# Teamcenter定制开发课程

#### Teamcenter custom development courses

## 第一课:入门综述与Rich Client开发入门 Getting started



作者:王江江 邮件: James.wang@vdior.com

-年五月|上海 二零

© Copyright Value Delivery Innovator 2011



#### TEAMCENTER

Siemens and the Siemens logo are registered trademarks of Siemens AG. Teamcenter is a registered trademark of Siemens Product Lifecycle Management Software Inc. This software and related documentation are proprietary to Siemens Product Lifecycle Management Software Inc. 2008 Siemens Product Lifecycle Management Software Inc.

All Rights Reserved.



## 课程目标

- 知道Teamcenter技术架构;
- 知道Teamcenter开发方式;
- 掌握Teamcenter Rich Client开发环境的搭建;
- 掌握Teamcenter Rich Client中插件工程的建立;





目录



我们能怎么客户化定制

Teamcenter体系结构概述

RCP开发原理介绍

Teamcenter中类结构介绍

开发环境部署



# 为什么要客户化定制

Teamcenter的标准功能基本可以满足客户使用功能,但是基于每家公司业务问题以及使用习惯,还有一些特殊的应用与数据整合,所以会产生不同的业务需求,这个时候我们就需要通过客户化去满足客户的应用需求。



## 我们能怎么客户化定制

#### ■ 服务端定制

通过结合使用集成工具包(ITK的)或服务的外部应用Teamcenter的Teamcenter的行为。比如前处理,后处理,SOA接口,ITK函数, Handler等等。

#### ■ 客户端定制

通过开发让用户操作更加方便点,如一些对话框,操作界面等等的定制。可以开发一些自己的功能模块进行提高用户的操作效率。

#### ■ 数据显示与表单定制

通过客定制开发,可以对系统中对象图标的显示,还有表单界进行一个详细的定制与开发。

#### ■ 数据共享

通过定制开发可以达到Teamcenter系统中的一些数据与别的系统进行共享或者Teamcenter共享别的系统中数据,比较典型的有报表,CAD 集成,ERP集成,CAPP集成等等。



## Teamcenter架构体系



**Rich client architecture** 



▶ RCP简介

RCP是Rich Client Platform的缩写。其实就是利用Eclipse核心平台和一些有用的插件。 进行应用系统的构建。只要新的代码符合插件的代码结构,将代码放入工作空间后,Eclipse 平台会自动加载新代码,构成新的Eclipse平台。



▶ RCP体系结构





#### > Eclipse平台在文件系统中的目录结构

E Clipse
E Configuration
settings
- Corg. eclipse. core. runtime
+ org. eclipse. equinox. app
org. eclipse. equinox. simpleconfigurator
org. eclipse. equinox. source
+ org. eclipse. osgi
org. eclipse. ui. intro. universal
The org. eclipse. update
in Han Francisco in the second
Gorge eclipse, cvs. source_1.1.100.v20030314-121312c3bj333317M0j02f
$\blacksquare$ $\Box$ org. eclipse. equinox, p2, user, ui, source_1, 1, 0, v2000000-1440-1440-10F05F0510512011K
General Street For Section 2 1 1 0 - 20000002-7-7-7-8-81 RB/2/7-2-8-28
$\blacksquare  \bigcirc  org. estimate interpret in the source 3, 5, 0, x20090527=2000-7x88FF a Francy a 33D (7, c)$
$\blacksquare \qquad \qquad$
The org eclipse nde source 3.5.0 x20090123-777 F8NFF-70VXbWiu-80b95
$\blacksquare \bigcirc org \ eclipse \ pde \ 3 \ 5 \ 0 \ x20090123 - 727 \ F8NFE - x0VXbWin - 80b95$
The org, eclipse, platform, source 3, 5, 0, v20090611a-9/EeG10Ft9cmRTh0403aR_foSMvTR2sT
The org eclipse platform 3.5.0 v20090611a-9gEeG1HFt9cmRTh0403aR foSMvTR2sT
T org. eclipse.rcp. source 3, 5, 0, v20090519-9SA0FwxFv6x089WEf-TWh11
Image: eclipse.rcp_3.5.0.v20090519-9SA0FwxFv6x089WEf-TWh11
Image: aclipse.sdk 3.5.0.v20090423-7Q7bA7DPQ1SM60YQ7wKYJhI-z-xZJCmLTBRhtZ0o61b1z
E plugins JH II I 30
\pm 🚞 org. apache. ant_1.7.1. v20090120-1145
표 🚞 org. eclipse. core. runtime. compatibility. registry_3. 2. 200. v20090429-1800
표 🚞 org. eclipse. equinox. launcher. win32. win32. x86_1. 0. 200. v20090519
T 😂



