

排除Cisco CVP Call Studio日志故障

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[问题：从何处获取Call Studio安装日志](#)

[解决方案](#)

[问题：从何处获取Call Studio许可证日志](#)

[解决方案](#)

[问题：从何处获取Call Studio应用程序的Call Studio错误日志](#)

[解决方案](#)

[问题：从何处获取Call Studio日志以进行应用验证](#)

[解决方案](#)

简介

本文档介绍要进行故障排除的常见Cisco Call Studio场景以及日志的获取位置。

先决条件

要求

Cisco 建议您了解以下主题：

- 思科Call Studio

使用的组件

本文档中的信息基于Cisco Call Studio 11.6和/或12.0。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

问题：从何处获取Call Studio安装日志

解决方案

可在C:\Temp目录中找到Call Studio安装日志。

以下是新安装日志的示例片段：

```
[10-20-2020 08:52:32] Informational: Starting Call Studio installation procedure... [10-20-2020 08:52:32] Informational: Attempting to read version file:
D:\CallStudio\Installer_Windows\VERSION [10-20-2020 08:52:32] Informational: Welcome to the installation of the Cisco Unified Call Studio, Release 12.0.1. [10-20-2020 08:52:34] Informational: User has passed the Copyrights notification screen. [10-20-2020 08:52:37] Informational: User has accepted the license agreement. [10-20-2020 08:52:38] Informational: User selected directory: C:\Cisco\CallStudio [10-20-2020 08:52:38] Informational: User has selected target directory = C:\Cisco\CallStudio [10-20-2020 08:52:39] Informational: LaunchApp - success running: Setting CALLSTUDIO_HOME environmental variable through setx:
C:\Windows\system32\setx.exe [10-20-2020 08:52:39] Informational: Installing the Vc++ redistributable exe from the path: C:\Users\ADMINI~1\AppData\Local\Temp\3\{7F6E4651-C47A-4607-8E9D-64A8B21F688F}\{763E1DF9-41BC-4C54-9705-A0C6D1594B26}\vcredist_x64.exe [10-20-2020 08:52:41] WARNING: LaunchApp - Error 5100 while running: Attempting to silently install Vcredist_x64 using cmd=
```

问题：从何处获取Call Studio许可证日志

解决方案

Call Studio许可证仅在版本11.6之前有效。从12.0及以后，Call Studio不需要任何许可证。

您可以在此处获取许可证日志：

以下是示例日志文件：**C:\Cisco\CallStudio\eclipse\workspace\metadata**

```
!SUBENTRY 1 com.audiumcorp.studio.core 4 4 2019-07-24 14:33:13.797 !MESSAGE This installation of Call Studio cannot be used until it has been activated. If you installed Call Studio more than 30 days ago, you are now required to activate it to use it. To activate, restart Call Studio, enter your licensing information and click on the 'Activate >>' button. !STACK 0
com.audiumcorp.license.client.LicenseException: This installation of Call Studio cannot be used until it has been activated. If you installed Call Studio more than 30 days ago, you are now required to activate it to use it. To activate, restart Call Studio, enter your licensing information and click on the 'Activate >>' button. at
com.audiumcorp.studio.core.license.LicenseManager.initialize(LicenseManager.java:184) at
com.audiumcorp.studio.core.CorePlugin.validateLicense(CorePlugin.java:325) at
com.audiumcorp.studio.core.CorePlugin.start(CorePlugin.java:308) at
org.eclipse.osgi.framework.internal.core.BundleContextImpl$1.run(BundleContextImpl.java:711) at
java.security.AccessController.doPrivileged(Native Method) at
org.eclipse.osgi.framework.internal.core.BundleContextImpl.startActivator(BundleContextImpl.java:702) at
org.eclipse.osgi.framework.internal.core.BundleContextImpl.start(BundleContextImpl.java:683) at
org.eclipse.osgi.framework.internal.core.BundleHost.startWorker(BundleHost.java:381) at
org.eclipse.osgi.framework.internal.core.AbstractBundle.start(AbstractBundle.java:299) at
org.eclipse.osgi.framework.util.SecureAction.start(SecureAction.java:440) at
org.eclipse.osgi.internal.loader.BundleLoader.setLazyTrigger(BundleLoader.java:268) at
org.eclipse.core.runtime.internal.adaptor.EclipseLazyStarter.postFindLocalClass(EclipseLazyStarter.java:107) at
org.eclipse.osgi.baseadaptor.loader.ClasspathManager.findLocalClass(ClasspathManager.java:463) at
org.eclipse.osgi.internal.baseadaptor.DefaultClassLoader.findLocalClass(DefaultClassLoader.java:216) at
org.eclipse.osgi.internal.loader.BundleLoader.findLocalClass(BundleLoader.java:400) at
org.eclipse.osgi.internal.loader.SingleSourcePackage.loadClass(SingleSourcePackage.java:35) at
org.eclipse.osgi.internal.loader.BundleLoader.findClassInternal(BundleLoader.java:473) at
org.eclipse.osgi.internal.loader.BundleLoader.findClass(BundleLoader.java:429) at
org.eclipse.osgi.internal.loader.BundleLoader.findClass(BundleLoader.java:417) at
org.eclipse.osgi.internal.baseadaptor.DefaultClassLoader.loadClass(DefaultClassLoader.java:107) at
java.lang.ClassLoader.loadClass(ClassLoader.java:358) at
com.audiumcorp.studio.builder.core.util.BuilderResources.isBuilderSupported(BuilderResources.jav
```

