

HP OpenView Configuration Server Using Radia

for the HP-UX, Linux, Solaris, and Windows server platforms

Software Version: 4.5.4

Radia Database Utility (RadDBUtil) Guide

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

© Copyright 1998-2005 Hewlett-Packard Development Company, L.P.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

Trademark Notices

Linux is a registered trademark of Linus Torvalds.

Microsoft®, Windows®, and Windows® XP are U.S. registered trademarks of Microsoft Corporation.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Acknowledgements

PREBOOT EXECUTION ENVIRONMENT (PXE) SERVER
Copyright © 1996-1999 Intel Corporation.

TFTP SERVER
Copyright © 1983, 1993
The Regents of the University of California.

OpenLDAP
Copyright 1999-2001 The OpenLDAP Foundation, Redwood City, California, USA.
Portions Copyright © 1992-1996 Regents of the University of Michigan.

OpenSSL License
Copyright © 1998-2001 The OpenSSLProject.

Original SSLeay License
Copyright © 1995-1998 Eric Young (ey@cryptsoft.com)

DHTML Calendar
Copyright Mihai Bazon, 2002, 2003

Support

Please visit the HP OpenView support web site at:

<http://www.hp.com/managementsoftware/support>

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

<http://www.managementsoftware.hp.com/passport-registration.html>

Documentation Updates

This manual's title page contains the following identifying information:

- Version number, which indicates the software version.
- Print date, which changes each time the document is updated.

To check for recent updates or to verify that you are using the most recent edition, visit:

http://ovweb.external.hp.com/lpe/doc_serv/

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

The following table indicates the changes that were made to this document since the previous edition.

Table 1 Document Changes

Chapter	Version	Changes
N/A	4.5.4	Page 9, revised the text of the bullets in the section, IMPORT.
N/A	4.5.4	Page 10, corrected the syntax in the examples.
N/A	4.5.4	Page 11, Table 3, revised the syntax for the COMMIT keyword.
N/A	4.5.4	Page 12, Table 3, revised the recommended syntax for the REUSE keyword.
N/A	4.5.4	Page 13, Table 3, revised the explanation for the ACCEPT, REJECT, and IGNORE keywords.
N/A	4.5.4	Page 14, in the section, Conditional Files, revised the description of the XPI and XPR files.
N/A	4.5.4	Page 18, in the section, Examples: <ul style="list-style-type: none">• revised the syntax of the examples,• restructured the numbering of the examples, and• added several new examples.
N/A	4.5.4	Page 18, added a Note indicating that two syntax formats are supported.
N/A	4.5.4	Page 20, revised Example 7.

Contents

- Introduction..... 6
 - Requirements & Prerequisites.....6
 - Product Versions6
 - Platforms6
 - Components & Processes6
 - Components6
 - Processes.....7

- Implementation Details7
 - HP OpenView Patch Manager Using Radia Considerations.....7
 - IMPORT.....7
 - EXPORT7
 - EDMPROF File Settings.....7
 - RadDBUtil Verbs.....8
 - General Syntax.....8
 - VERSION.....8
 - LOG9
 - IMPORT.....9
 - EXPORT14
 - DELETE16
 - RCS17
 - Return Codes.....18
 - Examples18
 - IMPORT Examples18
 - EXPORT Examples19

Introduction

This document provides information about the HP OpenView Using Radia database tool, `raddbutil.exe`, which manages Radia database updates (imports, exports, and deletions), as well as activity logging, Configuration Server communications, and version queries. This document focuses on the functionality, syntax, and common-use attributes of this tool, and provides examples of each of these capabilities.

- ▶ HP recommends creating a back up the Radia database prior to executing any of the commands that are shown in this document.

Requirements & Prerequisites

Product Versions

- HP OpenView Configuration Server Using Radia (Configuration Server), version 4.5.4
 - ▶ The Configuration Server does not need to be running in order for RadDBUtil to run.
- Radia database, version 3.0 and greater

Platforms

Table 2 Supported Operating Systems and Levels

Platform	Operating System and Level
Windows	NT 4.0 Server, Service Pack 6
	2000 Server, Service Pack 3
	2003 Server, Service Pack 1
	XP Professional, Service Pack 2
UNIX	HP-UX (PA-RISC 1.1 and 2.0), Version 10.20
	Red Hat Enterprise Linux, ES Version 3.0
	Red Hat Enterprise SuSE Enterprise Server, Version 9.0
	Solaris, Version 2.7

Components & Processes

This section details the components and processes that benefit from using RadDBUtil.