#### ■ Eclipse平台应用运行的界面(Eclipse为例)





#### ■ RCP应用运行的界面(Teamcenter为例)



© Copyright Value Delivery Innovator 2011



> RCP中视图和透视图说明

RCP中的每个视图都对应每个应用中的各个界面构件。每个透视图对应一个应用组合,如在Teamcenter中我的Teamcenter,结构管理器,分类管理等等。



■ Eclipse RCP开发具有的优点:

组件化:基于Eclipse的系统设计由被称为plug-ins的插件构成,可以通过扩展点进行 配置,也可以被不同应用程序共享。

便利性:Eclipse RCP对各个平台下的产品包装提供了强有力的支持,其开发的RCP甚至可以在嵌入式设备、掌上电脑上运行。Sun公司对Java的口号是"write once, run everywhere", Eclipse也可以说是"RCP run everywhere"。

智能安装和升级:Eclipse提供了专门的Update组件,可以实现通过HTTP、Web站

点、复制等多种方式进行安装和更新,一扫早期富客户端应用部署升级的麻烦。



可扩展性:Eclipse基于插件进行扩展的思想使得用户可以方便地搭建各种规模、类型和用途的应用程序。按照Eclipse官方的说法,Eclipse RCP一开始就被设计为可扩展的。 本地观感及使用体验:Eclipse为各种操作系统提供了本地图形接口包。当RCP运行时,Eclipse首先直接调用本机窗口组件,只有没有本机所需组件时才进行模拟。无论RCP在哪种操作系统上运行,都可以保持与本机一致的外观和行为。一个设计优良的富客户端,可以提供诸如拖曳操作、剪切板、导航等UI元素。UI设计者也可以利用各种界面工具,轻松设计出完美的用户界面。



■ Swing 概述

Java Swing 是 Java Foundation Classes (JFC)的一部分,它是试图解决 AWT 缺点的一个尝试。在 Swing 中, Sun 开发了一个经过仔细设计的、灵活而强大的 GUI 工具包。不幸的是,这意味着我们又要花一些时间来学习 Swing 了,对于常见的情况 来说, Swing 有些太复杂了。

Swing 是在 AWT 组件基础上构建的。所有 Swing 组件实际上也是 AWT 的一部分。Swing 使用了 AWT 的事件模型和支持类,例如 Colors、Images 和 Graphics。 Swing 组件、布局管理器以及事件总结如下。正如您可以看到的一样,这些组件集比 AWT 提供的组件集更为广泛,与 SWT 组件集相比也毫不逊色。



■ SWT 概述

与 AWT 的概念相比, SWT 是一个低级的 GUI 工具包。JFace 是一组用来简化使用 SWT 构建 GUI 的增强组 件和工具服务。SWT 的构建者从 AWT 和 Swing 实现中学习了很多经验,他们试图构建一个集二者优点于一体而没 有二者的缺点的系统。从很多方面来说,他们已经成功了。

