



Red Hat Developer Tools 1

Eclipse 4.14.0 の使用

Eclipse 4.14.0 のインストールとアプリケーション開発の最初のステップ

Red Hat Developer Tools 1 Eclipse 4.14.0 の使用

Eclipse 4.14.0 のインストールとアプリケーション開発の最初のステップ

Olga Tikhomirova
otikhomi@redhat.com

Supriya Takkhi

Peter Macko

Kevin Owen

Yana Hontyk

法律上の通知

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

Red Hat Developer Tools をインストールして使用を開始するユーザー向けの情報。

目次

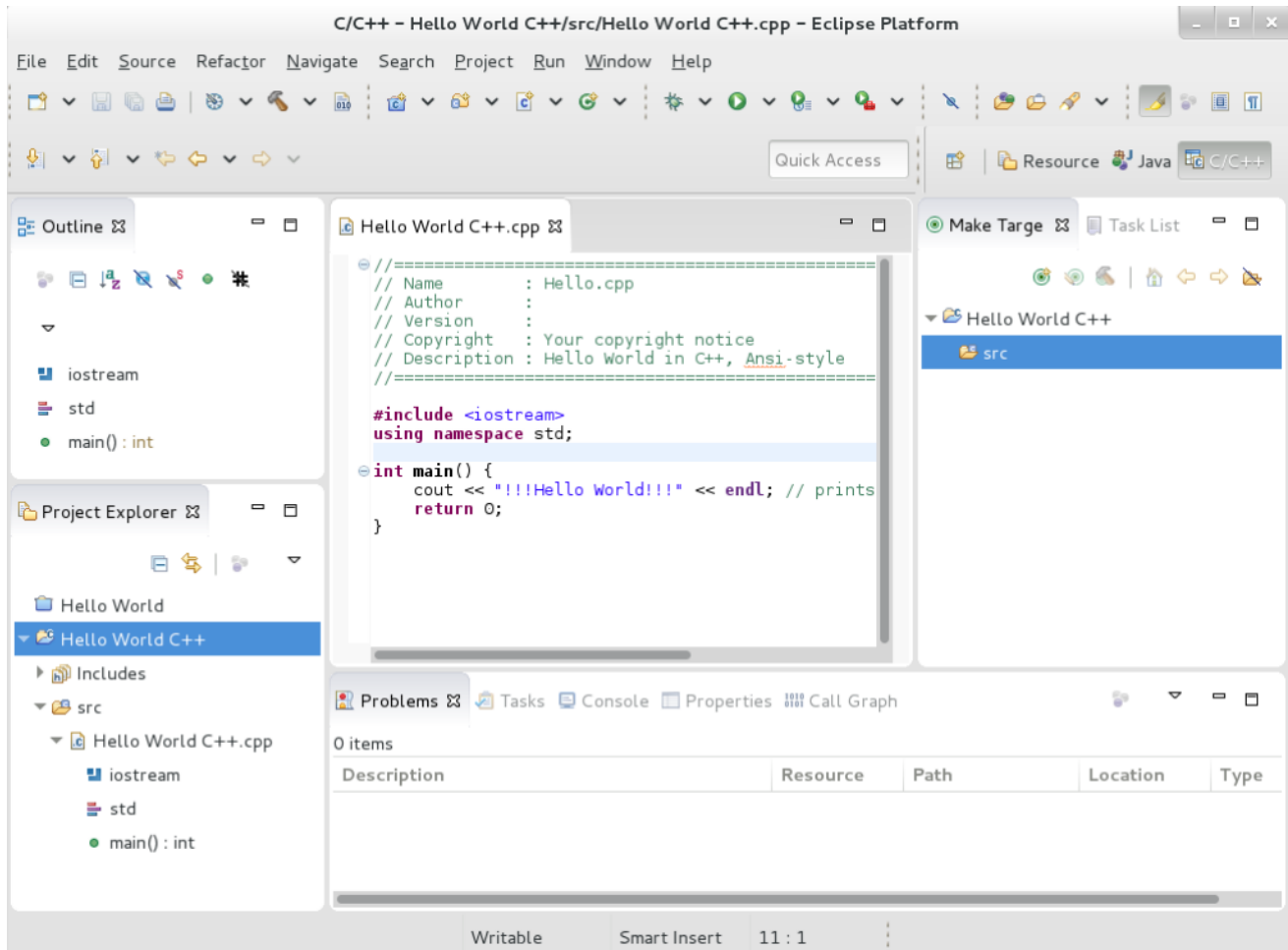
第1章 ECLIPSE 4.14.0	3
1.1. RED HAT ENTERPRISE LINUX 7 での ECLIPSE へのアクセス	5
1.2. ECLIPSE のインストール	6
1.3. ECLIPSE の使用	7
1.4. 関連情報	8
第2章 ECLIPSE 4.14.0 の変更点	10
第3章 ECLIPSE 4.14.0 の既知の問題	12
3.1. ECLIPSE が起動時にハングする	12
3.2. 古い ECLIPSE バージョンで作成されたワークスペースを実行すると、"INITIALIZING ECLIPSE ERROR REPORTING SYSTEM" エラーが発生する	12
3.3. サードパーティーソースからの一部のプラグインで NULLPOINTEREXCEPTION が発生する	12
3.4. DEBUGINFO 競合	12
3.5. RHSCCL RH-MAVEN33-SCLDEVEL と RH-MAVEN35-SCLDEVEL パッケージ間の競合	13
3.6. PIP 統合で PYDEV ユーザーに問題が発生する可能性がある	13
3.7. PYTHON モジュールを管理するための PYDEV ショートカットが機能しない	13
3.8. ECLIPSE SUBCLIPSE とベース RHEL SUBVERSION 間の非互換性	13
3.9. TYCHO 競合	14

