

March 2006



# **IBM Rational Software Development Platform: Practitioner product installation concepts**

**Ready for IBM Rational software validation  
Technical White Paper**

---

## Contents

---

<b>Ready for IBM Rational software validation Technical White Paper.....</b>	<b>1</b>
<b>Introduction .....</b>	<b>2</b>
<i>Practitioner products .....</i>	<i>2</i>
<i>Rational Software Development Platform.....</i>	<i>3</i>
<i>Practitioner product installation scenarios .....</i>	<i>3</i>
<b>Extending an IBM Software Development Platform Practitioner product.....</b>	<b>4</b>
<b>Extension installation using Update Manager .....</b>	<b>5</b>
<i>Implementation.....</i>	<i>5</i>
<i>User interaction with the install process.....</i>	<i>6</i>
<b>Extension installation using an installation routine.....</b>	<b>6</b>
<i>Implementation.....</i>	<i>7</i>
<i>User interaction with the install process.....</i>	<i>7</i>
<b>Rational Software Development Platform system markers .....</b>	<b>9</b>
<b>System markers .....</b>	<b>9</b>
<i>Windows platform example .....</i>	<i>11</i>
<i>Non-Windows platform example .....</i>	<i>11</i>
<b>Interpreting system markers.....</b>	<b>12</b>
<i>System marker structure.....</i>	<i>12</i>
<i>Pseudo-code for reading system markers .....</i>	<i>14</i>
<b>Eclipse product and extension marker files .....</b>	<b>15</b>
<b>Identifying optionally installed features .....</b>	<b>16</b>
<b>References.....</b>	<b>17</b>

### **Introduction**

The IBM Rational Software Development Platform is a modular set of tools for developing software-based systems. There are two types of products in the Rational Software Development Platform: Practitioner products and Process and Lifecycle products.

This document discusses how the Practitioner products themselves are installed, and how you can install additional tools as part of one of the installed Practitioner products. The Ready for IBM Rational software requirements identify the rules for how new function can be installed, but additional guidance on how you can define an installation process that is compatible with the Rational Software Development Platform can be helpful.

### **Practitioner products**

Several Rational Software Development Platform Practitioner products support the modeling and development process:

- Rational Software Modeler
- Rational Software Architect
- Rational Web Developer for WebSphere Software
- Rational Application Developer for WebSphere Software
- Rational Systems Developer
- WebSphere Development Studio Client for iSeries
- WebSphere Development Studio Client Advanced Edition for iSeries
- WebSphere Developer for zSeries

Each of these products offers a different set of function and capabilities. When you create a new tool it may be useful in any number of these products, but your tool could also depend on function that is available in only one or just a few of these products. That is your choice, and your installation process should ensure that your tool can only be installed as part of a product that can support the use of your tool.

## Rational Software Development Platform

When the Rational Software Development Platform Practitioner products are installed they first install an Eclipse-based platform for tool integration. Many of the Practitioner products build on a shared instance of the Eclipse-based Rational Software Development Platform, but a non-shared instance is used for the Rational Systems Developer product (see Figure 1).

