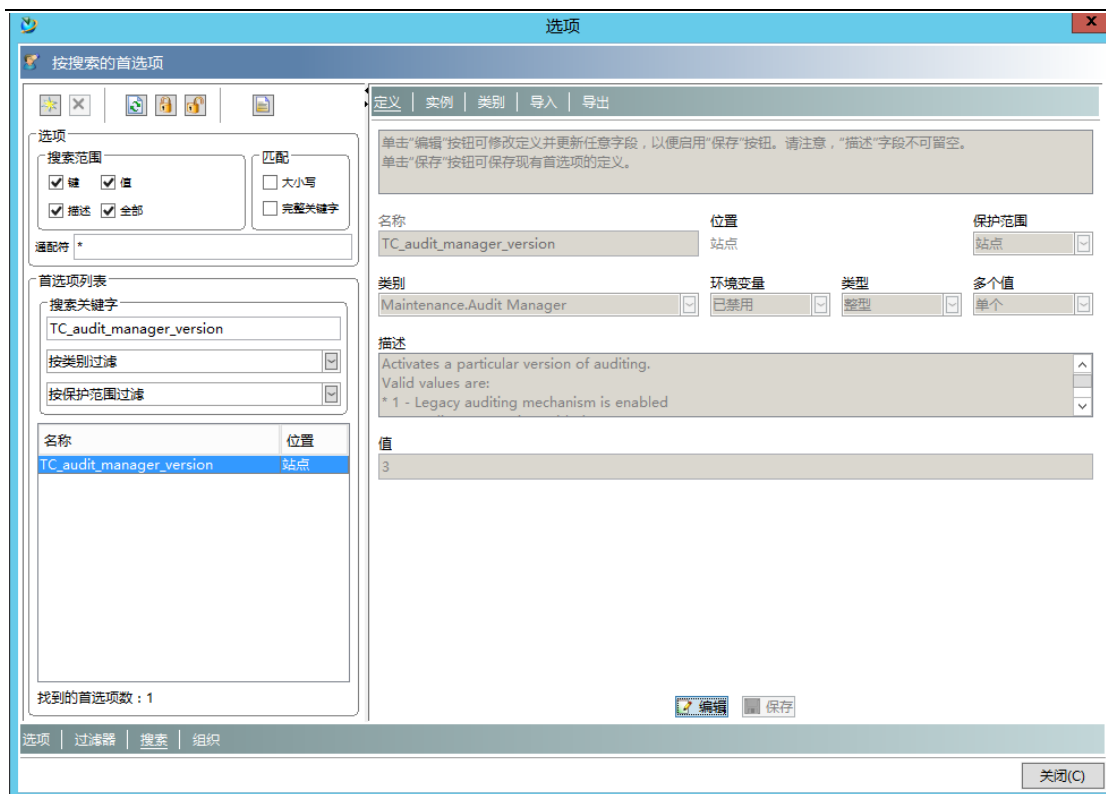


图 2

3、点击编辑，修改首选项的值为“2”，进行保存：

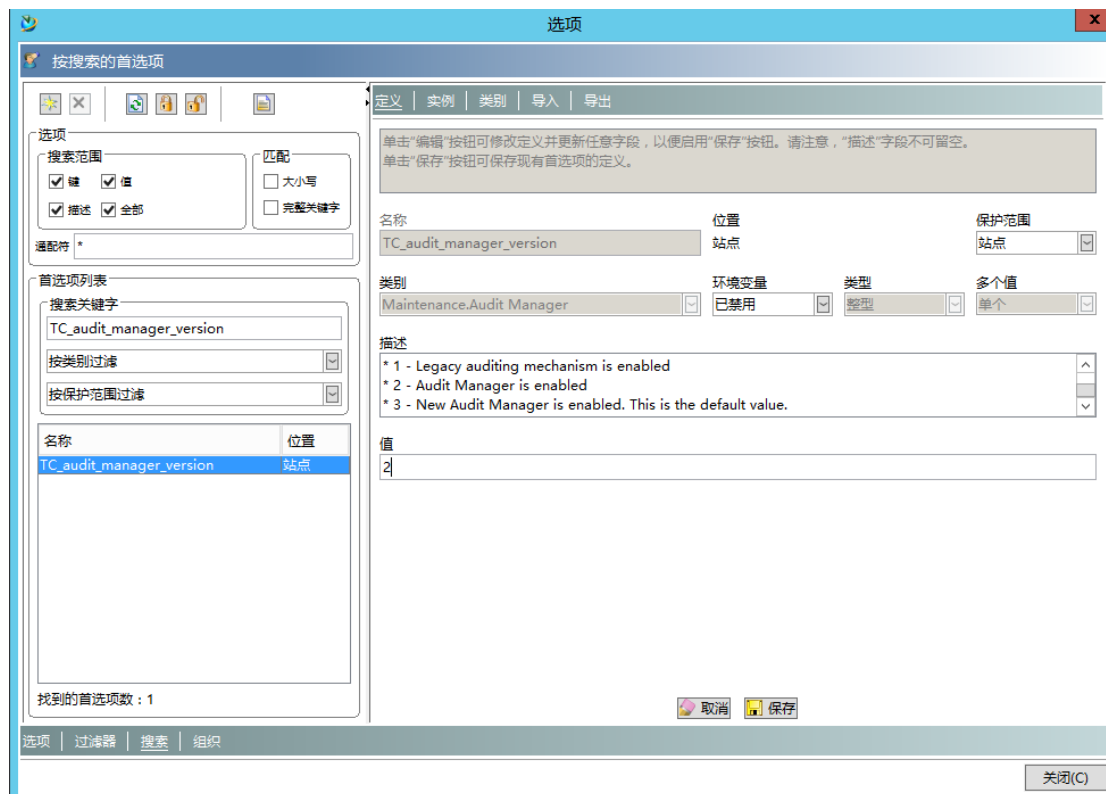


图 3

4、可以看到，视图那块的流程历史记录，已经自动消失了，并且也无法调出，同时，在工具栏内，点击视图，多了一个审计的功能。这样就切换为旧版的审计模块了。



图 4

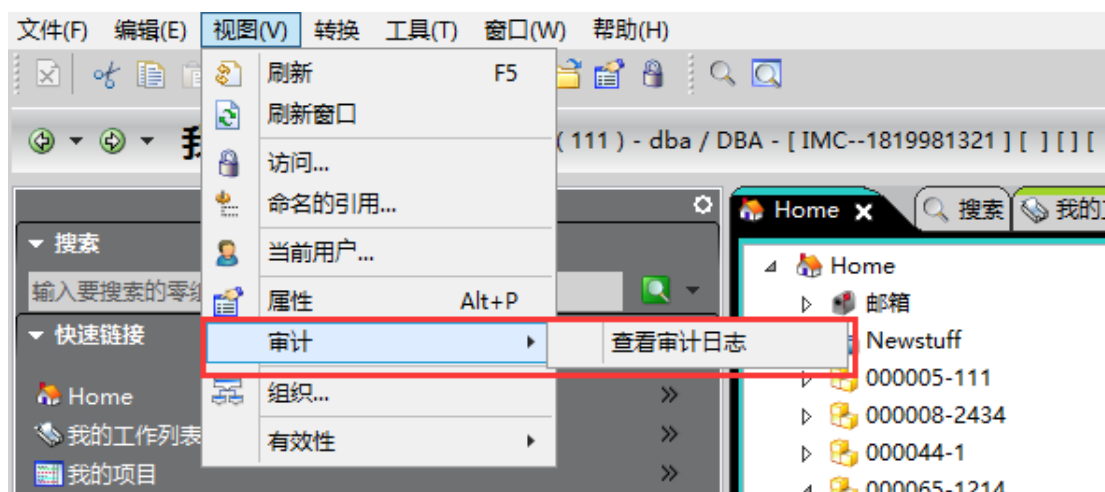


图 5

如何查询处于流程中的零组件

作者：郑煜鑫

审校：杨崇华

适用版本：TC11

在实际业务操作中，用户有时会需要去查询正在处于流程中的流程，用于统计审核对象数据，那么我们该如何去配置查询来实现此需求呢？

查询构建器配置如下图 1 所示：

名称: A_WF - 处于流程中的1零组件

描述: 该查询返回处理中的指定类型零组件。该查询支持 WF - 处理中的零组件报告。

查询类型: 本地查询

搜索类型: ItemRevision 显示提示

可修改的查询类型: 本地查询

版本规则:

属性选择

显示设置

当前位置代码

搜索准则 排序方式

	属性	用户条目本地...	用户条目名称		默认值
	object_name	Name	名称	=	
AND	items_tag.item_id	ItemID	零组件 ID	=	
AND	item_revision_id	Revision	版本	=	
AND	process_stage_list			!=	\$NULL
AND	object type	Type	类型	=	ItemRevision
AND	creation_date	CreatedAfter	创建时间晚于	>=	
AND	creation_date	CreatedBefore	创建时间早于	<=	
AND	owning_user.user_id	OwningUser	所有权用户	=	\$USERID

创建 删除 修改 清除

图 1

登录 Teamcenter 系统，使用该查询去进行查询，搜索条件可以输入名称，所有权者或创建时间去搜索，如图 2 所示。



图 2

搜索出的结果为正在审核的零组件，如下图 3 所示。

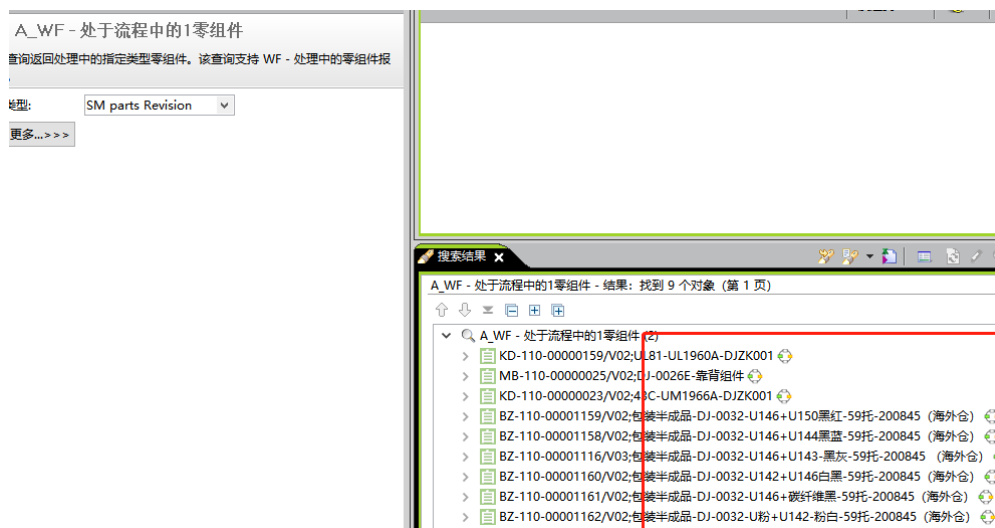


图 3

通过使用该查询，我们可以方便的查询到系统正在审核的物料，便于跟踪查询。

附录为查询文件可用于导入 Teamcenter 查询构建器中，其内容如下：

```
<?xml version="1.0" encoding="utf-8"?>
<!-- GENERATED BY: PLM XML SDK 7.0.5.491 -->
<plmxml_bus:PLMXMLBusinessTypes
xmlns="http://www.plmxml.org/Schemas/PLMXMLSchema"
xmlns:plmxml_bus="http://www.plmxml.org/Schemas/PLMXMLBusinessSchema"
language="en-us" time="11:05:54" schemaVersion="6" author="Teamcenter
V11000.2.5.61_20181118.00 - infodba@IMC--1833704369(-1833704369)" date="2020-07-08"
languages="en-us zh-cn">
  <plmxml_bus:SavedQueryDef id="id1" name="A_WF - 处于流程中的 1 零组件"
nameRef="#id2" descriptionTextRef="#id3" queryFlag="0" queryClass="ItemRevision">
    <Description>该查询返回处理中的指定类型零组件。该查询支持 WF - 处理中的零组
件报告。</Description>
    <ApplicationRef application="Teamcenter" label="u7U59pNKpsjT0D"
version="u7U59pNKpsjT0D"></ApplicationRef>
    <plmxml_bus:QueryClause id="id4" stringValue="SELECT qid FROM ItemRevision
WHERE &quot;object_name&quot; = &quot;${Name = }&quot; AND
&quot;items_tag.item_id&quot; = &quot;${ItemID = }&quot; AND
&quot;item_revision_id&quot; = &quot;${Revision = }&quot; AND &quot;object_type&quot; =
&quot;${Type = ItemRevision}&quot; AND &quot;creation_date&quot; &gt;=
&quot;${CreatedAfter = }&quot; AND &quot;creation_date&quot; &lt;= &quot;${CreatedBefore
= }&quot; AND &quot;owning_user.user_id&quot; = &quot;${OwningUser = $USERID}&quot;
AND &quot;owning_group.name&quot; = &quot;${OwningGroup = $GROUP}&quot; AND
&quot;process_stage_list&quot; !=
&quot;${NULL}&quot;"></plmxml_bus:QueryClause></plmxml_bus:SavedQueryDef>
    <plmxml_bus:Text id="id2" primary="en-us"></plmxml_bus:Text>
    <plmxml_bus:Text id="id3" primary="en-
us"></plmxml_bus:Text></plmxml_bus:PLMXMLBusinessTypes>
```

PDPS 部署问题-Oracle 字符集不匹配

作者：张崇宏

审校：钟耀洪

使用版本：Tecnomatix v15.1.2

PDPS 作为一个协同仿真，仿真数据共享平台，安装部署的过程中少不了安装数据库服务器和客户端的过程。在 PDPS 系统安装部署的过程中，我们常常会遇到各种问题。在往期的期刊中我们刊登了许多问题的解决办法，本期继续添加其他问题，希望在用户们遇到相类似问题时能够提供些帮助。

问题现象：在 PDPS 系统安装部署过程中，在绑定 Schema 时，提示如下图 1 所示的报警：The following error occurred while associating shcema:在 Oracle 客户端（eMserver 计算机上）和数据库字符集之间存在不匹配。

