

# Risoluzione dei problemi di Cisco CVP Call Studio con i log

## Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Problema: Dove ottenere i log di installazione di Call Studio](#)

[Soluzione](#)

[Problema: Dove ottenere i log delle licenze di Call Studio](#)

[Soluzione](#)

[Problema: Dove ottenere i log degli errori di Call Studio per le applicazioni Call Studio](#)

[Soluzione](#)

[Problema: Dove ottenere i log di Call Studio per la convalida delle applicazioni](#)

[Soluzione](#)

## Introduzione

Questo documento descrive gli scenari comuni di Cisco Call Studio da risolvere e dove ottenere i log relativi.

## Prerequisiti

### Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Cisco Call Studio

### Componenti usati

Il riferimento delle informazioni contenute in questo documento è Cisco Call Studio 11.6 e/o 12.0.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

## Problema: Dove ottenere i log di installazione di Call Studio

## Soluzione

I log di Call Studio Install sono disponibili nella directory **C:\Temp**.

Di seguito è riportato un frammento di log di esempio per una nuova installazione:

```
[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=
```

## Problema: Dove ottenere i log delle licenze di Call Studio

### Soluzione

La licenza di Call Studio è valida solo fino alla versione 11.6. A partire dalla versione 12.0, Call Studio non richiede alcuna licenza.

I log delle licenze possono essere visualizzati qui:

Di seguito è riportato un esempio di file registro: **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(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  
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)

```

## Problema: Dove ottenere i log degli errori di Call Studio per le applicazioni Call Studio

### Soluzione

La **visualizzazione Registro errori** consente di acquisire tutti gli avvisi e gli errori registrati dai plug-in. Il file di log sottostante è un file con estensione log archiviato nella sottodirectory dei metadati dell'area di lavoro. La vista Log degli errori è disponibile in **Finestra > Mostra vista > Log degli errori**.

Message	Plug-in	Date
<b>Session</b>		<b>2008-06-05 14:54:30.000</b>
✘ Could not load repository template extension repository	org.eclipse.mylyn.tasks.ui	2008-06-05 14:54:36.593
<b>Session</b>		2008-06-05 11:23:08.437
⚠ NLS unused message: classpath_cannotUseLibraryAsOutput	org.eclipse.osgi	2008-06-05 11:23:54.437
✘ Could not load repository template extension repository	org.eclipse.mylyn.tasks.ui	2008-06-05 11:23:44.531
⚠ The workspace exited with unsaved changes in the previous	org.eclipse.core.resources	2008-06-05 11:23:30.375
<b>Session</b>		2008-06-05 09:15:46.949
⚠ NLS missing message: GenerateHashCodeEqualsDialog_inst	org.eclipse.osgi	2008-06-05 09:59:37.637
✘ Resource is out of sync with the file system: 'ganymede-a	org.eclipse.wst.sse.core	2008-06-05 09:47:18.184
✘ Resource is out of sync with the file system: 'ganymede-a	org.eclipse.wst.sse.core	2008-06-05 09:47:18.184

### Ordinamento eventi

Gli eventi nella visualizzazione registro possono essere ordinati in base al **messaggio**, all'**ID del plug-in** o alla **data** in ordine crescente o decrescente. È sufficiente fare clic sull'intestazione di colonna su cui si desidera basare l'ordinamento. La freccia in giù nell'intestazione di colonna

indica l'ordine decrescente; mentre la freccia in su indica un ordine crescente.

## Raggruppamento eventi

Gli eventi nella visualizzazione registro possono essere raggruppati in base all'**ID sessione** o **plugin**. È sufficiente fare clic sulle frecce nella barra degli strumenti della visualizzazione e selezionare **Raggruppa per**.

## Filtro eventi

È possibile filtrare la visualizzazione per visualizzare gli eventi di un determinato tipo o sessione. È inoltre possibile limitare il numero di voci nella visualizzazione. Le opzioni di filtro sono disponibili in **Filtri...** dal menu a discesa della barra degli strumenti della visualizzazione.

## Importazione ed esportazione di log

Per **importare** un file *.log* arbitrario nella vista, premere il pulsante della barra degli strumenti **Importa log** o selezionare **Importa log...** dal menu di scelta rapida. Scegliere quindi un file *.log* dal file system.

Per **esportare** il contenuto della visualizzazione del log corrente in un file, premere il pulsante **Esporta log** sulla barra degli strumenti o selezionare **Esporta log...** dal menu di scelta rapida. Immettere un nome di file.


## Cancella ed elimina registri

Per **cancellare** il contenuto del log di visualizzazione senza eliminare il file *.log*, premere il pulsante **Cancella log** sulla barra degli strumenti o selezionare **Cancella Log Viewer** dal menu di scelta rapida.

Per **eliminare** in modo permanente il file *.log* sottostante, premere il pulsante **Elimina registro** sulla barra degli strumenti o selezionare **Elimina registro** dal menu di scelta rapida.

## Dettagli evento

I dettagli completi relativi a un evento specifico possono essere visualizzati nella finestra di dialogo **Dettagli evento** facendo doppio clic su una voce specifica oppure selezionando **Dettagli evento** dal menu di scelta rapida della voce. È possibile visualizzare la data, la **gravità**, il **messaggio**, la **traccia dello stack di eccezioni** (se disponibile) e i **dati della sessione** di ogni evento.

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
```