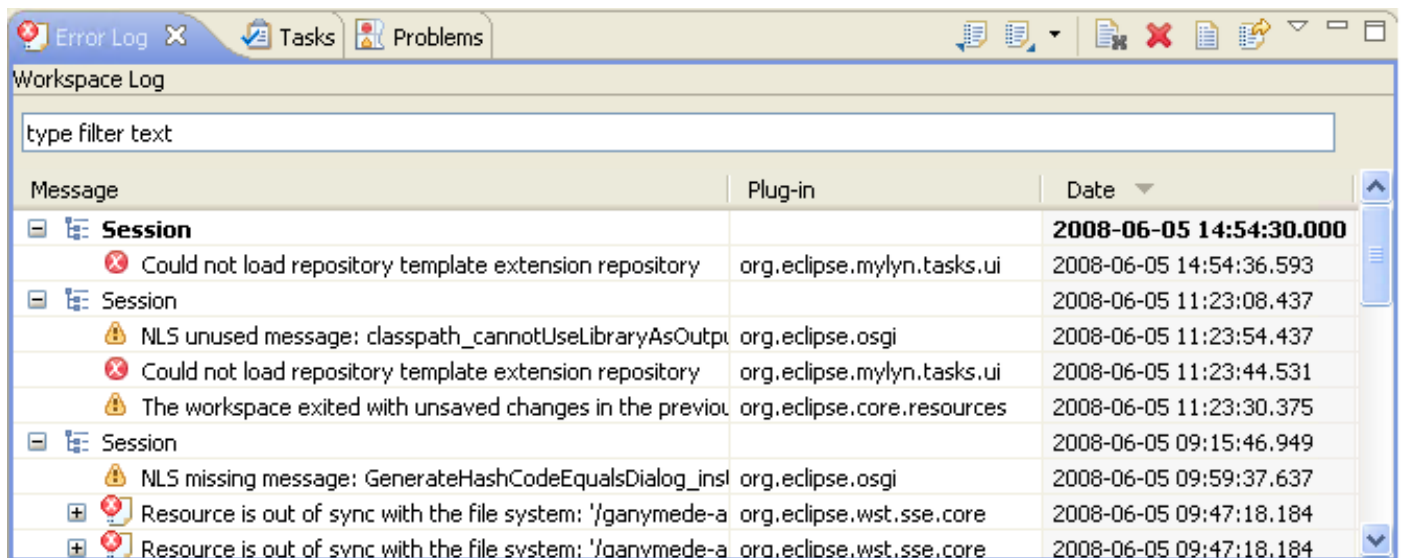
a:138) at com.audiumcorp.studio.builder.core.CorePlugin.validateLicense(CorePlugin.java:74) at
com.audiumcorp.studio.builder.core.CorePlugin.start(CorePlugin.java:60) at
org.eclipse.osgi.framework.internal.core.BundleContextImpl\$1.run(BundleContextImpl.java:711) at
java.security.AccessController.doPrivileged(Native Method) at
org.eclipse.osgi.framework.internal.core.BundleContextImpl.startActivator(BundleContextImpl.java
:702) at
org.eclipse.osgi.framework.internal.core.BundleContextImpl.start(BundleContextImpl.java:683) at
org.eclipse.osgi.framework.internal.core.BundleHost.startWorker(BundleHost.java:381) at
org.eclipse.osgi.framework.internal.core.AbstractBundle.start(AbstractBundle.java:299) at
org.eclipse.osgi.framework.util.SecureAction.start(SecureAction.java:440) at
org.eclipse.osgi.internal.loader.BundleLoader.setLazyTrigger(BundleLoader.java:268) at
org.eclipse.core.runtime.internal.adaptor.EclipseLazyStarter.postFindLocalClass(EclipseLazyStart
er.java:107) at
org.eclipse.osgi.baseadaptor.loader.ClasspathManager.findLocalClass(ClasspathManager.java:463)
at
org.eclipse.osgi.internal.baseadaptor.DefaultClassLoader.findLocalClass(DefaultClassLoader.java:
216) at org.eclipse.osgi.internal.loader.BundleLoader.findLocalClass(BundleLoader.java:400) at
org.eclipse.osgi.internal.loader.SingleSourcePackage.loadClass(SingleSourcePackage.java:35) at
org.eclipse.osgi.internal.loader.BundleLoader.findClassInternal(BundleLoader.java:473) at
org.eclipse.osgi.internal.loader.BundleLoader.findClass(BundleLoader.java:429) at
org.eclipse.osgi.internal.loader.BundleLoader.findClass(BundleLoader.java:417) at
org.eclipse.osgi.internal.baseadaptor.DefaultClassLoader.loadClass(DefaultClassLoader.java:107)
at java.lang.ClassLoader.loadClass(ClassLoader.java:358) at
java.lang.Class.getDeclaredConstructors0(Native Method) at
java.lang.Class.privateGetDeclaredConstructors(Class.java:2493) at
java.lang.Class.getConstructor0(Class.java:2803) at java.lang.Class.newInstance(Class.java:345)
at
org.eclipse.core.internal.registry.osgi.RegistryStrategyOSGI.createExecutableExtension(RegistryS
trategyOSGI.java:184) at
org.eclipse.core.internal.registry.ExtensionRegistry.createExecutableExtension(ExtensionRegistry
.java:905) at
org.eclipse.core.internal.registry.ConfigurationElement.createExecutableExtension(ConfigurationE
lement.java:243) at
org.eclipse.core.internal.registry.ConfigurationElementHandle.createExecutableExtension(Configur
ationElementHandle.java:55) at
org.eclipse.ui.internal.WorkbenchPlugin\$1.run(WorkbenchPlugin.java:268) at
org.eclipse.swt.custom.BusyIndicator.showWhile(BusyIndicator.java:70) at
org.eclipse.ui.internal.WorkbenchPlugin.createExtension(WorkbenchPlugin.java:264) at
org.eclipse.ui.internal.registry.EditorDescriptor.createEditor(EditorDescriptor.java:235) at
org.eclipse.ui.internal.EditorManager.createPart(EditorManager.java:875) at
