Eclipse中CVS界面功能描述

2009-10-23 来源：网络

现在，您已经对CVS有了一个基本的理解。接下来，我们将讨论Eclipse是如何为CVS使用者提供帮助的。在对其进行讨论之前，先略为详细地看一下Eclipse的CVS用户界面。在开始之前，Eclipse的CVS用户界面是毫无价值的。这是因为，与Eclipse中的很多操作相似，大多数的CVS资源库访问操作都在后台运行，这些操作不会对其他操作造成妨碍。诸如检出、提交、同步等可能需要长时间运行的操作并不会对您所要完成的其他任务造成妨碍。在下面所给出的这个简短列表中，列出了在Eclipse用户界面中使用CVS时会碰到的最常见任务。在练习5中，您会实践所有这些任务。

定义到CVS资源库的连接。

浏览CVS资源库。

将新的工作空间项目定义到CVS中。

从CVS中检出已有项目。

将工作空间中的更改提交给CVS。

使用CVS中的最新版本来更新自己的工作空间项目。

在合适的里程碑处进行版本设置。

用资源库中的资源来比较或替换工作空间中的资源。

5.4.1 CVS Repository Exploring透视图：资源库主页

如图5-1所示的CVS Repository Exploring透视图是在Eclipse中使用CVS的大本营。在该透视图中含有CVS Repositories视图、CVS Resource History视图以及编辑器区域。在CVS Repositories视图中会显示出您的工作空间所知道的CVS资源库。而在CVS Resource History视图中所显示的则是受CVS控制的任何已知文件的修改历史记录。在CVS Repository Exploring透视图中，您可以建立与CVS服务器的连接。

5.4.2 CVS Repositories视图

CVS Repositories视图扮演着资源库浏览器的角色。在CVS Repositories视图中，您可以对资源库中存在的项目进行选择，并可将这些项目检出到工作空间中。如图5-1所示，CVS Repositories视图位于CVS Repository Exploring透视图的左边窗格中。通过使用快捷菜单New | Repository Location...或选择Add CVS Repository 命令，您可以在自己的工作空间中定义CVS资源库的位置。Eclipse支持各种类型的CVS连接(见图5-2)。在CVS Repositories视图中，您可以定义多个CVS资源库。在浏览资源库时，您可以看到资源库中的内容。如图5-1所示，这些内容包括HEAD、Branches、versions目录中的项目以及Dates条目。您可展开目录树或使用工具栏中的Go Into 和Back 操作来往下一直看到具体文件。在工具栏上有一个Refresh View 操作，由于CVS Repositories视图不会与资源库中的内容更改自动保持一致，所以在做任何重要操作之前，您可能要使用Refresh View 按钮来刷新CVS Repositories视图。

出于方便的考虑，如果您从Resource透视图中打开了CVS Repositories视图，那么该视图将作为一个选项卡式视图显示在Navigator透视图的旁边。

在CVS Repositories视图中，一个最常见的任务是从资源库中将项目检出到您的工作空间中。要完成上述任务，请先在Branches、HEAD或Versions下选择一个项目，然后再在单击鼠标右键后所出现的快捷菜单中选择Check Out命令。

图5-1 CVS Repository Exploring透视图

图5-2 在工作空间中定义新的CVS资源库位置

在第一次展开CVS Repositories视图中的Branches目录时，您会看到该目录中没有任何内容。在使用了快捷菜单操作Refresh Branches…后会出现Refresh Branches对话框(见图5-3)。在Refresh Branches对话框中，您可以为一个或多个资源库项目执行CVS中的分支标记检索请求。CVS资源库中可以包含大量分支。而您可能仅对其中的部分分支感兴趣。在Refresh Branches对话框中，您可以对要进行分支标记信息检索的项目进行选择。在表5-1所示的参考列表中，列出了CVS Repositories视图中的所有可用的快捷菜单操作。其中有些操作依赖于您在图5-3的第3个栏目中的选择。一开始，您可能需要使用以下这些操作：New | Repository Location…、Check Out…、Compare with…以及Show in Resource History。现在，您不必理解这些操作，它们仅为您提供一个参考。

