

# TFS 使用手册(Java 版)

1. TFS 简介.....	3
1.1 为什么要用 TFS.....	3
1.2 TFS 的逻辑 workflow.....	3
1.3 TFS 体系结构.....	4
2. Eclipse 中安装支持 TFS.....	5
3. 在 TFS 上开展团队项目.....	7
3.1 连接到 TFS.....	7
3.2 将本地新建项目上传至 TFS.....	9
3.3 配置本地工作空间.....	13
4. TFS 源代码管理.....	15
4.1 如何与服务器同步.....	16
4.2 如何做 Check-In.....	16
4.3 什么是上架与下架.....	18
5. 项目中各角色对在 TFS 上的应用.....	20
5.1 项目经理及开发经理.....	22
5.1.1 使用 Project 分配任务.....	22
5.1.2 单独分配任务.....	27
5.1.3 风险跟踪.....	30
5.2 测试人员.....	31
5.3 开发人员.....	41

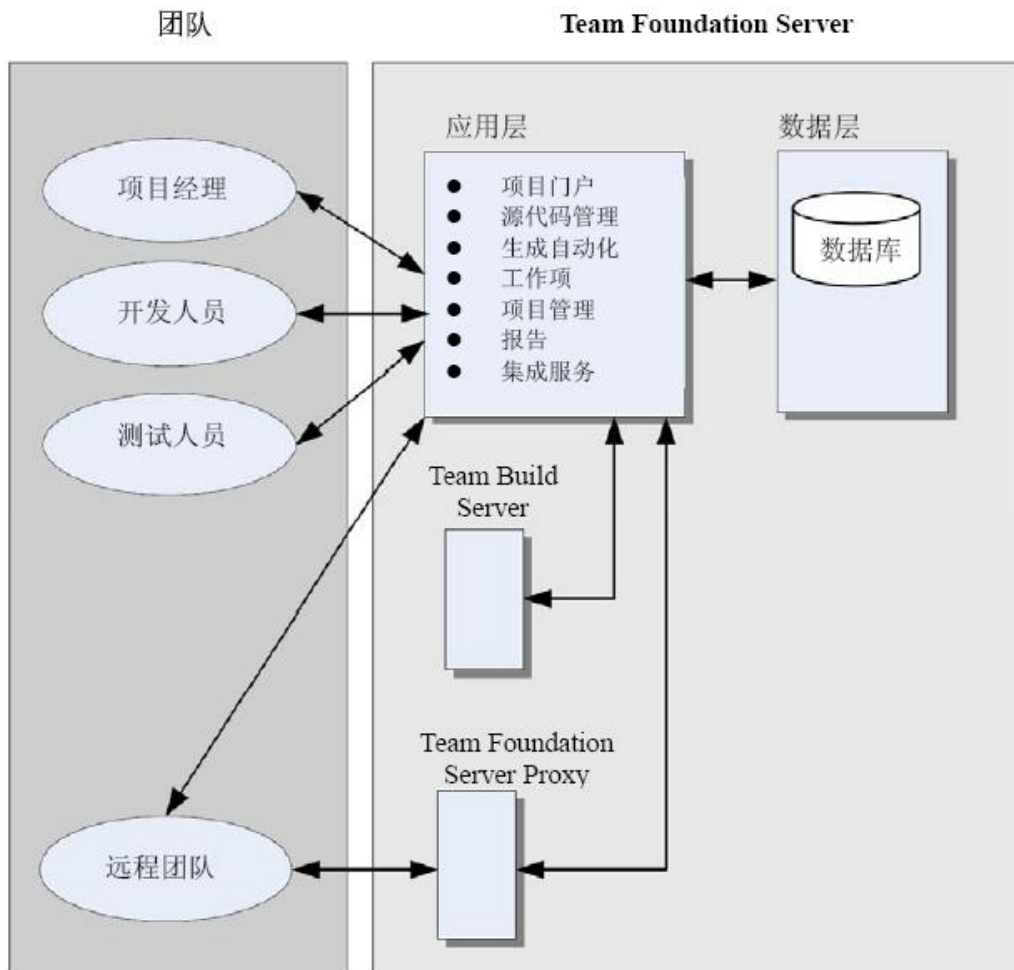
5.4	QA 人员.....	42
5.5	变更申请人.....	43
6.	生成.....	45

# 1. TFS 简介

## 1.1 为什么要用 TFS

TFS 不仅能像 SVN 一样做源代码管理, 它更优秀的地方在于能够将项目中的“任务分派”、“缺陷管理”、“变更管理”、“审批流程”、“问题跟踪”、“自动构建”、“风险控制”等项目管理中的关键活动集成在一起, 并很方便的展示给项目成员。

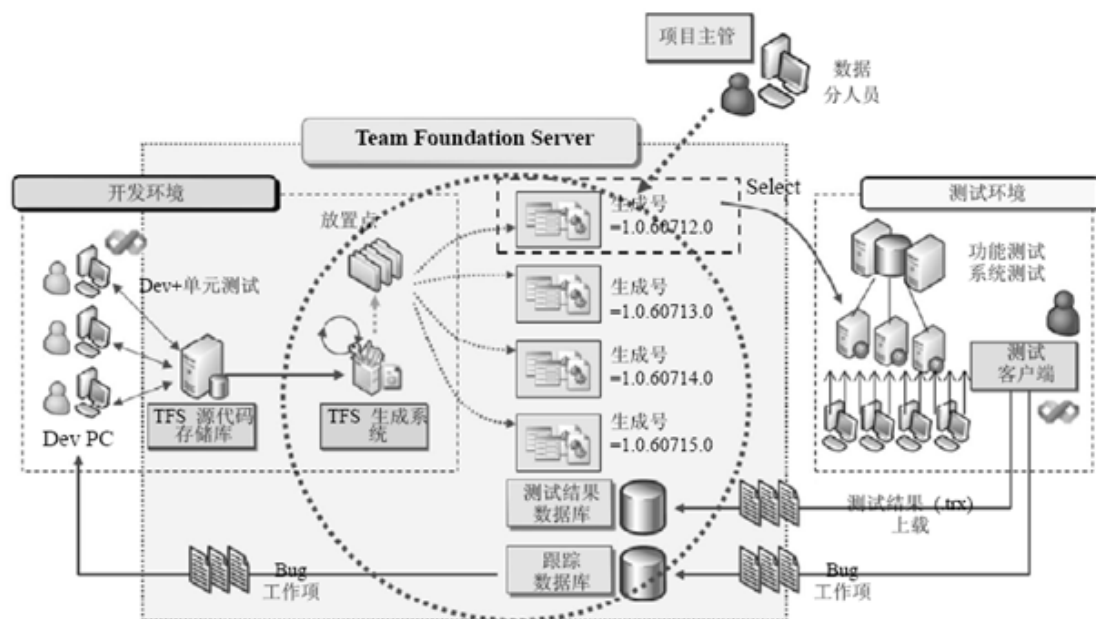
下图概括了 Team Foundation Server 的一个示例逻辑实现, 在图中, 它与软件工程和开发生命周期中最常见的角色关联。



## 1.2 TFS 的逻辑 workflow

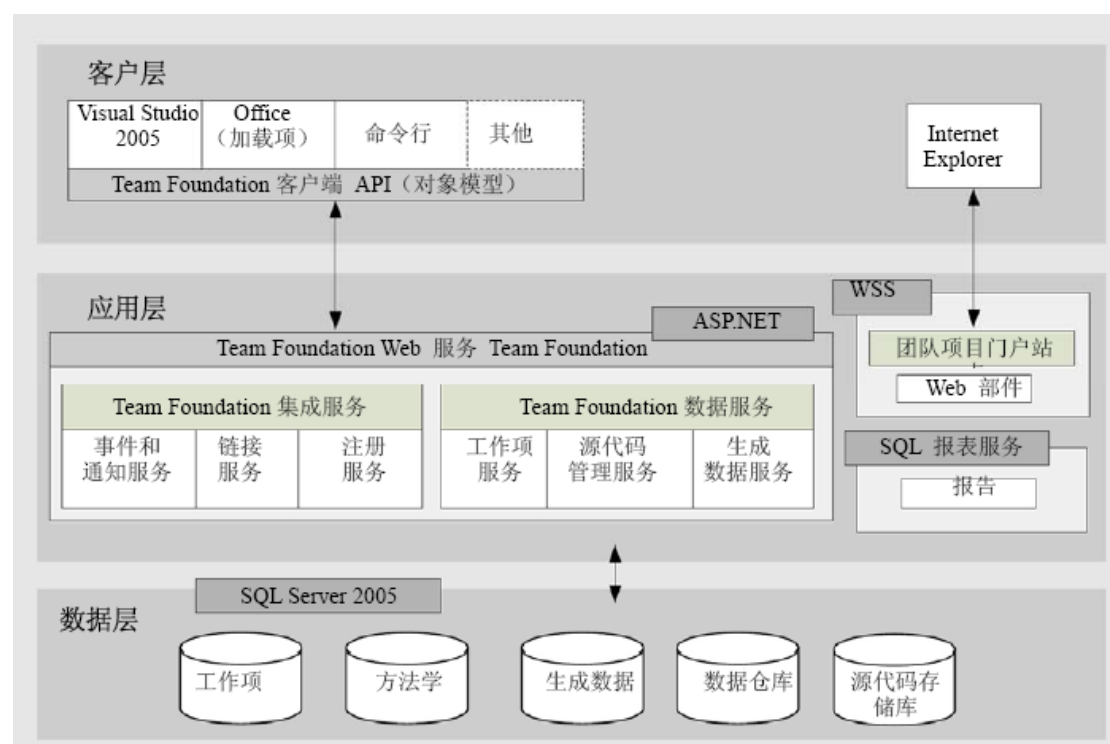
TFS 使开发团队能够将代码存储在一个集中管理的源代码存储库中。可以使用生成服务器, 通过此存储库创建生成, 随后可以将这些生成分发给测试团队。

下图展示了 TFS 的逻辑 workflows 以及开发和测试团队是如何连接在一起的。



## 1.3 TFS 体系结构

TFS 利用了一种逻辑三层体系结构，包括客户层、应用层和数据层。TFS 客户端通过各种 Web 服务与应用层交互；应用层又通过数据层中的各种数据库得到支持。下图展示了 TFS 各层组件及其交互。

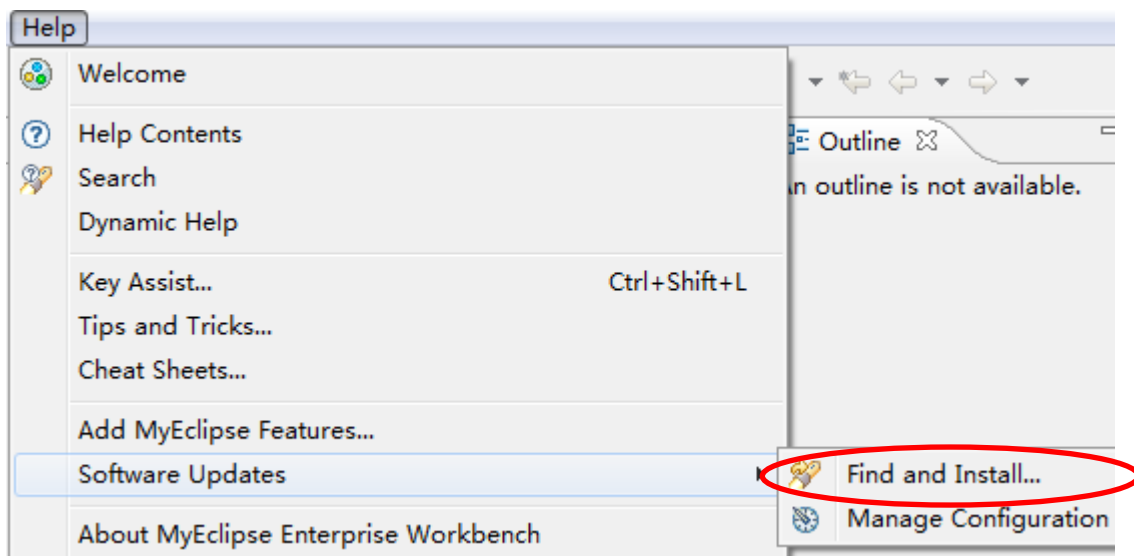


## 2. Eclipse 中安装支持 TFS

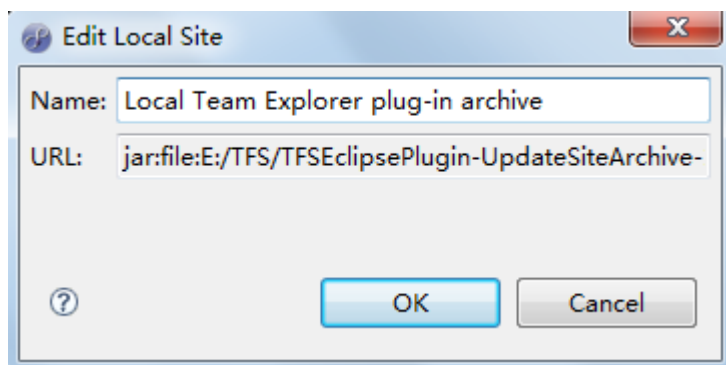
Eclipse 支持 TFS 功能,需下载安装 Team Explorer Everywhere 插件,最新的版本是 Team Explorer Everywhere 下载地址是: <http://www.microsoft.com/en-pk/download/details.aspx?id=47727>  
下载时候请选择对应的语言,并按照“安装指导”来逐步安装:

不同 Eclipse 版本安装该插件,过程一样,但某些设定可能不同,此处以 Eclipse6.5 版本为例做说明,安装步骤如下:

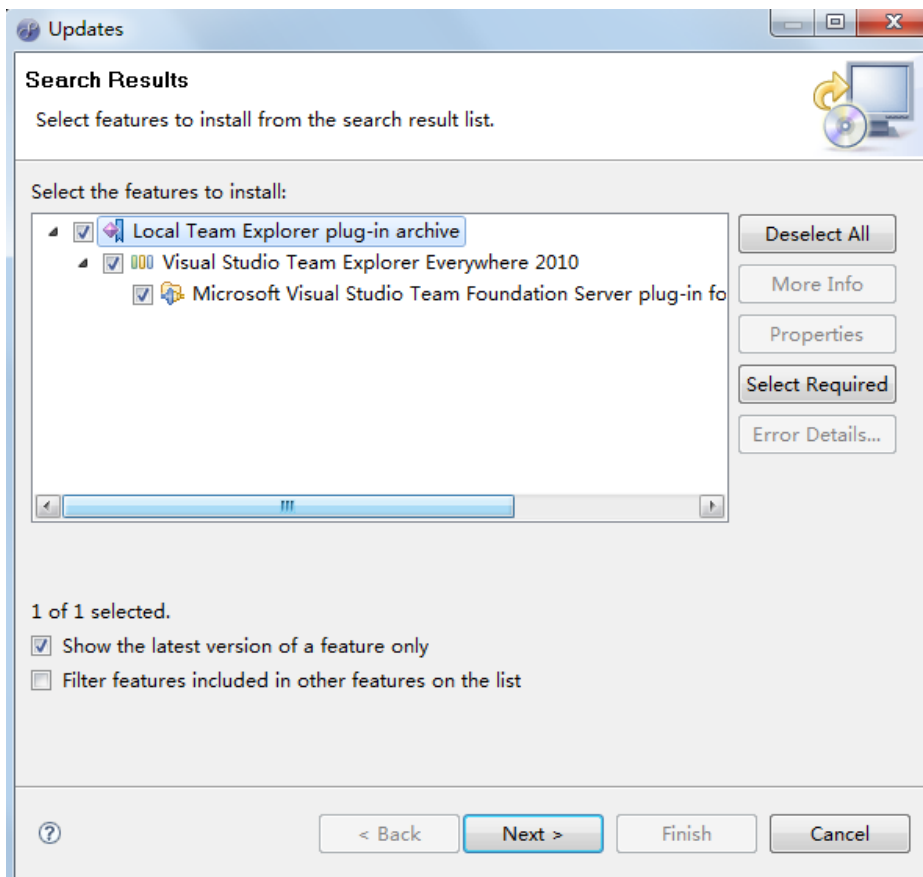
- a. 打开 Eclipse
- b. 在菜单栏选择 Help—>Software Updates→Find and Install,如下图:



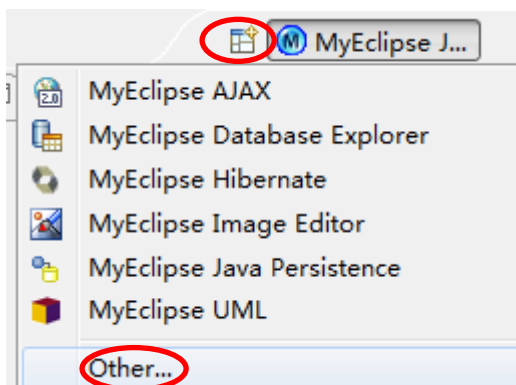
- c. 在弹出的 Install/Update 窗口中,选择 Search for new features to install, 点击 Next
- d. 选择 New Archived Site, 在弹出的窗口中选择要安装的插件, 点击“打开”
- e. 在弹出的 Edit Local Site 的 Name 中, 输入 Local Team Explorer plug-in archive, 点击 OK, 点击 Finish, 如下图:

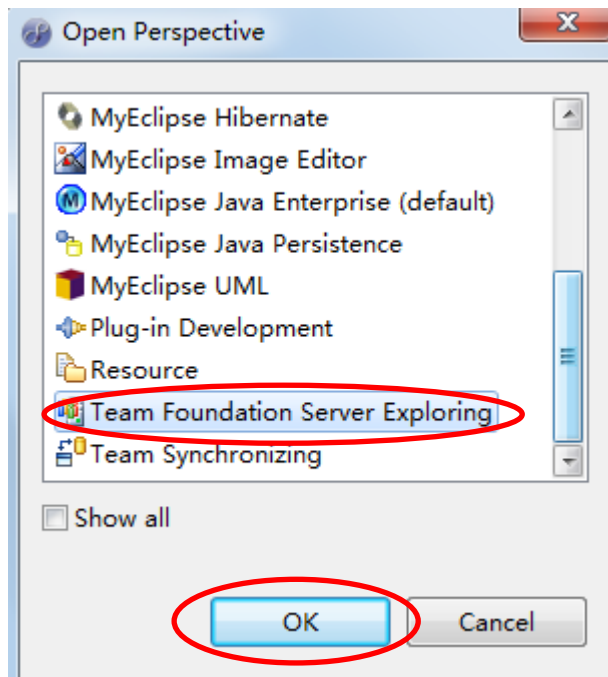


- f. 在弹出的 Search Results 窗口中,选中插件,注意三个选项都要选中, 点击 Next, 如下图:



- g. 在弹出的 Install 窗口中, 选择 I accept the terms in the license agreement, 点击 Next, 点击 Finish, 点击 Install All
- h. 安装完成后, 重启 Eclipse, 开启 Team Foundation Server Exploring, 如下图:

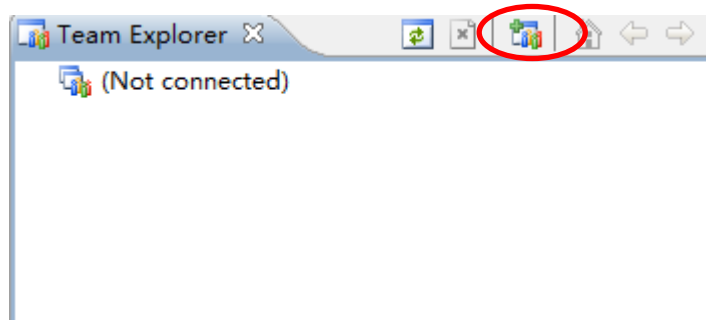




## 3. 在 TFS 上开展团队项目

### 3.1 连接到 TFS

通过 Eclipse 连接到 TFS，点击 TFS 视图中的添加项目按钮，如下图：



初次添加，需输入 Product Key: 7W3RJ-4WX3R-BV8JM-FC8P7-3W7QX，点击 Next

**Add Existing Team Project**

**Product Key**  
Enter your product key.

Thank you for your purchase of Microsoft® Visual Studio® Team Explorer Everywhere 2010.