SWT 也是基于一个对等体实现的,在这一点上它与 AWT 非常类似。它克服了 AWT 所面临的 LCD 的问题,方 法如下:定义了一组控件,它们可以用来构建大部分办公应用程序或开发者工具,然后可以按照逐个主机的原则,为 特定主机所没有提供的控件创建模拟 控件(这与 Swing 类似)。对于大部分现代主机来说,几乎所有的控件都是基 于本地对等体的。这意味着基于 SWT 的 GUI 既具有主机外观,又具有主机的性能。这样就避免了使用 AWT 和 Swing 而引起的大部分问题。特定的主机具有一些低级功能控件,因此 SWT 提供了扩充(通常是模拟的)版本(通 常使用 "C" 作为名字中的第一个字母),从而可以产生更一致的行为。



在大部分情况中,都是使用 Swing 与结合了 JFace 的 SWT 一起构建GUI的。通常来说,每个工 具包都非常完整且功能强大,足以构建功能完善的 GUI,但是 Swing 通常要比单独使用 SWT (不使 用 JFace 时)更好。Swing 具有内嵌于 Java 技术的优点,是完全可移植的,无可争议地是一种更好的 架构。Swing 也具有高级图形应用程序所需要的优点。SWT 具有可以作为本地应用程序实现的优点, 这可以提高性能,并利用基于 SWT 的 GUI 来实现本地兼容性。

如果只为一种平台来开发系统,那么 SWT 就具有主机兼容性方面的优点,包括与主机特性的集成,例如在 Windows 上对 ActiveX 控件的使用。

Teamcenter2007以上都是采用SWT与Swing结合进行客户端构建的。



- 客户化开发中常用到的插件
  - ✓ com.teamcenter.rac.aifrcp
  - ✓ com.teamcenter.rac.common
  - ✓ com.teamcenter.rac.external
  - ✓ com.teamcenter.rac.kernel
  - ✓ *com.teamcenter.rac.tcapps*
  - ✓ com.teamcenter.rac.util



#### com.teamcenter.rac.aifrcp

Teamcenter 基础客户化插件,一些主要的接口以及抽象类,入口类等都在该插件中进行了定义,如:

AbstractAIFApplication

AbstractAIFCommand

AbstractAIFDialog

AbstractAIFOperation

AIFDesktop

AIFPortal

AbstractAIFAction



#### com.teamcenter.rac.common

Teamcenter的一些动作和菜单都在该插件包中进行了定义,如菜单栏,工具栏,以及右键菜单等。

首先要说的是,界面上所有菜单,以及一些公共组件及Form的顶级实现。一般的菜单动作都在

com. teamcenter. rac. common. actions中进行了定义,如新建Item为例:

1. 在该包中建立NewItemAction类,并集成AbstractAIFAction类,实现

public void run()方法。

2. 在action.properties文件中进行了注册,如下所示:

```
# File -> New -> Item
# ------
newItemAction=com.teamcenter.rac.common.actions.NewItemAction
newItemAction.ICON=images/newitem_16.png
newItemAction.COMMAND=newItemCommand
newItemAction.ACCELERATOR=ctrl pressed T
newItemCommand=com.teamcenter.rac.commands.newitem.NewItemCommand
```

3. 相应的,在该插件中对应一个com.teamcenter.rac.commands.newitem包,创建Item的业务逻辑都在该包中进行了实现。



#### com.teamcenter.rac.external

Teamcenter插件中引用到的第三方类大部分都在该插件中进行了集成,可以说该插件为资源性插件。如

下图:

🛋 activation.jar 🗃 antlr-2.7.2.jar 🛃 Applicatic 🎦 Registry Client.jar avalon-framework-4.1.5.jar 🔊 a×is-ant.jar 🚂 a×is.jar 🔄 commons-codec.jar 🥁 🚂 commons-discovery-0.2.jar -------jacorb.jar -------jaxb-api.jar ----**ia** ja≍b-×j⊂.jar ---**--**\_\_ja×rp⊂.jar --------jgraph.jar 🔄 logdi 1.2 Size



