

Easy! Software LLC

Easy!Flow SCMClient for Eclipse

用户手册

Easy! Software LLC

本文档包含受版权保护的信息。禁止以任何形式发布、修改或出版本文档。本文档的使用必须获得 Easy! Software LLC 的授权

目录

目录.....	2
1. 概述.....	3
1.1. Easy!Flow SCMClient 是什么.....	3
1.2. 为什么要使用 Easy!Flow SCMClient.....	3
1.3. 编辑配置文件.....	4
1.4. 工作流程概览.....	4
2. 使用 SCMClient 进行开发.....	8
2.1. 登录 SCMClient.....	8
2.2. 打开 Easy!Flow 站点.....	8
2.3. 在 SVN 中创建产品.....	9
2.4. 创建分支.....	10
2.5. 检出本地副本并提交.....	11
2.6. 合并与冲突处理.....	12
2.7. 剔除合并分支.....	14
2.8. 创建标签.....	15
3. 常见问题.....	16

1. 概述

1.1. Easy!Flow SCMClient是什么

Easy!Flow SCMClient 是 Easy!Flow 的客户端软件, 用来访问与整个开发过程管理系统紧密集成的版本控制系统。它通过一个预定义的、可配置的软件开发过程, 帮助用户与对 SVN 版本控制系统进行规范化操作。Easy!Flow SCMClient 提供了标准的规范化的 SVN 版本库结构, 变更管理规则, 以及方便易用的代码分支合并与剔除工具。

Easy!Flow 提供两种形式的 SCMClient: (1) 独立运行的 SCMClient, (2) 基于 Eclipse 的 SCMClient。后者允许用户直接从 Eclipse IDE 访问 Easy!Flow 并进行相关的 SVN 操作。

1.2. 为什么要使用 Easy!Flow SCMClient

与其他 SVN 客户端相比, Easy!Flow SCMClient 具有以下优点。

1. 容易使用-- Easy!Flow SCMClient 为用户提供了一个简单易用的面向过程的 SVN 接口。使用 Easy!Flow SCMClient, 用户不需要在版本库结构、分支与标签策略、分支对应的版本号等细节上花费大量时间。
2. 服务器端 SVN 触发器 -- Easy!Flow SCMClient 提供了服务器端的 SVN 触发器, 防止用户进行违反标准开发过程的操作, 引导用户进行正确的操作。
3. 自动组织版本库结构 -- Easy!Flow SCMClient 定义了标准的、适用于变更管理和缺陷跟踪的版本库结构。版本库受 Easy!Flow 服务器端触发器的保护。
4. 与 Easy!Flow 缺陷跟踪系统和 Easy!Review 代码审查系统无缝集成 -- Easy!Flow SCMClient 维护缺陷跟踪记录和 SVN 版本库之间的数据一致性。Easy!Review 可以为代码分支自动生成代码审查包。

1.3. 编辑配置文件

选择 **SCMClient > 编辑配置文件**，在文本编辑器中打开配置文件后，设置 'easyflow_url' 和 'repos_url' 为与你的 Easy!Flow 服务器对应的 URL 地址。（'instance' 一般不需要进行设置，除非你安装了多个 Easy!Flow 实例）

下面是一个典型的 Easy!Flow SCMClient 配置文件：

```
# Easy!Flow 缺陷管理系统的 Web 地址.
easyflow_url=http://10.0.0.102:1080/EasyFlow/

# Easy!Flow 缺陷管理系统的实例.
# 不同实例用';'分隔
# 例如: instance=default;instance1
instance=default

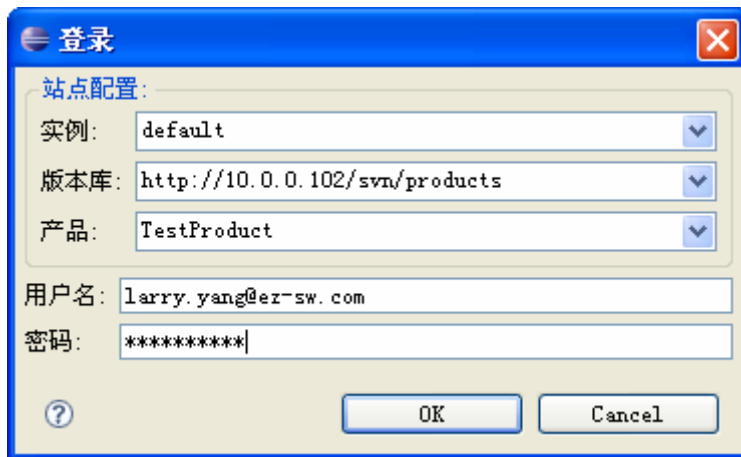
# SVN 资源库.
# 不同实例的 SVN 资源库用'|'分隔.
# 同一实例的 SVN 资源库用';'分隔.
# 例如: repos_url=<inst1 repos1>;<inst1_repos2>|<inst2_repos1>;<inst2_repos2>
repos_url=http://10.0.0.102:1080/svn/products/
```