Product Key: 7W3RJ 4WX3R BV8JM FC8P7 3W7QX

☐ Use a full featured trial version of Microsoft® Visual Studio® Team Explorer Everywhere

< Back **Next >** Finish Cancel

这里需要输入 TFS 的服务器名称或 URL。例如，172.16.1.248。

**Add Existing Team Project**

**TFS Server**  
Connect to a Team Foundation Server.

Login Details Advanced Proxy Details

Name or URL of Team Foundation Server:

Server: 172.16.1.248  
Example: tfs.example.com or http://tfs.example.com:8080/tfs

Authenticate as currently logged-in user

Username: gzcass

Domain: gzcass-HP

Password: ●●●●●●●●

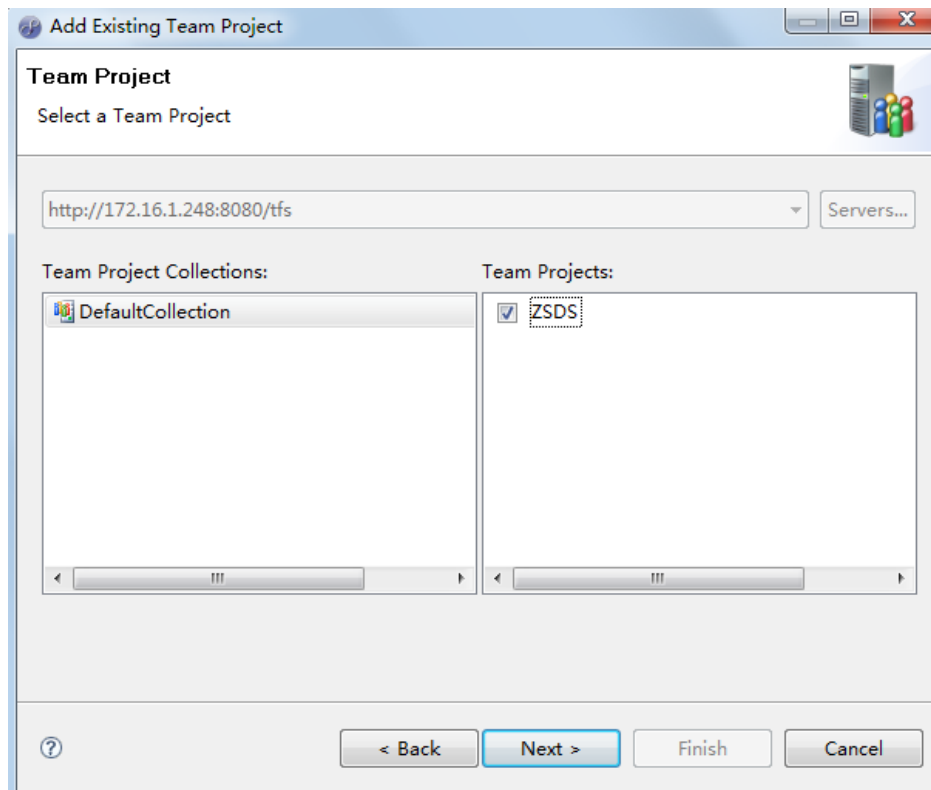
☐ Save password

< Back **Next >** Finish Cancel

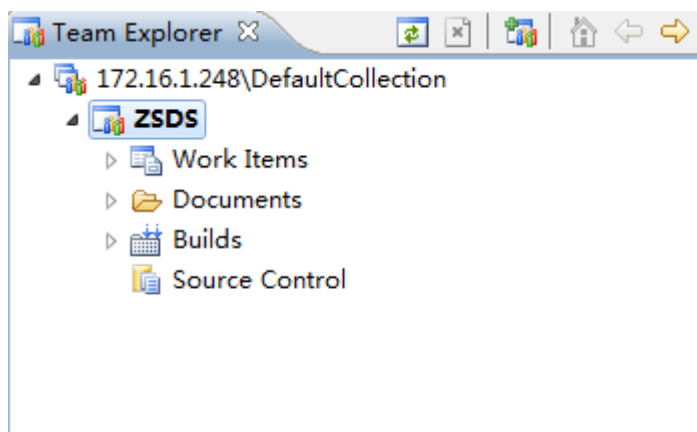
在这里，你可以从组合框里选择服务器，接着选择 DefaultCollection，然后选中连接的项目，点击



Next, 点击 Finish:



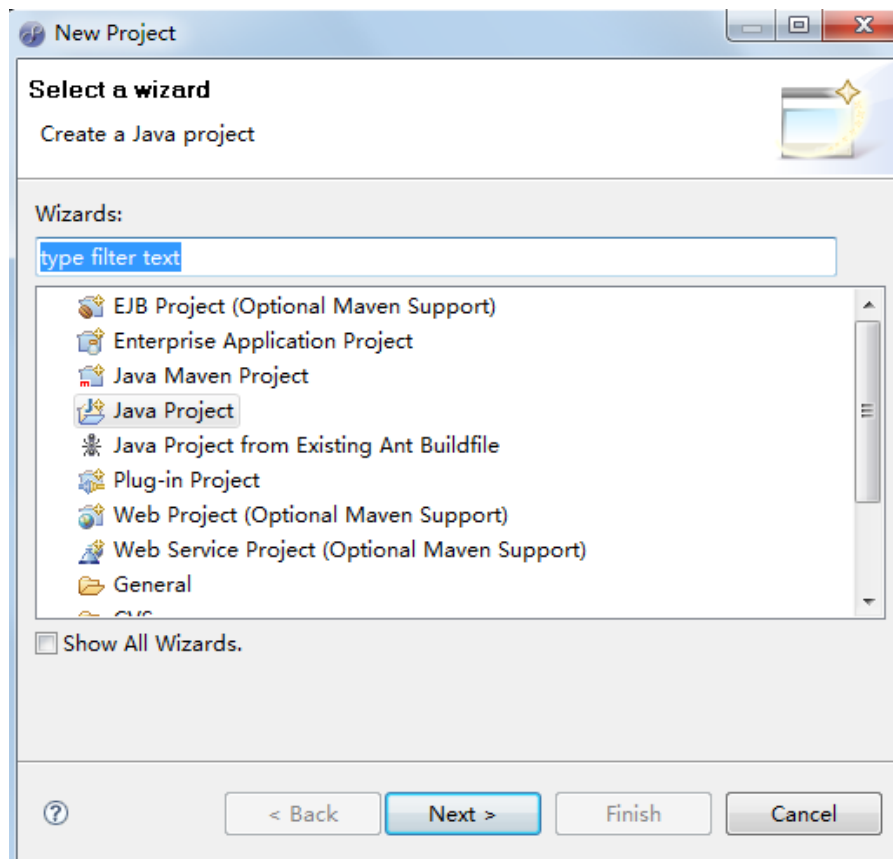
现在 Team Explorer 中有了服务器连接和 ZSDS 项目连接，如下图：



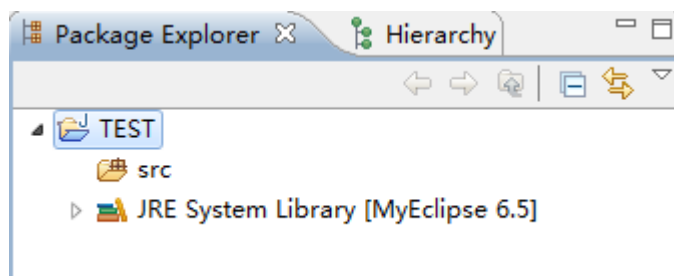
## 3.2 将本地新建项目上传至 TFS

在编码工作开始前，开发经理需要建立符合自己需求的项目，新建项目操作如下。

“File” —> “New” —> “Project” 会出现如下界面：



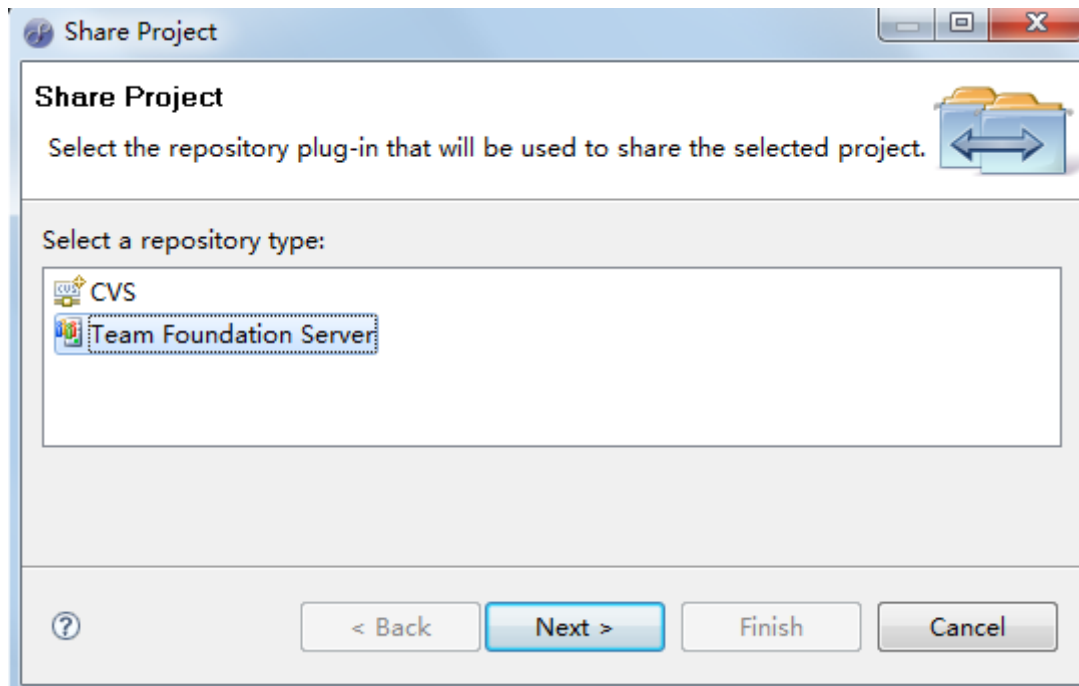
例如选择建立 “Java Project”，点击 Next，输入项目名 “TEST” 确定后所建项目会在 “Package Explorer” 中呈现，如图所示：