图5-3 Refresh Branches对话框

表5-1 在CVS Repositories视图中可使用的上下文操作

操 作 描 述 从快捷菜单中可获得该命令

Add to Branch

List… 将一个项目(但不包括该项目的内容)添加至一个新的或已经存在的分支上。该操作可代替Team | Branch操作。对于那些将在一个分支中单独开发的新项目来说，该操作是比较有用的 被选择的项目

Check Out 把从资源库中所选择的内容复制到自己的工作空间中。而且文件夹可被检出，同时该文件夹与项目的一致性会得到维护。对于含有独立部分的大型项目来说，该操作非常有用 被选择的项目，以及在Branches、HEAD或Versions下所选择的项目文件夹

Check Out As… 把从资源库中所选择的内容复制到自己的工作空间中，并以另外一个名称来命名。在对所做的更改进行提交时，该项目保持它在CVS中的初始一致性。当需要在工作空间中同时拥有某一项目的多个实例时，该操作非常有用 被选择的项目，以及在Branches、HEAD或Versions下所选择的项目文件夹

(续表)

操 作 描 述 从快捷菜单中可获得该命令

Compare 在选择了两个资源库文件后，执行该操作会显示出Compare视图 两个被选择的资源库

Compare With… 该操作会打开Compare with Branch or Verson对话框，并允许您将所选资源与Branches、HEAD或Versions中的资源进行比较 被选择的资源

Configure Branches and

Versions… 执行该操作后将显示出一个对话框。在所显示的对话框中会显示出资源中的分支或版本标记。而对话框中所显示的那些标记会显示在CVS Repositories视图中 各种对象

Copy to Clipboard 将连接字符串复制到写字板上。例如：pserver: anonymous@dev.eclipse.org:/home/eclipse 被选择的资源库

Discard location 从CVS Repositories视图中删除所选资源库的位置定义 被选择的资源库

New | Date Tag… 执行该操作后会显示出一个对话框以创建新的日期标记 被选择的资源库，在CVS Repositories视图中所选择的Dates条目

New | Repository Location…

执行该操作后将显示出一个对话框以定义新的CVS资源库位置。另外还可获得一个工具栏图标 任何对象

Open 执行该操作后会在编辑器中打开所选文件的最新版本 被选择的文件

Properties 显示所选资源库的属性及其连接信息 被选择的资源库

Refresh Branches… 执行该操作后将显示出一个对话框以允许您发现一个或多个项目的分支 在CVS Repositories视图中所选择的Branches条目，被选择(或没有被选择)的资源库位置

Refresh View 使用资源库的当前信息来刷新CVS Repositories视图 各种对象

Remove 将所选的日期标记从CVS Repositories视图删除掉 被选择的日期标记

Show Annotation 该操作会为所选文件显示CVS Annotate视图 被选择的文件

Show In Resource History 该操作会在CVS Resource History视图中显示所选文件的版本历史信息 被选择的文件

Tag As Version… 将一个版本标记应用到所选的资源及其子资源上 被选择的资源

Tag With

Existing… 该操作会将一已有的资源库版本或分支标记应用到所选的资源上。这是一个比较特殊的操作。有关何时使用该操作的更多信息请参见联机帮助中的Moving Version Tags专题 被选择的资源

5.4.3 CVS Annotate视图

CVS Annotate视图与CVS Repository Exploring透视图中的CVS Repositories视图使用的是同一界面空间。CVS Annotate视图用来标识各版本文件中所添加的新行以及被更改过的行。CVS Annotate视图还会指出做出这次修改的是谁。现在您知道是谁做出了这些难以解释的更新，或者是谁应该对这些难以解释的更新负责。对于所选择的任何版本来说，文件中所添加的新行会在编辑器中突出显示(见图5-4)。若要打开CVS Annotate视图，则从CVS Repositories视图、CVS Resource History视图的快捷菜单Team中选择Show Annotation操作。打开CVS Annotate视图的一种比较方便的方法是从Navigator透视图或Package Explorer视图中所显示的某一个文件上选择Team | Show Annotation操作。在打开CVS Repository Exploring透视图时会打开CVS Annotate视图，CVS Resource History视图，并且在编辑器中所打开的文件的版本是最新的。CVS Annotate视图中的选择操作会将编辑器复位到被更新的代码行，并且相应的版本会在CVS Resource History视图中突出显示出来。相反，如果在编辑器中选择了一行代码，那么在CVS Repository Exploring透视图的那些视图中就会适当地对所更改的对象进行突出显示。现在您就可以非常容易地对任何文本文件的、详细的更改历史信息进行检查。