第1章 ECLIPSE 4.14.0

Red Hat Enterprise Linux 7 で利用できる Red Hat Developer Tools には、Eclipse 4.14.0 が含まれています。これは、Eclipse Foundation の 2019-12 リリーストレインに基づいています。

Eclipse は、開発プロセスの各フェーズにツールを提供する強力な開発環境です。豊富な開発エクスペリエンスを実現するためにさまざまな異種ツールを統一された環境に統合し、完全に設定可能なユーザーインターフェイスを提供して、さまざまな方法で拡張できるプラグイン可能なアーキテクチャーを特長としています。たとえば、**Valgrind** プラグインを使用すると、プログラマーは、**Eclipse** ユーザーインターフェイスを介してコマンドラインで実行されるメモリープロファイリングを実行できます。

図1.1 Eclipse セッションの例



Eclipse は、コマンドラインツールによる従来の対話に代わるグラフィカルな開発環境を提供します。コマンドラインインターフェイスを使用しない開発者には代替手段となります。従来のほぼコマンドラインベースの Linux ツール (**gcc** や **gdb** など) および **Eclipse** は、プログラミングに対して 2 つの異なるアプローチを提供します。

Red Hat JBoss Middleware のアプリケーション開発方法、または OpenShift Tools のサポート方法については、[Red Hat Developer Studio](#) を参照してください。

表1.1 rh-eclipse パッケージコレクションに含まれる Eclipse コンポーネント:

パッケージ	説明
-------	----

パッケージ	説明
rh-eclipse-eclipse-cdt	C/C++ Development Tooling (CDT) は、これらの2つの言語で開発するための機能およびプラグインを提供。
rh-eclipse-eclipse-changelog	changelog ファイルを作成および維持できる ChangeLog プラグイン。
rh-eclipse-eclipse-dtp	Eclipse Data Tools Platform プロジェクト。
rh-eclipse-eclipse-egit	Eclipse のチームプロバイダーで、Git リポジトリと対話するための機能およびプラグインを提供する EGit。
rh-eclipse-eclipse-emf	EMF (Eclipse Modeling Framework) を使用すると、構造化データモデルに基づいたアプリケーションのビルドが可能。
rh-eclipse-eclipse-gcov	GCov テスト対象プログラムを Eclipse と統合する GCov プラグイン。
rh-eclipse-eclipse-gef	既存のアプリケーションモデルからリッチグラフィカルエディターを作成できるようにする GEF (Graphical Editing Framework)。
rh-eclipse-eclipse-gprof	Gprof パフォーマンス分析ユーティリティーを Eclipse と統合する Gprof プラグイン。
rh-eclipse-eclipse-jdt	Eclipse Java 開発ツール (JDT) プラグイン。
rh-eclipse-eclipse-jgit	Git リビジョン管理システムの Java 実装である JGit。
rh-eclipse-eclipse-manpage	Eclipse の man ページを表示できる man ページプラグイン。
rh-eclipse-eclipse-mpc	Eclipse Marketplace クライアント。
rh-eclipse-eclipse-mylyn	Eclipse のタスク管理システムである Mylyn。
rh-eclipse-eclipse-pde	Eclipse プラグインを開発するためのプラグイン開発環境。
rh-eclipse-eclipse-perf	perf ツールを Eclipse と統合する Perf プラグイン。
rh-eclipse-eclipse-pydev	Eclipse 向けの PythonIDE の完全な機能。

パッケージ	説明
rh-eclipse-eclipse-remote	拡張可能な remote-services フレームワークを提供する Remote Services プラグイン。
rh-eclipse-eclipse-rpm-editor	RPM 仕様ファイルの維持を可能にする Eclipse Spec File Editor。
rh-eclipse-eclipse-systemtap	SystemTap と Eclipse を統合する SystemTap プラグイン。
rh-eclipse-eclipse-subclipse	Subclipse は、Eclipse のチームプロバイダーで、Subversion リポジトリと対話するための機能とプラグインを提供。
rh-eclipse-eclipse-valgrind	Valgrind を Eclipse と統合する Valgrind プラグイン。
rh-eclipse-eclipse-webtools	Eclipse Webtools プラグイン。

1.1. RED HAT ENTERPRISE LINUX 7 での ECLIPSE へのアクセス

Eclipse は、Red Hat Developer Tools コンテンツセットの一部として配布されるオフラインで、Red Hat Enterprise Linux 7 をデプロイしているお客様が利用できます。Eclipse をインストールするには、Red Hat Subscription Management を使用して、Red Hat Developer Tools および Red Hat Software Collections リポジトリを有効にします。

1. システムのバリエーション (**workstation** または **server**) を選択し、以下のコマンドを変更します。



注記

開発者には、Red Hat Enterprise Linux Server を使用して、多岐にわたる開発ツールにアクセスすることを推奨しています。

2. **rhel-7-variant-devtools-rpms** リポジトリを有効にします。

```
# subscription-manager repos --enable rhel-7-variant-devtools-rpms
```

3. **rhel-variant-rhsc-7-rpms** リポジトリを有効にします。

```
# subscription-manager repos --enable rhel-variant-rhsc-7-rpms
```

4. **rhel-7-variant-optional-rpms** リポジトリを有効にします。

```
# subscription-manager repos --enable rhel-7-variant-optional-rpms
```

リポジトリが有効になります。「[Eclipse のインストール](#)」で説明されているように、Red Hat Eclipse をインストールしてください。

Red Hat Developer Tools debuginfo リポジトリの有効化



注記

この手順はオプションです。

Red Hat Developer Tools は、リポジトリに含まれるすべてのアーキテクチャー依存 RPM の **debuginfo** パッケージも提供します。これらのパッケージは、コアファイル分析や Eclipse 自体のデバッグに役立ちます。