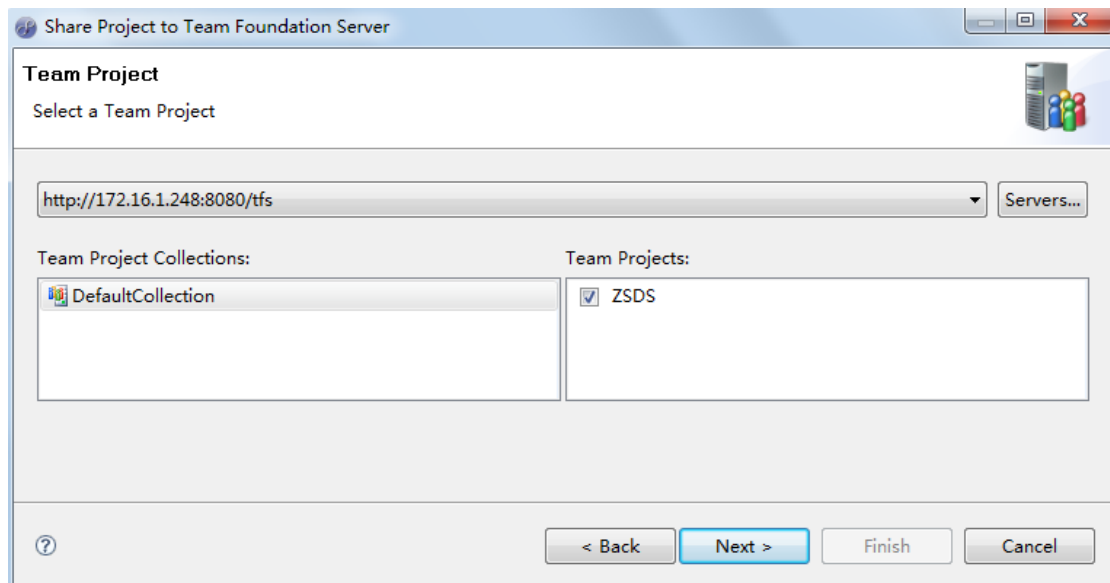
此时需要将本地项目上传至 TFS 服务器，操作步骤如下：

右键单击项目，选择 Team—>Share Project，在弹出的对话框中，选择 Team Foundation Server，

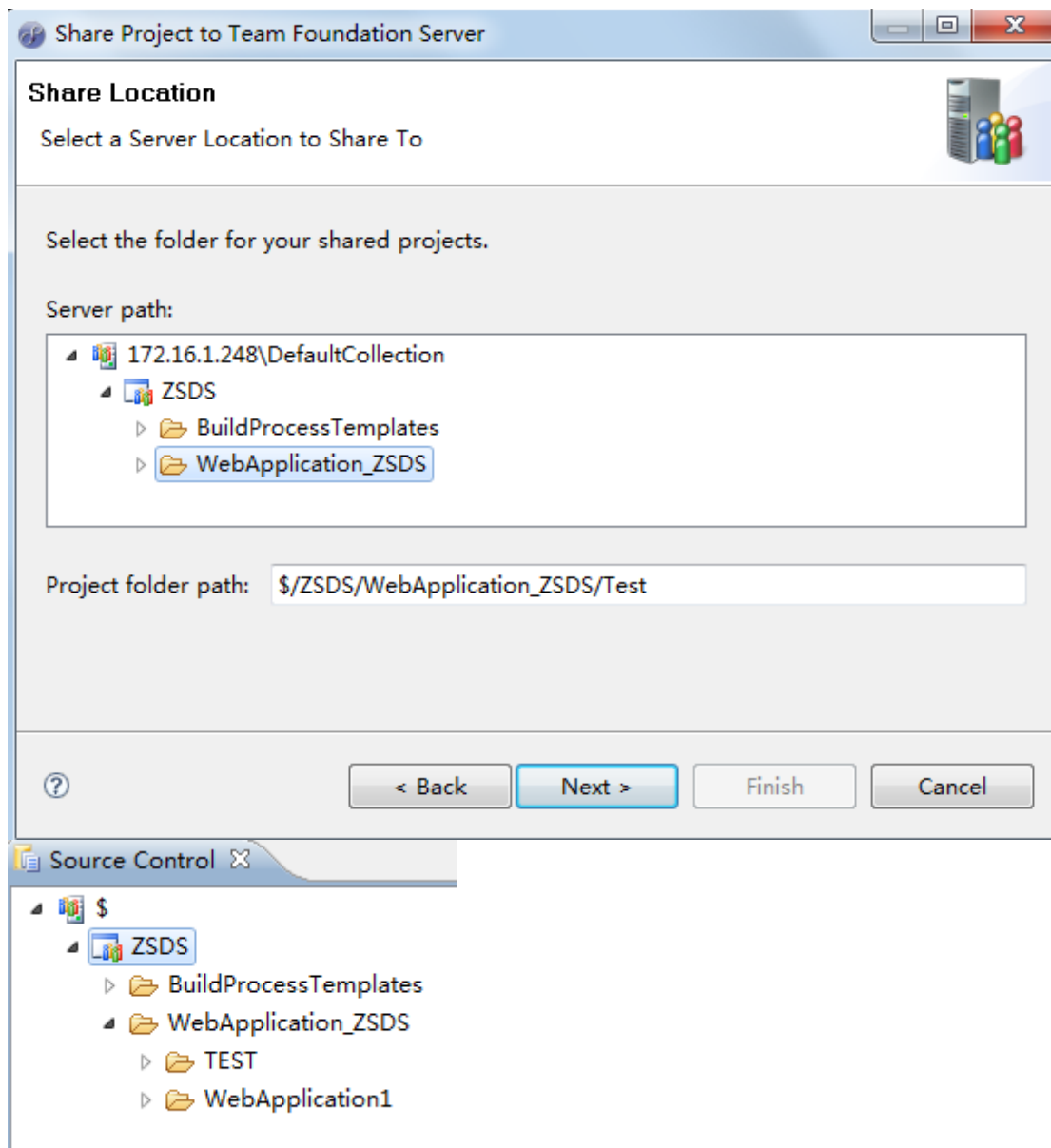
如下图：



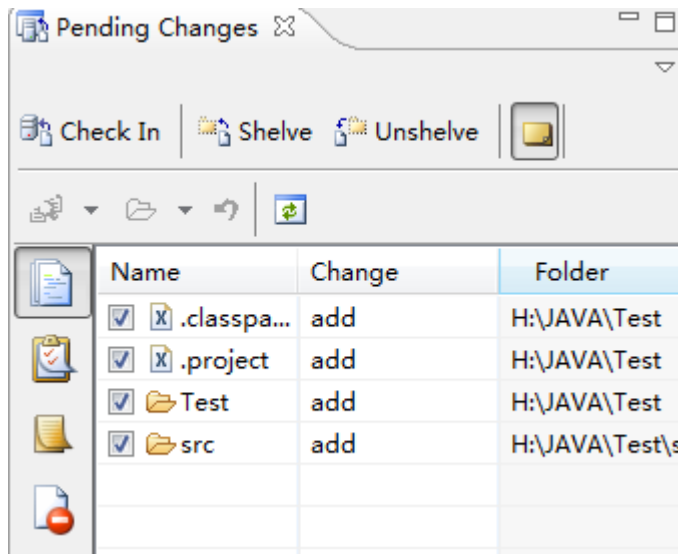
选中 TFS 中的团队项目，点击 Next，如下图：



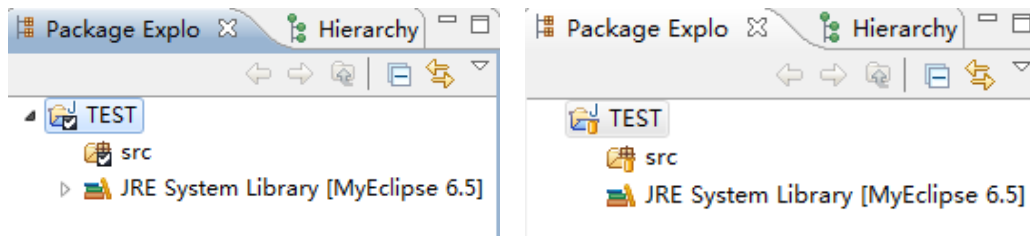
在 ZSDS 团队项目中选择要将此项目添加到哪个目录下，如选择 “WebApplication\_ZSDS” ,点击 Next，点击 Finish。如下图：



此时，会在 WebApplication\_ZSDS 的 Source Control 多了 Test 项目，将整个项目迁入后即可把本地项目提交至服务器。在 Pending Changes 中，点击 Check In，进行迁入。



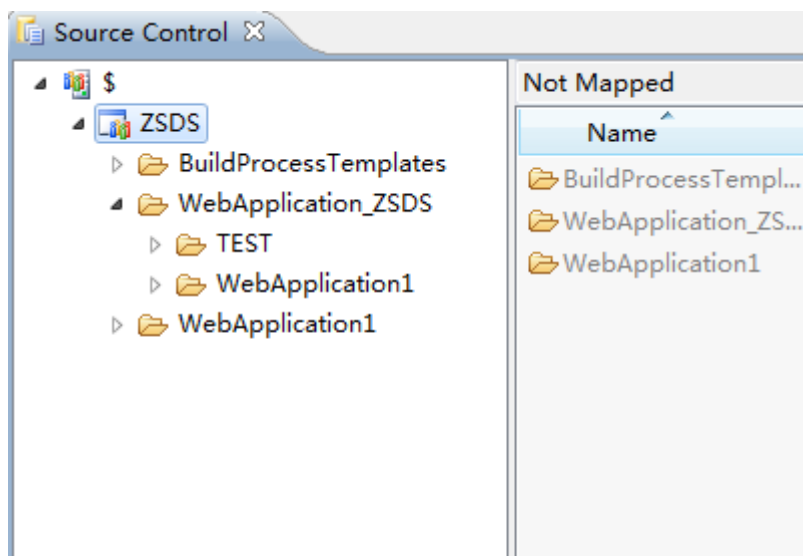
提交后原来“√”的标记变为“锁”的标记，如图所示。



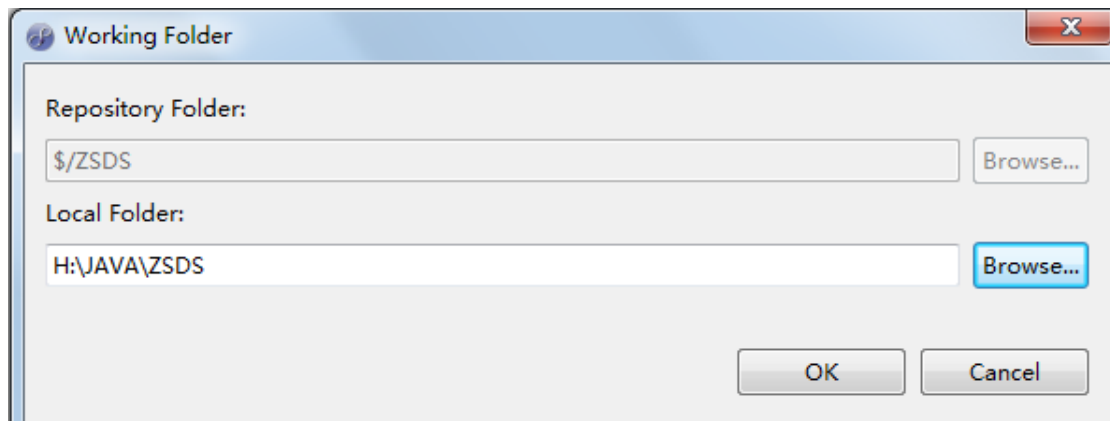
### 3.3 配置本地工作空间

虽然 TFS 提供了很好的团队合作模式，但是为了工作安全和方便，日常的开发还是需要在本本地工作空间进行。下面讲述如何根据服务器上的项目设定本次工作区：Team Explorer 中双击 ZSDS 的

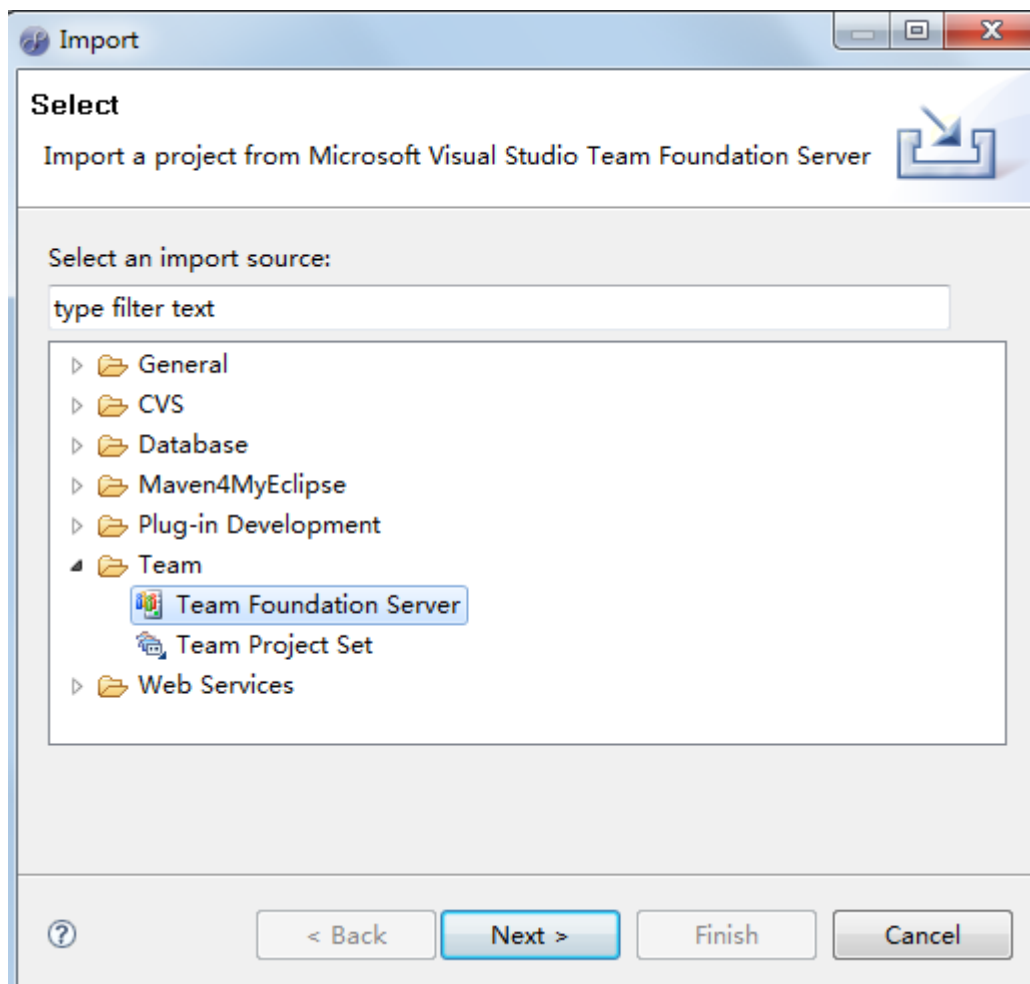
“Source Control” 会出现以下界面：



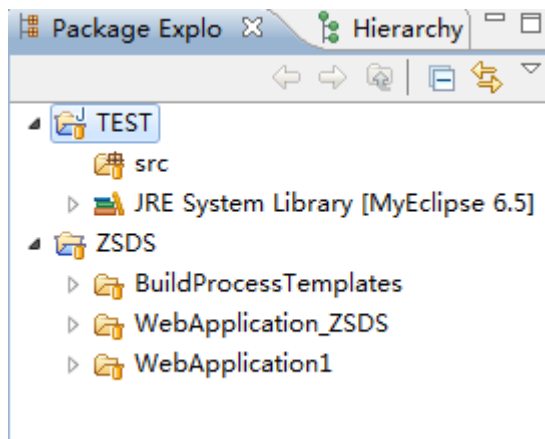
在 ZSDS 项目右键快捷键，选择 Set Working Folder，在弹出的窗口中，选择本地目录，如下图：



在 Package Explorer 空白区，右键选择 Import，在弹出的对话框中，选择 Team Foundation Server，如下图：



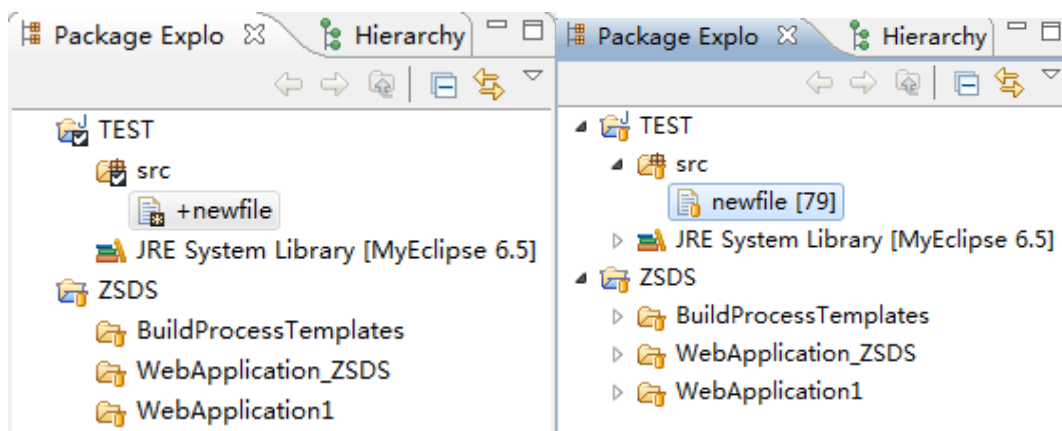
选中 ZSDS 项目，点击 Next，点击 Finish 关闭。此时在 Package Explorer 中出现 ZSDS 项目：



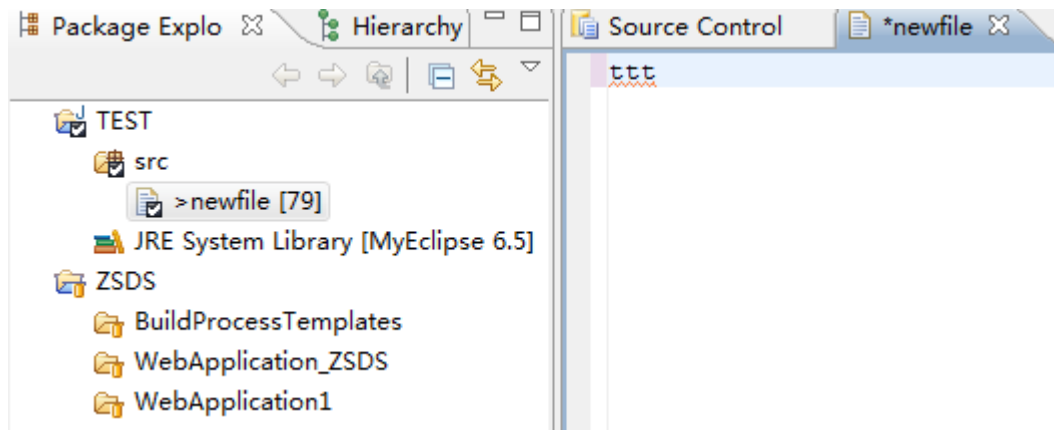
## 4. TFS 源代码管理

上一章我们将到了怎样将本地新建项目上传至 TFS，下面我们看看，如果向已经加入源代码管理的项目中，添加一个文件，会是什么情况：

- 首先，项目会被自动 Check-Out（签出），在项目前面有一个的打钩图标(下面左图)
- 其次，新增的文件前面有一个\*，表示这是一个挂起修改的文件（下面左图），签入后如下面右图所示



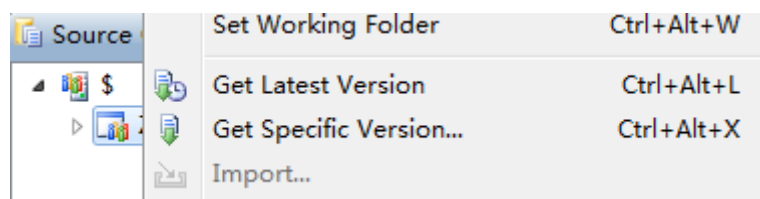
先把 newfile 文件签入，然后我们试着双击打开 newfile 文件，输入一行内容，我们发现编辑文件后，文件会被自动签出（Check-Out），由于文件被 Check-Out，所以项目的状态也更新成 Check-Out 了：



## 4.1 如何与服务器同步

在团队协同工作的环境中，许多人都在修改同一个项目中的代码，你需要把项目中最新的修改下载到本地。如下的操作，可以让你的工作空间和服务器上的最新版本同步：

- 在 Source Control 中，选择 Team Project 或 Team Project 下的某个 Project，右键点击，选择“Get Latest Version”



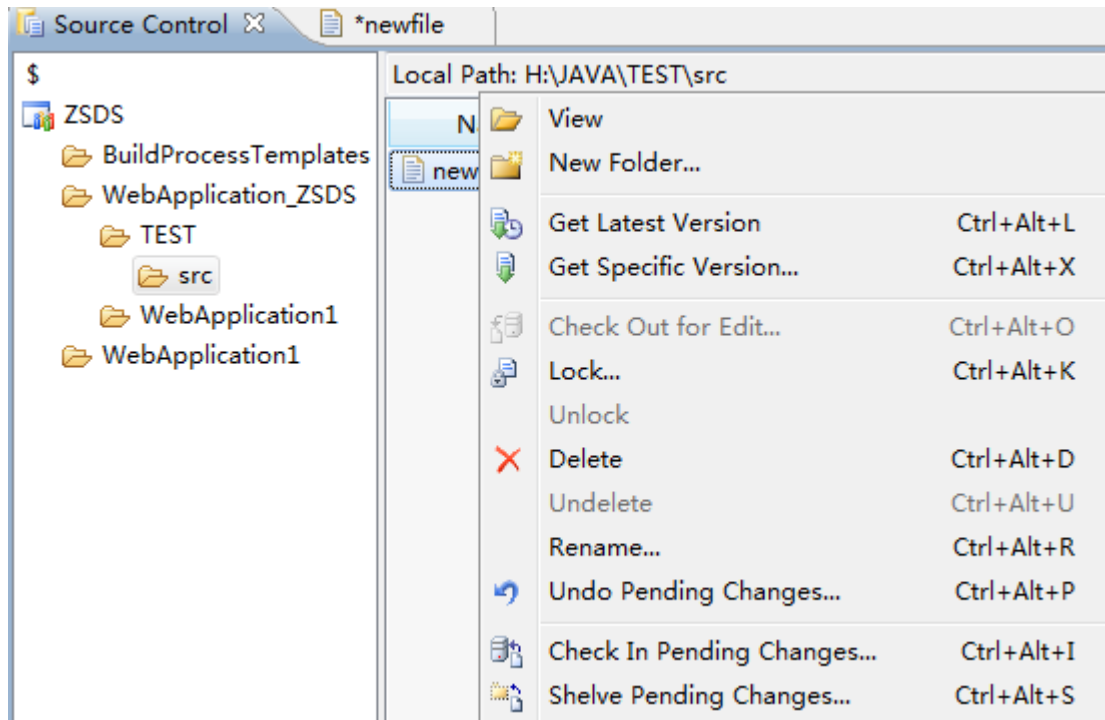
在签入你自己的代码前，最好把服务器上最新的版本同步下来，这个过程需要解决可能的版本冲突问题，然后你要构建项目，保证没有问题后，再签入代码。这是为了尽量避免你签入的代码，导致在服务器和其他开发人员的机器上出现构建失败。