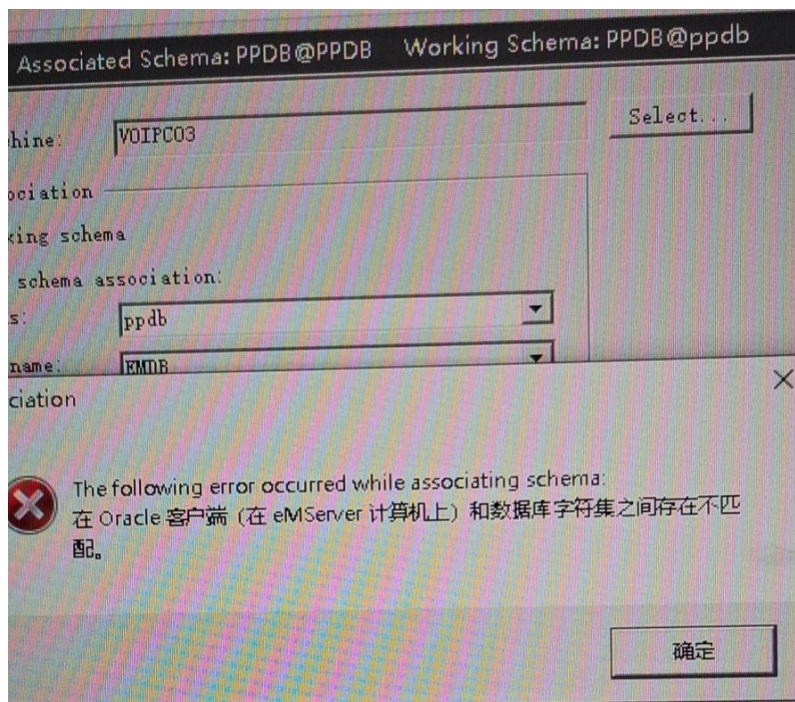


图 1

或登陆时遇到如图 2 所示提示报警

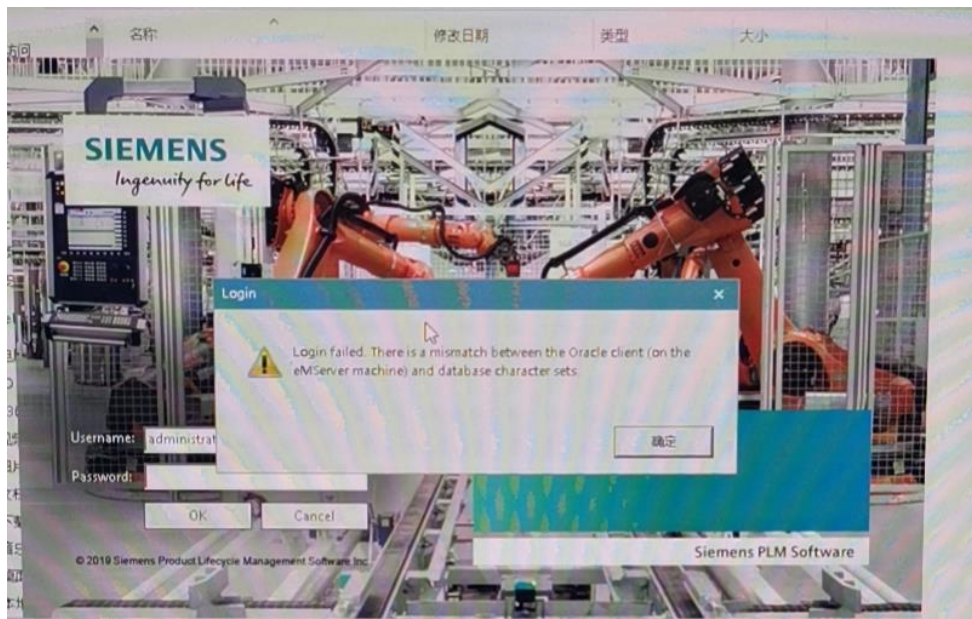


图 2

原因：出现该问题的原因是 Oracle 客户端和 Oracle 服务器的 NLS_LANG 注册表值不一致，导致 Oracle 客户端和数据库字符集之间存在不匹配的错误。