1. Red Hat Developer Tools の **debuginfo** リポジトリを有効にします。**variant** は、Red Hat Enterprise Linux システムバリエーション (**server** または **workstation**) に置き換えます。

```
# subscription-manager repos --enable rhel-7-variant-devtools-debug-rpms
# subscription-manager repos --enable rhel-variant-rhsc7-7-debug-rpms
```

debuginfo パッケージのインストール、概要、および使用方法の詳細は、[実行中のアプリケーションのデバッグ](#) を参照してください。

関連情報

- Red Hat Subscription Management を使用してシステムを登録し、サブスクリプションに関連付ける方法は、[Red Hat Subscription Management](#) のガイドを参照してください。
- Red Hat Software Collections のサブスクリプションに関する詳細は、[Red Hat Developer Toolset ユーザーガイド](#) の [セクション 1.4 Red Hat Developer Toolset へのアクセス](#) を参照してください。

1.2. ECLIPSE のインストール

Eclipse 開発環境は、RPM パッケージのコレクションとして提供されます。

利用可能なコンポーネントのリストについては、[表 1.1 「rh-eclipse パッケージコレクションに含まれる Eclipse コンポーネント」](#) を参照してください。



注記

Eclipse は C、C++、および Java 開発を完全にサポートしますが、Fortran プログラミング言語はサポートされません。



注記

rh-eclipse コレクションには **rh-maven35** コレクションが必要です。

前提条件

- 「[Red Hat Enterprise Linux 7 での Eclipse へのアクセス](#)」に従ってリポジトリを有効にしている。

手順

1. Eclipse をインストールするには、**root** ユーザーとして次のコマンドを実行します。

```
# yum install rh-eclipse
```

1.3. ECLIPSE の使用

Eclipse を起動する方法:

1. **Applications > Programming > Red Hat Eclipse** の順にクリックするか、シェルプロンプトで以下を入力します。

```
$ scl enable rh-eclipse eclipse
```

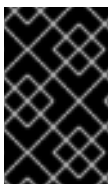
2. 起動中に、Eclipse はプロジェクトを保管するディレクトリーである **ワークスペース** を選択するように求められます。デフォルトのオプションである **~/workspace/** を使用するか、**Browse** をクリックしてファイルシステムを閲覧し、カスタムディレクトリーを選択します。さらに、**Use this as the default and do not ask again** のチェックボックスを選択すると、次回この開発環境を実行するときに Eclipse がこのダイアログボックスを表示しないようにすることもできます。
3. **OK** をクリックして選択を確定し、起動に進みます。

1.3.1. Red Hat Developer Toolset ツールチェーンの使用

Red Hat Developer Toolset の **GNU Compiler Collection** および **binutils** をサポートする Eclipse を使用するには、以下を実行します。

1. **devtoolset-9.0-toolchain** がインストールされていることを確認し、「[Eclipse の使用](#)」の説明に従ってアプリケーションを実行します。**rh-eclipse** コレクションは、デフォルトで Red Hat Developer Toolset ツールチェーンを使用します。

システムに **devtoolset-9.0-toolchain** パッケージをインストールする方法の詳細な手順は、[Red Hat Developer Toolset ユーザーガイドのセクション 1.5 Red Hat Developer Toolset のインストール](#) を参照してください。



重要

GNU コンパイラコレクション の Red Hat Enterprise Linux バージョンを使用して以前にビルドしたプロジェクトの以前のビルド結果をすべて破棄します。破棄するには、Eclipse でプロジェクトを開き、メニューから **Project > Clean** を選択します。

1.3.2. Red Hat Enterprise Linux ツールチェーンの使用

Red Hat Enterprise Linux で配布されるツールチェーンへのサポートのある Eclipse を使用するには、プロジェクトの設定を変更して、Red Hat Enterprise Linux システムバージョンの **gcc**、**g++**、および **as** への絶対パスを使用します。

現在のプロジェクトでツールの Red Hat Enterprise Linux システムバージョンを明示的に使用するように Eclipse を設定するには、以下を実行します。

1. C/C++ パースペクティブで **Project > Properties** をクリックし、プロジェクトプロパティーを開きます。
2. ダイアログボックス左側のメニューで、**C/C++ Build > Settings** をクリックします。