#### com.teamcenter.rac.kernal

该插件上基本上对业务对象进行了组件的定义,基本上系统中每个业务对象都在该插件中有相应的类去实现。如Folder对象, 系统中为文件夹对象,该对象对应的系统类是:

1. TCComponentFolder 该类集成了TCComponent类。扩展定义了 Folder的获取以及和属性的修改方法。基本上所有的业务对象 都集成于TCComponent类。相应的我们可以联想到, Item 对应的组件类为TCComponentItem; Dataset 对应的组件类为 TCComponentDataset。

2. 一个业务类对应的组件类相应的也会对应一个业务类型类,如Folder对应的业务类型组件类为TCComponentFolderType。该类集成 于TCComponentType类。相应的我们可以联想到,Item对应的组件类型类为TCComponentItemType;Dataset对应的组件类型类为 TCComponentDatasetType。该类主要扩展定义了相应的业务对象的创建以及另存为等方法。



■ 接下来,我们可以看看怎么通过业务对象对应的类型组件类去创建业务对象,还是以Folder为例:

TCComponentFolderType t = (TCComponentFolderType)

session.getTypeComponent("Folder");

TCComponentFolder f = t.create("My Folder Name",

"My Folder Description", "My Folder Type");



#### com.teamcenter.rac.tcapps

Teamcenter中部分应用的基础实现都在该类中进行了实现,这个主要是遗留问题,虽然现在的应用大部分都是以相应的插件进行了区别与分类,但是大部分应用的业务逻辑都在该插件中进行实现,还有一些公共组件等。如:

com.teamcenter.rac.cme.mpp 对MSE应用进行了实现

com.teamcenter.rac.pse 对结构管理器进行了实现

com.teamcenter.rac.querybuilder 对查询构建器进行了实现

com.teamcenter.rac.explorer 对MyTeamcenter进行了实现



#### com.teamcenter.rac.util

- 1.该插件是Teamcenter对大部分Swing组件进行了继承与重写。如: iTextField继承于JTextField; iComboBox类为 Teamcenter自己定义的组件。
- 2. 该插件中还定义了一些公共工具类,如:TcLogger日志控制类, Registry注册控制类等。



- 开发工具准备
- 1.JDK1.6.0\_14
- 2. Eclipse 3.5



▶开发环境配置

# ①在Eclipse中,选择Window→Preferences,打开Preferences对话框。

e Preferences		
type filter text	Java	
<ul> <li>General</li> <li>Ant</li> <li>Helpo</li> <li>Install/Update</li> <li>Plug-in Development</li> <li>Run/Debug</li> <li>Team</li> </ul>	General settings for Java development: Action on double click in the Package Explorer © Go into the selected element © Expand the selected element When opening a Type Hierarchy © Open a new Type Hierarchy Perspective © Show the Type Hierarchy View in the current perspective Refactoring Java code © Save all modified resources automatically prior to refactori © Rename in editor without dialog Search © Use geduced search menu Java dialogs Clear all 'do not show again' settings and show all hidden dialogs again <u>Restore Default</u>	ng       
?	OK	Cancel



# ②在左边的树列表中,双击Java,选择Installed JREs。

a. 在 Installed JREs 列表,点击Add 按钮。

e Preferences				
type filter text	Installed JREs			$\Leftrightarrow \bullet \bullet \bullet \bullet \bullet$
⊕…General ⊕…Ant ⊕…Help ⊕…Install/Update	Add, remove or edit JF build path of newly cr Installed JREs:	E definitions. By default, the chec eated Java projects.	ked JRE i	s added to the
⊡-Java	Name	Location	Type	<u>A</u> dd
H Build Path	🗹 🛋 jrel. 6.0_05	C:\Program Files\Java\jre1.6.0_05	Standar	7.11.1
Code Style				<u>E</u> art
i Compiler				Dupligate
±. Editor				Remove
H-Installed JAEs				Search
Properties Files Edit				
+ Plug-in Development				
⊞ Kun/Debug				
	•			
(?)		0	К	Cancel



b.在 JRE Type 对话框, 选择 Standard VM点击 Next.