Components

- Radia database
- Configuration Server
- HP OpenView Distributed Configuration Server Using Radia (Distributed Configuration Server)

Processes

Importing/exporting to/from the Radia database; output materials produced by, or during, importing and exporting; deleting instances and resources from the Radia database; querying and manipulating the Configuration Server lock status; version information queries; activity logging

Implementation Details

RadDBUtil has a dependency on the **edmprof file**, and should be placed in the same directory as it—typically the Configuration Server's `bin` directory on Windows, and `exe` directory on UNIX. Additionally, RadDBUtil must be able to find a valid Radia database and the Configuration Server `log` directory.

- ▶ HP recommends creating a back up the Radia database prior to executing any of the commands that are shown in this document.

HP OpenView Patch Manager Using Radia Considerations

This section contains important information and warnings about using RadDBUtil to import and export HP OpenView Patch Manager Using Radia (Patch Manager) bulletins.

- ▶ Important information regarding the deleting of Patch Manager bulletins is detailed in the section, [Deleting Bulletins from a Database](#), on page 17.

IMPORT

- ⚠ When using the **-domain** switch, all instances and resources (of the XPI file that is specified for INPUT) will be imported to the target domain. Therefore, it is imperative that the necessary domains and classes exist in the target domain.

EXPORT

- ▶ In order for the RadDBUtil tool to successfully export bulletins, in the `PATCHMGR.ZSERVICE` class template there must be an attribute named `SYNC` of the type `CONNECTION` with the appropriate default value.
See [Export](#) on page 20 for an example of the `SYNC-CONNECTION` attribute being specified for a Patch Manager bulletin.
If this attribute is not present in the `PATCHMGR.ZSERVICE` class template, it must be added.

EDMPROF File Settings

The following `edmprof` file settings affect the operation of RadDBUtil. Be sure to verify these settings and adjust them accordingly.

MGR_LOG_DIRECTORY

specifies where the activity and audit logs will be generated. For more information, see [Standard Files](#) on page 13.

MGR_DIRECTORIES.DBPATH

determines the Radia database location.

MGR_STARTUP.MGR_ID

is the unique, three-character identifier of the Configuration Server that was specified during its installation.

MGR_STARTUP.MGR_NAME

is the identifying name of the Configuration Server that was specified during its installation.

RadDBUtil Verbs

In this section, each of the six RadDBUtil verbs is detailed with syntax options, syntax descriptions, and verb-specific considerations.

**UNIX Note: Using Quotation Marks**

When working in a UNIX environment, it is important that any string that contains special characters—such as parentheses, (and)—be enclosed within quotation marks.

General Syntax

- The executable, verbs, keywords, and values are not case-sensitive.

**UNIX Note: Filename Case-sensitivity**

The RadDBUtil executable, `raddbut il`, must be specified exactly as it appears; otherwise it will not work.

- The verbs are VERSION, LOG, IMPORT, EXPORT, DELETE, and RCS.
- The verbs can be specified in either of the following formats:

-keyword value

If this syntax is used, pairs of combinations must be separated by a space, as in:

-keyword value -keyword value -keyword value

or

keyword=value

If this syntax is used, pairs of combinations must be separated by a space, a comma, or both, as in:

keyword=value keyword=value,keyword=value, keyword=value



A double-dash (--) indicates the end of the options.

The acceptable Boolean values are:

TRUE: 1, YES, ON, and TRUE

FALSE: 0, NO, OFF, and FALSE

VERSION

The VERSION verb produces build information for the RadDBUtil tool and all embedded executables. For examples of the syntax, see Examples on page 18.

Syntax

The syntax of the verb, VERSION, is shown below.

```
Radddbutil version
```

LOG

This verb places RadDBUtil-specific messages in the audit log and activity log.