3. **Tool Settings** タブをクリックします。
4. C プロジェクトで作業している場合は、以下を行います。
 - a. **GCC C Compiler** または **Cross GCC Compiler** を選択し、**Command** フィールドの値を以下のように変更します。

```
/usr/bin/gcc
```
 - b. **GCC C Linker** または **Cross GCC Linker** を選択し、**Command** フィールドの値を以下のように変更します。

```
/usr/bin/gcc
```
 - c. **GCC C Assembler** または **Cross GCC Assembler** を選択し、**Command** フィールドの値を以下のように変更します。

```
/usr/bin/as
```
5. C++ プロジェクトで作業している場合:
 - a. **GCC C Compiler*** または ***Cross G Compiler** を選択し、**Command** フィールドの値を以下のように変更します。

```
/usr/bin/g++
```
 - b. **GCC C Compiler** または **Cross GCC Compiler** を選択し、**Command** フィールドの値を以下のように変更します。

```
/usr/bin/gcc
```
 - c. **GCC C Linker*** または ***Cross G Linker** を選択し、**Command** フィールドの値を以下のように変更します。

```
/usr/bin/g++
```
 - d. **GCC C Assembler** または **Cross GCC Assembler** を選択し、**Command** フィールドの値を以下のように変更します。

```
/usr/bin/as
```
6. **OK** をクリックして設定の変更を保存します。

1.4. 関連情報

Eclipse とそのすべての機能の詳細な説明は、このドキュメントの対象外です。詳細は、次のリソースを参照してください。

インストールされているドキュメント

- **Eclipse** には、統合された各機能とツールに関する幅広いドキュメントを提供するヘルプシステムが組み込まれています。

関連項目

- [2章Eclipse 4.14.0 の変更点](#) では、Developer Tools の以前のリリースに含まれている **Eclipse** 開発環境の機能および改善点の一部を紹介しています。
- ユーザーガイドの [Red Hat Developer Toolset](#) の章では、Red Hat Developer Toolset の概要と、システムにインストールする方法の詳細について説明しています。
- ユーザーガイドの [GNU コンパイラコレクション \(GCC\)](#) の章では、コマンドラインの C、C++、および Fortran で書かれたプログラムをコンパイルする方法について説明しています。

第2章 ECLIPSE 4.14.0 の変更点

Red Hat Developer Tools は、Eclipse 4.14.0 および 2019-12 リリーストレインのプラグインと共に配布され、多数のバグ修正と機能拡張を提供します。

以下は、このリリースの新機能と互換性の変更点をまとめたリストです。

これらの新機能の使用方法は、組み込み Eclipse ドキュメントを参照してください。

重要なパッケージの更新

- **eclipse** 4.13.0 → 4.14.0 : Eclipse IDE および JDT/PDE プラグインが 4.14 に更新されました。主な機能は次のとおりです。
 - Java™ 13 のサポート。詳細は、<https://www.eclipse.org/eclipse/news/4.14/jdt.php> を参照してください。
 - **Quick Access** が改善され、**Find Actions** と呼ばれるようになりました。改善点の中には、ワークスペースファイルの検索、またはファイルコンテンツ内のテキスト検索があります。詳細は、<https://www.eclipse.org/eclipse/news/4.14/platform.php> を参照してください。
 - 問題マーカをインラインで表示します。この機能を有効にするには、**General > Editors > Text Editors** 設定ページをクリックします。詳細は、<https://www.eclipse.org/eclipse/news/4.14/platform.php> を参照してください。
 - バックスペース/削除では、スペースをタブとして扱うことができます。この機能を有効にするには、**General > Editors > Text Editors** 設定ページをクリックします。詳細は、<https://www.eclipse.org/eclipse/news/4.14/platform.php> を参照してください。
 - 一貫した暗いテーマの色。
- **eclipse-cdt** 9.9.0 → 9.10.0: C/C++ Development Tooling が 9.10.0 に更新されました。変更には、新しいコード分析チェッカーまたは Compilation Database Parser が含まれます。詳細は、<https://wiki.eclipse.org/CDT/User/NewIn910> にあるアップストリームのリリースノート参照してください。
- **eclipse-linuxtools** 7.4.0 → 7.5.0: Linuxtools プラグインが、バグ修正に重点を置いた 7.5.0 に更新されました。詳細は、https://wiki.eclipse.org/Linux_Tools_Project/News/NewIn75 にあるアップストリームのリリースノート参照してください。
- **eclipse-pydev** 7.3.0 → 7.4.0: **PyDev** プラグインが 7.4.0 に更新され、デバッガーの改善、Python 3.8 のサポートが行われるようになりました。詳細は、アップストリームのリリースノート (http://www.pydev.org/history_pydev.html) を参照してください。
- **eclipse-egit/jgit** 5.5.0 → 5.6.0: Git 統合プラグインが 5.6.0 に更新されました。History ビューでの並列または選択可能なブランチのプルなど、さまざまな改善が行われています。詳細は、アップストリームのリリースノート (<https://projects.eclipse.org/projects/technology.jgit/releases/5.6.0> および <https://projects.eclipse.org/projects/technology.egit/releases/5.6.0>) を参照してください。