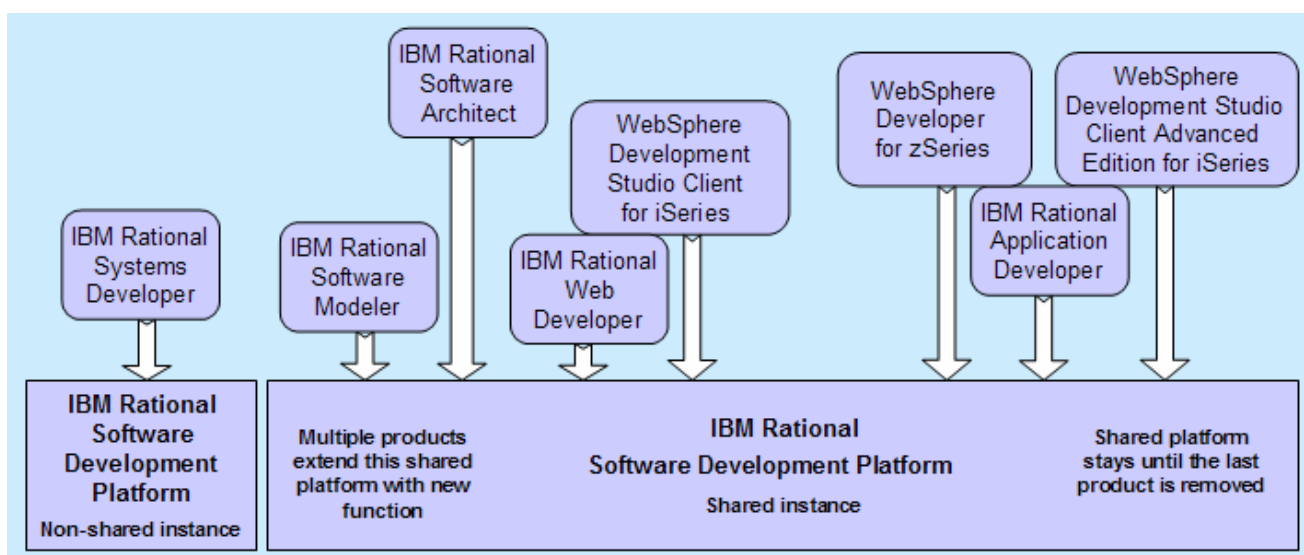


Figure 1 IBM Rational Software Development Platform products

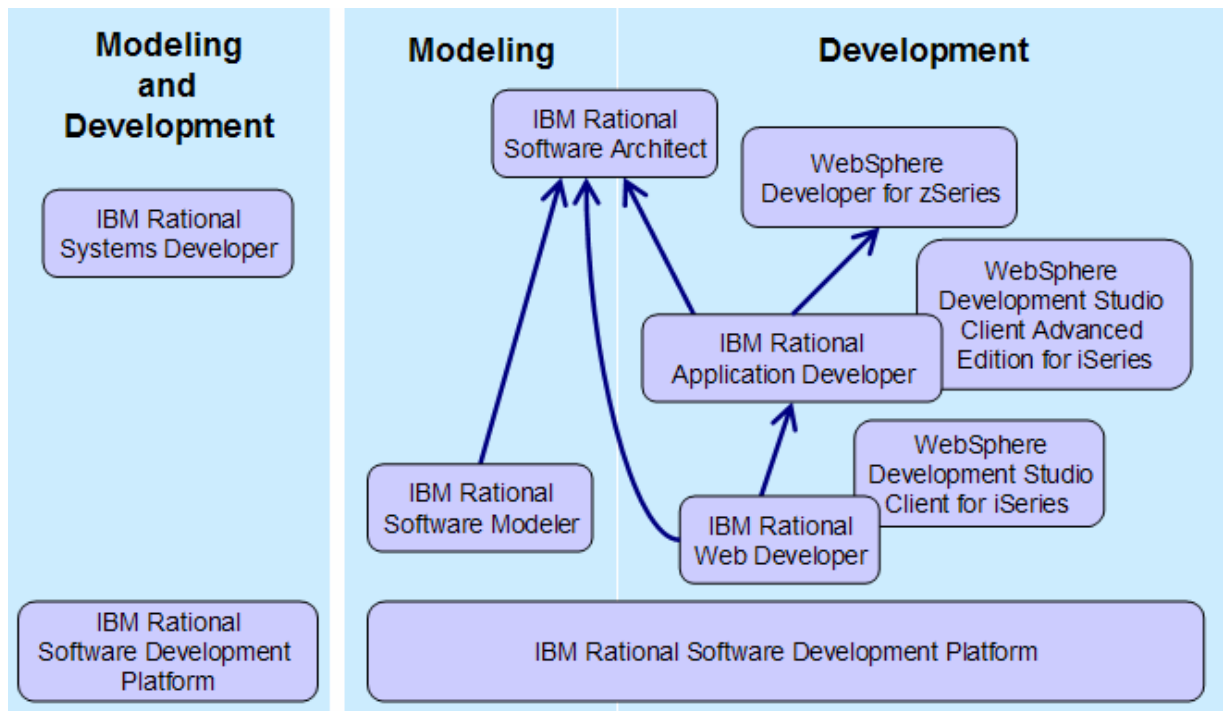
The addition of well-formed Eclipse extensions to the Rational Software Development Platform Practitioner products is encouraged. Your software solution can be added to any of these products, but you may choose to only support a subset that satisfies your prerequisites.

## Practitioner product installation scenarios

If a single Rational Software Development Platform Practitioner product is installed it also installs the Eclipse-based Rational Software Development Platform when required. Depending on the first product you installed, the second product will manage its installation as part of the system. This means the second product could either:

- Be added to the shared platform; for example, when Rational Software Modeler is added to Rational Web Developer.
- Upgrade the existing shared platform product; for example, when Rational Application Developer replaces the previously installed Rational Web Developer.
- Be installed as a prerequisites; for example, the WebSphere Developer for zSeries installer will first install Rational Application Developer, if it is not already available.
- Be installed separately; for example, when you install Rational Systems Developer as the second, or even as the first product, it will always create a non-shared instance of the Rational Software Development Platform.