org.eclipse.ui.internal.EditorReference.createPartHelper(EditorReference.java:609) at
org.eclipse.ui.internal.EditorReference.createPart(EditorReference.java:465) at
org.eclipse.ui.internal.WorkbenchPartReference.getPart(WorkbenchPartReference.java:595) at
org.eclipse.ui.internal.EditorAreaHelper.setVisibleEditor(EditorAreaHelper.java:271) at
org.eclipse.ui.internal.EditorManager.setVisibleEditor(EditorManager.java:1459) at
org.eclipse.ui.internal.EditorManager\$5.runWithException(EditorManager.java:972) at
org.eclipse.ui.internal.StartupThreading\$StartupRunnable.run(StartupThreading.java:31) at
org.eclipse.swt.widgets.RunnableLock.run(RunnableLock.java:35) at
org.eclipse.swt.widgets.Synchronizer.runAsyncMessages(Synchronizer.java:135) at
org.eclipse.swt.widgets.Display.runAsyncMessages(Display.java:4140) at
org.eclipse.swt.widgets.Display.readAndDispatch(Display.java:3757) at
org.eclipse.ui.application.WorkbenchAdvisor.openWindows(WorkbenchAdvisor.java:803) at
org.eclipse.ui.internal.Workbench\$33.runWithException(Workbench.java:1600) at
org.eclipse.ui.internal.StartupThreading\$StartupRunnable.run(StartupThreading.java:31) at
org.eclipse.swt.widgets.RunnableLock.run(RunnableLock.java:35) at
org.eclipse.swt.widgets.Synchronizer.runAsyncMessages(Synchronizer.java:135) at
org.eclipse.swt.widgets.Display.runAsyncMessages(Display.java:4140) at
org.eclipse.swt.widgets.Display.readAndDispatch(Display.java:3757) at
org.eclipse.ui.internal.Workbench.runUI(Workbench.java:2609) at
org.eclipse.ui.internal.Workbench.access\$4(Workbench.java:2499) at
org.eclipse.ui.internal.Workbench\$7.run(Workbench.java:679) at
org.eclipse.core.databinding.observable.Realm.runWithDefault(Realm.java:332) at
org.eclipse.ui.internal.Workbench.createAndRunWorkbench(Workbench.java:668) at
org.eclipse.ui.PlatformUI.createAndRunWorkbench(PlatformUI.java:149) at

```
org.eclipse.ui.internal.ide.application.IDEApplication.start(IDEApplication.java:123) at
org.eclipse.equinox.internal.app.EclipseAppHandle.run(EclipseAppHandle.java:196) at
org.eclipse.core.runtime.internal.adaptor.EclipseAppLauncher.runApplication(EclipseAppLauncher.j
ava:110) at
org.eclipse.core.runtime.internal.adaptor.EclipseAppLauncher.start(EclipseAppLauncher.java:79)
at org.eclipse.core.runtime.adaptor.EclipseStarter.run(EclipseStarter.java:344) at
org.eclipse.core.runtime.adaptor.EclipseStarter.run(EclipseStarter.java:179) at
sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method) at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57) at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) at
java.lang.reflect.Method.invoke(Method.java:606) at
org.eclipse.equinox.launcher.Main.invokeFramework(Main.java:622) at
org.eclipse.equinox.launcher.Main.basicRun(Main.java:577) at
org.eclipse.equinox.launcher.Main.run(Main.java:1410) at
org.eclipse.equinox.launcher.Main.main(Main.java:1386)
```