## 4.2 如何做 Check-In

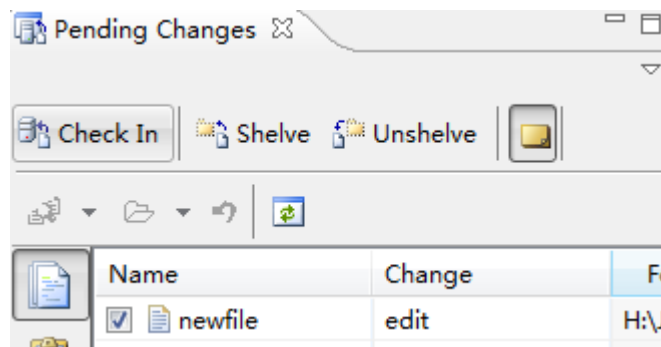
以下三种方法都可以做签入（Check-In）

- 在 Package Explorer 中，右键点击修改后的文件，然后选择 Team→Check In Pending Changes
- 在 Source Control 中，右键点击修改后的文件，然后选择“Check In Pending Changes”



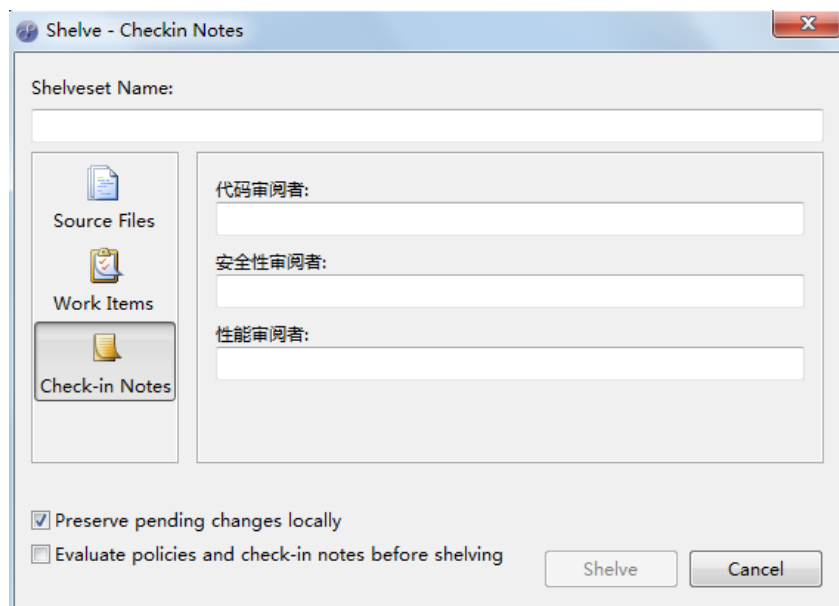


- 在 Pending Changes 中，选择要签入的文件后，点击 Check In 按钮：

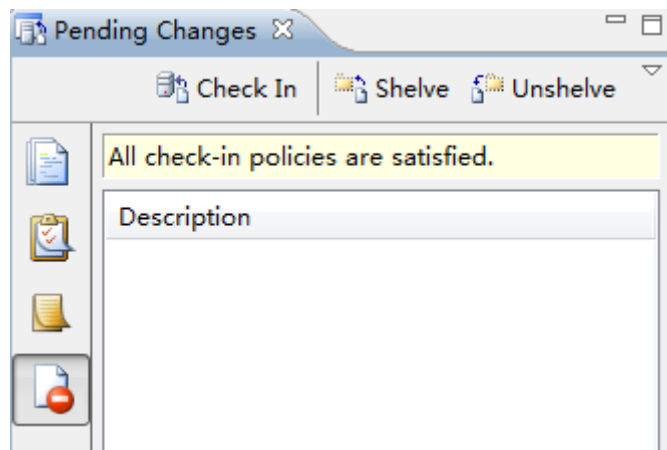


这种方法可以让我们做快速的签入，默认情况下，所有被修改的文件，都会自动列出，根据需要确定要签入的文件后，可以填写签入的注释，对签入的修改关联对应的工作项。

填写代码的相关审阅者：

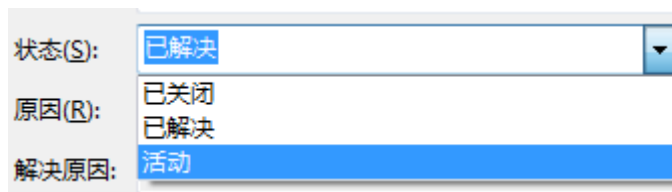


查看签入是否已经满足预定的策略：



当用户选择签入后，所有选中的文件，签入说明，以及与此次签入相关联的工作项，都将被存储到数据库中，作为一个新的更改集（**Changeset**）。一个更改集是文件版本、相关工作项、以及源代码管理元数据（**Metadata**）组成的一个单独的实体。

如果工作项有相关的流程处理规则，这些规则会修改工作项的状态。例如，一个团队可能会定义如下规则：如果当你签入时管理了工作项，并选择“签入操作”为“解决”，则工作项会从“活动的”变为“解决”：



## 4.3 什么是上架与下架

上架（**Shelve**）和下架（**UnShelve**）命令听起来不容易理解。我们不妨想象一下，你正在办公桌上伏案画图，假设是用工笔画红楼梦群芳夜宴图，大大小小的美女草图铺满了桌面，这是你接到命令要做另外一件事：泼墨画，而你只有一个办公桌，万一泼墨到美女们怎么办？

于是你就把目前的所有图纸卷起来，放到书架上。这就相当于上架。

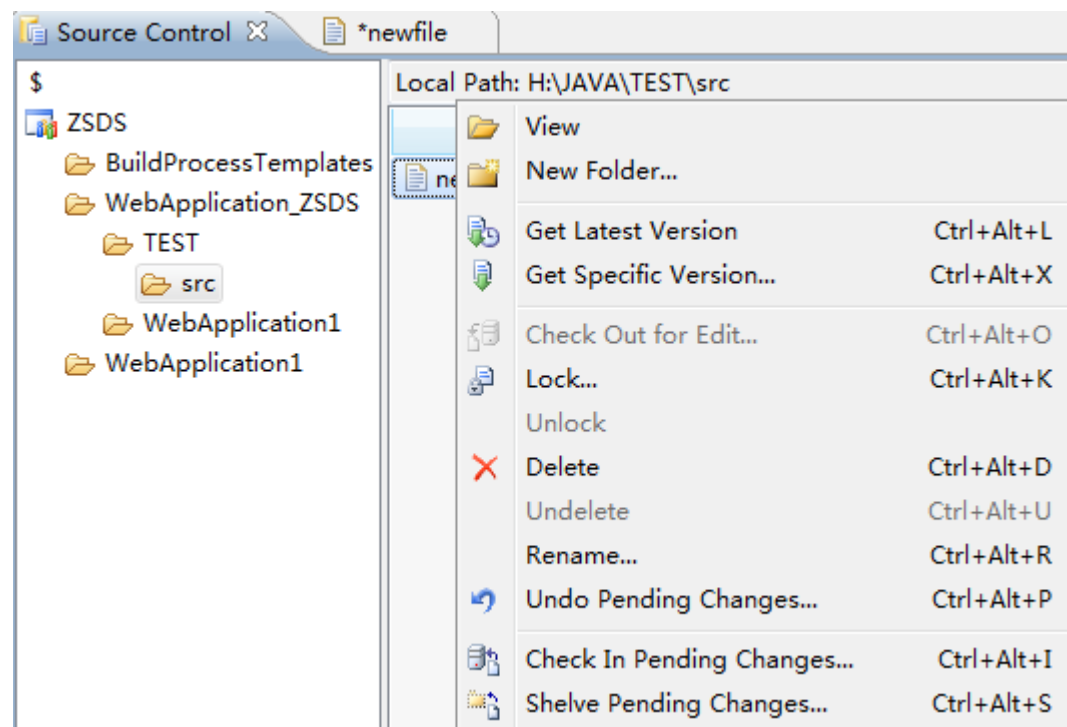
接下来你开始泼墨画的工作。尽情挥洒之后，清理桌面，从书架上把刚才收起的图纸都拿下来，铺开，继续你的群芳夜宴图。这就相当于下架。

这两个命令主要用于：

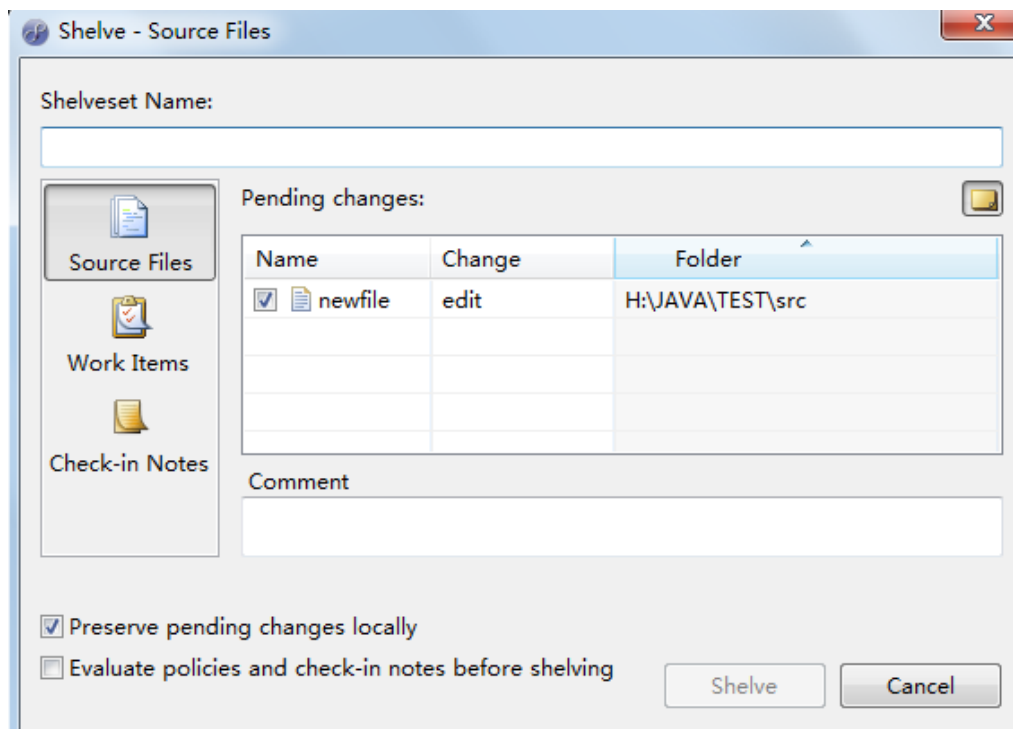
- 保存目前的工作，切换到另一个任务
- 其他人开始代码复审（你将修改上架了，别人可以在自己的环境中，把你上架的内容下架，然后做复审）
- 集成别人的修改（两人的修改互相依赖，这是可以通过上架/下架命令把修改集成起来）

这位多人协作提供一种便利：不需要每次 **Check-In**，然后别人 **Check-Out**，而是多次上架下架后，最后来一次 **Check-In** 即可。

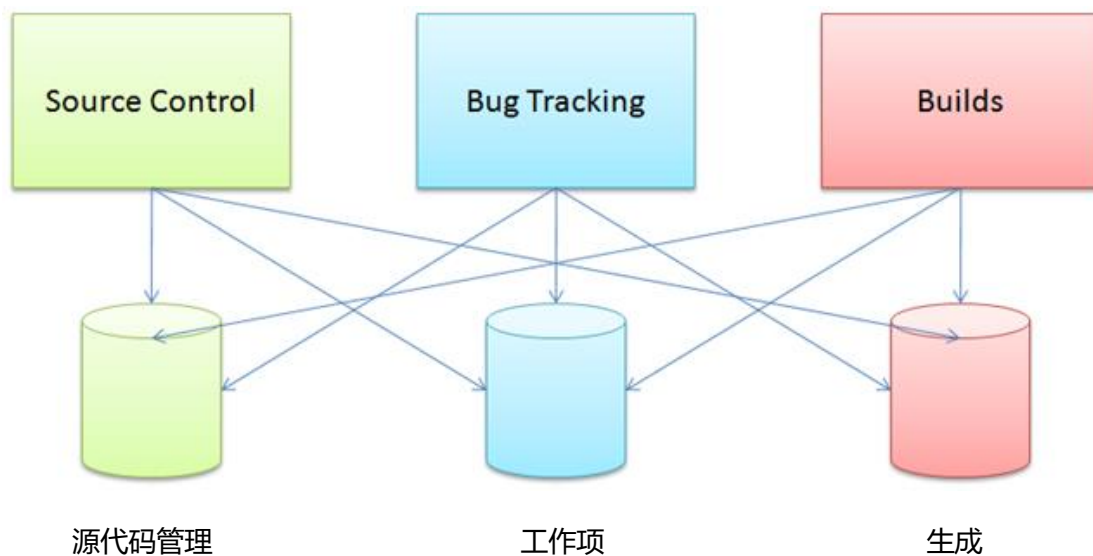
上架的操作十分类似于签入操作，在源代码管理器中，右键点击要上架的文件，选择 **Shelve Pending Changes** 或者在 **Pending Changes** 窗口中点击 **Shelve** 按钮：



所有的参数都与签入操作类似，不同的是，修改不会做冲突检查，也不会影响源代码（即没有真正被 **Check-In**，这可以在上架操作结束后，验证文件前面还有红色打钩标志来确认），而是暂时存储在 **TFS** 服务器上，以便其他人做下架处理：



## 5. 项目中各角色对在 TFS 上的应用

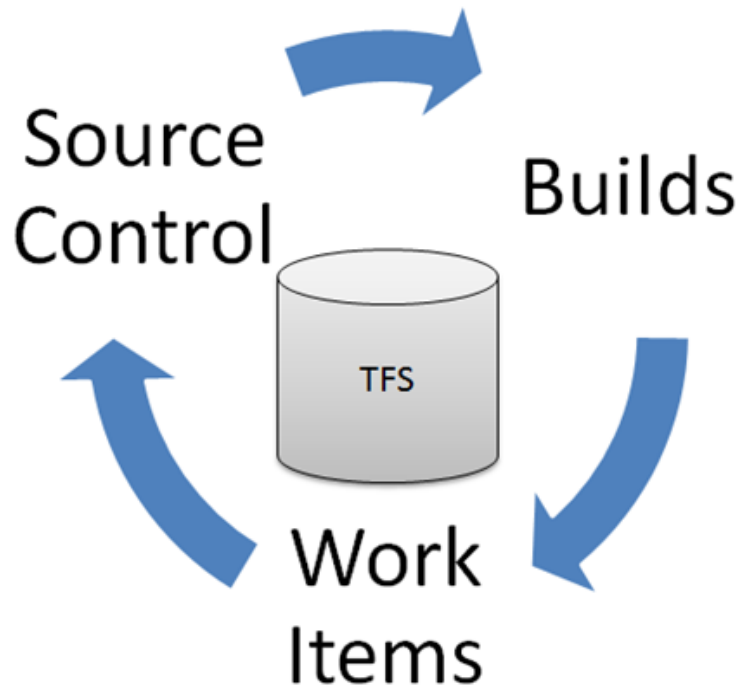


TFS 有对复杂环境的良好支持，比如，报表、SharePoint 的整合，支持跨域，分布式数据库等。

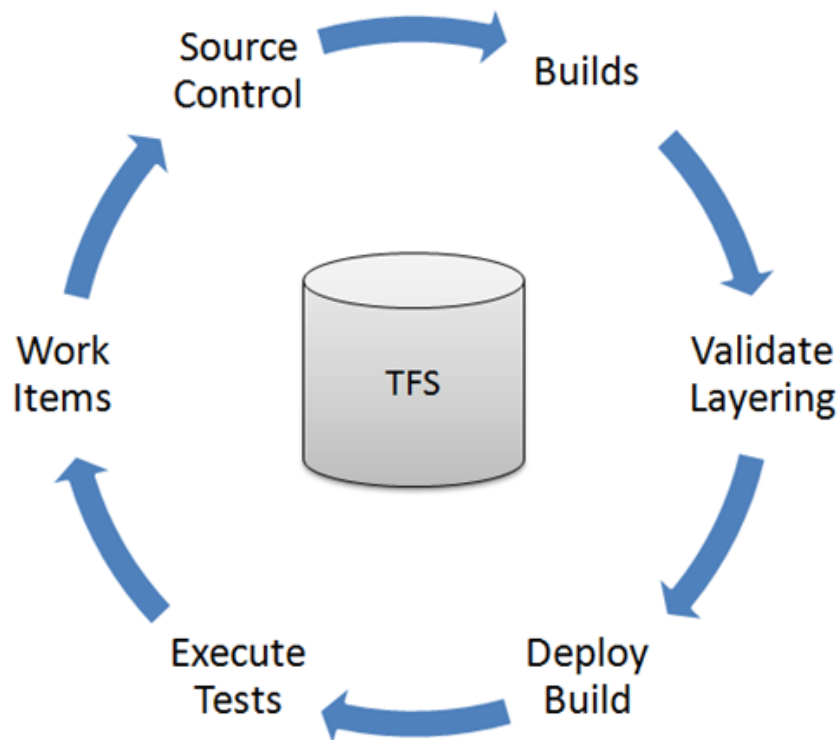
在上图中，每个系统都有独立的存储空间，资源标识集，命令和工具集。要让整个系统工作