See Figure 2 for the various installation paths and the relationship between the different products and the supporting Rational Software Development Platform instances.



**Figure 2 IBM Rational Software Development platform product installation paths**

Some products include the capabilities of other products; for example, the WebSphere Development Studio Client for iSeries product includes the function provided by Rational Web Developer for WebSphere Software. Products that provide more function can still upgrade these products.

You can independently install the WebSphere Development Studio Client for iSeries and WebSphere Development Studio Client Advanced Edition for iSeries products. You can even extend a WebSphere Development Studio Client Advanced Edition for iSeries configuration with the installation of Rational Software Architect.

In summary, you can install each Practitioner product individually, or install multiple products, but the end result depends on the products chosen. There will be at least one instance of the Rational Software Development Platform available, but there could be two if more than one product was installed and one of them was Rational Systems Developer.

### **Extending an IBM Software Development Platform Practitioner product**

Your goal is to get your software installed as part of the installed Rational Software Development Platform Practitioner product(s). Choose between these software installation options:

- Extension Installation using Update Manager
- Extension Installation using an Installation Routine

Each option requires that you use an approach that will pass the Ready for IBM Rational software validation requirements. Just using the Update Manager or a custom installation routine is not enough; you have to use a specific technique to satisfy the requirements.

### ***Extension installation using Update Manager***

If you apply the right technique, you can use the Eclipse Update Manager to install your software as part of an existing Rational Software Development Platform-based product. You are required to define your root feature(s) so that they can control the install location used as part of the Rational Software Development Platform. When done correctly, this technique satisfies the Ready for IBM Rational software validation requirements.

#### **Implementation**

If you want to use this technique there is only one thing you have to do that differs from the standard use of Eclipse and an update site. The `feature.xml` for all your root features (features that are not included by another feature) must include this attribute to identify a specific target feature included in the Rational Software Development Platform base:

```
colocation-affinity="com.ibm.rational.sdp.isvinstall"
```

When you use the `colocation-affinity` attribute as shown above and install your software as part of the Rational Software Development Platform, the Eclipse Update Manager can only write your software to this relative location in the directory where the product is installed:

```
<install_location>\sdpisv\eclipse
```

You can still install your software in other Eclipse-based products when you add the `colocation-affinity` attribute to your `feature.xml`. When the feature identified by the `colocation-affinity` attribute is not included in the target product, the install proceeds and the user can choose any target installation location. If you do not want to make this modification to your existing feature(s), you can always add a new feature that includes your existing feature(s) that has the `colocation-affinity` attribute. Use the new feature when installing on the Rational Software Development Platform.

There is a reason for wanting to place your software in the specified target location. The Rational Software Development Platform uninstall routine is designed to leave this target location and any associated software untouched. This way your software will not be uninstalled if the Rational Software Development Platform is uninstalled, which could possibly happen as part of service processing that is performed on the product by the customer.

You may still need to use a standalone installation routine because that is your standard, or you have other non-Eclipse-based software to install, but this is your decision. The use of Update Manager is a viable installation technique that can meet the requirements of the Ready for IBM Rational software validation offering.

### **User interaction with the install process**

You can use the guidance provided by Eclipse to package your function on an Update Site, be it a server based remote site or a file based local site.

This is a high-level description of what you would do to use the Update Manager to install your tool as part of a Rational Software Development Platform Practitioner product:

1. Start the Install/Update wizard using the Help > Software Updates > Find and Install... menu option and choose the Search for new features to install option.
2. Use the appropriate Add update site button to identify the URL for the Update Site you want to access; this could be a local or remote site depending on your design.
3. Use the wizard to open the Update Site, locate the feature(s) that you want to install, and request installation. The wizard will only allow you to use the required target location when the installed feature uses the appropriate colocation-affinity value.

When you select Finish, the features are then downloaded, written to the required target location, and configured as part of the Rational Software Development Platform.

This is the same process for any Update Manager based install; the only difference is that your feature is controlling the target location in the Rational Software Development Platform used to add the extension.

### ***Extension installation using an installation routine***

Choosing to use an installation routine may be your only option; this really depends on the needs of your software (you may have software beyond what you add as a Rational Software Development Platform extension to install for example).

Be advised that you could install your software physically on the target machine with an installation routine and then use the Eclipse Update Manager to install your extension; the update site could be included in your installed software on the local or even on a server machine within the users domain. This option would satisfy the Ready for IBM Rational software requirements.

There are several Ready for IBM Rational software requirements that must be satisfied when you use an installation routine to add your tool as a Rational Software Development Platform extension. You have to use the right technique to find the Rational Software Development Platform target, pick an appropriate file system target for your software, and integrate your extension with the active Rational Software Development Platform Practitioner product(s). See section “3.1 Installation”, of the “Ready for IBM Rational Software Integration Requirements” document for a discussion of these requirements.