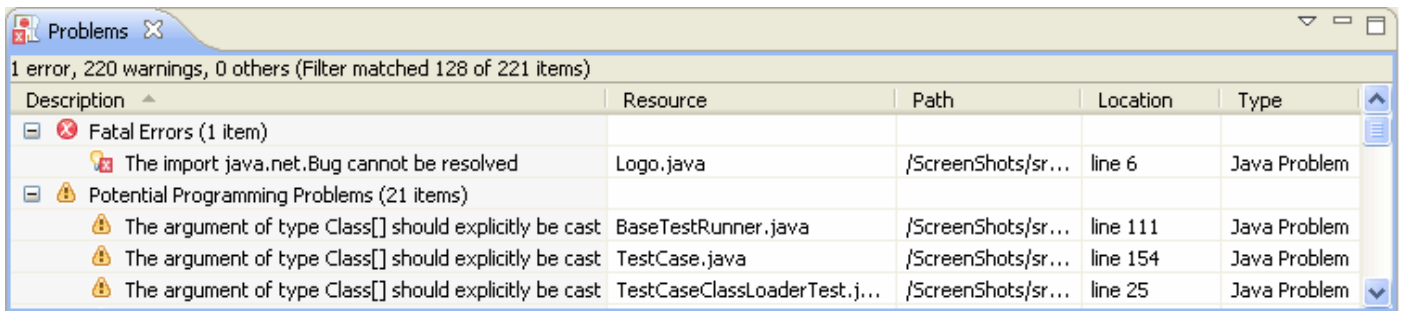
È possibile spostarsi da una voce all'altra tramite i pulsanti freccia **Su** e **Giù**.

Per copiare l'errore negli Appunti, premere il pulsante con l'immagine degli Appunti.