问题：从何处获取Call Studio应用程序的Call Studio错误日志

解决方案

“错误日志”视图捕获插件记录的所有警告和错误。基础日志文件是存储在工作区的.metadata子目录中的.log文件。“错误日志”视图位于“窗口”>“显示视图”>“错误日志”下。



事件排序

日志视图中的事件可按消息、插件ID或日期按升序或降序排序。只需单击要排序所基于的列标题。列标题中的向下箭头表示降序；而向上箭头则表示升序。

事件分组

日志视图中的事件可以按会话或插件ID分组。只需单击视图工具栏上的V形，然后选择“分组依据”。

事件过滤

您可以过滤视图以显示特定类型或会话的事件。此外，您还可以限制视图中的条目数。从视图的工

具栏下拉菜单的“过滤器.....”(Filters...)下方提供过滤选项。

导入和导出日志

要将任意.log文件导入到视图中，请按导入日志工具栏按钮或从上下文菜单中选择导入日志.....。然后，从文件系统中选择.log文件。

要将当前日志视图内容导出到文件中，请按“导出日志”工具栏按钮，或从上下文菜单中选择“导出日志.....”。然后，输入文件名。

清除和删除日志


要清除视图日志内容而不删除基础.log文件，请按清除日志工具栏按钮或从上下文菜单中选择清除日志查看器。

要永久删除基础.log文件，请按“删除日志”工具栏按钮，或从上下文菜单中选择“删除日志”。

事件详细信息

通过双击某个特定条目或从该条目的上下文菜单中选择“事件详细信息”(Event Details)，可在“事件详细信息”(Event Details)对话框中查看有关特定事件的完整详细信息。可以查看每个事件的日期、严重性、消息、异常堆栈跟踪（如果可用）和会话数据。