图5-4 CVS Repository Exploring透视图中的CVS Annotate视图

5.4.4 CVS Resource History视图

CVS Resource History视图可显示某一具体文件的所有版本的详细信息(见图5-5)。CVS Resource History视图是CVS Repository Exploring透视图的一部分。在CVS Repositories视图中先选择一个文件，然后再在单击右键所出现的快捷菜单中选择Show in Resource History命令，这样您就可以看到该文件的历史版本信息。在受CVS控制的项目上单击鼠标右键后所出现的Team快捷菜单项中也有Show in Resource History命令。作为一种查看文件历史的快速方法，您可以将受CVS控制的工作空间中的某一文件拖放到CVS Resource History视图以直接查看该文件的历史信息。

图5-5 CVS Resource History视图

对于任何被选择的修订版本来说，与该修订版本相关的分支和版本标记，连同该版本的文件被提交到资源库时所提供的注释，都会一起显示在CVS Resource History视图的底部(见图5-5)。通过在CVS Resource History视图的下拉菜单中进行相应选择，我们可以隐藏该视图中的“Tags”项和“Comment”项。

文件的历史记录可能比较长，这时，您可以使用CVS Resource History视图工具栏上的Filter History 按钮来过滤CVS Resource History视图。也可以根据作者、注释值、日期变化等来对CVS Resource History视图中的显示内容进行过滤。CVS Resource History视图工具栏上的Link with Editor操作 会使CVS Resource History视图与当前活动编辑器(仅对CVS控制之下的文件)保持同步。

对于在CVS Resource History视图中被选择的那些版本来说，您对它们可以执行的操作是依赖于资源是否已被复制到工作空间而发生变化的。这些操作被列在表5-2中。

表5-2 CVS Resource History视图中的操作

操 作 描 述

Compare 该操作会对所选的两个版本进行相互比较

Get Contents 该操作会用所选版本的内容来替换该资源的本地工作副本中的内容

Get Sticky Revision 该操作会将资源的本地工作副本还原为一个先前版本。本地副本会作为资源库修订版，并会保留相同的版本标记。需要指出的是，只有在对CVS非常熟悉的情况下，您才可以使用该操作。该操作的后果之一是，除非“sticky revision”标记被移走(直接使用CVS命令)，否则您不能提交任何进一步的修改。当然，您可使用Replace With | Latest from Repository操作返回到原版本。通常，除非您是一个熟练的CVS用户，否则您可能不会对“sticky revision”标记感兴趣

Open 该操作会在编辑器中打开所选的修订版文件。这时，您是不能对该版本文件作出任何更改的

(续表)

操 作 描 述

Refresh View 该操作会用CVS资源库中的内容来更新CVS Resource History视图

Show Annotation 该操作会使Eclipe切换到CVS Repository Exploring透视图，并使得CVS Annotate视图变为当前焦点以显示出所选文件的注释

Tag with Existing… 该操作会将已有的资源库版本或分支标记应用到所选的资源上。该操作是一个特殊的操作。请查看联机帮助中的Moving Version Tags专题以获得何时使用该操作的详细信息

5.4.5 查看CVS命令

Eclipse Console视图可显示出CVS与Eclipse之间的所有的命令和响应；在您碰到问题的时候，该视图是非常有用的。在Eclipse Console视图中还显示了所发布的供用户使用的任何Eclipse CVS操作的所有CVS命令。在Team | CVS | Console首选项页面中，您可以对该控制台的外观和行为进行配置。