The audit log information is:

- BuildInfo
- CommandLine
- CompletionCode
- Timestamp (ISO)

There are two messages per command—one when RadDBUtil begins and one when it ends.

```
20050126 15:24:04 C:/RadDBUtil/radddbutil.exe 60 --> import
20050126 15:24:04 C:/RadDBUtil/radddbutil.exe 60 <-- rc=8
```

Syntax

The syntax of the LOG verb is shown below. For examples of the syntax, see Examples starting on page 18.

```
RadDBUtil LOG "record this information"
```

The RadDBUtil entry in the log can be accompanied by customized text.

The text must be enclosed within quotation marks, as shown above. For example, to delineate a nightly log entry for a specific date, specify:

```
RadDBUtil log "Nightly Log for June 22, 2005"
```

IMPORT



HP recommends creating a back up the Radia database prior to executing any of the commands that are shown in this section.

The IMPORT verb simplifies the importing of materials into the Radia database. It can import materials from one domain into a domain of a different name within a Radia database, and from one Radia database to another. It offers the following options in order to optimize the import operation.

- Automatically recognize and re-use the instances that exist in the target domains, and
- Import only those elements that do not exist in the target domains.

Additionally, by having a feature that allows the IMPORT verb to identify packages and dialogs that exist in the database and to dynamically adapt to them, RadDBUtil reduces the size of the targeted domain—as well as Distributed Configuration Server execution times—without impacting the integrity of the imported materials. Other features of the IMPORT verb are:

- Parameter validation
- Database receptiveness to receiving the import materials
- Deck verification (XPI and XPR)
- Attributes that are dropped from the import operation will be noted with a warning level (--?) in the activity log.



For the IMPORT verb, an “all-or-nothing” rule applies—that is, if RadDBUtil *cannot* do *all* of that which is requested, it will do *none* of that which is requested.

If a RadDBUtil IMPORT operation fails, the Configuration Server log will have an entry reflecting this and the return code will be rc=8.

Syntax

The syntax of the IMPORT verb is shown in this section. For examples of the syntax, see Examples on page 18.

- The keywords and values are not case-sensitive, as can be seen in the following examples.



Optional keyword-value combinations are in parentheses.
Default values are underlined.

```
Radddbutil import -input FileName (-output value) (-domain value)
(-commit false) (-root=*) (-accept a) (-reject u+d) (-ignore s)
raddbutil IMPORT INPUT=value(,OUTPUT=value)(,DOMAIN=domain_name
{:REUSE})(,COMMIT=TRUE/FALSE)(,ROOT=*)(,ACCEPT=A)(,REJECT=U+D)(,IGNORE=S)
```

IMPORT Keywords

Table 3 below lists and defines the keywords for the verb, IMPORT.

Table 3 RadDBUtil IMPORT Keywords

Keyword	Explanation
INPUT	<p>This is the prefix (the fully specified drive and path) of the input files.</p> <ul style="list-style-type: none"> • If a fully specified drive and path is not provided with the input file name, the current directory is used. • If .xpi is specified, it will be stripped off. • This keyword is mandatory; it does not have a default value. • The associated resource file must be in the same location and have the same name, but with the extension .XPR. <p> This is the only location that will be searched if any of the import Instances require resource data.</p> <ul style="list-style-type: none"> • If the fully specified drive and path contains blanks or other special characters, it is necessary to enclose in quotation marks the entire file identifier, including the drive letter and all directories.