The detailed installation guidance that follows assumes that you are comfortable with the basic concepts of extending the Eclipse platform; See the [Product Extensions](#) topic in the Eclipse help system for more information.

## **Implementation**

If you want to install your tool using an installation routine this routine must do more than simply write your software to the target system. Your installation routine is required to properly identify what instance(s) of the Rational Software Development Platform might be available, prompt the user for guidance on which should be extended, and adjust the configuration so that your tool is included in the appropriate Rational Software Development Platform instance.

Your installation routine must find the available installation targets and then present the appropriate target, or a list of possible targets, to the customer for confirmation. If the routine identifies more than one installed Rational Software Development Platform instance, present the list of possible targets to the user. You then add your extension by modifying the Rational Software Development Platform configuration for the selected installation target.

You must use the Rational Software Development Platform system markers to identify the available installation targets and their location on the file system. When a Practitioner product is installed it creates system markers that describe what was installed and where it is located on the file system. For a full description of the system markers and how they should be interpreted see the “Rational Software Development Platform System Markers” section.

If you need to determine if the currently installed product set meets any non-standard pre-requisites defined for your software you may have to inspect the configuration before installation. See the “Identifying Optionally Installed Components” section for details.

## **User interaction with the install process**

Your installation routine should support a flow and user interaction that is close to this:

1. The user runs your installer. Your installer identifies if one or more instances of the Rational Software Development Platform, or if required, a particular Practitioner product are installed.
2. You then prompt the user for guidance on which Rational Software Development Platform instance(s) your tool should extend. Identify the instance(s) using the associated product(s). If your tool required a specific Practitioner product then this might not be a prompt, but instead just an information page in your installer that says you found the required product. Your installer may wish to terminate if it cannot find the required product or a valid Rational Software Development Platform instance.
3. Your installer should then recommend an install location for your extension, but you should allow the user to override this default value. Your installer must ensure that the chosen extension install directory is outside the Rational Software Development Platform product directory tree structure.

4. Your installer then writes your extension to the target install directory and then adds the extension to the configuration of the target the Rational Software Development Platform instance(s) you want to extend. This can be done in one of two ways:

- o Adds a link file to the `\eclipse\links` directory where the Rational Software Development Platform has been installed.

The link file associates your Extension with the Rational Software Development Platform base. Typically, the link file name includes your feature id (`a.b.c.link`); the install directory used for your extension determines the exact content of the link file. This is an example of the content of a link file:

```
path=E:/Tool_Install_Directory_Name
```

The directory identified contains an eclipse directory structure with features and plug-ins for the extension and an appropriate `.eclipseextension` file.

- o Issues an Update Manager `addSite` command using the Eclipse instance in the target Rational Software Development Platform to be extended.

The `addSite` command provided by the [Update Manager API](#) allows you to add your Extension directly to the active configuration. This is the format of the command:

```
java.exe -cp startup.jar org.eclipse.core.launcher.Main  
-data tempwork  
-application org.eclipse.update.core.standaloneUpdate  
-command addSite -from E:\extensionInstallDir
```

The `extensionInstallDir` must contain an `.eclipseextension` marker file for the `addSite` command to succeed.

**Note:** Using a link file is the recommended approach. A link file is more robust than the `addSite` command approach. A link file will be processed when an existing configuration is deleted and rebuilt at the next startup. If you use the `addSite` approach, after rebuilding a deleted configuration any previous `addSite` commands may need rerun by the user.

Just remember that the location of the link file you create, or the context for the `addSite` command to be executed, is based not on the product, but the location of the Rational Software Development Platform instance to be extended.

Your installer may do more work beyond extending a Rational Practitioner product; the installer may even support the discovery of other Eclipse-based products that you may also want to extend. Nevertheless, it is important to recognize that the Ready for IBM Rational software requirements are specific in this area; you must use the system markers to discover the location of the Rational Software Development Platform instance(s) you wish to extend. Your installer may not ask the user to identify the target or to assume a default location.

### ***Rational Software Development Platform system markers***

The term system marker is used to identify the method used to record what Rational Software Development Platform Practitioner products have been installed on a given target system. This includes the products, components, and though this data the physical location of the Rational Software Development Platform base instance that you want to extend using an installation routine.

The technique used to implement system markers is platform dependant. On the Windows platform system markers are implemented as registry keys. These keys are prefixed with:

```
HKEY_LOCAL_MACHINE\SOFTWARE\  
IBM\Rational\Software Development Platform
```

On non-Windows platform system markers are implemented in a file system structure. The file system structure begins with:

```
etc\IBM\Rational\SoftwareDevelopmentPlatform
```

There are two types of system markers, product and component. Product system markers identify what products are installed; component system markers provide additional detail about the install. The most important component system marker is the Rational Software Development Platform shared base instance system marker.