5.4.6 CVS是如何管理工作空间中项目的

图5-6显示了Package Explorer视图中一个处在CVS控制之下的项目。通过观察图标上的标记修饰符以及附加的文字信息，您可以很明显地看出这一点。只有在启用了Workbench | Label Decorations首选项设置后，Eclipse才会在Package Explorer视图中显示附加文字信息。对于那些受版本控制所有资源来说，在它们的资源图标上都有一个很小的磁盘标志修饰符 。对于那些CVS所不知道的、被添加到项目中的文件来说，在这些文件的图标中也有一个很小的标志修饰符 。在如图5-6所示的Package Explorer视图中，紧接着项目名后面的是该项目所属CVS资源库服务器的名称(位于方括号中)。如果该项目是被作为一个分支部分来使用的，那么该分支的名称也会被包含在方括号中。文本修饰符>表明本地资源是一个传出更改(outgoing change)，这就意味着该本地资源已经在本地修改过，并与CVS中相应内容不一致。Package Explorer视图中受CVS控制的文件夹(直至项目文件夹)也具有这个特征。资源名后面的圆括号中所显示的文本表示的是该资源的文件格式(ASCII格式或二进制格式)。我们知道，对于ASCII文件来说，在CVS文件实例和本地文件实例之间的并行修改可被合并。而二进制文件只能被替换。

图5-6 Label Decorations首选项被启用时受CVS控制的项目

在Team | CVS | Label Decorations首选项页面中，您可以对CVS标签修饰的表示和规则进行修改。

5.4.7 在Team快捷菜单中可获得的CVS操作

Team菜单可从任何工作空间资源的快捷菜单中获得。有几个CVS操作是与Team菜单相关的。我们将对使用CVS进行工作维护中最常用的操作进行总结。Team菜单下的具体子菜单随着所选择的是否是一个项目而发生变化。只有在项目没有被CVS或任何其他资源库管理时，所选择的项目才会有一个Share Project…操作。表5-3所提供的是一个所有CVS小组操作的参考列表。其中有些操作可能是您不熟悉的。我们将在本章后面对这些操作进行讨论。通常情况下，您最常用的操作是Share Project…、Synchronize with Repository…、Update…以及Commit…。Share Project操作是一个允许您将项目加到CVS中的综合性向导。从定义CVS资源库连接到提交项目资源之间的每一个步骤都包括在该向导中。

表5-3 Team菜单中的CVS操作

操 作 描 述

Add to .cvsignore… 该操作会将所选的项目文件排除在CVS管理之外。这些被排除的文件不会出现在Synchronize视图之中，也不会被提交给CVS资源库。CVS在您的项目中维护了一个名为.cvsignore的文件。在该文件中会标识出那些被排除的文件实例。一开始，.cvsignore文件是不存在的；在该操作第一次执行时，才会创建.cvsignore文件。您必须将.cvsignore文件添加到版本控制中，并在CVS中维护该文件

Add to Version Control 该操作会将所选文件置于CVS控制之下。典型情况下，在项目中创建新文件时会使用该操作。如果您没有将资源显式地添加到版本控制中，那么在执行提交操作时，Eclipse会对您作出提示

Apply Patch… 该操作将使用由Create Patch操作所创建的补丁文件来修补程序

Branch… 该操作会创建一个分支。这样，项目(或所选资源)可独立于其他开发过程而被修改

Change ASCII/Binary

Property… 执行该操作后会显示Set Keyword Substitution向导。该向导允许您为所选文件选择期望的CVS关键字替换模式。它对CVS如何解释ASCII或二进制文件进行了定义

Commit… 如果没有冲突发生，该操作会使用您的本地更改来更新CVS资源库，并会提示您给出一个注释。如果执行该操作时所选择的操作对象是一个项目或文件夹，那么项目或文件夹中的所有被修改资源都将提交到CVS中。如果CVS中的相应文件取代了所要提交的文件，那么Commit…操作将失败

Create Patch… 该操作会基于本地资源和CVS之间的不同而创建一个补丁文件。其他人可共享该补丁文件。这就允许在资源库之外共享您的工作

Disconnect… 该操作会在所选工作空间项目中删除所有CVS连接信息

(续表)

操 作 描 述

Edit 该操作只可用于那些启用了Watch/Edit(可在项目的CVS属性页面或CVS首选项设置中启用该功能)的项目。该操作会通知CVS服务器，让它知道您正在更新该文件。如果有其他人(一个或多个)正在编辑该文件，那么您将被提示，以就是否继续该操作作出决定。此时，您可忽略该提示并对文件进行修改。但是，在您试图对修改进行提交时可能会碰到冲突。如果您只是想先打开所要编辑的文件，那么该文件会被打开。但是在试图对该文件进行修改时，您会得到相同的提示。如果要使用外部编辑器，那么您首先必须要做的是取消该文件的只读状态，然后再使用外部编辑器打开该文件。在后面，我们会对该操作继续进行讨论