解决方法：

1.在安装了 Oracle Client 应用的计算机上打开【运行】。可使用组合键 Win+R 打开，如图 3。

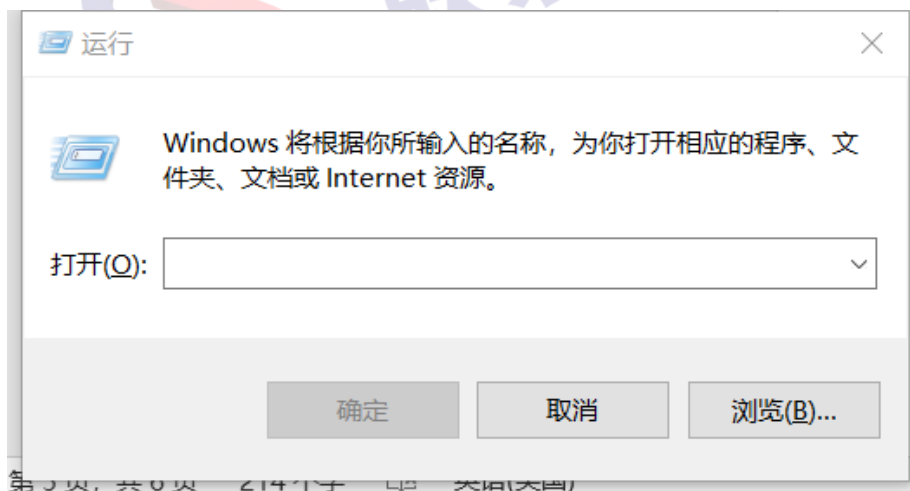


图 3

2.在打开一栏输入“regedit”，点【确认】打开，如图 4。

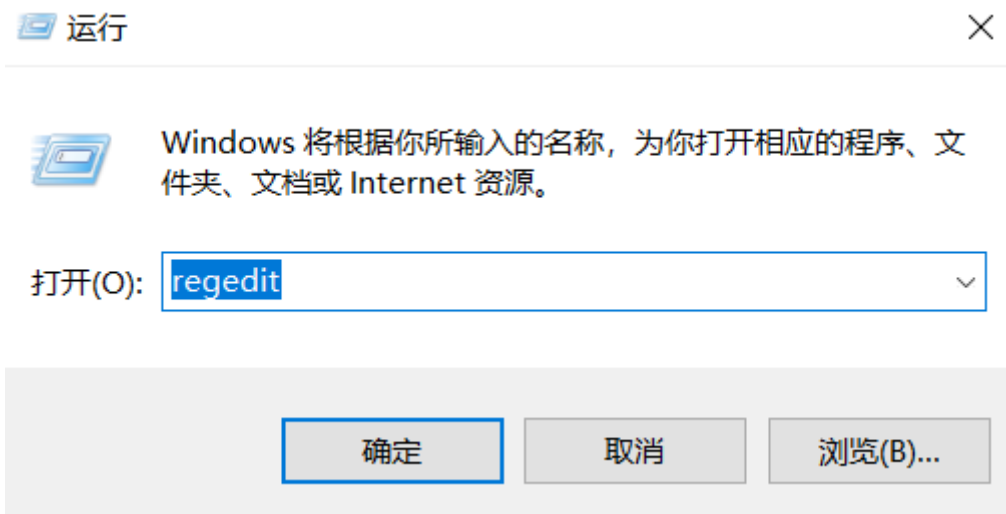


图 4

3. 如图 5，打开以下注册表路径：