非推奨

1.削除済みパッケージ

m2e 拡張機能を表す次のパッケージは、コレクションから削除されています。

- **eclipse-m2e-apt**
- **eclipse-m2e-egit**
- **eclipse-m2e-mavenarchiver**
- **eclipse-m2e-tycho**
- **eclipse-m2e-wtp**

それぞれの拡張機能は、m2e マーケットプレイスで入手できます。Eclipse で、**Find Actions** メニューを使用して m2e Marketplace を開きます。

1. **Ctrl+3** をクリックして、**Find Actions** フォームを開きます。
2. **m2e Marketplace** と入力します。
3. m2e Marketplace アクションが表示されたらクリックします。

2.Dockerfile エディター

eclipse-linuxtools によって提供される Dockerfile エディターは、アップストリームプロジェクトによって非推奨になりました。これは、次の場所で入手できる新しい Dockerfile エディターに置き換えられます。

- Docker ツール更新 Web サイト: <https://download.eclipse.org/linuxtools/update-docker>.
 - この事前設定された更新 Web サイトは、デフォルトでは無効になっていることに注意してください。有効にするには、**Available Software Sites** の設定を変更します。
- マーケットプレイス: <https://marketplace.eclipse.org/content/eclipse-docker-tooling>.

この新しいエディターは **Node.js** に依存しています。エディターを使用する方法:

1. RHSCCL の **rh-nodejs12** コレクションをインストールします。

```
# yum install rh-nodejs12
```

2. Eclipse を再起動します。

第3章 ECLIPSE 4.14.0 の既知の問題

このセクションでは、Eclipse 4.14.0 の既知の問題を説明します。

3.1. ECLIPSE が起動時にハングする

原因: スレッドデッドロックが原因で、Eclipse が起動時にハングする可能性があります。

結果: ワークスペース選択ダイアログを表示せずに Eclipse スプラッシュスクリーンのみが表示されます。

回避策: PID を特定し、プロセスの kill シグナルを送信することで、問題のある Eclipse プロセスを強制終了します。以下に例を示します。

```
$ jps
30066 Jps
31631 org.eclipse.equinox.launcher_1.5.300.v20190308-0658.jar
$ kill 31631
```

結果: 問題のあるプロセスが強制終了されると、Eclipse を正常に再起動できます。

3.2. 古い ECLIPSE バージョンで作成されたワークスペースを実行すると、"INITIALIZING ECLIPSE ERROR REPORTING SYSTEM" エラーが発生する

回避策: **-clean** オプションを使用して Eclipse を再起動し、依存関係解決キャッシュを消去します。

```
$ scl enable rh-eclipse "eclipse -clean"
```