Keyword	Explanation
OUTPUT	<p>This is the prefix of the output files (the merged “accepts” and “ignores”). Specify a filename to be used for the result of importing this and any modifications to the input media that were required due to events such as duplicate OIDs, different domains, different parentage, etc.</p> <ul style="list-style-type: none"> • This value is required if any of the output of the modified XPI and XPR decks is required. It defines the directory and name, in XPI and XPR, of the generated, modified output. • If <code>.xpi</code> is specified, it will be stripped off. <p> Although this keyword is optional, there is no default— if it is omitted or specified without a valid value, no output files other than logs will be available after the completion of RadDBUtil.</p>
COMMIT	<p>Specifies whether the Radia database will be updated.</p> <p>-commit true allows those elements that meet the import criteria to be passed into the Radia database for processing.</p> <p>-commit false (the default) forces RadDBUtil to only scan and reconcile the input files and the database.</p>
ROOT	<p>Specifies the root instance of the model (either <code>FILE.DOMAIN.CLASS</code>, <code>.INSTANCE</code>, <code>DOMAIN.CLASS.INSTANCE</code>, <code>CLASS.INSTANCE</code>, or <code>CLASS</code>).</p> <ul style="list-style-type: none"> • Only matching instances and those that are referenced (directly or indirectly) by root will be processed. • The default is <code>*</code>. • <code>COMPONENT</code> classes and instances cannot be specified.
DOMAIN	<p>Specifies an existing Radia database domain into which this import deck is to be placed.</p> <ul style="list-style-type: none"> • If the specified value does not correctly identify an existing domain in the target database, RadDBUtil will end with an error. <p> If a domain is specified, the input files will be imported into that Radia database domain. However, if no domain is specified (the default), the input files will be imported into the domains that are specified in the decks.</p> <ul style="list-style-type: none"> • All of the instance and resource data that are contained in the <code>INPUT-</code> specified decks will be examined and considered for import into the single Radia database domain that is specified by this keyword. • Domain names in the Radia database are limited to eight characters. <p> The -domain switch is designed to put the contents of the input decks into a single destination. Therefore, when importing a Patch Manager bulletin, do not use this switch because if the instances are coming from multiple domains they must be imported back into multiple domains.</p>

Keyword	Explanation
REUSE	<p data-bbox="487 210 1039 241">Specify as: -domain domain_name:REUSE</p> <p data-bbox="487 252 1461 409">This is an optional directive that allows RadDBUtil to compare the target database and import deck for matching package instances and, when present, re-use those that are in the database. This allows RadDBUtil to avoid creating duplicates of package instances and resources in the target domain if they already exist, in either the original source or target domain.</p> <ul style="list-style-type: none"> <li data-bbox="535 430 1461 525">• <i>For each non-root and non-component element in the deck:</i> the domain that is specified by the DOMAIN keyword is searched first, followed by the domain that is identified in the package instance of the import deck. <ul style="list-style-type: none"> <li data-bbox="568 535 1429 598">— If a package instance has the <i>same name</i> as the package instance that is found in the Radia database, RadDBUtil will: <ul style="list-style-type: none"> <li data-bbox="609 619 1429 682">– ABORT if it is determined that the contents are different from that which is in the import deck. <li data-bbox="609 693 1445 850">– REUSE it if it is determined that the contents are identical to that which is in the import deck. This means that the existing database package instance will be referenced by the appropriate connection attributes in the import deck, and the matching Package Instances in the import deck will be removed. <p data-bbox="568 861 1461 1060"> Although the :REUSE option might remove duplicates from the import process, the output files (INPUT.XPI and INPUT.XPR) will contain the instances and resources that were in the original import deck, and which could have been used to update the database but were deferred as a result of this option.</p>

Keyword	Explanation
ACCEPT, REJECT, and IGNORE	<p>The instances in the import deck are compared to the instances in the database. These optional keywords dictate the action—based on that comparison—that RadDBUtil is to take on the import deck and database instances.</p> <p>There are four instances <i>types</i>. They are:</p> <ul style="list-style-type: none"> • Adds (A) are instances that are in the import deck and are to be added to the database. The default behavior is to ACCEPT these additions. • Deletes (D) are instances that are to be deleted from the database. The default behavior is to REJECT these deletions. • Sames (S) are instances in the import deck that are identical to those in the database. The default behavior is to IGNORE these instances. • Updates (U) are instances in the database that will be updated by a matching instance in the import deck. The default behavior is to ACCEPT these instances. <p>▶ These actions (ACCEPT, REJECT, and IGNORE) are applicable only to these four instance types (A, D, S, and U), and act on only those instance types that have been specified for each.</p> <p>The following points are additional items for consideration when using these keywords.</p> <ul style="list-style-type: none"> • An instance type cannot be specified for multiple actions in a single RadDBUtil execution. That is, Adds cannot be configured to be accepted <i>and</i> ignored. • Multiple instance types can be specified for one action (as in, -reject d+u). <p>▶ If multiple instance types are specified for an action, a plus sign (+) must separate them.</p> <ul style="list-style-type: none"> • Any operations that match the REJECT parameters will cause the tool to not commit, and will return a non-zero return code when processing has completed.