Date: 2008-06-04 16:12:58.727

Severity:  Error

Message: Could not load library: jWinHttp.dll

Exception Stack Trace:

```
java.lang.UnsatisfiedLinkError: C:\eclipses\eclipse-SDK-3.4RC3-win32\eclipse\configuration\
at java.lang.ClassLoader$NativeLibrary.load(Native Method)
at java.lang.ClassLoader.loadLibrary0(Unknown Source)
at java.lang.ClassLoader.loadLibrary(Unknown Source)
at java.lang.Runtime.loadLibrary0(Unknown Source)
at java.lang.System.loadLibrary(Unknown Source)
at org.eclipse.core.internal.net.WindowsProxyProvider.<clinit>(WindowsProxyProvider.java:
at java.lang.Class.forName0(Native Method)
at java.lang.Class.forName(Unknown Source)
at org.eclipse.core.internal.net.ProxyManager.<init>(ProxyManager.java:73)
at org.eclipse.core.internal.net.ProxyManager.getProxyManager(ProxyManager.java:88)
at org.eclipse.core.internal.net.Activator.start(Activator.java:178)
at org.eclipse.osgi.framework.internal.core.BundleContextImpl$2.run(BundleContextImpl.java:
at java.security.AccessController.doPrivileged(Native Method)
at org.eclipse.osgi.framework.internal.core.BundleContextImpl.startActivator(BundleConte:
at org.eclipse.osgi.framework.internal.core.BundleContextImpl.start(BundleContextImpl.java:
at org.eclipse.osgi.framework.internal.core.BundleHost.startWorker(BundleHost.java:346)
at org.eclipse.osgi.framework.internal.core.AbstractBundle.start(AbstractBundle.java:265)
at org.eclipse.osgi.framework.util.SecureAction.start(SecureAction.java:400)
```