1.4. 工作流程概览

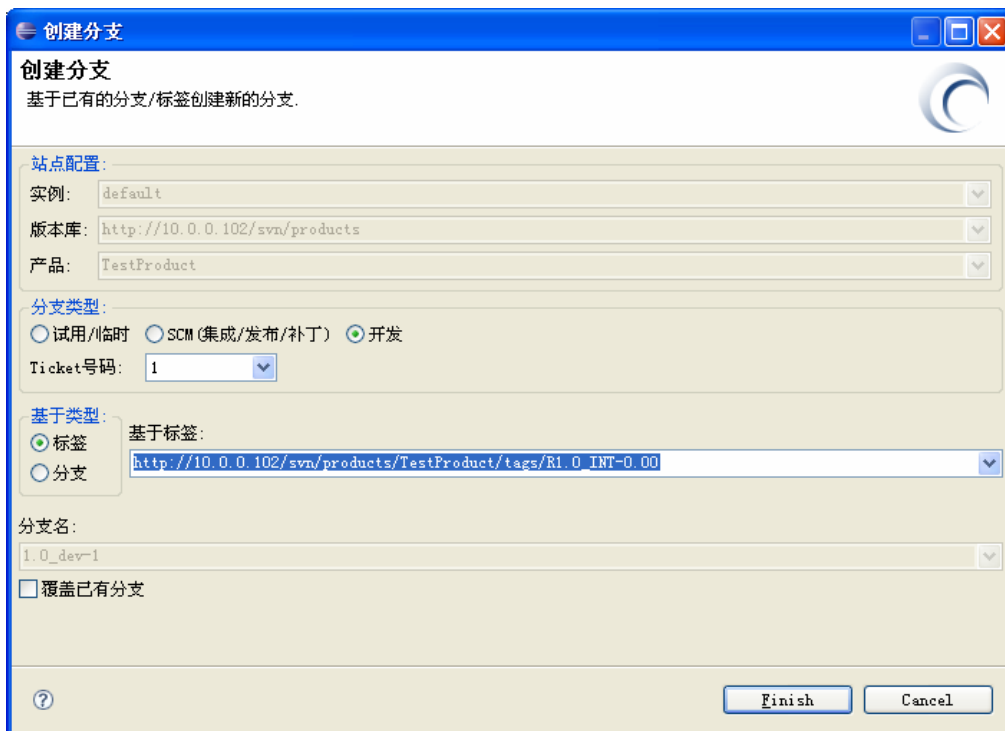
下面是使用基于 Eclipse 的 SCMClient 进行开发工作的一个简单例子。

假设 Larry 正在解决 Ticket# 1，需要修改一些文件。他使用 Easy!Flow SCMClient 进行文件修改与集成。

步骤 1: Larry 登录 SCMClient:

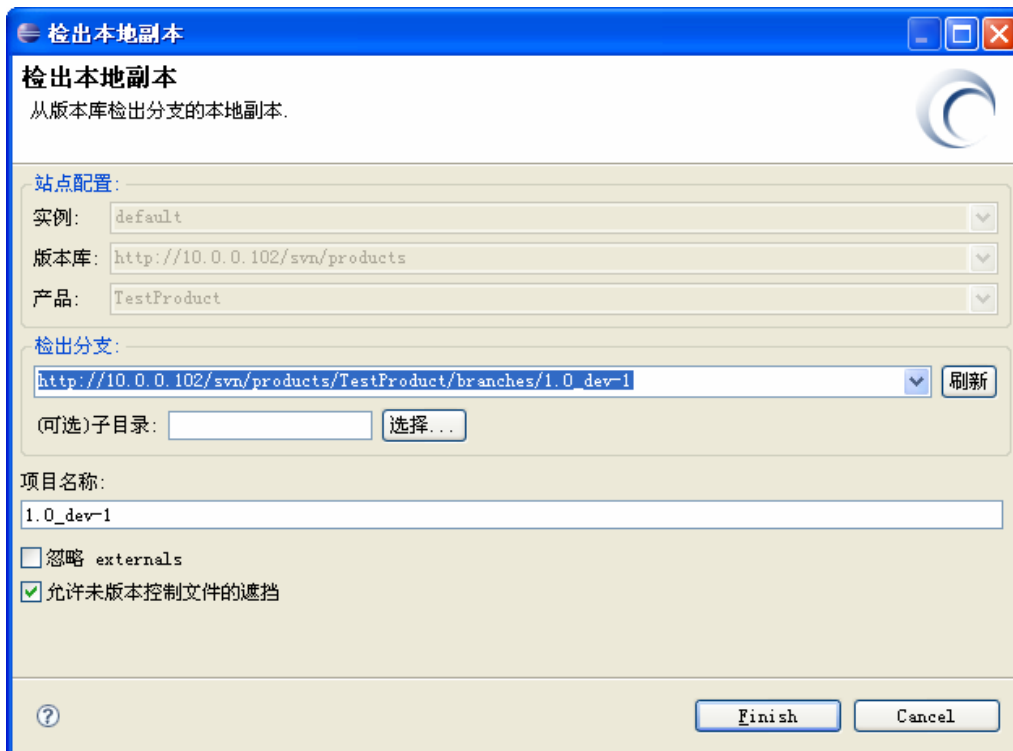


步骤 2: Larry 创建一个开发分支，他的所有修改将在该分支上完成。

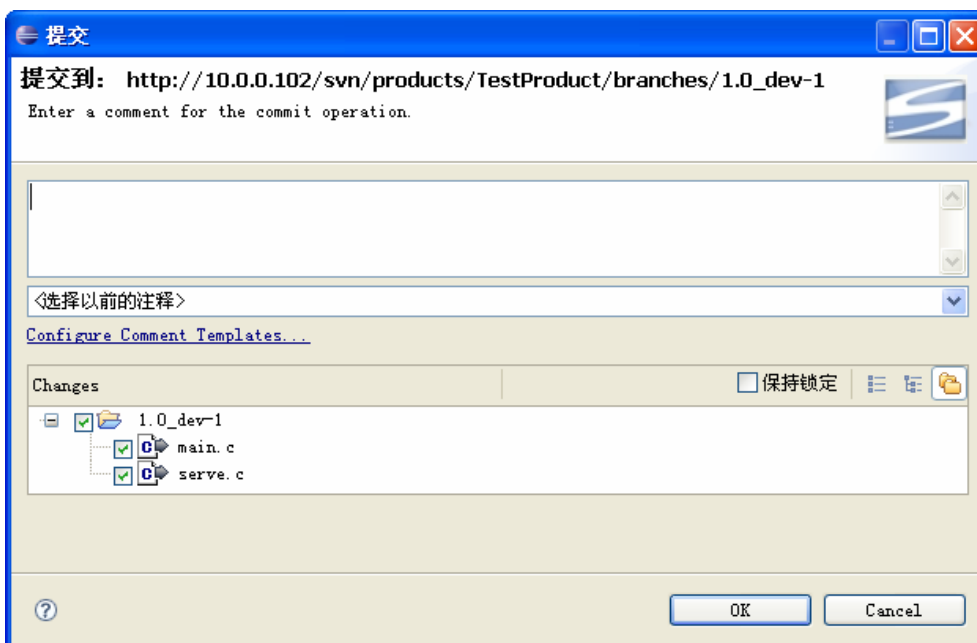


注意: SCMClient 运行时会在 Eclipse IDE 的底部自动打开 SVN 终端以显示运行信息。

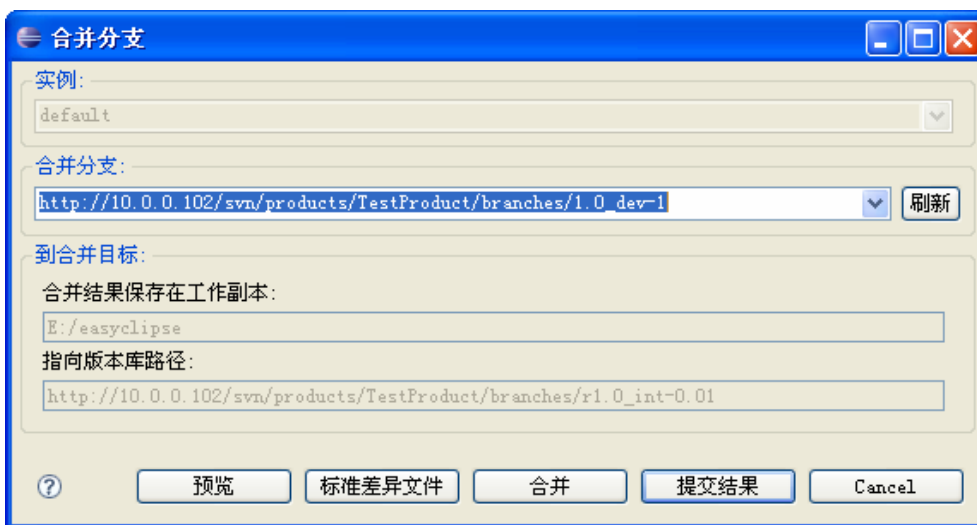
步骤 3: Larry 检出他新创建的开发分支的本地副本:



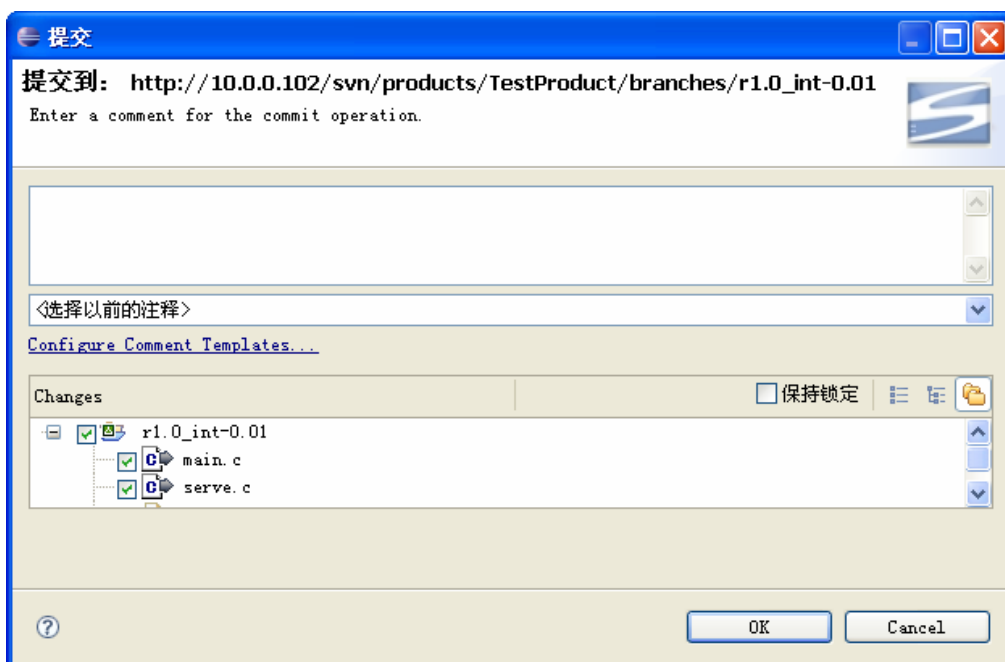
步骤 4: Larry 在本地副本中完成文件修改并提交到 SVN 版本库。



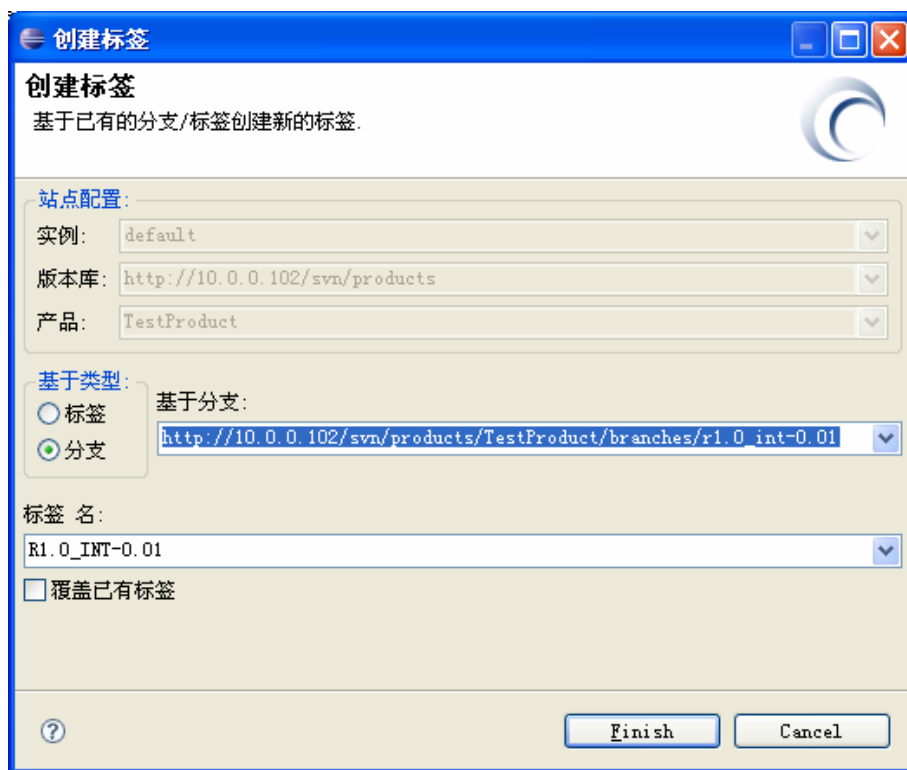
步骤 5: Larry 检出集成分支的本地副本，把他的修改从开发分支合并到集成分支的本地副本。



步骤 6: 如果合并过程没有发现冲突，Larry 把集成分支的本地副本提交到 SVN 版本库。（如果发现冲突，请参考本文 2.6 小节的内容）



步骤 7: 合并所有相关的开发分支到集成分支后, 系统管理员在集成分支上创建标签, 该标签将作为下一个迭代过程的基础。



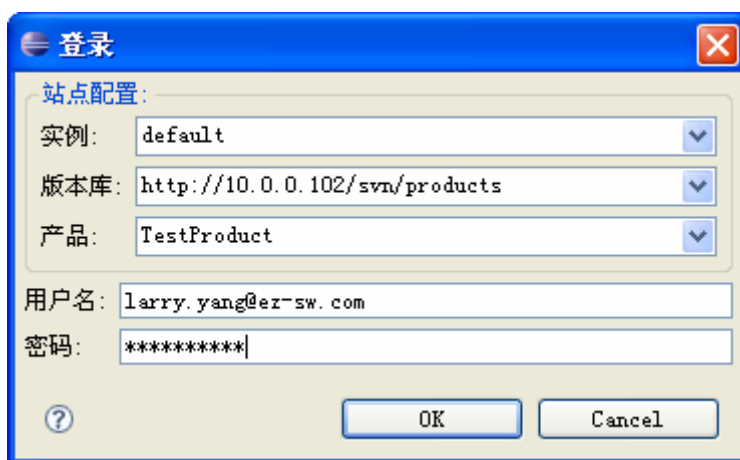
在上面的过程中, 产品源代码的管理, SVN 触发器对用户操作的检查, 缺陷跟踪系统的记录, SVN 操作全部由 Easy!Flow SCMClient 自动完成。这样开发人员就不用费心研究 SVN 的工作机制与使用技巧, 也不用担心自己的操作是否违反了公司的开发流程, 他们可以把精力更多的集中在产品的研发上面。

2. 使用SCMClient进行开发

2.1. 登录 SCMClient

菜单: **SCMClient > 登录**

SCMClient 和 Easy!Flow 缺陷跟踪系统共享登录账号。SCMClient 所有的 SVN 操作只有在当前登录用户有相应权限时才会被执行。在登录窗口中, 选择你的产品所在的 Easy!Flow 实例, SVN 版本库, 以及产品名。



注意: 可选的实例和 SVN 版本库在 SCMClient 配置文件中定义, 选择 **SCMClient > 编辑配置文件** 可以编辑 SCMClient 配置文件。