The location attribute for the Rational Software Development Platform shared base system marker identifies the physical location of the Rational Software Development Platform shared base instance that has been installed. This shared instance can be used by several Rational Software Development Platform Practitioner products. This shared instance, and the possibly the non-shared Rational Software Development Platform base which can exist when the Rational Systems Developer product has been installed, contain the Eclipse Platform you will extend using your installation routine.

**Note:** While other product system markers may have location information, when the Rational Software Development Platform component system marker exists it is the only system marker that should be used to identify the install location for the shared instance of the platform. See the “Interpreting System Markers” topic for additional detail on when and why the Rational Software Development Platform system marker may not exist.

### ***System markers***

Table 1 lists the products, system markers, and associated features that can be used to identify what products are installed and the location of the associated Rational Software Development Platform instance.

**Table 1 Rational Software Development Platform system markers**

<b>Product</b>	<b>System Marker Key</b>	<b>Feature ID</b>
Rational Software Development Platform	components\com.ibm.rational.sdp	com.ibm.rational.sdp.feature
Rational Web Developer	products\com.ibm.rational.web.developer	com.ibm.rational.web.developer
Rational Application Developer	products\com.ibm.rational.application.developer	com.ibm.rational.application.developer
Rational Systems Developer	products\com.ibm.rational.systems.developer	com.ibm.rational.systems.developer
Rational Software Modeler	products\com.ibm.rational.software.modeler	com.ibm.rational.software.modeler
Rational Software Architect	products\com.ibm.rational.software.architect	com.ibm.rational.software.architect
<hr/>		
WebSphere Development Studio Client	products\com.ibm.rwdi	com.ibm.rwdi.prod.feature
WebSphere Development Studio Client Advanced Edition	products\com.ibm.radi	com.ibm.radi.prod.feature
WebSphere Developer for zSeries	products\com.ibm.websphere.developer.zseries	com.ibm.websphere.developer.zseries.product
<hr/>		
Rational Software Development Platform Trial	No key defined when a trial product is installed	com.ibm.rational.sdp.feature
Rational Web Developer Trial	products\com.ibm.rational.web.developer.trial	com.ibm.rational.web.developer
Rational Application Developer Trial	products\com.ibm.rational.application.developer.trial	com.ibm.rational.application.developer
Rational Systems Developer Trial	"no marker defined"	com.ibm.rational.systems.developer
Rational Software Modeler Trial	products\com.ibm.rational.software.modeler.trial	com.ibm.rational.software.modeler
Rational Software Architect Trial	products\com.ibm.rational.software.architect.trial	com.ibm.rational.software.architect

The technique used to implement system markers is platform dependant. On the Windows platform system markers are implemented as registry keys. These keys are prefixed with:

```
HKEY_LOCAL_MACHINE\SOFTWARE\  
IBM\Rational\Software Development Platform
```

On non-Windows platform system markers are implemented in a file system structure. The file system structure begins with:

```
etc\IBM\Rational\SoftwareDevelopmentPlatform
```

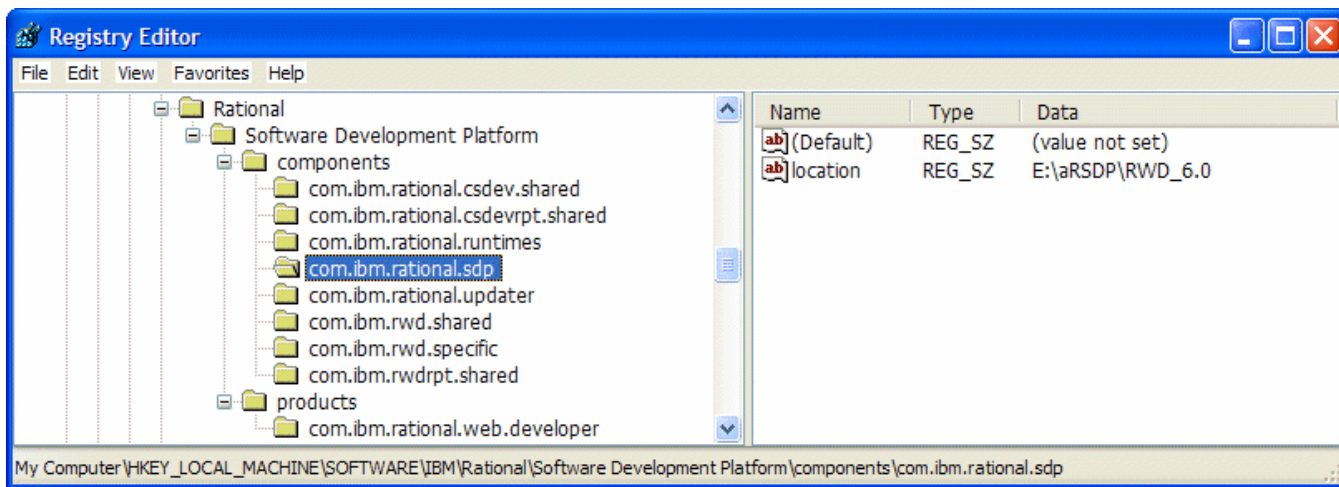
When one product upgrades another, a single product key in the registry will identify the last product installed. If the user has installed Rational Web Developer and then later installed Rational Application Developer, then Rational Application Developer is the only product listed. Some products do not upgrade, but instead enhance the existing product. When this occurs, then more than one product key will be found. Figure 2 showed these installation and upgrade paths.