起来，就像把一组自定义立体组件联接在一起：可以实现，但工作量巨大，而且可能在一些地方出现纰漏。

他们之间工作流流程是



这是一种很普遍的工作场景，在一天之中，编写代码，编译一个可工作的软件提交测试，测试发现的 bug，修复他们，这样不断的重复，整个过程的支持需要有一个统一的存储库进行支持，然后所有项目可以进行关联。使用 TFS 的 Basic 安装就可以完成这样的功能，比纯粹的像 VSS, SVN 这样的版本控制控制软件有很大的改善，这也能够回答有的网友问 TFS 有什么优势的疑问了。在 TFS 的完整版本中还包括自动化测试，虚拟实验室和架构验证的功能。完整版本的流程图是这样子的：



下面我们来看看 TFS 的 Basic 安装的三大主要功能 Work Items, Builds 和 Source Control, 是怎样融合在我们的开发过程中:

可以直接在 Visual Studio 里用团队资源管理器或者通过网页前端和 Test and Lab Management 工具来创建工程项目。打开团队资源管理器, 并展开工作项下的 Team Queries 项来浏览你的工程项目。也可以通过双击任意查询选项 (例如 Active Bugs) 来浏览任意你所能看到的项目。

## 5.1 项目经理及开发经理

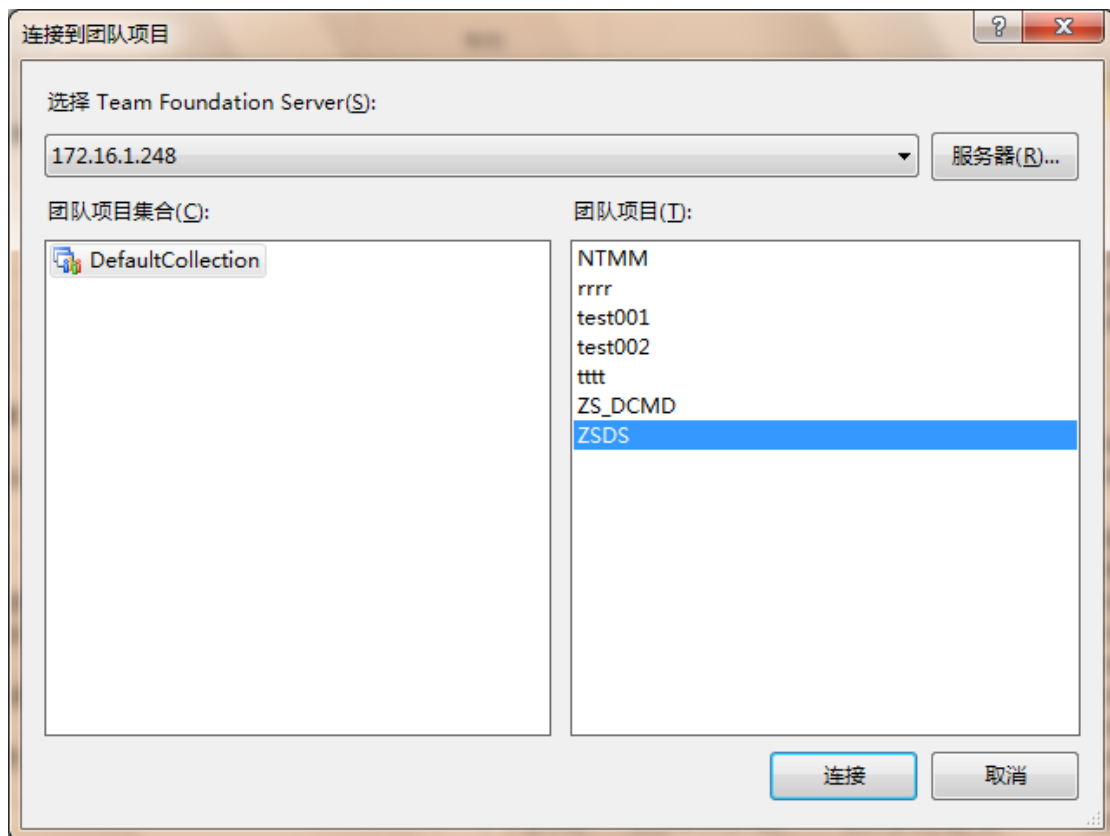
### 5.1.1 使用 Project 分配任务

新建一个 Project 文件, 此时打开的文件为一个空的 Project 文件, 可以在此文件中进行计划规划。

选择 Project 的团队选项卡, 点击选择团队项目, 如图:



然后再弹出的对话框中选择所需要发布的项目

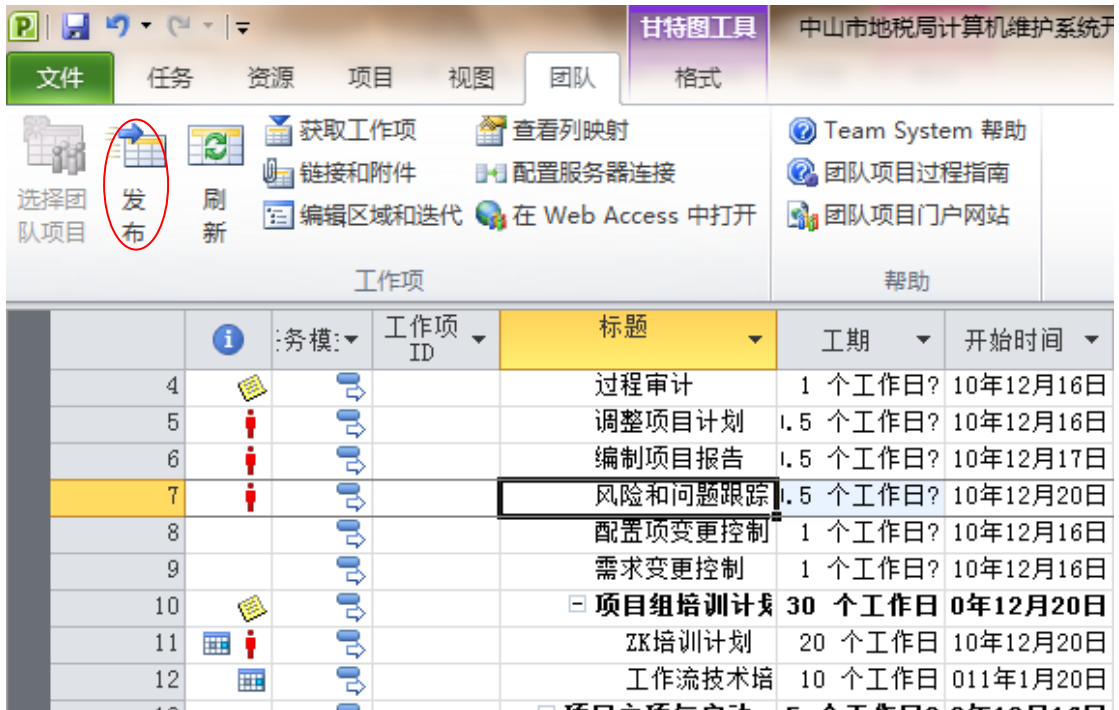


确认后 Project 会默认加入 TFS 相关的标题字段以及功能选项。例如：列“工作项 ID”，“工作项类型”



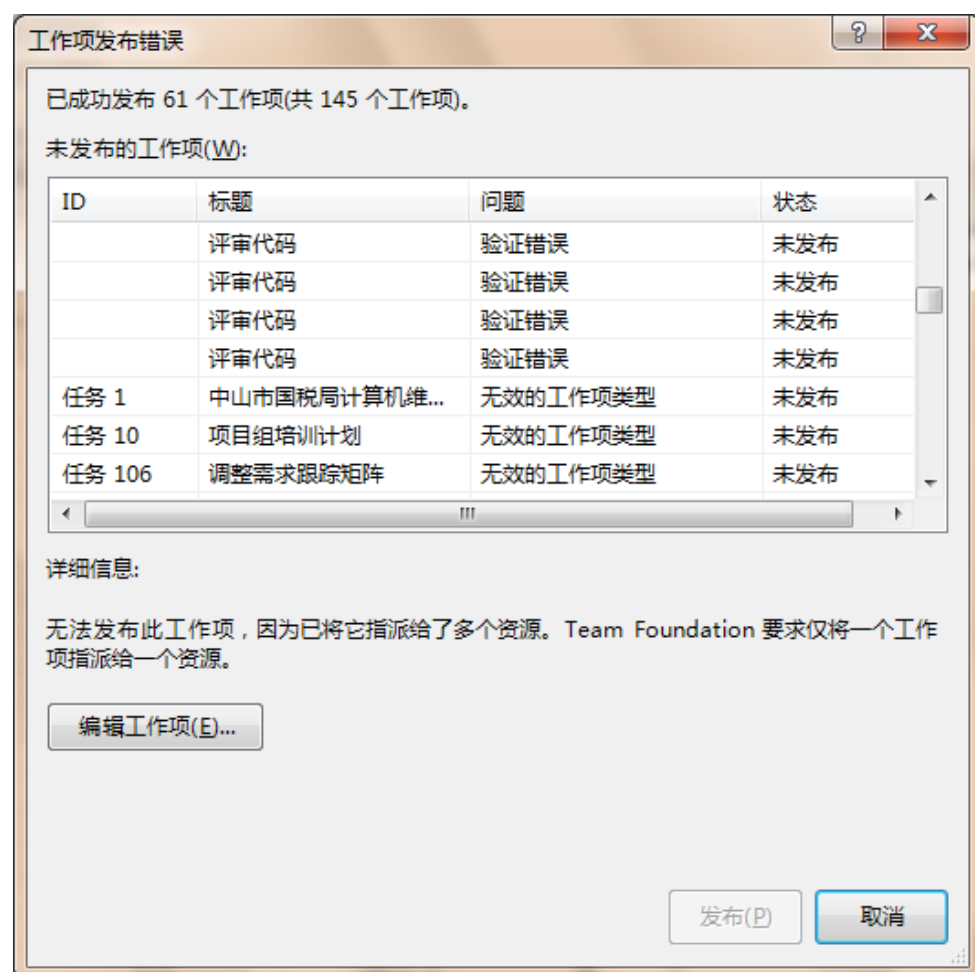


解。同步操作十分简单，只需要点击团队选项卡下的发布按钮即可。



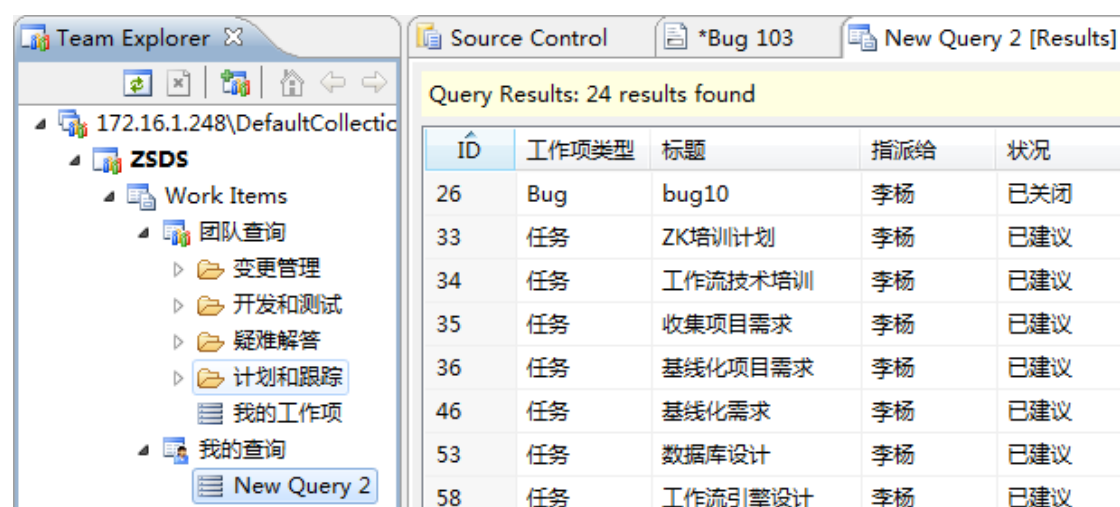
注意：若填写的项目计划在同步中出现异常或者不符合要求，则会自动弹出错误的项以及建

议修改的说明。如图：



发布完成后，可以通知各小组组长对计划进行更详尽的划分。

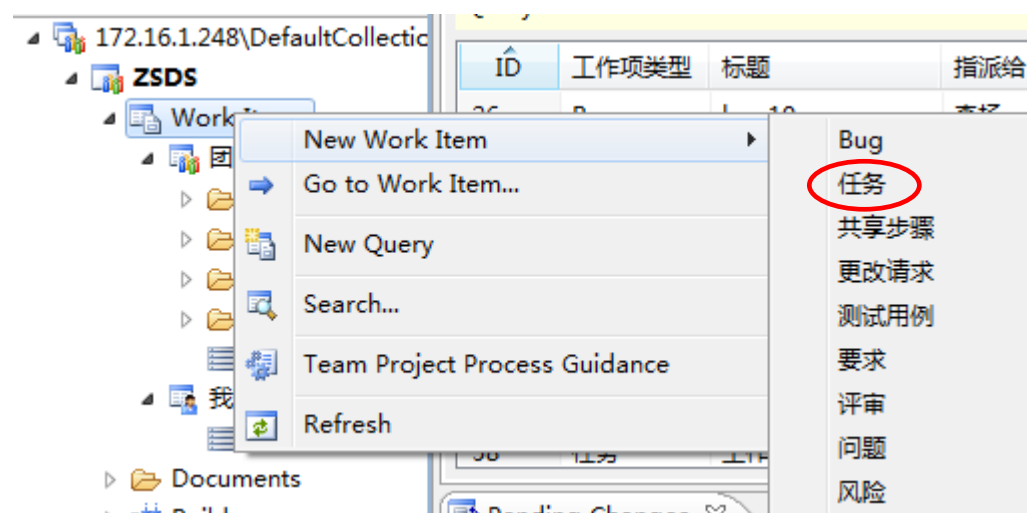
任务分派完毕后项目组成员可以在“Team Explorer”中通过定义“我的查询”来查看已分配任务，操作如图：



## 5.1.2 单独分配任务

除了用 Project 分配任务外，项目经理或开发经理还可以通过“新建工作项”来分配任务。

例如，开发经理给开发人员分配开发任务，右键点击“Work Items”——>“New Work Item”——>“任务”来实现。如下图所示：



进入新建任务界面后按要求填入任务内容点击“保存工作项”即可。此时系统会自动分派一个任务 ID，并且在历史记录中记录修改信息。



任务分派完后开发人员就会在“我的工作项”中看到自己被分派到的任务。

Source Control 我的工作项 [Results] X					
Query Results: 2 results found					
ID	堆..	优先级别	工作项类型	状况	标题
103		2	Bug	已解决	99999
104		2	任务	已建议	TTT

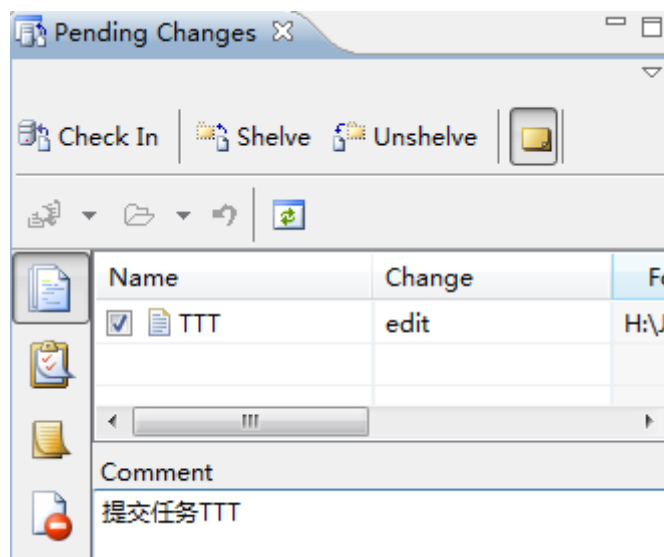
开发人员按照任务进行代码开发工作。并及时改变任务状态，以便任务分派人员跟踪。例如：  
将状态置为“活动”，将原因置为“已接受”并保存。此时历史记录里就会记录状态转换步骤。最终状态就会变为“已解决”最终等待评审或测试。