e Add JBE	
JKE Type Select the type of JRE to add to the wor	kspace.
Installed JRE <u>Typ</u> es: Execution Environment Description Standard 1.1.x VM	
Standard VM	
< Back	lext > Einish Cancel



c.在 JRE Definition 对话框,指定到JDK的安装目录。

e Add JBE	
JRE Definition Specify attributes for a JRE	
JRE home:     C:\Program Files\Java\jdk1.6.0_05       JRE name:     jdk1.6.0_05       Default VM Arguments:     Image: Comparison of the second s	Direct <u>o</u> ry
C:\Program Files\Java\jdk1.6.0_05\jre\lib\resource C:\Program Files\Java\jdk1.6.0_05\jre\lib\rt.jar C:\Program Files\Java\jdk1.6.0_05\jre\lib\jsse.jar C:\Program Files\Java\jdk1.6.0_05\jre\lib\charsets C:\Program Files\Java\jdk1.6.0_05\jre\lib\charsets C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\dnsr C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\dnsr C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum C:\Program Files\Java\jdk1.6.0_05\jre\lib\ext\sum	Add External JARs Javadoc Location Source Attachment Remove Up Down <u>R</u> estore Default
(?) < <u>Back</u> <u>Mext</u> > <u>Finis</u>	h Cancel



## d.点击 Finish。

## e.在Preferences对话框,选择新定义的JRE。

Freferences				
type filter text	Installed JREs			← → → → →
⊕ General ⊕ Ant ⊕ Help ⊕ Install/Update	Add, remove or edit J build path of newly o Installed JREs:	RE definitions. By default, the check reated Java projects.	ked JRE i	is added to the
E. Java	Name	Location	Type	<u>A</u> dd
H Suild Path	☑ ➡ jdk1.6.0_05 □ ➡ jre1.6.0_05	C:\Program Files\Java\jdk1.6.0_05 C:\Program Files\Java\jre1.6.0 05	Standar Standar	<u>E</u> dit
. Compiler 				Dupli <u>c</u> ate
Editor		N		Remove
JUNIT				Search
⊕…Flugin Development ⊕…Run/Debug ⊕…Team				
	•			
?		0	К	Cancel



#### ③<u>在Preferences</u>对话框,双击 Plug-in Development,选择 Target Platform。 \_ 🗆 🗡 🚝 Preferences type filter text **Target Platform** ← → ⇒ → ▼ +-General Add, edit and remove target definitions. The active target definition will be 🗄 - Ant used as the target platform which workspace plug-ins will be compiled and tested 🗄 - Help against. New definitions are stored locally, but they can be moved to a project in the workspace. . ∔ Java Target Definitions: - Plug-in Development 🗹 🛑 Running Platform (Active) Reload. . . API Baselines API Errors/Warnings Compilers Add. . . --- Editors OSGi Frameworks Edit... Target Platform Remove + Run/Debug i i Team Move.. Locations Restore Defaults Apply OK Cancel

?



# ④在 Target Platform 对话框,点击 Add。

Preferences		
type filter text	Target Platform	$ \diamondsuit \bullet $
⊞General ⊕Ant ⊕Help ⊕Install/Update ⊕.Java	Add, edit and remove target definitions. The active target defini used as the target platform which workspace plug-ins will be compi against. New definitions are stored locally, but they can be move in the workspace.	tion will be led and tested d to a project
- Plug-in Development	larget Definitions:	Reload
		Add
Editors OSGi Frameworks		<u>E</u> dit
Target Platform		Remove
		Move
	Locations	
	Restore Defaults	Apply
?	ок	Cancel