There are other special considerations for how you can interpret the system markers. These are covered in the “Interpreting System Markers” topic after the examples.

### Windows platform example

The system marker concept is implemented on the Windows platform using the Windows Registry. A registry key structure is used to represent the installed components, products, and the location of the IBM Software Development Platform.

Figure 3 shows the system markers that are created as part of the installation of Rational Web Developer on a Windows platform, the components branch that identifies the location of the IBM Software Development Platform is selected to show the location attribute.



**Figure 3 Rational Web Developer System Markers Implemented as Registry Keys**

In this example the target directory used during the install was E:\aRSDP\RWD\_6.0. The Eclipse Platform instance would be found at E:\aRSDP\RWD\_6.0\eclipse.

### Non-Windows platform example

The system marker concept is still valid on other platforms but the file system is a substitute for the Windows Registry. Regardless of target (AIX, Linux), a folder and file structure is used to represent the same awareness of installed components, products, and the location of the IBM Rational Software Development Platform.

This is the system marker set for an install of Rational Application Developer on a Linux platform:

---

```
etc\IBM\Rational\SoftwareDevelopmentPlatform\components
  com.ibm.rad.shared
  com.ibm.rad.specific
  com.ibm.rad.rsm.shared
  com.ibm.rational.csdev.shared
  com.ibm.rational.csdevrpt.shared
  com.ibm.rational.runtimes
  com.ibm.rational.sdp
  com.ibm.rational.updater
  com.ibm.rwd.shared
  com.ibm.rwdrpt.shared

etc\IBM\Rational\SoftwareDevelopmentPlatform\products
  com.ibm.rational.application.developer
```

---

The file `com.ibm.rational.sdp` contains the following information:

```
location=/opt/IBM/Rational/SDP/6.0
```

The file `com.ibm.rational.application.developer` contains the following information:

```
location=/opt/IBM/Rational/SDP/6.0
version=6.0
name=IBM Rational Application Developer V6.0
```

In this example `/opt/IBM/Rational/SDP/6.0` was the target directory used for the install. The Eclipse Platform instance would be found at `/opt/IBM/Rational/SDP/6.0/eclipse`.

### ***Interpreting system markers***

The system marker data shown for the Windows and non-Windows platforms is straightforward in that only one product was installed. It is easy to identify the product and the Eclipse Platform location when only one product is installed. When multiple products have been installed things get a bit more complex. First, we will describe the system marker structure and provide pseudo-code to guide you when reading the system markers in your installation routine.

### **System marker structure**

Depending on the products installed, and the order of product installation, there can be subtle differences in the system markers that will exist on a target system.

When the WebSphere Development Studio Client for iSeries or WebSphere Development Studio Client Advanced Edition for iSeries products are the first products installed they do not create this system marker for the shared Rational Software Development Platform base:

```
\components\com.ibm.rational.sdp
```

In this situation the location of the shared Rational Software Development Platform base can only be determined by reading the value of the associated product system marker; the product system marker for WebSphere Development Studio Client Advanced Edition for iSeries is shown below:

```
IBM\Rational\Software Development Platform  
  \products\com.ibm.radi
```

When you install the Rational Systems Developer product, it creates a non-shared Rational Software Development Platform base instance. The Rational Systems Developer installer ignores any shared Rational Software Development Platform base instance that may have been installed by another product. To identify the install location for the Rational Systems Developer product you must use the Rational Systems Developer product system marker:

```
IBM\Rational\Software Development Platform  
  \products\com.ibm.rational.systems.developer
```