## Problema: Dove ottenere i log di Call Studio per la convalida delle applicazioni

### Soluzione

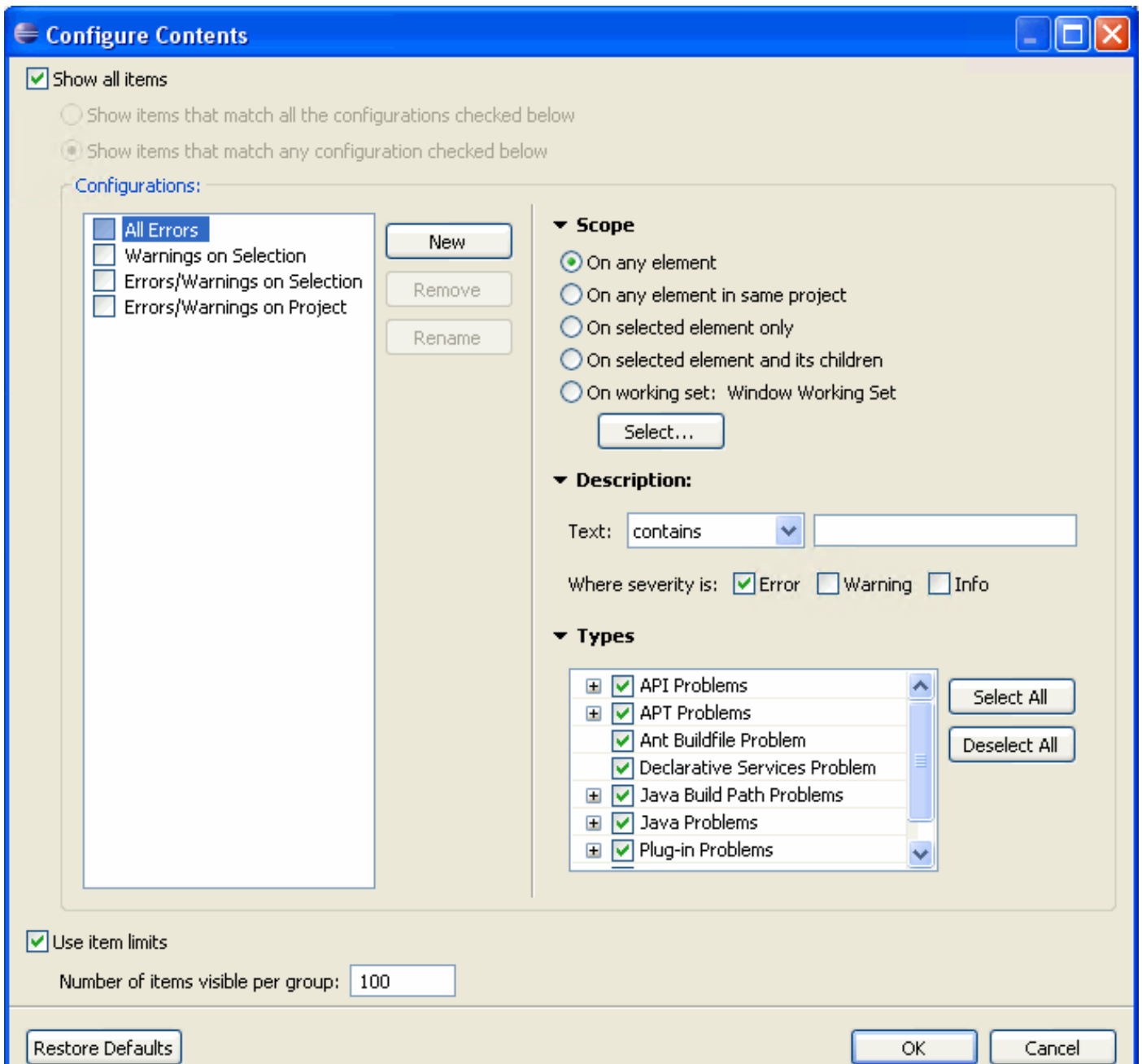
Quando si utilizzano le risorse nel workbench, diversi costruttori possono registrare automaticamente problemi, errori o avvisi nella visualizzazione Problemi. Ad esempio, quando si salva un file di origine Java che contiene errori di sintassi, questi vengono registrati nella visualizzazione Problemi. Quando si fa doppio clic sull'icona relativa a un problema, un errore o un avviso, l'editor della risorsa associata viene aperto automaticamente alla riga di codice corrispondente.



Per impostazione predefinita, la vista Problemi raggruppa i problemi in base alla gravità. È inoltre possibile raggrupparli per tipo oppure non raggrupparli affatto. Alcuni componenti aggiungono il proprio raggruppamento. Ad esempio, il supporto per gli strumenti di sviluppo Java (JDT) aggiunge un gruppo Java Problem Type. Il raggruppamento può essere selezionato utilizzando il menu **Raggruppa per**.