MSI Files

RadDBUtil has an option that allows an import error to be overruled if inconsistent MSI files are discovered. This might occur if the package with the materials matching the IDX file has been renamed (and, as a result, cannot be found by either import or tree export), or where the ACP file might not yet have been imported into the database, but is in a set of imports that is to be processed subsequent to the current import materials.

The command to overrule the import error is IGNORE=BADMSI.

Output Files

This section provides information about the log files that will be automatically generated by the IMPORT verb. Also, the section, Conditional Files (on page 14), discusses additional logs and files, and the conditions under which they might be generated.

- **Standard Files**

— **RADDBUTIL.LOG**

located in the directory specified by `DIRECTORY` in the `MGR_LOG` section of the `edmprof` file. This is a text file that records the actions (activity log) taken by RadDBUtil in processing each command invocation, and the results of these actions. Each execution of RadDBUtil overwrites the previous log.

- ▶ To save the logs of previous `RADDBUTIL.EXE` executions, rename the (`RADDBUTIL.LOG`) file.

This log queries the `edmprof` file and identifies the location of the `edmprof` file, the Radia database, and the Configuration Server log. It also contains return codes and summary information about execution results.

— **RADDBUTIL.AUDIT.LOG**

located in the directory specified by `DIRECTORY` in the `MGR_LOG` section of the `edmprof` file. This file contains a record of all RadDBUtil calls and the corresponding return codes; it is designed for archival reference only.

— **STDERR**

contains the same information as `RADDBUTIL.LOG` but, by default, is directed to the console. This log can be redirected as desired.

- **Conditional Files**

XPR and XPI Files

If RadDBUtil import updates the database, these two files will be created in the directory that is optionally specified by the keyword `OUTPUT`. The contents of these files can be returned to the customer's digital source library (DSL) as a record of the materials as imported into the target Radia database.

EXPORT

- ▶ HP recommends creating a back up the Radia database prior to executing any of the commands that are shown in this section.

The `EXPORT` verb allows for the simultaneous exporting of the XPC, XPI and, optionally, the XPR decks that are needed to ensure that the exported portions are accurately reproduced in another database. This includes class templates, instances and, optionally, the resources. It also allows the specifying of the entire database, or individual parts of the database (such as domain, class, instance, package, and service) to be exported. `EXPORT` offers the ability to:

- Perform deletions based on the results of object resolution.
- Specify multiple inputs, such as exporting four services on one export operation.

Additionally, exporting can include:

- The associated resources, and
- All required packages (similar to a client resolution).

Syntax

The syntax of the `EXPORT` verb is shown below. For examples of the syntax, see Examples on page 18.

- ▶ Optional keyword-value combinations are in parentheses.
Default values are underlined.

```
RADDBUTIL EXPORT ( ,DATA=TRUE | FALSE) ( ,WALK=TRUE | FALSE)  
( ,OUTPUT=stemname) INPUT
```

```
Raddbutil export (-data 0/1) (-walk 0/1) (-output stemname) input
Raddbutil export (-data NO/YES) (-walk no/yes) (-output stemname)
input
Raddbutil export (-data false/true) (-walk FALSE/TRUE) (-output
stemname) input
```

EXPORT Keywords

Table 4 below lists and defines the keywords for the EXPORT verb.

Table 4 RadDBUtil EXPORT Keywords

Keyword	Explanation
DATA	Indicates whether to export the resource files in the Radia database. The default is 0 (FALSE, NO).
WALK	Indicates whether to do a resolution—Radia database should be traversed. The default is 1 (TRUE, YES). ▶ INPUT is not a keyword, like the others; it is a documentation placeholder. A value must be specified without “input” being used as an indicator, as shown in this table.
OUTPUT	This is the prefix of the output files. <ul style="list-style-type: none"> • If .xpi is specified, it will be stripped off. • This keyword does not have a default value.
<i>INPUT</i>	This, the only mandatory parameter, indicates the Radia database Instances that are to be exported. ▶ INPUT is not a keyword, like the others; it is a documentation placeholder. A value must be specified without “input” being used as an indicator, as shown below. Raddbutil export (-data 0/1) (-walk 0/1) (-output <i>stemname</i>) <i>file.domain.class.*</i> <ul style="list-style-type: none"> • The format can be either: <ul style="list-style-type: none"> — FILE.*.*.*, — FILE.DOMAIN.*.*, — FILE.DOMAIN.CLASS.*, — FILE.DOMAIN.CLASS.INSTANCE, or — FILE.DOMAIN.CLASS.INSTANCE(msg). • Wildcards (*) are valid values for the domain, class, and instance keywords. • More than one database instance can be specified; multiples must be separated a blank space. ▶ The INPUT value must be specified at the end of the command line; otherwise the operation will fail.