You may want to extend both Rational Systems Developer and some other shared Rational Software Development Platform base instance product; this means you will have to use different system markers to identify the location of the Rational Software Development Platform instances you wish to extend:

- For Rational Systems Developer:  
 \products\com.ibm.rational.systems.developer
- For a product that uses a shared Rational Software Development Platform instance:  
 \components\com.ibm.rational.sdp
- When WebSphere Development Studio Client for iSeries or WebSphere Development Studio Client Advanced Edition for iSeries is installed first and adds the shared Rational Software Development Platform instance:  
 \products\com.ibm.radi

Therefore, as you can see, the logic required in your installer must address each of these situations if you intend to support all the products that are based on the Rational Software Development Platform. Table 2 provides additional detail on using the system markers to locate the platform or a specific product.

**Table 2 System markers used to locate Rational Software Development Platform base**

<b>Product</b>	<b>System Marker to locate Base</b>	<b>Notes</b>
Rational Software Modeler	\components\com.ibm.rational.sdp	
Rational Web Developer	\components\com.ibm.rational.sdp	
Rational Application Developer	\components\com.ibm.rational.sdp	
Rational Software Architect	\components\com.ibm.rational.sdp	If added to a WebSphere Development Studio Client for iSeries or WebSphere Development Studio Client Advanced Edition for iSeries install, the WebSphere Development Studio Client for iSeries or WebSphere Development Studio Client Advanced Edition for iSeries product system marker is used to locate the shared Rational Software Development Platform base.
Rational Systems Developer	\products\com.ibm.rational.systems.developer	
WebSphere Development Studio Client	\products\com.ibm.rwdi	If Rational Web Developer was installed first, then the standard Rational Software Development Platform base component system marker would be used.
WebSphere Development Studio Client Advanced Edition	\products\com.ibm.radi	If Rational Application Developer was installed first, then the standard Rational Software Development Platform base component system marker would be used.
WebSphere Developer for zSeries	\components\com.ibm.rational.sdp	The Rational Application Developer installation that precedes the installation of WebSphere Developer for zSeries adds this system marker.

**Pseudo-code for reading system markers**

There are many improbable installation scenarios that will probably never be seen (such as installing Rational Software Architect and then WebSphere Development Studio Client for iSeries), but there is a strategy that should account for all possible situations. This is a pseudo-code rendering of that strategy:

---

```
// Check first for Rational Systems Developer as it is unique
if (\products\com.ibm.rational.systems.developer exists) {
    1) Offer Rational Systems Developer as a product target
    2) Use the location attribute of the product Rational Systems Developer
       key as the target for configuration updates
}

// Check for the shared base, it drives location when found
if (\components\com.ibm.rational.sdp exists) {
    1) Offer the entries under \products\, less Rational Systems Developer,
       as a product target
    2) Use the location attribute for the component key as the target
       for configuration updates

// Check for WebSphere Development Studio Client Advanced Edition for iSeries
} else if (\products\com.ibm.radi exists) {
    1) Offer WebSphere Development Studio Client Advanced Edition
       for iSeries as a product target
    2) Use the location attribute of this key as the target
       for configuration updates

// Check for WebSphere Development Studio Client for iSeries
} else if (\products\com.ibm.rwdi exists) {
    1) Offer WebSphere Development Studio Client
       for iSeries as a product target
    2) Use the location attribute of this key as the target
       for configuration updates
}
```

---

### ***Eclipse product and extension marker files***

All Eclipse-based products are required to include an `.eclipseproduct` marker file in the `\eclipse` directory that contains the Eclipse Platform base and launch point (`eclipse.exe`). Extensions to Eclipse or an Eclipse-based product are required to include an `.eclipseextension` marker file in the `\eclipse` directory that contains the additional features and plug-ins added to an Eclipse configuration. The existence of an `.eclipseextension` marker file is a requirement in Ready for IBM Rational software validation program.

Each marker file contains values for the name, id, and version property keys. These values reflect the installed product or extension. As multiple products can share a Rational Software Development Platform instance, the associated `.eclipseproduct` file will never reflect any one product, but only the Rational Software Development Platform itself.

There are example installation routines on [eclipse.org](http://eclipse.org) that use the `.eclipseproduct` marker file as the method to locate an Eclipse-based product. While a file system scan does work, it is not a valid technique for the discovery of a Rational Software Development Platform instance. If you want your installation routine to pass the Ready for IBM Rational software validation, the routine must use the system markers to detect and locate a Rational Software Development Platform-based product to extend. The system markers are much faster than a file system scan as well.