Merge… 该操作会将某一具体分支中的修改合并回该分支之前的代码库中

Restore from

Repository… 对于那些被删除的项目文件来说，如果它们仍存在于项目中，并允许您将它们恢复回原工作空间，则执行该操作后会列出这些被删除的文件。在本章后面，我们将会对该操作进行详细讨论

Share Project… 只有项目才能使用该操作。而且，只有在那些不受资源库管理的项目上才可以看到该操作。执行该操作后会显示向导。在该向导中，您可以完成的任务有：将项目与一个可以使用的资源库相关联，以及提交项目资源。其他资源库提供程序也可能会使用该操作

Show Annotation 执行该操作后将切换到CVS Repository Exploring透视图，并会将焦点集中到CVS Annotate视图以显示出所选文件的注释

Show Editors 该操作适用于那些启用了Watch/Edit(可在项目的CVS属性页面或CVS首选项设置中启用该功能)的项目中的文件夹和文件。在执行该操作后会打开CVS编辑器视图。在该视图中会标识出其他正在编辑该文件或该文件夹中内容的用户。只有那些启用了Watch/Edit的项目才可能显示在CVS编辑器视图中

Show in Repository

History 该操作为所选资源显示CVS Repository History视图

Synchronize with

Repository… 该操作会将所选择的文件与CVS资源库中的对应文件进行比较，并切换到Team Synchronizing透视图。比较后所发现的任何差异都会显示在Synchronize视图中。对于项目或文件夹来说，在执行该操作时，它们里面所包含的全部资源都将与CVS中的对应资源进行比较。而那些与CVS中对应资源相比发生了变化的资源会被显示在Synchronize视图中。在您检查所有更改、与资源库实例进行比较，以及解决冲突时，这是一种最安全的CVS使用方法

Tag as Version… 该操作会为所选项目、文件夹(或文件)以及所有的子文件资源的所有CVS实例添加版本标志。您也可以在CVS Repositories视图中进行版本标识

(续表)

操 作 描 述

Unedit 该操作只可用于那些启用了Watch/Edit(可在项目的CVS属性页面或CVS首选项设置中启用该功能)的项目文件。该操作会将您从当前正在编辑某一文件的那个用户列表中删除掉。执行该操作后，您会从当前活动用户列表中被删除，并且所做的任何更改都会被取消，而所编辑的文件会恢复到执行任何更新之前的内容。在向CVS提交文件时，如果工作空间副本与服务器副本完全相同，那么Unedit操作会被隐式调用。在本章后面部分，我们还将对该操作进行讨论

Update… 该操作会使用CVS中所有最新版本来更新本地资源。CVS中的更新将与本地资源(二进制文件除外)进行合并。建议您在使用Commit…操作前执行Update…操作。如果其他人在您的最近一次更新后又进行了提交操作，那么您的提交操作将会失败。建议您用Synchronize with Repository…操作来代替Update…操作

5.4.8 Team Synchronizing透视图和Synchronize向导

Team Synchronizing透视图是您对工作空间与CVS(其他资源库提供程序可被集成到Team Synchronizing透视图中)进行同步的大本营。在选择了Team | Synchronize with Repository…操作后，您会被提示并切换到Team Synchronizing透视图。在Team Synchronizing透视图中含有Synchronize视图和一个编辑器区域。所有被打开的编辑器都会显示在该编辑器区域中。图5-7所显示的就是一个Team Synchronizing透视图的例子。在Synchronize视图中，您可以对本地项目和该项目的CVS副本之间的差别进行观察和操作。其他想使用该透视图的资源库提供程序也可使用该视图。在打开一个文件的同时，Eclipse也会打开一个适当的比较编辑器(文本、Java、XML或图像)。在比较编辑器中会显示出工作空间文件和CVS中该文件的最新版本之间的差别。这样，您就可以在将文件提交到CVS之前浏览和最终确定该文件的内容。