## ⑤在Target Definition框,确定 Nothing是选择的,并点击Next。

New Target Definition				>
Target Definition				
Create a new target definition.				$\odot$
Initialize the target definition with:				
Nothing: Start with an empty target definition				
C Default: Default target for the running platform				
C Current Target: Copy settings from the current target platfo	5 xw			
○ <u>T</u> emplate: Base RCP (Binary Only) 🔽	0			
(?)	< Back	<u>N</u> ext >	<u>F</u> inish	Cancel



# ⑥在Target Content 对话框,修改名称为TcTarget并点击 Add。

Hew Target Definition	
arget Content	
Lait the name, description, and plug-ins contained in a target.	$\bigcirc$
ame: New Target	
Locations Content Environment Arguments Implicit Dependencies	
The following list of locations will be used to collect plug-ins for this target definition.	
	<u>A</u> dd
	<u>E</u> dit
	Remove
	Remove All
Show Plug-in Content	
(?) < Back Mext > Fin:	ish Cancel



⑦在 Add Content 对话框选择 Directory 并点击 Next.

i:	🚝 Add Content	
E	Add Content Select a source of plug-ins.	
>f	Directory Installation Features Software Site	
	A directory in the local file system. The plug-ins found in the directory will be added to the target definition.	



## ⑧指定到TC\_ROOT\portal 目录并点击完成Finish。

🚝 Add Content		
Add Divectory		
Select a directory of plug-ins to add to the target.		
Location: D:\UGS\Siemens\Teamcenter8\portal		•
	Browse	<u>V</u> ariables
A Mext >     A Mext >     A	Finish	Cancel



## ⑨点击Finish 按钮。然后选择刚才添加的平台TcTarget,并点击OK.

🚝 Preferences		
type filter text	Target Platform	$\Leftrightarrow \bullet \to \bullet \bullet \bullet$
General     General	Add, edit and remove target definitions. The active target defin used as the target platform which workspace plugins will be comp against. New definitions are stored locally, but they can be mov- in the workspace. <u>T</u> arget Definitions:	ition will be iled and tested ed to a project
API Baselines API Baselines API Errors/Warnings Compilers Editors	☑ ● Bunning Platform (Active) ☑ ◎ TcTarget	Reload
OSGi Frameworks 		<u>E</u> dit
i+i⊷ Kun/Debug i+i⊷ Team		<u>M</u> ove
	Locations	
	D:\UGS\Siemens\Teamcenter8\portal = D:\UGS\Siemens\Teamcente	r8\portal (617
	Restore Defaults	Apply
?	OK	Cancel



## ▶通过Eclipse去运行Rich Client。

## ① 在Eclipse中, 选择 Run→Run Configurations.

🚝 Run Configurations		×
Create, manage, and run co	nfigurations	
★ Provide State Stat	ColSigure launch settings from this dialog: - Press the 'New' button to create a configuration of the selected type. - Press the 'Duplicate' button to copy the selected configuration. - Press the 'Delete' button to remove the selected configuration. - Press the 'Filter' button to configure filtering options. - Edit or view an existing configuration by selecting it. Configure launch perspective settings from the <u>Perspectives</u> preference page.	
Filter matched 6 of 6 items		
?	<u>R</u> un.	Close



②在 Create, manage, and run configurations 对话框的左边,双击Eclipse Application, 然后选择 New\_configuration,并且修改名称为tcRun。