Session Data:

```
eclipse.buildId=I20080530-1730
java.version=1.6.0_06
```

您可以通过向上和向下箭头按钮从一个条目导航到下一个条目。

要将错误复制到剪贴板，请按带剪贴板图像的按钮。

问题：从何处获取Call Studio日志以进行应用验证

解决方案

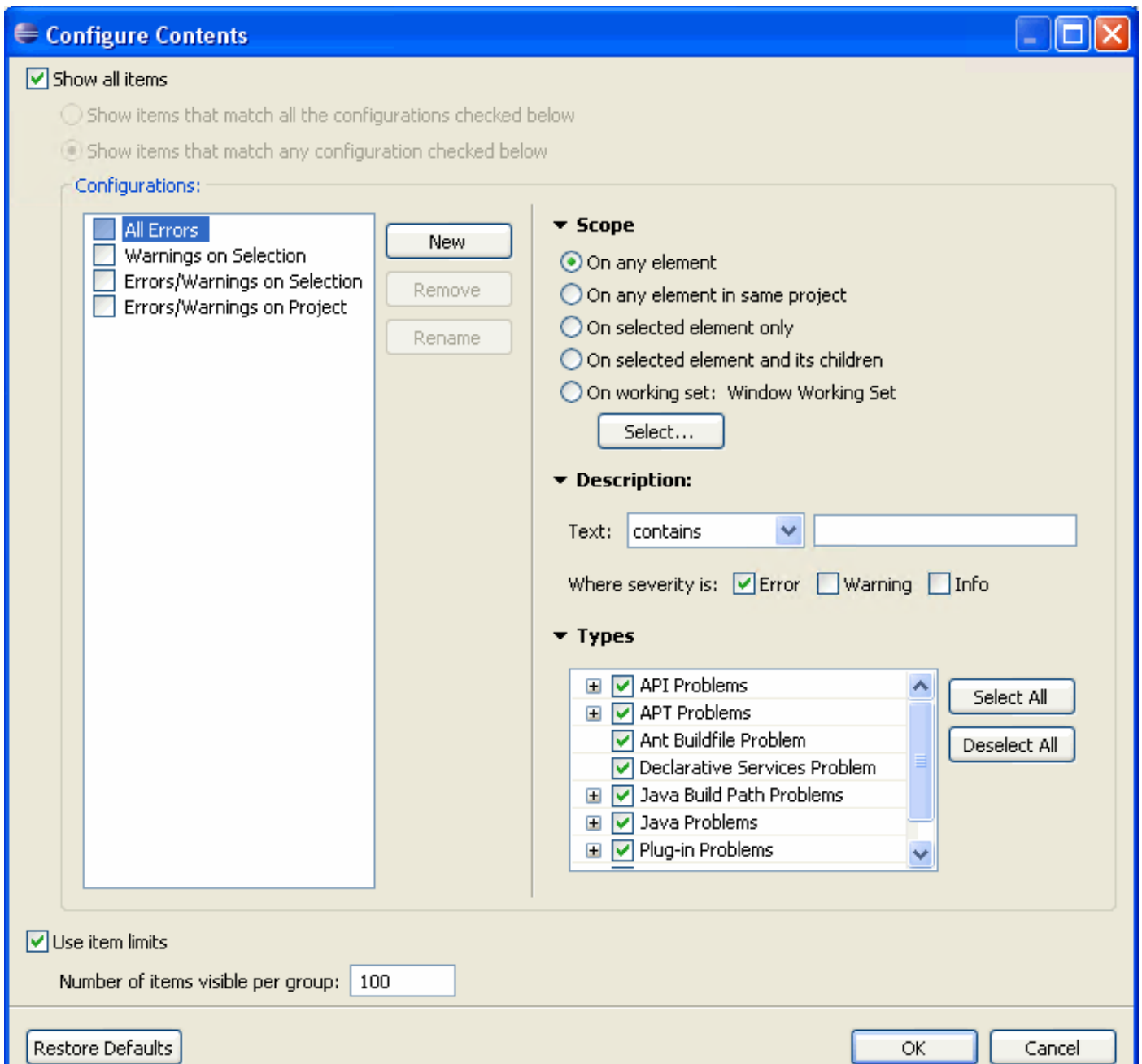
在工作台中处理资源时，各构建商可以在“问题”视图中自动记录问题、错误或警告。例如，当保存包含语法错误的Java源文件时，这些错误将记录在“问题”视图中。当您双击问题、错误或警告的图标时，关联资源的编辑器会自动打开到相关代码行。

Description	Resource	Path	Location	Type
Fatal Errors (1 item)				
The import java.net.Bug cannot be resolved	Logo.java	/ScreenShots/sr...	line 6	Java Problem
Potential Programming Problems (21 items)				
The argument of type Class[] should explicitly be cast	BaseTestRunner.java	/ScreenShots/sr...	line 111	Java Problem
The argument of type Class[] should explicitly be cast	TestCase.java	/ScreenShots/sr...	line 154	Java Problem
The argument of type Class[] should explicitly be cast	TestCaseClassLoaderTest.j...	/ScreenShots/sr...	line 25	Java Problem