在Synchronize视图的工具栏上有如下一组按钮：Incoming Mode (从CVS中更新)、Outgoing Mode (提交到CVS)、Incoming and Outgoing Mode 以及Conflicts Mode 。通过使用上述按钮，您可以对同步信息进行过滤。在使用了Incoming Mode按钮后，您会看到有哪些操作正在进行。如果小组中只有您一个人，那么可能只有Outgoing Mode按钮才是您感兴趣的。若要接受所有的传入更改，则使用Synchronize视图工具栏上的Update All Incoming Changes…按钮 。若要提交您的所有更改，则使用Synchronize视图工具栏上的Commit All Outgoing Changes…按钮 。工具栏操作Pin Current Synchronization 允许您对先前同步中所使用的一组资源进行引用。Synchronize视图的工具栏上的这些按钮允许您通过工作集来过滤内容。您可对比较标准进行调整，并可使用Layout菜单在树、压缩文件夹以及提交集(Commit Set)之间进行视图切换。提交集允许您查看资源库更改。这些更改会按照提交者、注释和提交日期来分类显示。在确定传入更改集的更详细信息时，这种分类非常有用。Synchronize视图拥有自己的Preferences对话框。在Synchronize视图的菜单中，您可以打开该Preferences对话框。在Synchronize视图的菜单中，您还可以执行Schedule…操作。该操作允许您按照所选择的某一重复进度来自动执行资源库的同步操作。

图5-7 显示传入更改和传出冲突文件三方比较的Team Synchronizing透视图

1. Synchronize向导

从工作台工具栏(或Synchronize视图菜单栏)的下拉菜单中选择Synchronize…操作 后会出现Synchronize向导。在Synchronize向导中，您可以对要与CVS(以及其他那些被允许使用Eclipse小组同步支持的资源库提供程序)进行同步的项目进行选择。如图5-8所示，您可对工作空间、所选资源或者一个工作集进行同步。在Synchronize向导结束后，您会返回到Team Synchronizing透视图。

5.4.9 处理同一文件的并发更新

首先，我们讨论一下一个被建议用来进行修改操作的协议。当在工作空间中使用资源时，您的操作是独立于CVS资源库的。因为发生在资源库中的更改可能是在您不知道的情况下发生的，所以非常重要的一点是，如果还未检查资源库中是否有更新取代了您的更改，那么您一定不要提交任何更改。Team | Update…操作会用资源库中那些取代了您的更改的任何更改来更新项目的本地副本。对于文本文件(包括Java类)来说，Team | Update…操作会自动对任何更改进行合并(从资源库到工作空间)，即使这些更改可能造成冲突。在遇到冲突时，合并操作会用特定的CVS标记文本来标识文件中的冲突，以帮助您识别出冲突行。标记文本并不是与任何类型的文件都兼容，例如它可能导致编译错误。

Update…操作是一个非常强大的操作，我们在练习时一定要小心。在下面这个例子中，Pat和Lynn更新了一个简单的文本文件。无需知道它，它们都对以“B”和“C”开始的行进行了更新。标记文本会对冲突的那些行以及引入了冲突的版本号(1.2)进行了标识。对于一个带有众多冲突的复杂Java文件来说，您可能很难解决其中的冲突。

A is for apple

<<<<<<< sample.txt

B is for bird (updated by pat)

C is for crow (updated by pat)

=======

B is for bobcat (updated by lynn)

C is for cow (updated by lynn)

>>>>>>> 1.2

D is for dog

E is for excellent

F is for farm

G is for goat...

Team | Synchronize with Repository…操作也支持更新，但是该操作不会自动进行更新。在执行该操作后会显示Synchronize视图，并会列出工作空间副本和资源库最新资源之间那些不一致的资源。在Synchronize视图中，您可以对这些差别进行检查，并决定所要采取的操作。由于Team | Synchronize with Repository…操作可使您拥有更多的控制权，所以我们建议您使用该操作来代替Update…操作。

下面总结一下文件并发更新的基本规则：在对工作空间中的文件进行修改之前，您必须用资源库中的所有更改来更新工作空间中的资源。为了安全起见，此时请使用Team | Synchronize with Repository…操作。在Synchronize视图中，您可执行更新操作，还可提交更改，并可协调冲突。