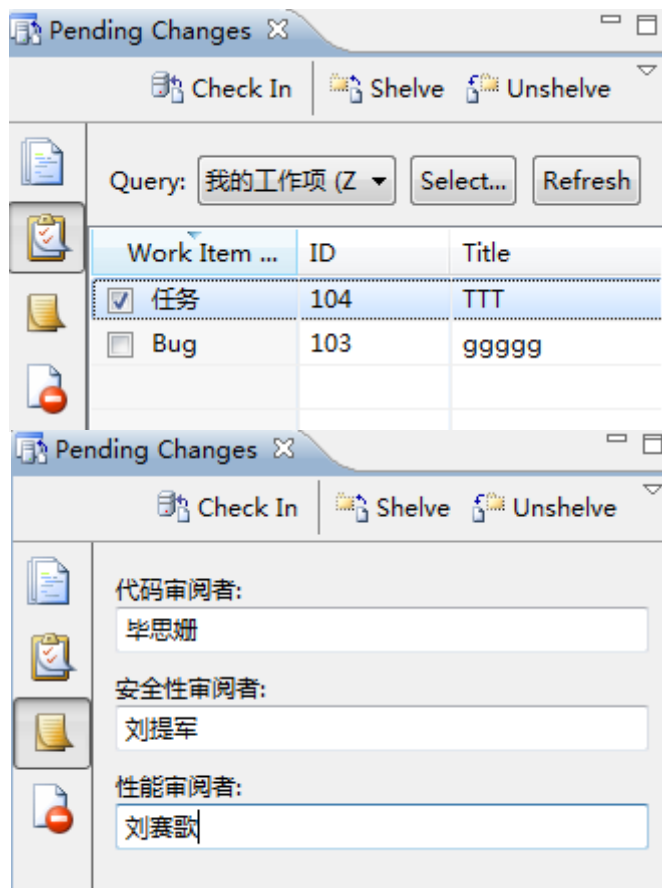
Source Control		我的工作项 [Results]		任务 104 X																									
任务 104 : TTT																													
标题(T): TTT			任务类型(Y): 计划的																										
指派给(G): 毕思嫻 状态(S): 活动 原因(R): 已接受			分类 区域(A): ZSDS 迭代(R): ZSDS 专业(D):																										
计划 优先级别(P): 2    会审(I): 挂起    被阻止(B): 否			投入(小时) 初始估计(E):    剩余工作(W):    已完成工作(P):																										
详细信息    实现    其他    所有链接    附件																													
说明(D): TTT			历史记录(H): Type your comment here.																										
<table border="1"> <thead> <tr> <th colspan="3">04/08/11 11:55:23 AM Edited (已建议 to 活动) by 毕思嫻</th> </tr> <tr> <th>Field</th> <th>Old Value</th> <th>New Value</th> </tr> </thead> <tbody> <tr> <td>激活日期</td> <td></td> <td>04/08/11 11:55:23 AM</td> </tr> <tr> <td>修订</td> <td>1</td> <td>2</td> </tr> <tr> <td>原因</td> <td>新建</td> <td>已接受</td> </tr> <tr> <td>状况</td> <td>已建议</td> <td>活动</td> </tr> <tr> <td>状态更改日期</td> <td>04/08/11 11:52:38 AM</td> <td>04/08/11 11:55:23 AM</td> </tr> <tr> <td>激活者</td> <td></td> <td>毕思嫻</td> </tr> </tbody> </table>						04/08/11 11:55:23 AM Edited (已建议 to 活动) by 毕思嫻			Field	Old Value	New Value	激活日期		04/08/11 11:55:23 AM	修订	1	2	原因	新建	已接受	状况	已建议	活动	状态更改日期	04/08/11 11:52:38 AM	04/08/11 11:55:23 AM	激活者		毕思嫻
04/08/11 11:55:23 AM Edited (已建议 to 活动) by 毕思嫻																													
Field	Old Value	New Value																											
激活日期		04/08/11 11:55:23 AM																											
修订	1	2																											
原因	新建	已接受																											
状况	已建议	活动																											
状态更改日期	04/08/11 11:52:38 AM	04/08/11 11:55:23 AM																											
激活者		毕思嫻																											

如果从项目网站上看的话会得到更多的信息。如图所示



当任务完成意味着此开发任务已经结束，此时需要将开发的代码连同这个开发任务一同签入，如图所示：

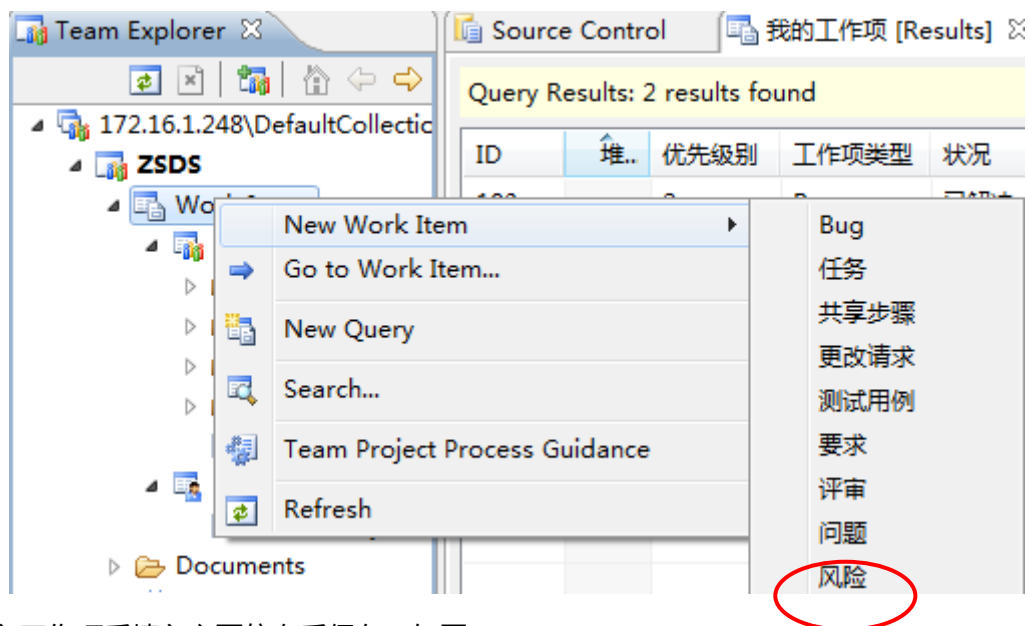




### 5.1.3 风险跟踪

风险工作项可以通过右键点击“Team Explorer”中“Work Items”——>“New Work Item”

——>“风险”来实现。如图：



进入工作项后填入必要信息后保存，如图：

Source Control 我的工作项 [Results] \*New 风险 3

New 风险: 需求风险

标题(T): 需求风险 概率(P): 10

状态

指派给(G): 李杨

状态(S): 已建议

原因(R): 新建

分类

区域(A): ZSDS

迭代(I): ZSDS

计划

优先级(P): 2 严重级别(S): 3 - 中 被阻止(B): 否 初始估计(E):

详细信息 缓解操作 应变计划 所有链接 附件

说明(C):

1

2

3

历史记录(H):

Type your comment here.

风险工作项的状态可以按如下状态流转:



## 5.2 测试人员

测试人员可以通过 Test and Lab Manager 来访问 TFS (没有必要安装 VS), 在浏览器中键

入如下地址: mtm://172.16.1.248:8080/tfs/ 会出现如下界面:

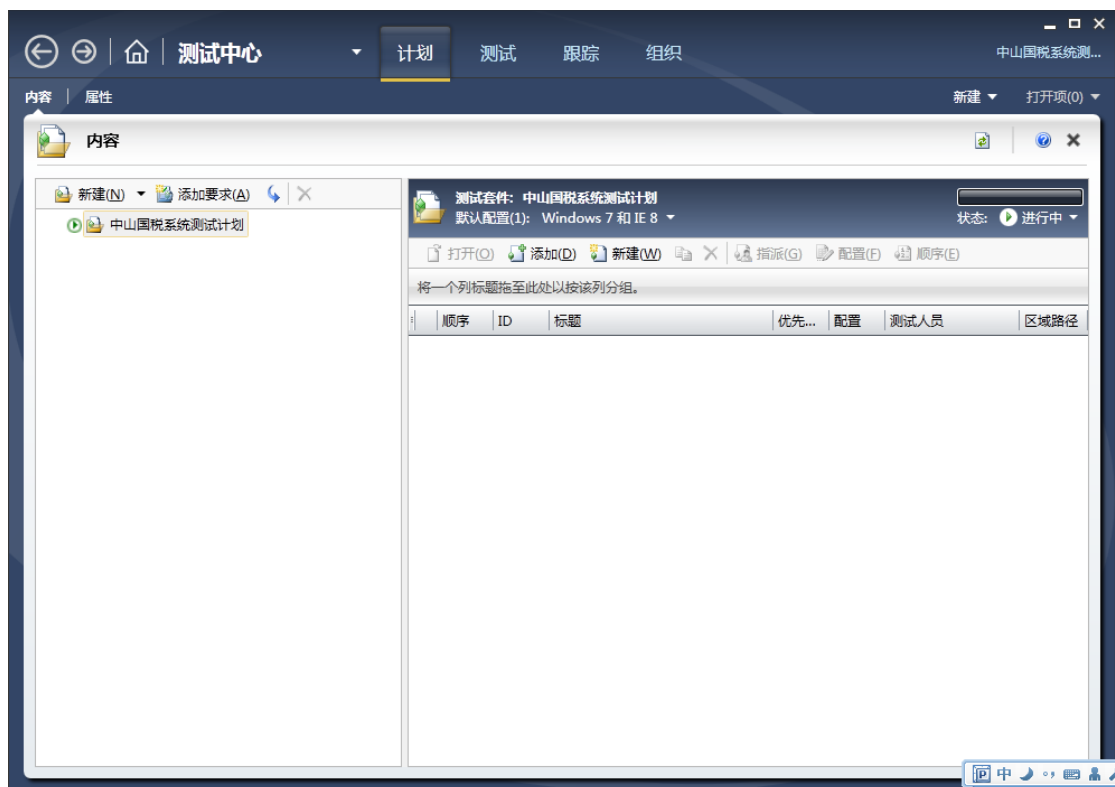


双击需要测试的项目，此时需要为这个项目添加测试计划，如图：

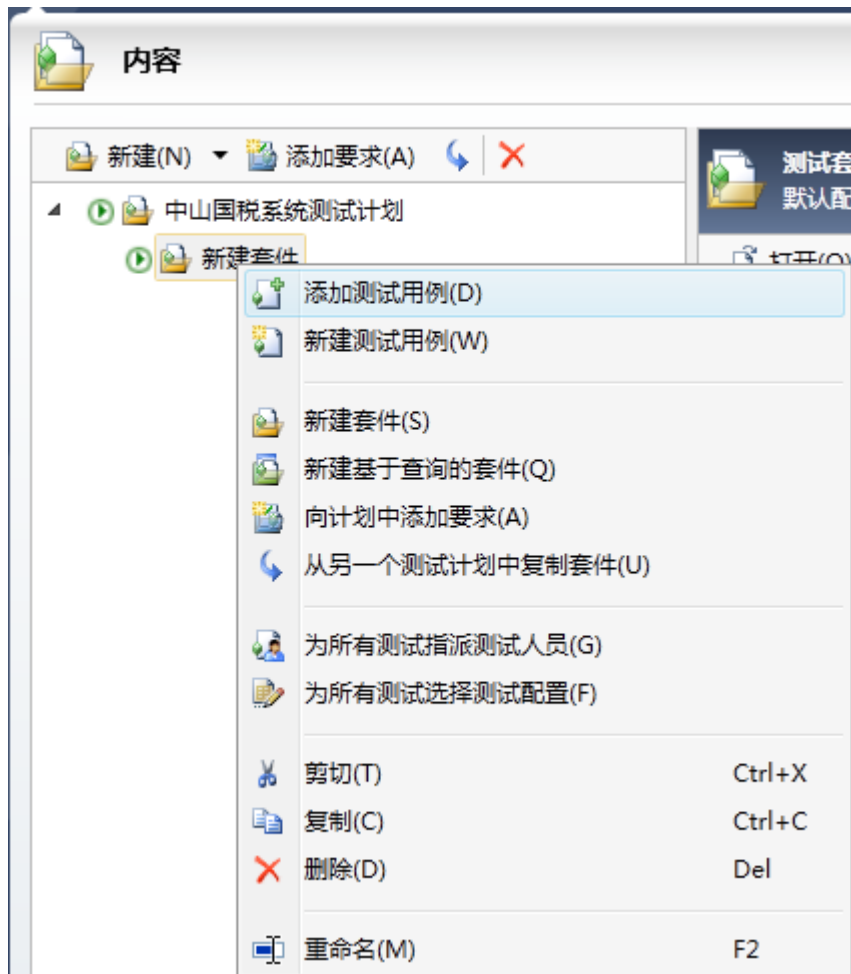




点击“添加”后双击刚创建的计划，可进入如下界面：



添加套件和测试用例，将计划补充完整。



测试用例界面如下，测试人员需要在里面输入详细测试步骤等信息，测试用例中也可插入共享步骤，或链接一些要求：

测试用例 105: mm

标题(T): mm

状态

指派给(G): 毕思嫻

状态(S): 设计

优先级别(P): 2

自动化状态(U): 未自动运行

分类

区域(A): ZSDS

迭代(R): ZSDS

步骤 摘要 经过测试的要求 所有链接 附件 关联的自动化

插入步骤 插入共享步骤 插入参数

操作	预期结果
1. c	kkk
单击此处添加步骤	

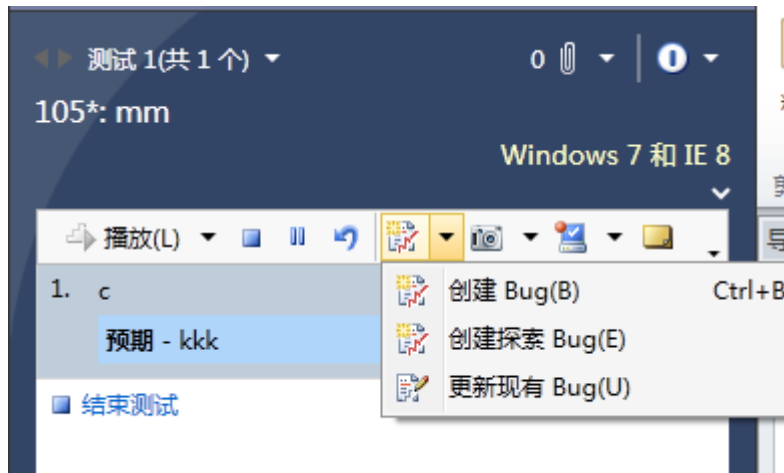
参数值:

删除迭代 重命名参数 删除参数

当做完计划后就可以试着运行了，运行测试用例时会出现如下界面：



点击“开始测试”，在测试过程中可以将用例中的步骤做标记，以表明测试是否通过。例如测试没有通过，这时测试人员可以在注释中写明结果。并提一个 Bug 单。



此 Bug 单中的很多信息系统已自动记录。如“重现步骤”，“系统信息”，“测试用例”和“所有链接”如下图所示：

新建 Bug\*

标题(T): <必选>

根源(R): 未知

状态

指派给(G): 毕思姗

状态(S): 已建议

原因(R): 新建

解决原因:

分类

区域(A): ZSDS

迭代(R): ZSDS

计划

优先级(P): 2

严重级别(S): 3 - 中

会审(T): 挂起

被阻止(B): 否

重现步骤

详细信息

系统信息

修复

测试用例

其他

所有链接

附件

重现步骤(R):

微软雅黑

2

B I U

2011/4/8 14:11:09

Bug 于 "mm" 记录

步 骤 号	结果	标题
1	未通过	<div>c</div> <div>预期的结果: kkk</div> <div>注释: TTT</div>

测试配置: Windows 7 和 IE 8

输入数据/配置 输出/结果

新建 Bug\*

标题(T): TTT

根源(R): 未知

状态

指派给(G): 毕思姗

状态(S): 已建议

原因(R): 新建

解决原因:

分类

区域(A): ZSDS

迭代(R): ZSDS

计划

优先级(P): 2

严重级别(S): 3 - 中

会审(T): 挂起

被阻止(B): 否

重现步骤

详细信息

系统信息

修复

测试用例

其他

所有链接

附件

环境

发现环境(F):

发现途径(H):

系统信息(T):

微软雅黑 2 B I U

GZCSS-HP 中的系统信息

用户名	gzcass
计算机名称	GZCSS-HP
用户域名	gzcass-HP
操作系统名称	Microsoft Windows 7 家庭普通版
操作系统版本	Microsoft Windows NT 6.1.7600.0

新建 Bug\*

标题(T): TTT 根源(R): 未知

状态

指派给(G): 毕思嫻

状态(S): 已建议

原因(R): 新建

解决原因:

分类

区域(A): ZSDS

迭代(R): ZSDS

计划

优先级(P): 2 严重级别(S): 3 - 中 会审(T): 挂起 被阻止(B): 否

重现步骤 详细信息 系统信息 修复 测试用例 其他 所有链接 附件

链接到(L)

ID	工作项类型	标题
测试方 (1 items)		
105	测试用例	mm
结果附件 (3 items)		
		TC105_ActionLog.3.html
		TC105_ActionLog.3.txt
		SystemInformation.3.xml

测试结束后可以查看测试结果，如图：

顺...	ID	标题	测试人员	配置	优先...	自动
未通过 (2)						
1	96		李杨	Windows 7 和 IE 8		
2	105	mm	毕思嫻	Windows 7 和 IE 8 2	否	

测试结果信息如下：

mm 的测试结果 - 2011/4/8 14:05:13
 保存并关闭

---

^ 摘要(未通过)

---

运行类型: 手动      运行者: 毕思姗      相关项: 查看

故障类型: 无      配置: Windows 7 和 L...      注释:

解决方法: 无

---

^ 详细信息

---

^ 分析

---

结果: 未通过      开始日期: 2011/4/8 14:0...      测试持续时间: 00:04:35

注释:

---

^ 测试步骤详细信息

---

1 c

Expected - kkk

TTT

---

v 附件(8)

---

v 链接(1)

---

v 结果历史记录(2 结果)

此时开发人员已经收到一个测试人员提的 Bug。

Source Control   
 我的工作项 [Results]   
 Bug 107

---

Bug 107 : TTT

标题(T): TTT      根源(R): 编码错误

状态

指派给(G): 毕思姗

状态(S): 已建议

原因(R): 新建

解决原因:

分类

区域(A): ZSDS

迭代(R): ZSDS

计划

优先级(P): 2      严重级别(S): 3 - 中      会审(I): 挂起      被阻止(B):

重现步骤    详细信息    系统信息    修复    测试用例    其他    所有链接    附件

重现步骤(R):