2.2. 打开 Easy!Flow 站点

菜单: **SCMClient > 打开 Easy!Flow 站点**

用户可以选择 **SCMClient > 打开 Easy!Flow 站点** 在 Eclipse IDE 中打开 Easy!Flow 缺陷跟踪系统的用户界面。

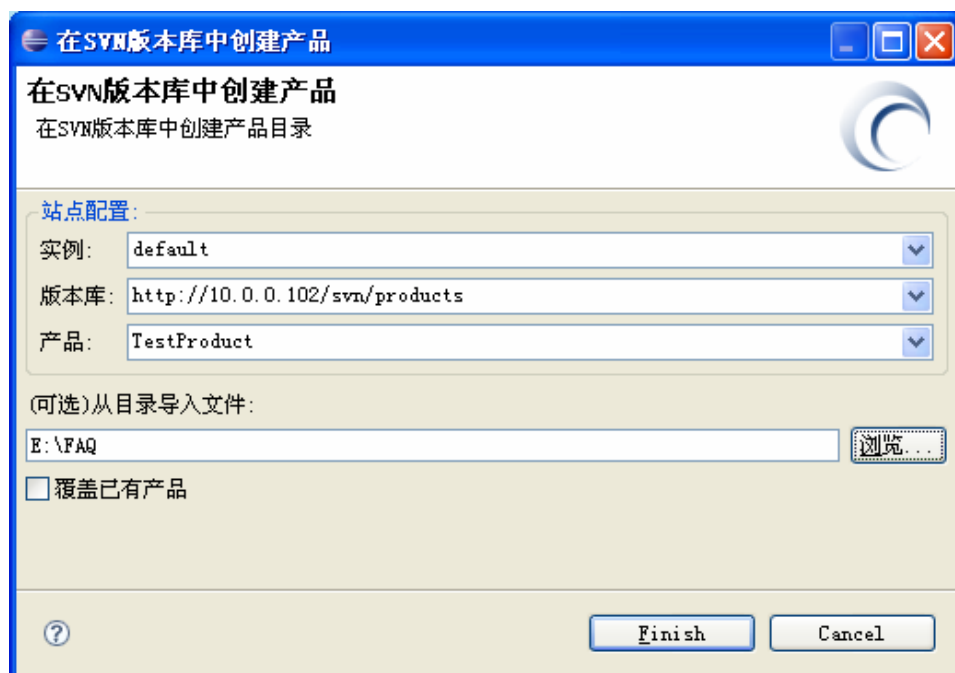


2.3. 在 SVN 中创建产品

菜单: **SCMClient > 在 SVN 中创建产品**

在 Easy!Flow 缺陷跟踪系统中创建一个新产品之后, 系统管理员需要在 SVN 版本库中创建该产品。

如果你有初始产品文件需要导入到新产品, 点击“浏览...”按钮以选择包含初始产品文件的目录。



2.4. 创建分支

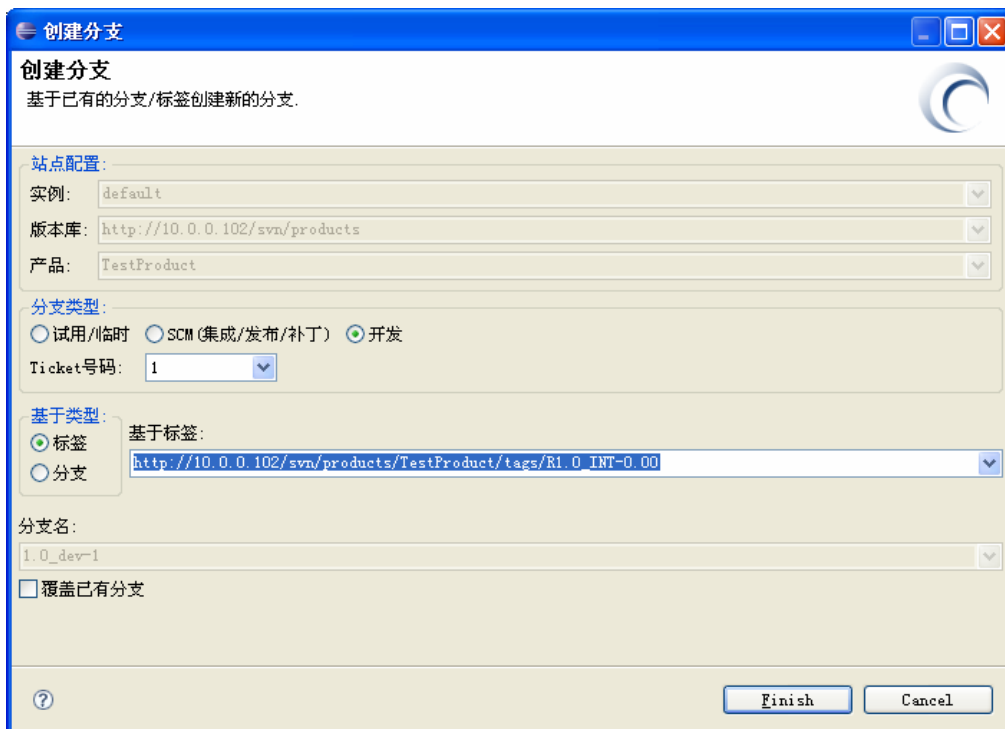
菜单: **SCMClient > 创建分支**

分支是现代变更管理系统中用于实现多用户同时修改的主要策略。Easy!Flow SCMClient 有三种类型的分支：临时分支，开发分支和 SCM 分支。任何用户都可以创建临时分支，其修改不会被集成。一个开发分支对应 Easy!Flow 缺陷跟踪系统中的一个 Ticket，其修改将被合并到 Ticket 的目标里程碑对应的集成 SCM 分支，以形成一个新的发布版本或基线。

下面是三种类型的分支的命名规则。Easy!Flow SCMClient 会通过用户提供的信息自动生成分支名称。

分支类型	命名规则	举例
临时	private_tmp-<...> 或其他名称	private_tmp-test-by-larry
开发	<产品版本号>_dev-<Ticket 号码>	1.0_dev-1
SCM	与产品的目标里程碑同名，但字母为小写形式	r1.0_int-0.01