5.4.10 使用CVS的Watch/Edit支持以避免对同一文件的并行更新

CVS的Watch/Edit支持是Eclipse本身就提供的。如果您同意使用CVS的Watch/Edit支持，那么在提交资源时，您和小组其他成员之间就可以避免冲突。在启用了CVS的Watch/Edit支持之后，您的项目小组可以更好地理解在所有具体项目中，当前谁正更新文件。相反，前面所讨论的CVS Annotate视图为您所显示的是先前的更改以及谁做的更改。CVS的Watch/Edit支持是在项目层次上被启用的。如果在Team | CVS >Watch/Edit的首选项设置页面中选中了Configure projects to use Watch/Edit on checkout复选框，那么在项目被检出时，Watch/Edit支持将被自动激活。您也可以通过设置项目的CVS属性在受CVS管理的现有项目上启用Watch/Edit支持。在CVS的Watch/Edit支持被启用时，该项目中的文件处于只读状态。此时，如果您在编辑器中打开了某文件后又试图对该文件进行修改，那么CVS服务器会被询问。在编辑文件时，如果其他人也正在编辑该文件，那么就会出现一个对话框以通知您小组中其他成员也正在编辑该文件(见图5-9)。这时，您可以继续对该文件进行编辑，但是在您提交该文件时需要进行冲突管理。在文件被修改时，检查标记修饰符 会被添加到该文件上。

图5-9 显示其他人正在编辑同一文件的对话框

在项目、文件夹以及文件上执行Team | Show Editors操作后会打开CVS Editors视图。CVS Editors视图中列出的是所有正在被其他人更新的资源(见图5-10)。无论是否处于Watch/Edit的控制之下，您在任何项目及项目资源上都可以使用Team | Show Editors操作。但是Team | Show Editors操作只会对启用了Watch/Edit支持的项目资源进行标识。

图5-10 列出了哪个用户正在更新文件夹中文件的CVS Editors视图

您可以使用Team | Edit操作来显式地通知CVS服务器，让CVS服务器知道您想对文件进行更新。您还可以使用Team | Unedit操作来通知CVS服务器，让CVS服务器知道您不想再编辑该文件。这样，您所做的任何更改都会被从CVS资源库中检出。表5-3给出了有关这些操作的信息。在向CVS提交了更改之后，Watch/Edit列表中的资源会被隐式删除掉。

Watch/Edit是一个非常有用的工具。但是该工具的使用需要遵循一些规则，并要注意一些考虑事项。首先，如果要使Watch/Edit功能高效，那么小组中的每个成员都必须在存在潜在冲突的项目中启动Watch/Edit功能。其次，您必须连接到CVS服务器上。如果您没有和CVS服务器相连，并且想更新一个文件，那么您将不得不显式地将该文件的只读状态(从该文件的属性页面中可获得该属性)取消掉。当然，该支持现在是折中的。这是因为，没有任何人知道该更新，而且在提交过程中存在潜在的冲突。如果您删除了一个项目并且该项目资源的Watch/Edit首选项设置没有被启用，那么CVS服务器将不会被通知，并且CVS服务器会继续向外报告您正在对这些资源进行编辑。如果您的小组计划使用Watch/Edit支持，那么明智的做法是小组中所有人都启用被标注为Configure projects to use Watch/Edit on checkout的CVS Watch/Edit首选项设置。

5.4.11 恢复被删除的文件

有时候，在文件被删除并且更改被提交给CVS后，您可能需要恢复所删除的文件。幸运的是，CVS不会将所删除的文件从该文件所在的资源库中移除。您可以使用Team | Restore from Repository操作来恢复所删除的文件。如果被删除的文件在资源库中存在，那么您将看到一个对话框。在该对话框中，您可以选择所要恢复的文件的具体修订版(见图5-11)。而文件的最新修订版则是一个对所删除文件的引用。这是因为文件删除是一个CVS修订操作。

图5-11 显示可恢复的被删除文件的Restore from Repository对话框

当一个文件被恢复到工作空间中的时候，该文件被看作是一个新文件。在提交该文件之前或提交该文件时，必须将该新文件添加到版本控制中。

5.4.12 其他CVS用户界面元素

1. Checkout向导

在New wizard对话框中获得的是CVS向导中的检出项目。通常，您会使用CVS Repositories视图来检出项目。但是，有些CVS资源库是不允许被浏览的。在这种情况下，检出向导可能就比较有用。但是在这种情况下，您必须预先知道该项目的名称。

2. CVS备忘单

若要打开Cheat Sheet Selection对话框，则选择Help | Cheat Sheets…命令。对于CVS任务来说，备忘单用来为分支和合并操作提供帮助。