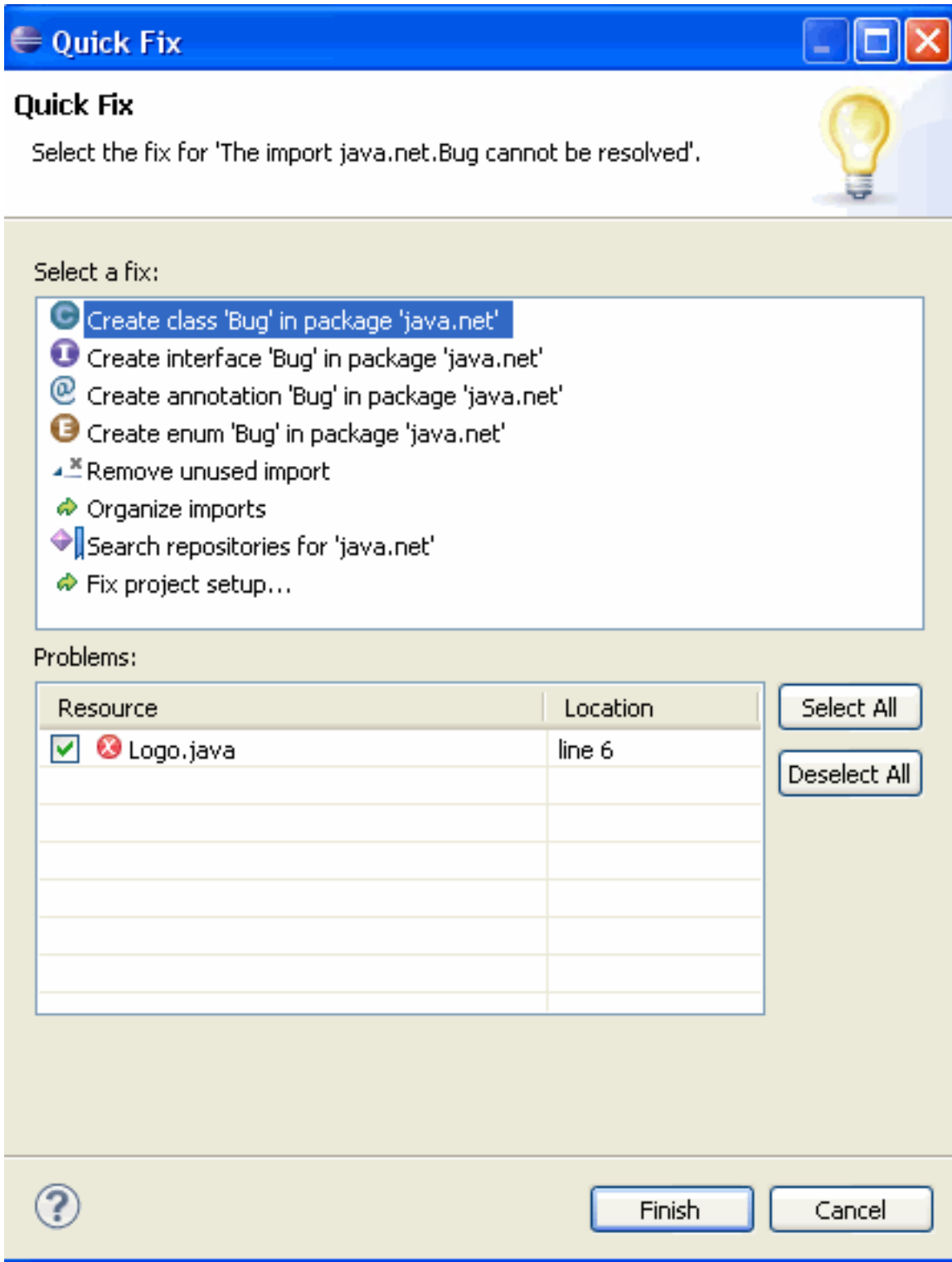
Nella prima colonna della visualizzazione Problemi viene visualizzata un'icona che indica il tipo di elemento riga, la categoria e la descrizione. Fare clic con il pulsante sinistro del mouse sull'elemento per aprire il file in un editor ed evidenziare la riga contenente il problema.

È possibile configurare il contenuto della visualizzazione Problemi in modo da visualizzare solo gli avvisi e gli errori associati a una particolare risorsa o gruppo di risorse. A tale scopo, utilizzare la finestra di dialogo **Configura contenuto** disponibile dal menu a discesa. È possibile aggiungere più filtri alla visualizzazione Problemi e attivarli o disattivarli in base alle esigenze. I filtri possono essere additivi (vengono visualizzati tutti i problemi che soddisfano almeno uno dei filtri attivati) o esclusivi (vengono visualizzati solo i problemi che soddisfano tutti i filtri). I due filtri più diffusi (Tutti gli errori e gli avvisi nella selezione) sono forniti per impostazione predefinita.



I problemi possono essere risolti selezionando **Correzione rapida** dal menu di scelta rapida. Viene presentato l'elenco delle possibili risoluzioni.





Per aggiungere la vista Problemi (Problems) alla prospettiva corrente, fate clic su Finestra (Window) > Mostra vista (Show View) > Altro... > Generale (General) > Problemi (Problems).