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\oracle\KEY_
OraClient11g_home1

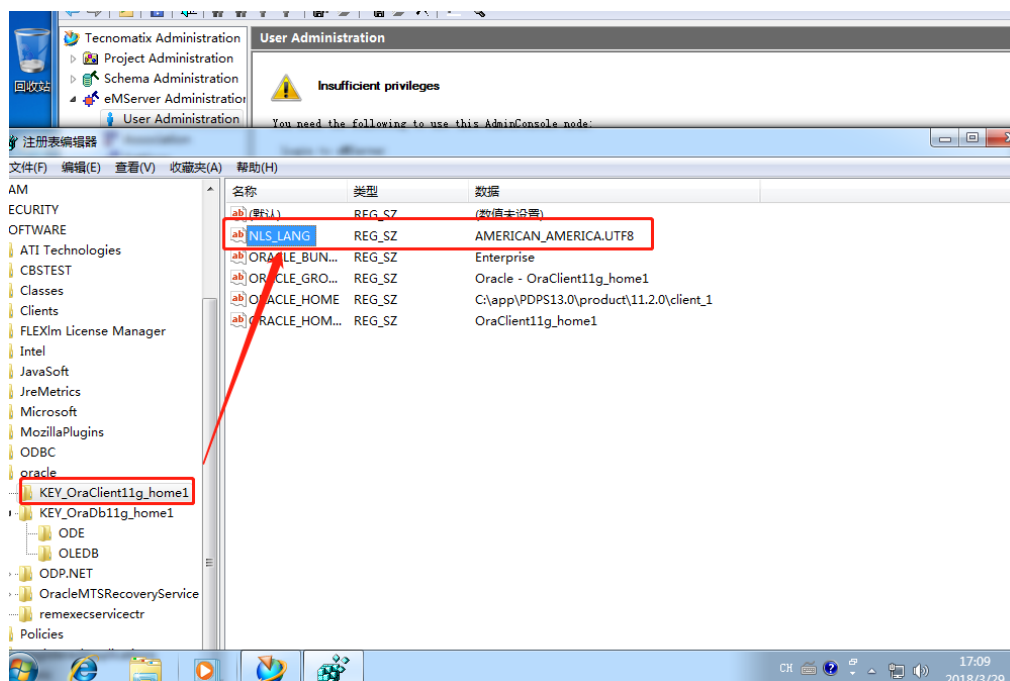


图 5

4.修改 NLS_LANG 的值为 AMERICAN_AMERICA.UTF8。

5.报错消除，重新执行想要的操作即可。

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