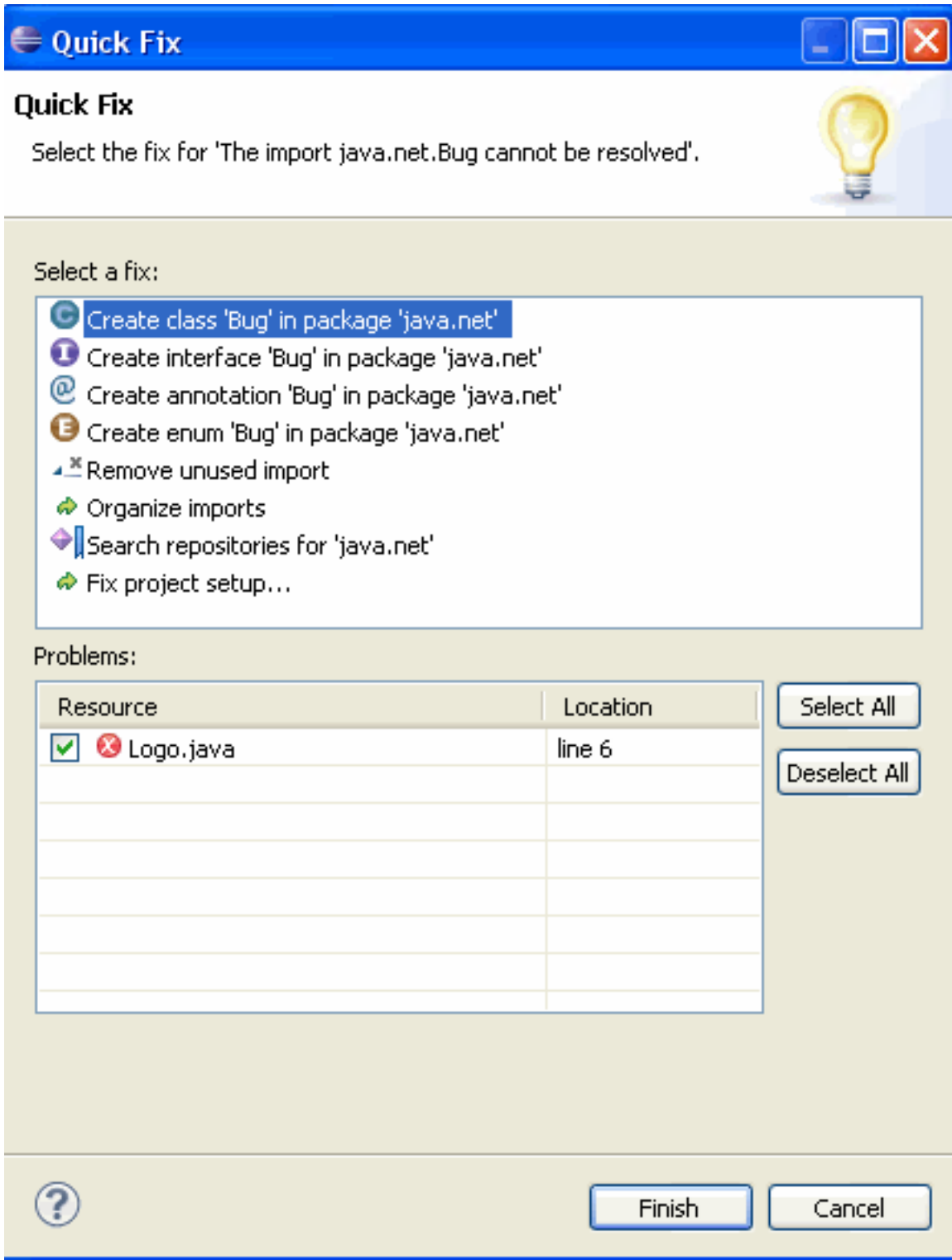
默认情况下，“问题”视图按严重性对问题进行分组。您也可以按类型或根本不按类型对它们进行分组。某些组件会添加自己的分组。例如，Java开发工具(JDT)支持会添加Java问题类型组。可使用“分组依据”(Group By)菜单选择分组。

“问题”视图的第一列显示一个图标，该图标表示行项目的类型、类别和说明。左键单击项目以在编辑器中打开文件，并突出显示包含问题的行。

您可以配置“问题”视图的内容，以仅查看与特定资源或资源组关联的警告和错误。使用下拉菜单中的“配置内容”对话框完成。您可以向“问题”视图添加多个过滤器，并根据需要启用或禁用这些过滤器。过滤器可以是加性的（将显示满足至少一个已启用过滤器的任何问题），也可以是排它的（仅显示满足所有过滤器的问题）。默认情况下，提供两个最常用的过滤器(All Errors and Warnings on Selection)。



从上下文菜单中选择“快速修复”(Quick Fix)可以解决问题。列出了可能的解决方案。



要将“问题”视图添加到当前透视中，请单击“窗口”>“显示视图”>“其他.....”>“常规”>“问题”。