Eun Configurations	×
<b>Create, manage, and run configura</b> Create a configuration to launch an D	ipse application.
type filter text type filter text Eclipse Application Java Applet Java Applet Junit JUnit Plug-in Test OSGi Framework	ane: tcRun Main @= Arguments @ Plug-ins & Configuration Tracing @ Environment Common Workspace Data Location: \${workspace_loc}//runtime=New_configuration © glear: @ gorkspace © log only Workspace File System Variables Ask for confirmation before clearing Program to Run @ Run a groduct: com.teamcenter.rac. aifrcp.product @ Run an application: org eclipse.ui.ide.workbench Java executable: @ default @ java @ Egecution environment: CDC-1 0/Foundation-1.0 (jrel 8.0_05) @ Runtimg JRE: jdkl.8.0_05 @ Runtimg JRE: jdkl.8.0_05 @ Runtimg environment: CDC-1 0/Foundation-1.0 (jrel 8.0_05) @ Jagalled JREs Bgotstrap entries:
Filter matched 7 of 7 items	
(?)	<u>R</u> un Close



## ③确保Run a product 列表选择为com.teamcenter.rac.aifrcp.product。

🗬 Bun Configurations	×
<b>Create, manage, and run configura</b> Create a configuration to launch an B	tions cclipse application.
Image: Second system         type filter text         Image: Second system         Image: Second system	Mame: tcRud Main & Arguments Plug-ins Configuration Tracing Environment Common Workspace Data Location: \$ {workspace_loc}//runtime-New_configuration Clear: © gorkspace C log only Xorkspace File System Variables Ask for confirmation before clearing Program to Run © Run a groduct: com. teamcenter. rac. aifrcp. product © Run an application: org eclipse. ui.ide.workbench Java executable: © default © java © Egecution environment: CDC-1.0/Foundation-1.0 (jre1.6.0_05) © Runting JRE: jdkl.6.0_05 Bgotstrap entries: Apply Regert
?	Run Close



## ④点击 Arguments页,并在VM arguments 框中输入: - Xms256m - Xmx1024m

Run Configurations		2
Create, manage, and run configura	ations	
Create a configuration to launch an 3	Eclipse application.	
type filter text Eclipse Application Eclipse Applet Java Applet Java Applet JujUnit JUJUnit OSGi Framework	<pre>Mane: tcRun Mane: tcRun Main (************************************</pre>	Varjables Variables Variables
Filter matched 7 of 7 items		



×

#### 开发环境安装与部署

## ⑤点击Envirement页面,新建PATH和FMS\_HOME选项 FMS\_HOME=TC\_ROOT\fcc

#### PATH=%FMS\_HOME%\bin;%FMS\_HOME%\lib;%TPR%;%PATH%

- Aun Configurations		
Create, manage, and run configurations		
Create a configuration to launch an Eclipse application.		

type filter text Eclipse Application	Name: tcRun	wints ( 🌸 Plug-ins ( 🛜 Configuration ( 🚔 Tracing ( 🚾 Env. ables to <u>s</u> et:	ironment <sup>»</sup> 1
Tava Applet	Variable	Value	N <u>e</u> w
Java Application	FMS_HOME	D:\UGS\Siemens\TEAMCE~1\fcc D:\UGS\Siemens\TEAMCE~1\fcc\bin;D:\UGS\Siemens\	Se <u>l</u> ect
JUnit Plug-in Test			E <u>d</u> it
OSGi Framework		Image: Sector	Rem <u>o</u> ve
Filter matched 7 of 7 items	• <u>Append</u> envir Replace nati	onment to native environment ve environment with specified environment Apply	Reyert
rifter matched   of   items			
?		Run	Close



▶编写服务端部署脚本。

1.在TC\_Root\portal 目录中,打开portal.bat,修改脚本为:

**	▼   <sup>2</sup>		
	portalServer.bat		
	1	@echo off rem	
14 👘 👘 👅 📡	- - 10	rem TPR is short for TC_PORTAL_ROOT to reduce command line length title Teamcenter Rich Client	
=	13	start "TAO ImR" /min cmd /c "D:\UGS\Siemens\Teamcenter8\iiopservers\start_imr.bat"	
{+} ■I			
+			

2.另存该脚本为portalServer.bat,通过Eclipse启动TC客户端时,必须通过该脚本 先启动TAO控制台。