結果: Eclipse はこのエラーメッセージなしで起動します。

3.3. サードパーティーソースからの一部のプラグインで NULLPOINTEREXCEPTION が発生する

サードパーティーの更新サイトからプラグインがインストールされていると、Eclipse が起動に失敗し、ワークスペースのログファイルに **NullPointerException** が記録されることがあります。

回避策: **-clean** オプションを使用して Eclipse を再起動し、依存関係解決キャッシュを消去します。

```
$ scl enable rh-eclipse "eclipse -clean"
```

結果: Eclipse は通常どおりに起動します。

3.4. DEBUGINFO 競合

原因: ユーザーがこのコレクションの **debuginfo** パッケージ (例: **rh-eclipse-eclipse-cdt-debuginfo**) をインストールしようとする時、内容が以前のコレクションにある同じパッケージと競合する可能性があります。

結果: **rh-eclipse48** コレクションの同じ **debuginfo** パッケージがインストールされている場合、**rh-eclipse debuginfo** パッケージのインストールに失敗する可能性があります。

回避策: `rh-eclipse` コレクションに `debuginfo` を使用するには、コマンド `yum remove rh-eclipse48*debuginfo` を使用して、最初に `rh-eclipse48` コレクションの `debuginfo` パッケージをアンインストールする必要があります。

結果: `rh-eclipse` コレクションからの `debuginfo` パッケージのインストールが正常に実行されます。

3.5. RHSCCL RH-MAVEN33-SCLDEVEL と RH-MAVEN35-SCLDEVEL パッケージ間の競合

RHSCCL `rh-maven33-scldevel` と `rh-maven35-scldevel` パッケージの間に競合があります。これは `rh-eclipse-scldevel` パッケージに影響します (このパッケージはデフォルトではインストールされないことに注意してください)。

`rh-eclipse-scldevel` パッケージと `rh-eclipse46-scldevel` パッケージを并存させることはできません。

3.6. PIP 統合で PYDEV ユーザーに問題が発生する可能性がある

Python SCL によって事前にインストールされるモジュールはアンインストールできません。これを試行すると、パーミッション拒否エラーが発生します。

3.7. PYTHON モジュールを管理するための PYDEV ショートカットが機能しない

`Ctrl+2` ショートカットを使用して `pip` を選択しても、適切に機能しません。

結果: このショートカットを使用しようとする、ワークスペースログに `SWTException` が記録されます。

回避策: ユーザーは、`PyDev > Interpreters` 設定を使用して Python モジュールを引き続き管理できます。

3.8. ECLIPSE SUBCLIPSE と ベース RHEL SUBVERSION 間の非互換性

Eclipse Subclipse で作成した Subversion リポジトリのワーキングコピーは、Subversion のベース RHEL バージョンと互換性がありません。このような作業コピーで `svn` コマンドを使用すると、以下のエラーが発生する可能性があります。

```
$ svn up
svn: E155021: This client is too old to work with the working copy
```

回避策: コマンドラインで Eclipse Subclipse によって使用される Subversion の Pure-Java 実装を使用します。

```
# yum install rh-eclipse-svnkit-cli # Command line support for SVNKit
```

これで、`svn` コマンドを通常使用する場所で `svn` を使用できます。

```
$ jsvn up
Updating '!':
At revision 16476.
```

3.9. TYCHO 競合

原因: `rh-eclipse-tycho` パッケージが以前のコレクションに含まれる同じパッケージと競合しています (例: `rh-eclipse48-tycho`)。

結果: `rh-eclipse48-tycho` パッケージがすでにインストールされている場合、`rh-eclipse-tycho` パッケージのインストールに失敗することがあります。

回避策: Tycho を必要とするのは、Eclipse やそのプラグインをビルドまたは再ビルドするユーザーのみであることに注意してください。必要に応じて、`rh-eclipse-tycho` パッケージをインストールする前に、`yum remove rh-eclipse48-tycho` コマンドを使用して `rh-eclipse48-tycho` パッケージをアンインストールします。

結果: `rh-eclipse-tycho` パッケージのインストールが正常に実行されます。