**Note:** The installation of a Rational Software Development Platform instance may result in the existence of both the `.eclipseproduct` and `.eclipseextension` marker files in the Rational Software Development Platform target directory. The `.eclipseproduct` file contents may not even reflect the Rational Software Development Platform name, id, and version. While this data is invalid, it will not be a problem for your installer as the system markers are the defined technique for locating the Rational Software Development Platform install location and any associated products.

## Identifying optionally installed features

Your software may need features that are not always available in the target product. In this situation, your installation routine should validate that any required features are in the list of available features before attempting to install your extension. If you do not find the required features, you may need to either install these features or terminate the installation. You may want to install these features as a separate extension.

You can identify if optional features are available in the product configuration. The Eclipse 3.0 platform has a command API that allows you to request a list of all configured features. This command:

```
java.exe -cp startup.jar org.eclipse.core.launcher.Main  
-data tempwork -application org.eclipse.update.core.standaloneUpdate  
-command listFeatures
```

Shown below is the output of the `listFeatures` command when run against an installed instance of Rational Web Developer (list edited for brevity):

---

```
Site: file:/E:/aRSDP/RWD_6.0/rwd/eclipse/  
  Feature: com.ibm.wtp.server.tomcat 6.0.0 enabled  
  Feature: com.ibm.debug.javascript 6.0.0 enabled  
  Feature: com.ibm.etools.ftp 6.0.0 enabled  
  ...  
Site: file:/E:/aRSDP/RWD_6.0/sdpisv/eclipse/  
  Feature: com.ibm.rational.sdp.isvinstall 6.0.0 enabled  
Site: file:/e:/aRSDP/RWD_6.0/eclipse/  
  Feature: com.ibm.rational.welcome.feature 6.0.0 enabled  
  ...  
  Feature: org.eclipse.platform 3.0.1.1 enabled  
  ...  
  Feature: org.eclipse.pde 3.0.1 enabled  
  ...  
  Feature: org.eclipse.jdt 3.0.1.1 enabled  
  Feature: com.ibm.rational.common.activities.feature 6.0.0 enabled  
Site: file:/E:/aRSDP/RWD_6.0/csdevrpt_shared/eclipse/  
  Feature: org.eclipse.xsd 2.0.1 enabled  
Site: file:/E:/aRSDP/RWD_6.0/csdev/eclipse/  
  Feature: com.ibm.xtools.common.ui.properties 6.0.0 enabled  
  Feature: com.ibm.xtools.comparemerge.feature 6.0.0 enabled  
  Feature: org.eclipse.gef 3.0.1 enabled
```

```
Site: file:/E:/aRSDP/RWD_6.0/rwdrpt_shared/eclipse/  
...  
Feature: org.eclipse.hyades 3.0.1.1 enabled  
...  
Site: file:/E:/aRSDP/RWD_6.0/rwd_prod/eclipse/  
Feature: com.ibm.support.rwd.doc 1.0.0 enabled  
Feature: com.ibm.rwd.nav.doc.feature 6.0.0 enabled  
Feature: com.ibm.rational.web.developer.product 6.0.0 enabled  
Feature: com.ibm.rwd.prod.welcome.feature 6.0.0 enabled  
Command completed successfully.
```

---

The install site is where the Rational Software Development Platform base is installed is:

```
Site: file:/e:/aRSDP/RWD_6.0/eclipse/
```

### Notes:

- IBM Software Development Platform products offer a variety of features, including many from eclipse.org. If you need certain eclipse.org features then you must establish a prerequisite for the corresponding IBM Rational Software Development Platform product that includes these features.
- If you want to add an eclipse.org feature that is not available in any of the IBM Software Development Platform products, you may do so, but these eclipse.org features must be installed in the same location as your Extension; you are responsible for supplying maintenance to both your software and any eclipse.org features that you add.

### References

There topics you can review if some of the concepts discussed in this document are not familiar to you. See these links for more information:

- Eclipse Update Manager API - [http://help.eclipse.org/help30/topic/org.eclipse.platform.doc.isv/reference/misc/update\\_standalone.html](http://help.eclipse.org/help30/topic/org.eclipse.platform.doc.isv/reference/misc/update_standalone.html)
- Eclipse Product Extensions - [http://help.eclipse.org/help30/topic/org.eclipse.platform.doc.isv/guide/product\\_extension.htm](http://help.eclipse.org/help30/topic/org.eclipse.platform.doc.isv/guide/product_extension.htm)
- Eclipse Update Sites – Eclipse.org article “How to Keep Up To Date” <http://www.eclipse.org/articles/Article-Update/keeping-up-to-date.html>



© Copyright IBM Corporation 2006  
Produced in the United States of America  
03-2006  
All Rights Reserved

IBM, the IBM logo, and Rational trademarks of IBM Corporation in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries or both.

Other company, product and service names may be trademarks or service marks of others.