一个新的分支必须基于一个已有的分支或标签。Easy!Flow SCMClient 会自动列出所有已有的分支/标签，用户只需要从下拉列表中选择一个。

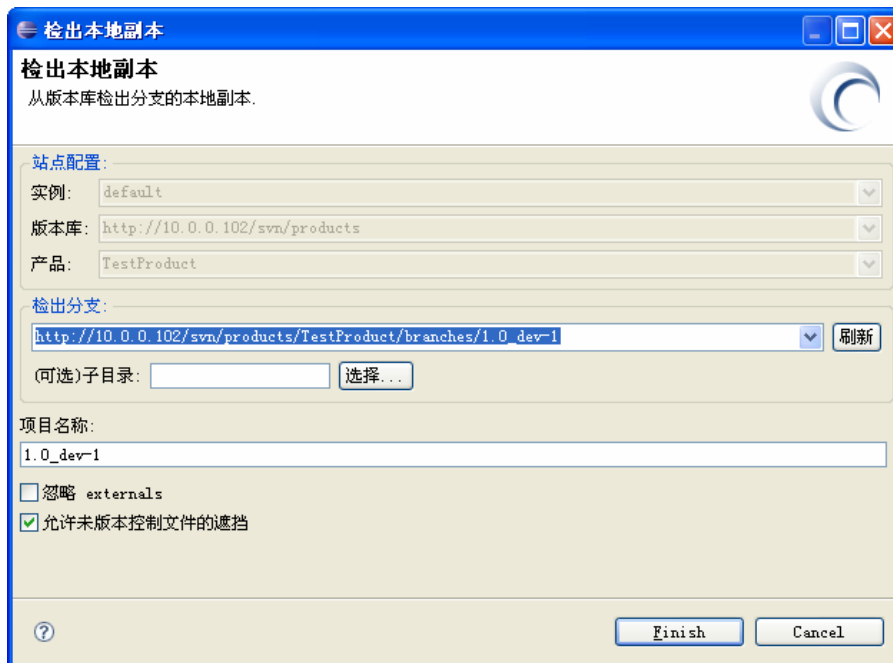


如果同名分支已经存在，用户必须选中“覆盖已有分支”单选框，这样 SCMClient 将首先从 SVN 版本库中删除同名分支后再创建新分支。

2.5. 检出本地副本并提交

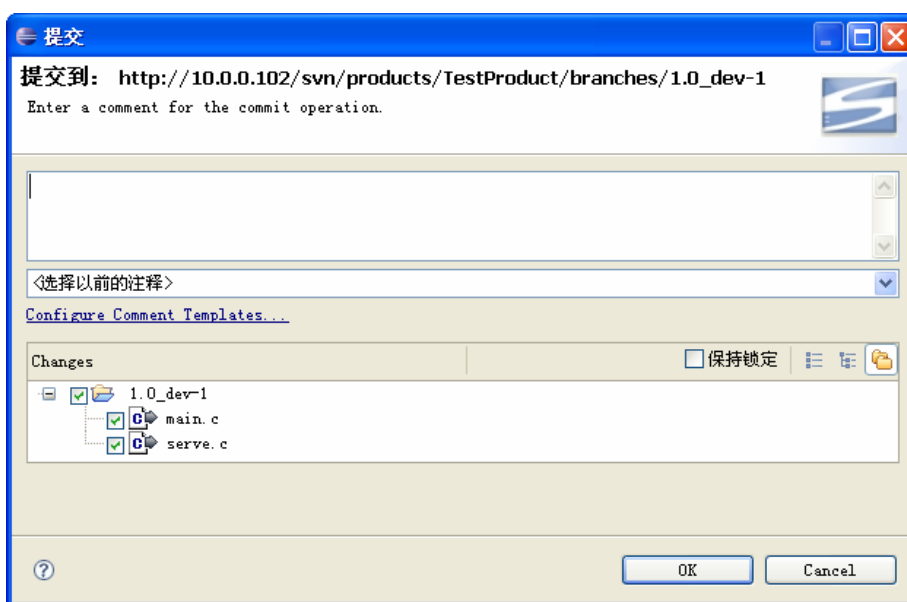
菜单: **SCMClient > 检出本地副本**; **SCMClient > 提交...**