2011/4/8 14:11:09      Bug 于 “mm”记录

步骤号	结果	标题
1	失败	

缺陷可按如下状态流转：

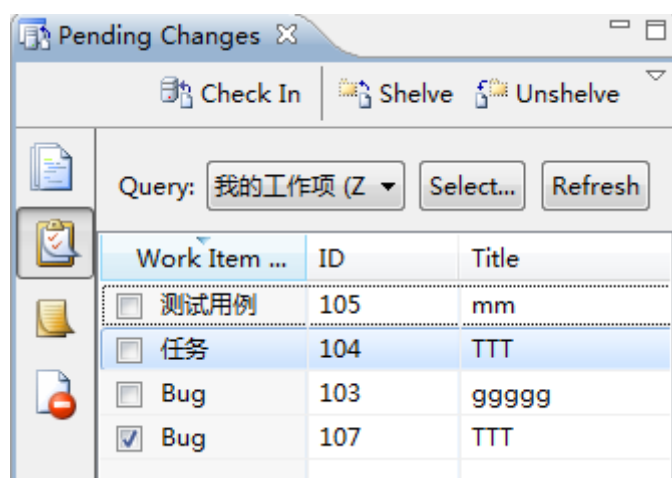
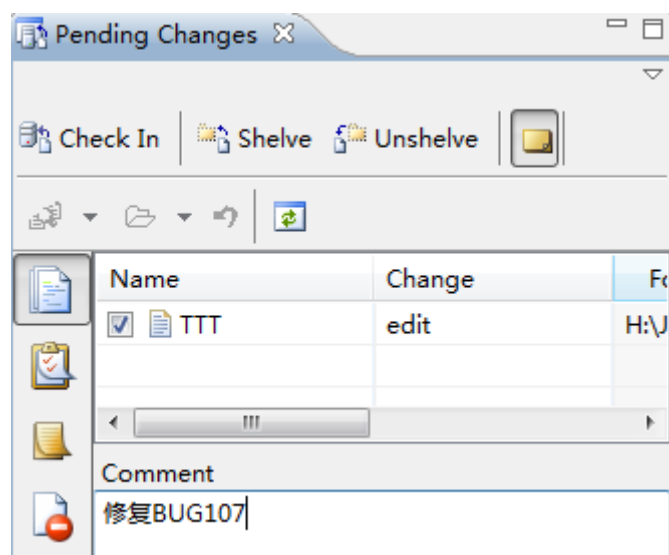




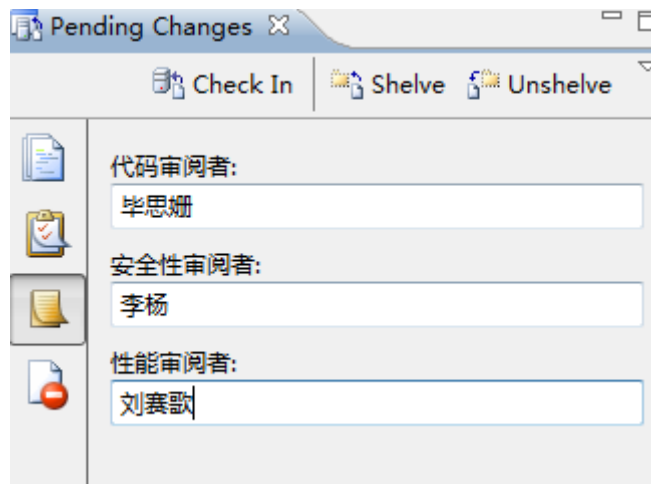
## 5.3 开发人员

开发人员的开发任务一般有两个条件触发，一个是项目经理或开发经理分派的“任务”，另一个是测试人员分派的“Bug”。这里需要强调的是

当任务完成意味着此开发任务或 Bug 修复已经结束，此时需要将开发的代码连同这个开发任务或 Bug 一同签入，如图所示：

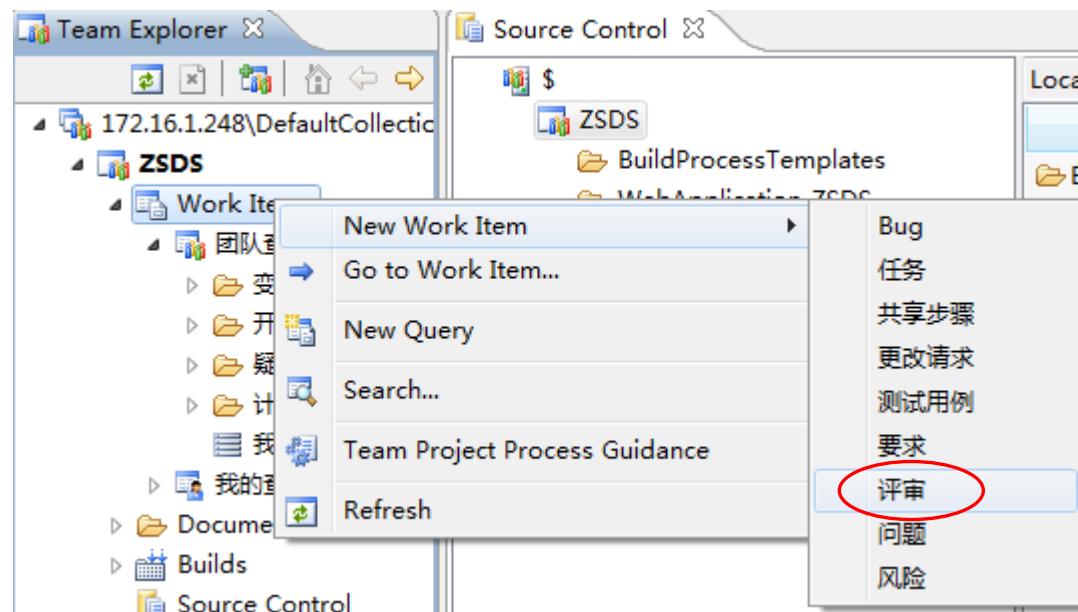


代码签入时有可能还要求有代码审阅者等角色的介入。



## 5.4 QA 人员

QA 人员用的最多的是分派“评审”工作项。建立过程如下：



例如新建一个项目计划的评审，按实际情况填好“详细信息”，“会议记录”，“与会者”等内容后保存。

Source Control \*New 评审 4

New 评审 : 项目计划评审

标题(T): 项目计划评审 会议类型(Y): 会议

状态

指派给(G): 毕思姗

调用方(B): 毕思姗

状态(S): 活动

调用日期(D):

原因(R): 新建

详细信息 会议记录 注释 与会者 所有链接 附件

目标(D): 项目计划评审

历史记录(H):

Type your comment here.

保存后“调用日期”一栏自动记录保存该评审工作项时的时间，如下：

会议类型(Y): 会议

参与实施者

调用方(B): 毕思姗

调用日期(D): 4/8/11 2:41:55 PM

评审 109 (Modified) : 项目计划评审

标题(T): 项目计划评审 会议类型(Y): 会议

状态

指派给(G): 毕思姗

调用方(B): 毕思姗

状态(S): 活动

调用日期(D): 4/8/11 2:41:55 PM

原因(R): 新建

详细信息 会议记录 注释 与会者 所有链接 附件

必填	可选	实际
指定与会者 1(A): 刘堤军	非指定与会者 1(A): 李杨	实际与会者 1(A): 刘堤军
指定与会者 2(A):	非指定与会者 2(A):	实际与会者 2(A): 李杨
指定与会者 3(A):	非指定与会者 3(A):	实际与会者 3(A):

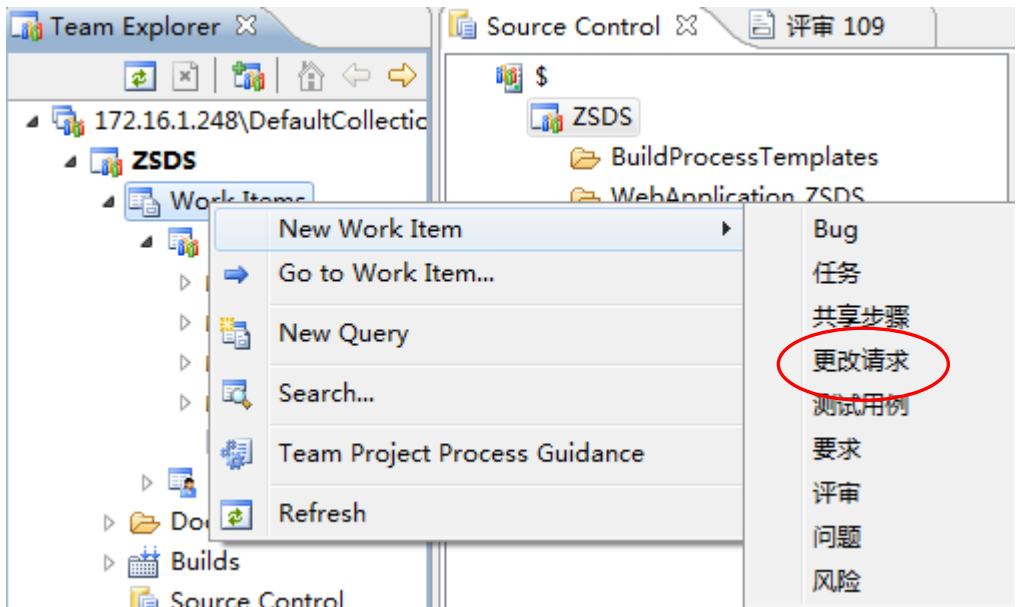
评审的流程相对简单，例如：



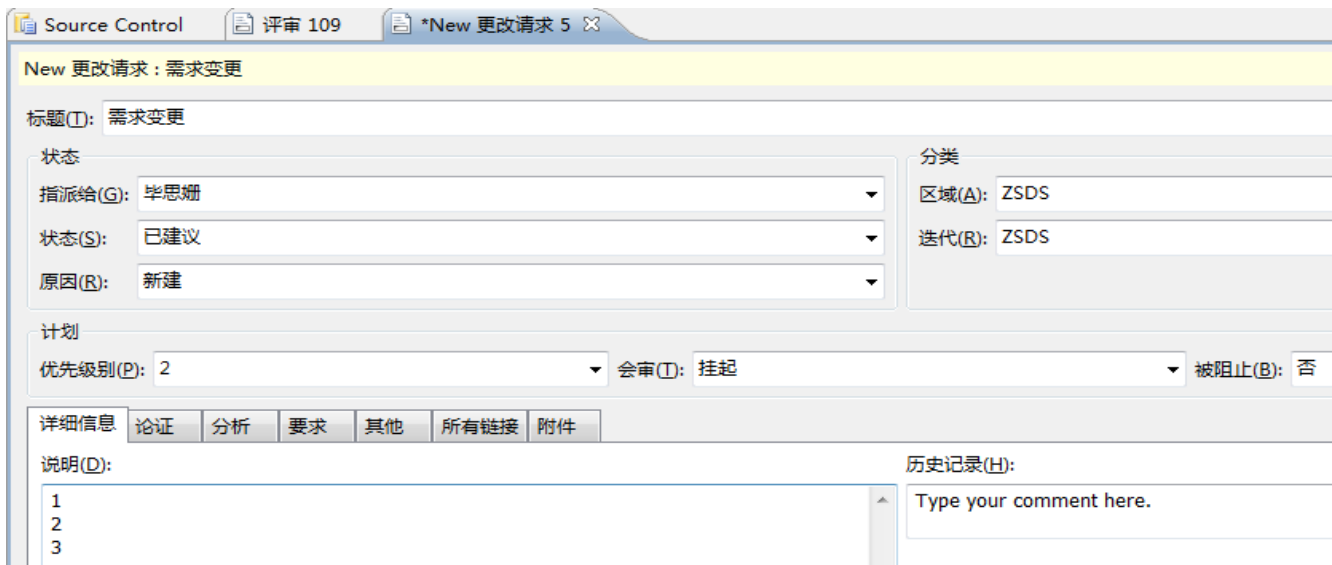
## 5.5 变更申请人

这部分主要介绍变更管理流程，因为项目中任何一个人都有可能是变更的发起者，所以这里

单独描述。变更申请人提交“变更”请求单的操作如下：



填入详细信息后保存工作项。



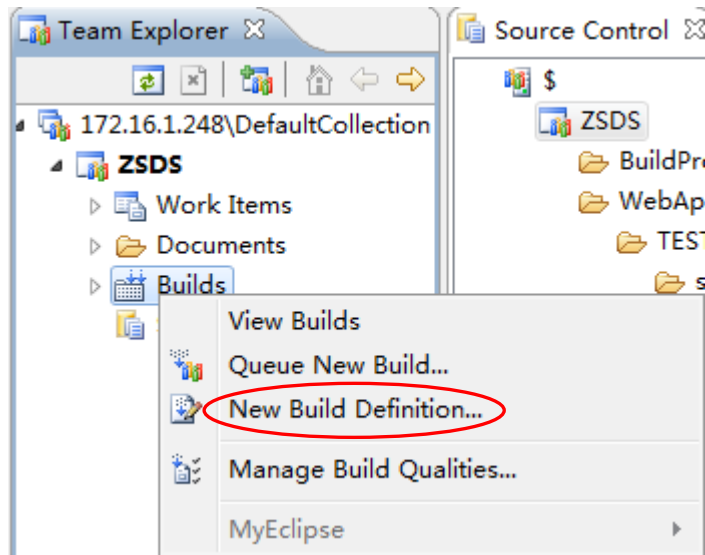
变更的状态转换图可能如下：



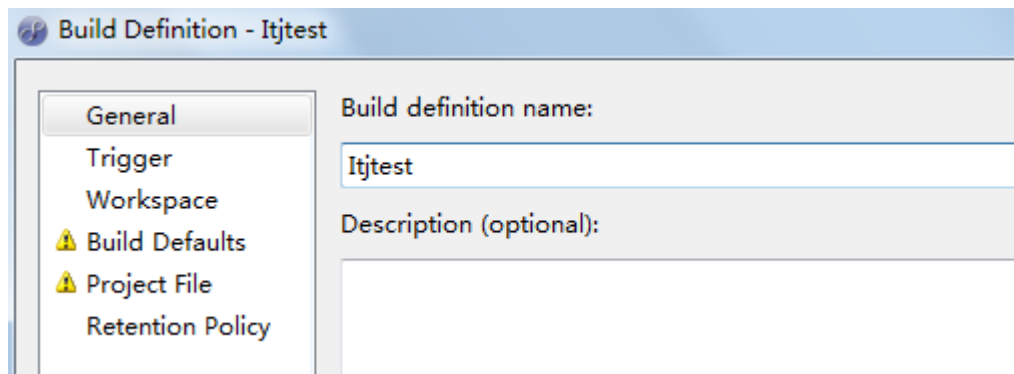
## 6. 生成

工作流程的下一个典型的环节就是自动生成产品。

通过 “Team Explorer ”的 “Builds” 定义一个构建：

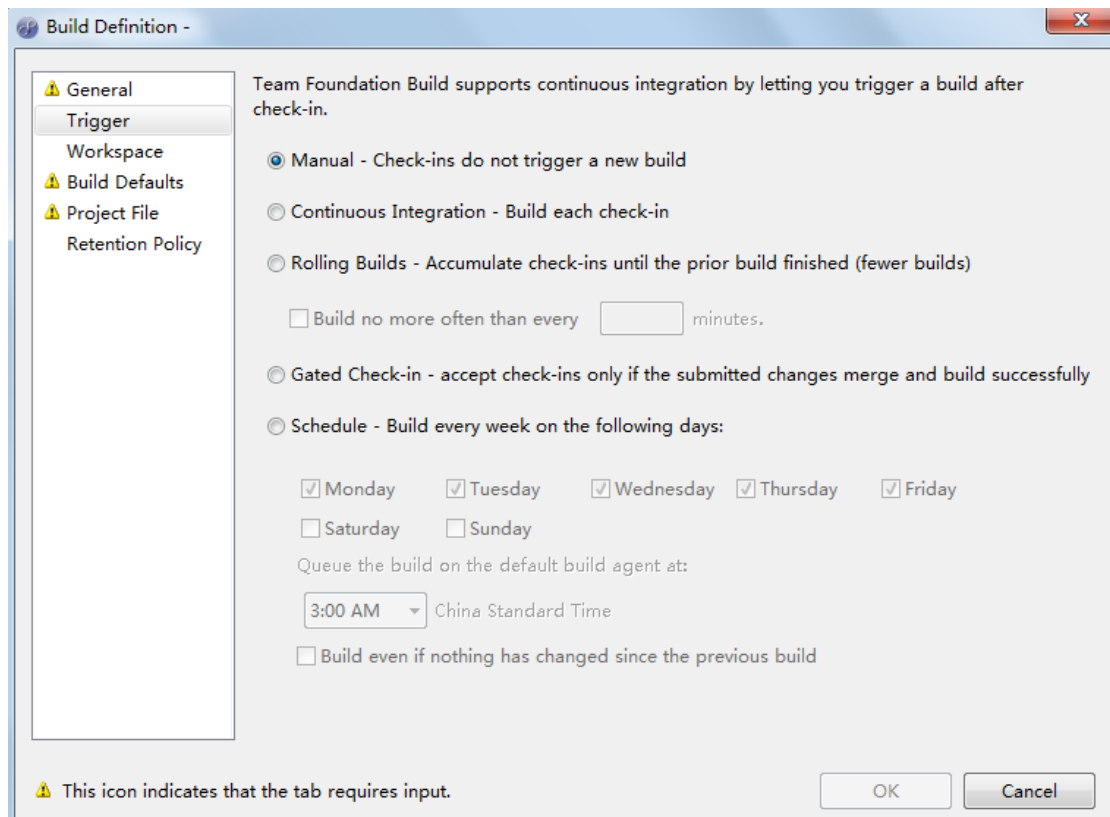


有一系列的定义需要填，就像一个代码项目的属性页：



触发页面使我们能够决定生成何时开始。你可以从下列项中选择：

手工、持续集成、滚动式的构建、每个 Check in 触发一个构建、定时进行构建



默认情况下使用手动项。我们必须以这一项开始我们自己的生成。

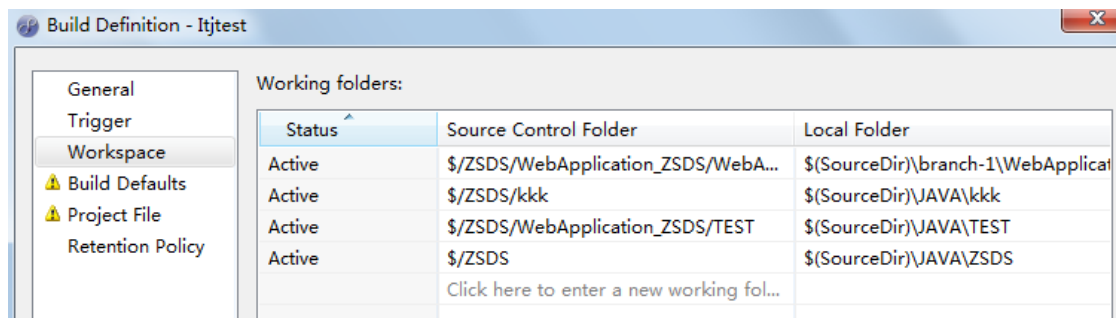
每一次签入后，当你想拥有一个新的生成的时候，持续集成是非常有用的。它允许你立刻验证新的签入而不用等很多的签入混合在一起后再验证。