为了在分支上修改文件，你需要创建分支的一个本地副本，在本地副本中修改文件并最终提交到 SVN 版本库。



如上图所示，从“检出分支”下拉列表中选择一个要检出的分支。

在本地副本中完成修改后，提交修改到 SVN 版本库：



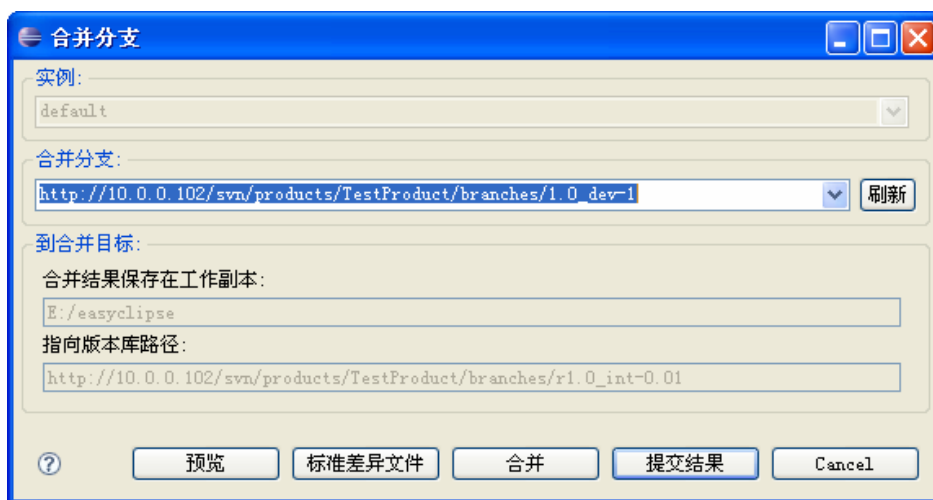
2.6. 合并与冲突处理

菜单: **SCMClient > 合并分支**; **SCMClient > 编辑冲突**; **SCMClient > 标记为已解决**

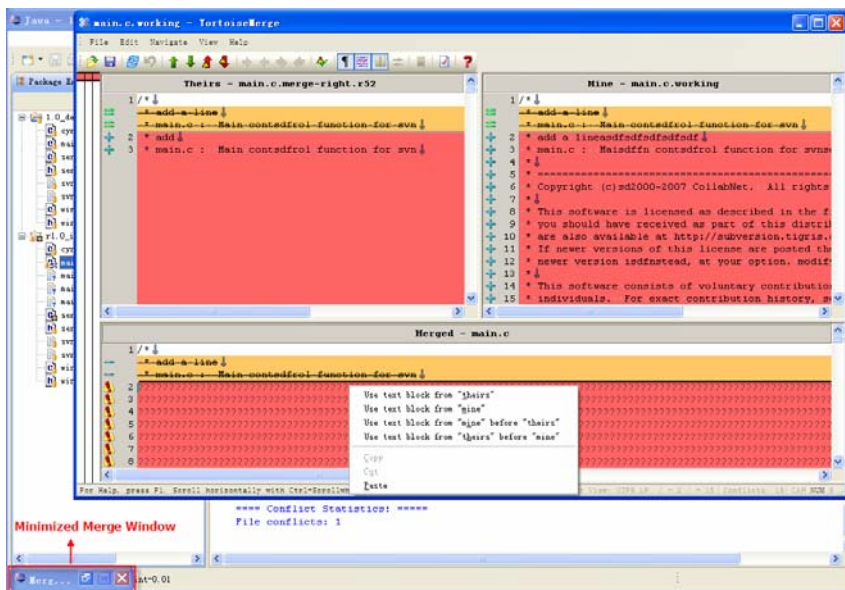
在合并一个分支到目标分支之前，必须首先检出目标分支的本地副本，然后选中目标分支的本地副本，选择 **SCMClient > 合并分支**。在合并分支窗口中，在“合并分支”下拉列表中，选择需要合并的分支，目标分支相关的信息会根据你选中的本地副本自动填充。点击“合并”按钮进行合并。

如果合并成功，没有发现冲突，点击“提交结果”按钮提交合并结果到 SVN 版本库。

把开发分支 1.0_dev-1 合并到集成分支 r1.0_int-0.01 的截图：



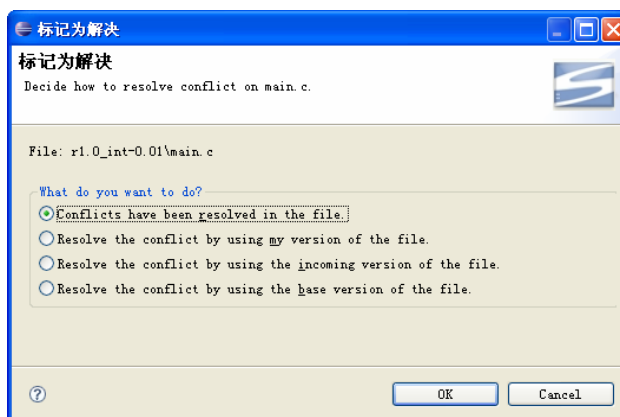
如果两个开发分支分别修改了同一文件的同一部分，并且被合并到同一个集成分支，就会发生合并冲突。例如，假设开发分支 1.0_dev-1 和 1.0_dev-2 修改了同一文件的同一部分，1.0_dev-1 首先合并到集成分支，那么 1.0_dev-2 合并到集成分支时就会发生冲突。合并 1.0_dev-2 到集成分支的用户需要在集成分支的本地副本中解决冲突，标记冲突文件为已解决，最后提交合并结果。在合并发生冲突的情况下，用户可以暂时把合并窗口最小化，然后开始解决冲突（选中冲突文件并选择 **SCMClient > 编辑冲突**）：



默认情况下，**SCMClient > 外部冲突处理器** 是被选中的，这样编辑冲突时就会调用如上图所示的冲突处理窗口。在冲突处理窗口中有 3 个子面板：

- 左上角的面板显示合并分支对基础文件的修改；
- 右上角的面板显示目标分支对基础文件的修改(一般是之前合并到目标分支的所有分支对基础文件的修改的合集)；
- 底部的面板显示冲突发生的位置，你可以采用下面任何一种方法修改文件以解决冲突：
 1. 右击发生冲突的红色区域，从右键菜单中选择一个冲突解决办法。
 2. 手工编辑冲突行。

解决冲突文件中的所有冲突后，保存文件，关闭冲突处理窗口，并标记当前文件为已解决 (选择 **SCMClient > 标记为已解决**)：



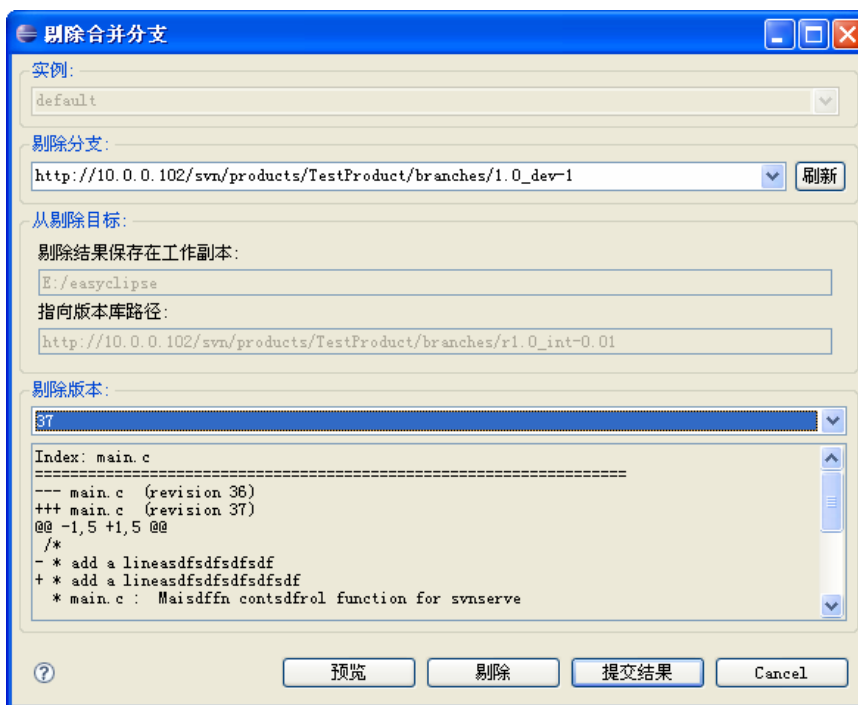
所有冲突文件都解决完毕后，恢复刚才被最小化的合并窗口，点击“提交结果”按钮提交合并结果到 SVN 版本库。

2.7. 剔除合并分支

菜单: **SCMClient > 剔除合并分支**

Easy!Flow SCMClient 支持从集成分支剔除已经合并的分支。例如，Larry 把开发分支 1.0_dev-1 合并到了集成分支 r1.0_int-0.01，但后来发现 1.0_dev-1 引入了一个难以在短时间内解决的问题，为了不影响整个产品发布，Larry 使用 Easy!Flow SCMClient 的“剔除合并分支”功能从集成分支 r1.0_int-0.01 剔除开发分支 1.0_dev-1 的修改。

下面的例子展示了如何进行“剔除合并分支”操作。首先在 Eclipse IDE 中选中集成分支 r1.0_int-0.01 的本地副本（作为剔除目标），然后选择 **SCMClient > 剔除合并分支**。在“剔除合并分支”窗口中，选择 1.0_dev-1 作为“剔除分支”，剔除目标相关的信息会根据你选中的本地副本自动填充。然后从“剔除版本”下拉列表中选择需要剔除的版本（在 1.0_dev-1 合并到 r1.0_int-0.01 时生成）并点击“剔除”按钮。剔除结果保存在 r1.0_int-0.01 的本地副本中，点击“提交结果”按钮提交剔除结果到 SVN 版本库。（如果剔除过程中发现冲突，则需要首先解决冲突）

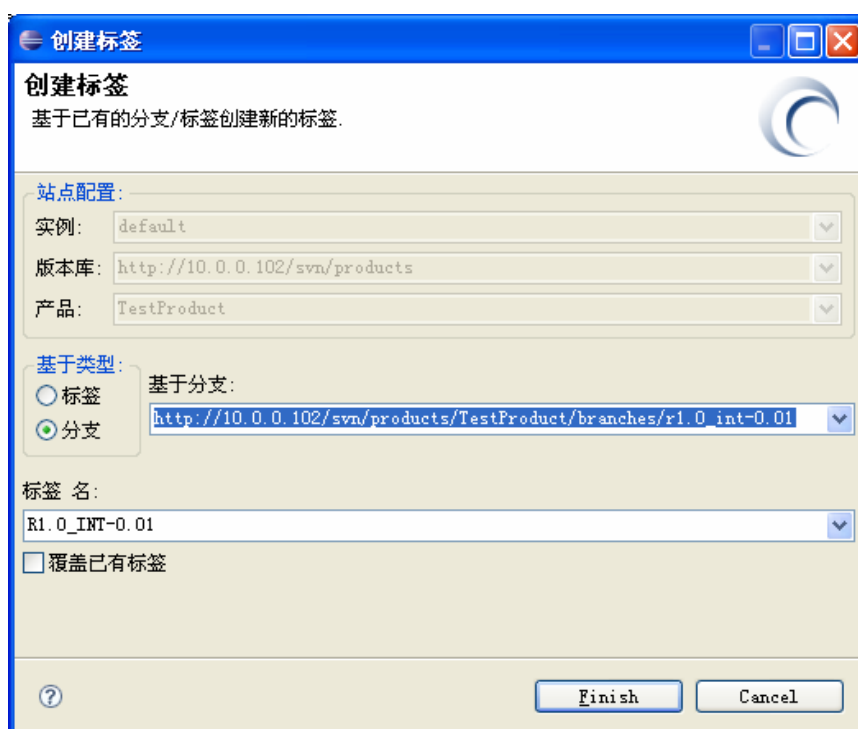


2.8. 创建标签

菜单: **SCMClient > 创建标签**

标签用于在 SVN 版本库中标记产品的基线。SCMClient 允许用户在分支或已有标签上创建标签。在完成代码集成之后,系统管理员通常需要在集成分支上创建一个标签作为下一个迭代过程的基础。

注意: 只有系统管理有可以在 SCM 分支上创建 SCM 标签。SCM 标签与 SCM 分支同名,但所有字母都大写,例如 R1.0_INT-0.01。



3. 常见问题

- [我可以选择SCMClient > 提交...，而不是在合并/剔除窗口中通过点击“提交结果”按钮来提交合并/剔除结果吗？](#)
- [在合并/剔除窗口中，为什么合并分支/剔除分支下拉列表列出的是文件/目录而不是分支路径？](#)

我可以选择 **SCMClient > 提交...**，而不是在合并/剔除窗口中通过点击“提交结果”按钮来提交合并/剔除结果吗？

不可以。只有在合并/剔除窗口中点击“提交结果”按钮，才会在提交窗口中自动输入固定格式的注释（comment），Easy!Flow 需要该注释信息才能在数据库中记录该合并/剔除操作。

在合并/剔除窗口中，为什么合并分支/剔除分支下拉列表列出的是文件/目录而不是分支路径？

在打开合并/剔除窗口前，你没有选中目标分支的本地副本（Eclipse 项目）。