滚动生成提供了一种批处理改变的方法，当开始生成要花一点时间并且你无法去做每一项时，这种方法很便利。

封闭签入让你确保所有的签入传入 TFS 前被生成。并确保你不会对你项目组的其它成员造成生成破坏。

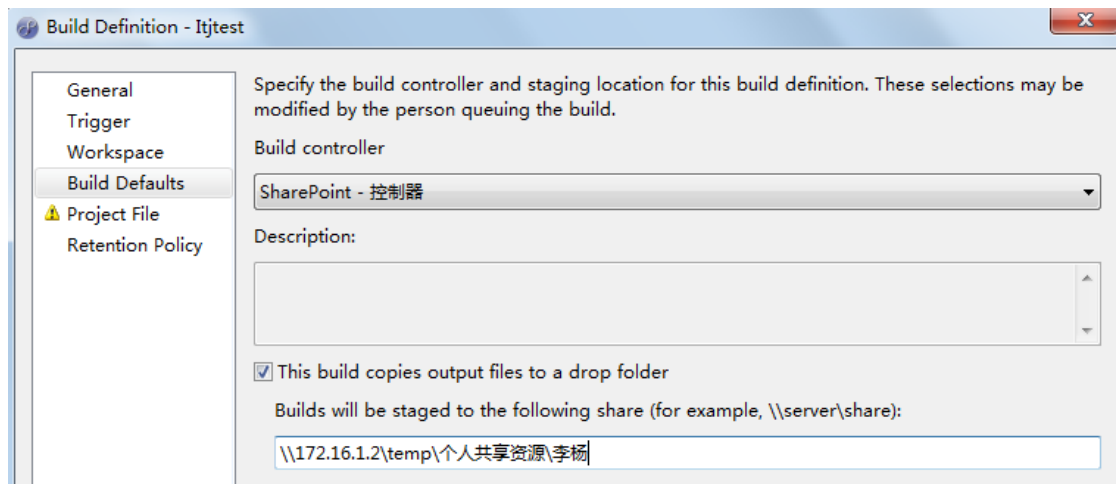
计划生成是整个团队做每日生成进行了有益的尝试

可以创建和使用多个不同的生成定义，允许您根据不同的目的来使用不同类型的生成。

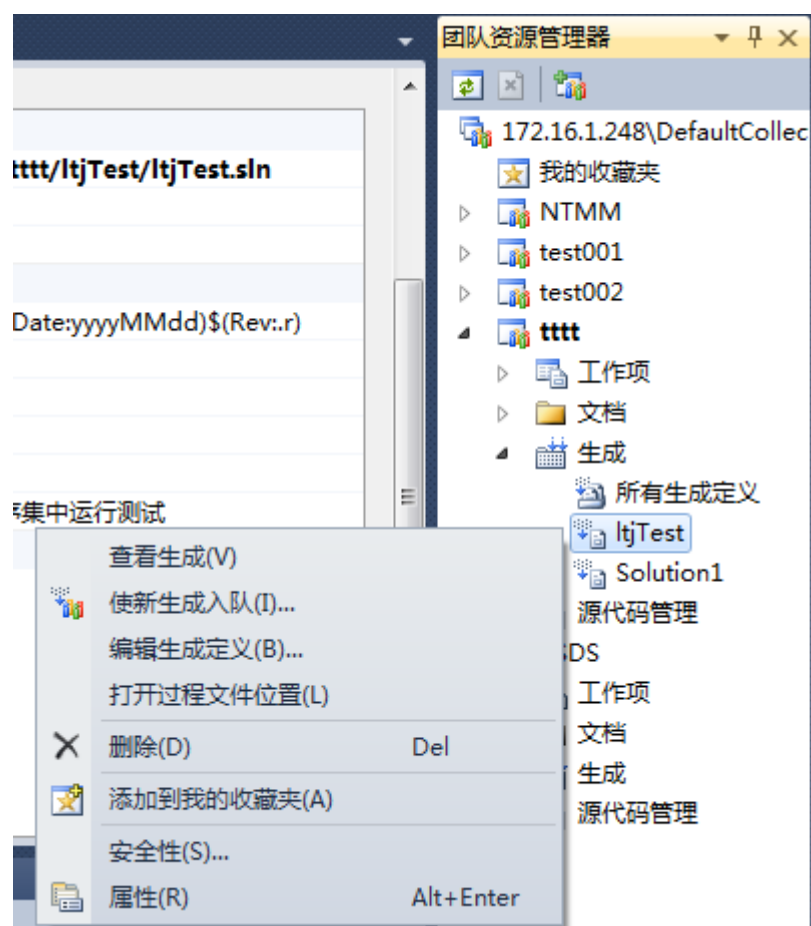


你需要在工作区中指定活动的工作区。

你看到“生成默认值”有一个黄色的警告标签，每一个 Build 都会被打上有一个标签，需要有一个地方来存放每次 Build 的结果，默认没有设置，所以是一个警告标记，我这里使用服务器上的一个共享目录进行存放。



现在可以保存生成定义到 TFS。如果回到 Team Explorer，我们可以“使新的生成入队”：



出来一个确认对话框，可以直接按下“排队”按钮。



将生成“tttt”排队

常规 参数

生成定义(F):

ltjTest

您希望生成什么(W)?

最新源

生成控制器(C):

SharePoint - 控制器

队列中的优先级(P):

正常

位置(S):

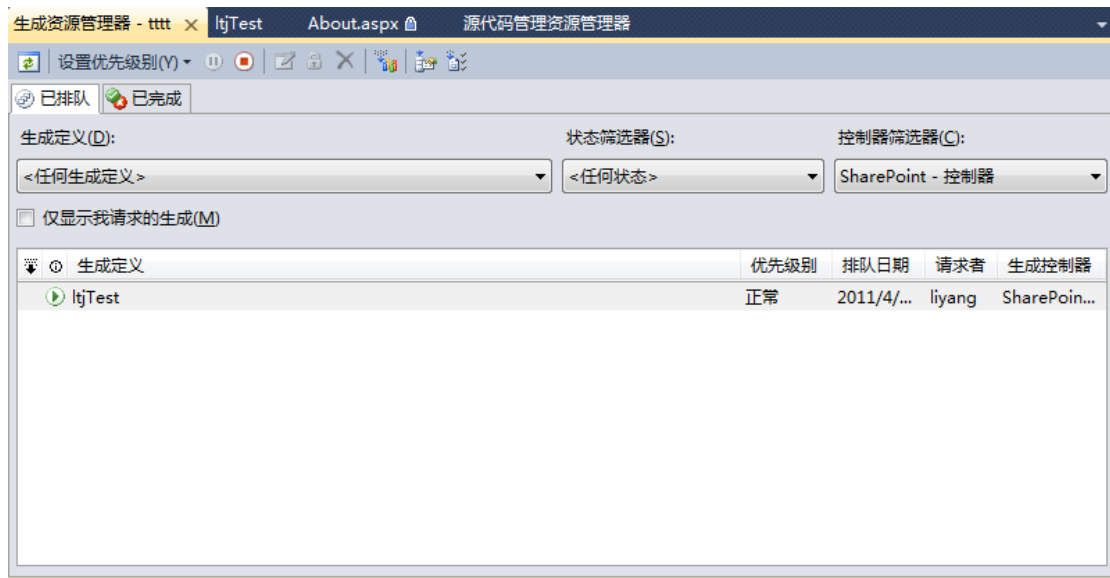
1

该生成的放置文件夹(D):

\\172.16.1.2\temp\个人共享资源\李杨

排队(Q) 取消

我们可以看到构建的状态



双击就可看到详细的构建报告



Build WindowsFormsApplication1\_20110122.1 x Build Explorer - TestProject1 Bug 5 Active Bugs [Results]

WindowsFormsApplication1\_20110122.1 - Build failed - <No Quality Assigne... ▾

[View Summary](#) | [View Log](#) - [Open Drop Folder](#) | [Delete Build](#)

fzq triggered WindowsFormsApplication1 (TestProject1) for changeset 122  
Ran for 26 seconds (zjst-server - Controller), completed 89 seconds ago

**Latest Activity**

Build last modified by TFSBUILD 89 seconds ago.

**Summary**

**Other Errors and Warnings**

▼ 1 error(s), 0 warning(s)

TF14044: Access Denied: User ZJST-SERVER\TFSBUILD needs the CreateWorkspace global permission(s).

Build WindowsFormsApplication1\_20110122.2 x Build Explorer - TestProject1

WindowsFormsApplication1\_20110122.2 - Build succeeded - <No Quality Assigne... ▾

[View Summary](#) | [View Log](#) - [Open Drop Folder](#) | [Retain Indefinitely](#) | [Delete Build](#)

fzq triggered WindowsFormsApplication1 (TestProject1) for changeset 122  
Ran for 41 seconds (zjst-server - Controller), completed 5 seconds ago

**Latest Activity**

Build last modified by TFSBUILD 5 seconds ago.

**Summary**

**Debug | x86**

0 error(s), 0 warning(s)

► 1 projects/solutions compiled

No Test Results

No Code Coverage Results

**Impacted Tests**

从这里你可以看到警告和错误，日志文件，导航到 Drop 的目录等。例如，如果你选择“查看日志文件”，你能看到执行生成的脚本（子集）：

Build WindowsFormsApplication1\_20110122.1 x Build Explorer - TestProject1 Bug 5 Active Bugs [Results]

**WindowsFormsApplication1\_20110122.1 - Build failed** - <No Quality Assigne...>

[View Summary](#) | [View Log](#) - [Open Drop Folder](#) | [Delete Build](#)

fzq triggered WindowsFormsApplication1 (TestProject1) for changeset 122  
 Ran for 26 seconds (zjst-server - Controller), completed 89 seconds ago

Activity Log | **Next Error** | Next Warning

	Duration
Overall Build Process	00:17
Update Build Number	00:00
Create the Drop Location	00:00
Run On Agent (reserved build agent zjst-server - Agent1)	00:16
Delete Test Results Directory	00:00
Delete Binaries Directory	00:00
Delete Workspace	00:00
Delete Sources Directory	00:00
Create Workspace	00:00
TF14044: Access Denied: User ZJST-SERVER\TFSBUILD needs the CreateWorkspace global permission (s).	

生成 ItjTest\_20110405.4 x F:\TFS\TEST003\...aultTemplate.xaml 源代码管理资源管理器 生成资源管理器 - tttt

**ItjTest\_20110405.4 - 生成已部分成功** - <未分配质量>

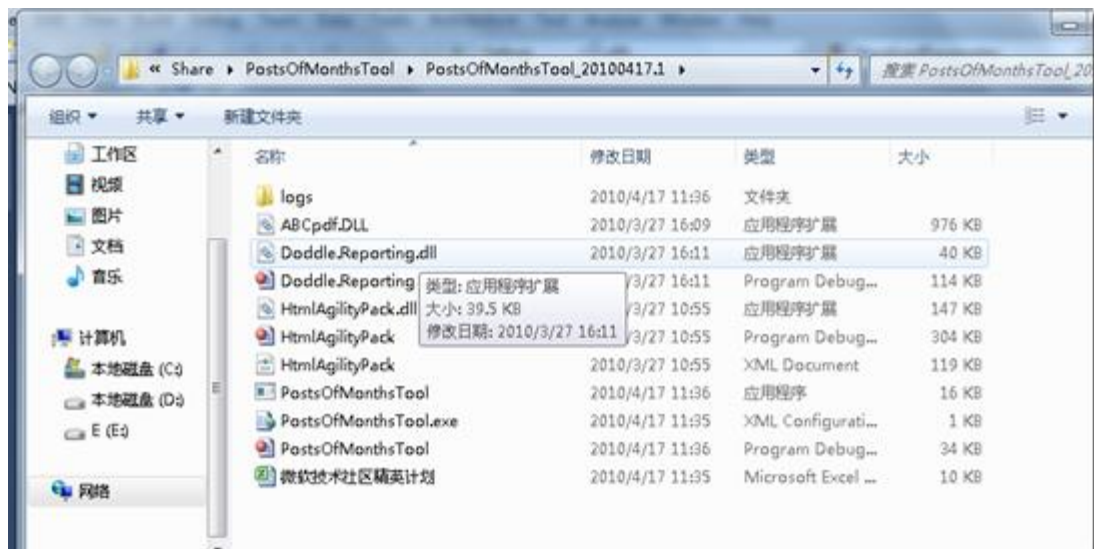
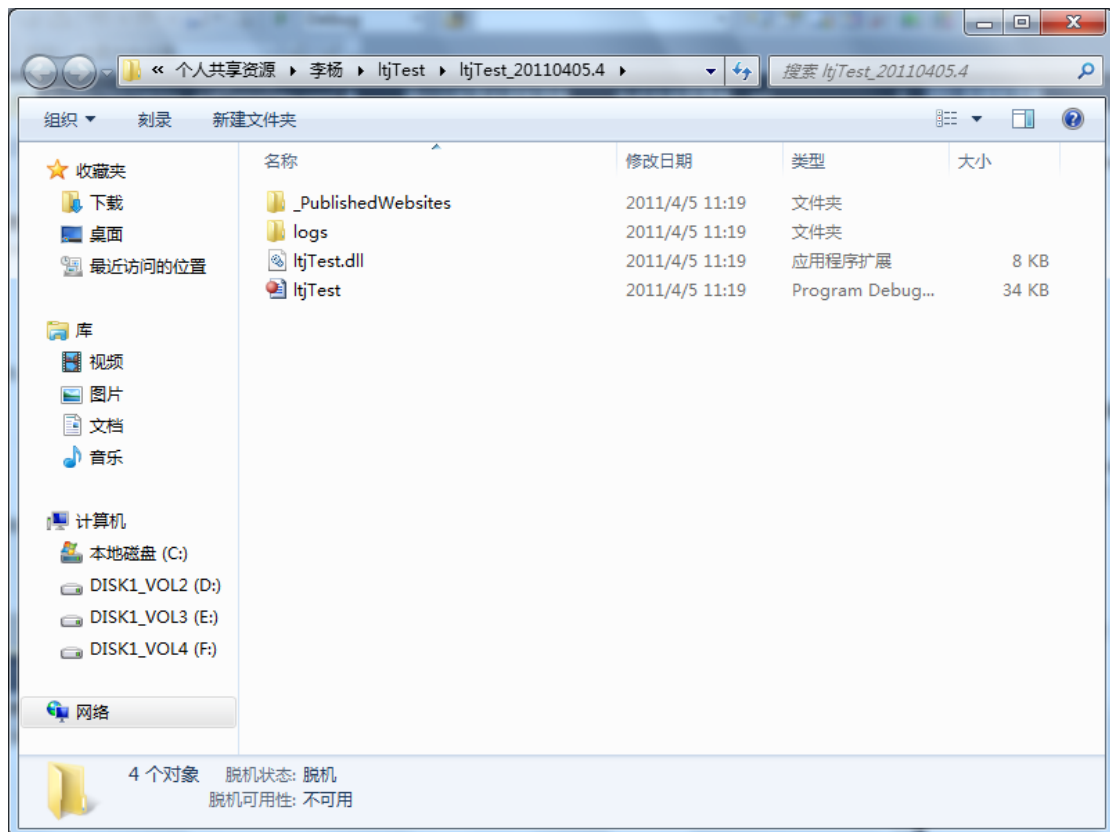
[查看摘要](#) | [查看日志](#) - [打开放置文件夹](#) | [无限期保留](#) | [删除生成](#)

liyang 已为 变更集 46 触发 ItjTest (tttt)  
 运行了 5 秒(SharePoint - 控制器), 于 37 秒之前完成

活动日志 | **下一个错误** | 下一个警告

	持续时间
总体生成过程	00:05
更新版本号	00:00
创建放置位置	00:00
在代理上运行 (已保留生成代理 SharePoint - 代理1)	00:05
删除测试结果目录	00:00
删除二进制文件目录	00:00
删除工作区	00:00
删除源目录	00:00
创建工作区	00:00

通过 “打开放置文件夹” 就可以打开我们编译的文件的存放位置



现在任何人都可以选择生成来做他们日常测试，或发布给客户等。