Output Files

The EXPORT verb will always generate XPC and XPI files. If -data is specified, an XPR file will also be generated.

DELETE

- ▶ HP recommends creating a back up the Radia database prior to executing any of the commands that are shown in this section.

This verb deletes instances and resources from the Radia database. It offers the ability to perform deletions based on:

- The results of object resolution.
- The contents of an XPI file.

Syntax

The syntax of the DELETE verb is shown below. For examples of the syntax, see Examples on page 18.

- ▶ Optional keyword-value combinations are in parentheses. Default values are in underlined.

```
RADDBUTIL DELETE INPUT=value(,PREVIEW=TRUE | FALSE)(,FILE=value)  
(,WALK=TRUE | FALSE)(,IGNORE=value)
```

```
Raddbutil delete -input (-preview value) (-file value) (-walk value)  
(-ignore value)
```

DELETE Keywords

Table 5 below lists and defines the keywords for the verb, DELETE.

Table 5 RadDBUtil DELETE Keywords

Keyword	Explanation
PREVIEW	Indicates whether to preview the changes only, or make the deletions. The default is 0 (FALSE, NO, OFF) .
FILE	The name of XPI file that specifies what to is to be deleted from the Radia database.
WALK	Indicates whether to do a resolution—Radia database should be traversed. The default is 1 (TRUE, YES, ON) .
IGNORE	This keyword specifies (in) which Radia database instances should not be deleted. <ul style="list-style-type: none">• The format can be either F.D.C.I or f.d.c.i format.• Wildcards (*) are valid in the domain, class, and instance specifications.• More than one database instance can be specified; multiples must be separated a plus sign (+).

Keyword	Explanation
<i>INPUT</i>	<p>This parameter indicates which Radia database instances are to be deleted.</p> <p> INPUT is not a keyword, like the others; it is a documentation placeholder. A value must be specified without “input” being used as an indicator, as shown below.</p> <pre>Raddbutil delete file.domain.class.* (-walk 0/1)</pre> <ul style="list-style-type: none"> The format can be either: <ul style="list-style-type: none"> — FILE.*.*.*, — FILE.DOMAIN.*.*, — FILE.DOMAIN.CLASS.*, — FILE.DOMAIN.CLASS.INSTANCE, or — FILE.DOMAIN.CLASS.INSTANCE(msg). Wildcards (*) are valid in the domain, class, and instance specifications. <p>More than one database instance can be specified; multiples must be separated a blank space.</p>

Deleting Bulletins from a Database

To permanently delete Patch Manager bulletins from the Radia database, specify the following classes with the IGNORE keyword.

- PRIMARY.PATCHMGR.CMETHOD
- PRIMARY.PATCHMGR.OPTIONS
- PRIMARY.PATCHMGR.METADATA
- PRIMARY.SYSTEM.ZMETHOD
- PRIMARY.SYSTEM.PROCESS
- PRIMARY.PATCHMGR.PRODUCT
- PRIMARY.PATCHMGR.SP
- PRIMARY.PATCHMGR.RELEASE
- PRIMARY.PATCHMGR.PATCHARG

The following example shows these classes being included with the IGNORE option.

```
raddbutil delete -walk 1 -ignore PRIMARY.PATCHMGR.CMETHOD.*
+PRIMARY.SYSTEM.ZMETHOD.*+PRIMARY.SYSTEM.PROCESS.*
+PRIMARY.PATCHMGR.OPTIONS.*+PRIMARY.PATCHMGR.METADATA.*
+PRIMARY.PATCHMGR.PRODUCT.*+PRIMARY.PATCHMGR.SP.*
+PRIMARY.PATCHMGR.RELEASE.*+PRIMARY.PATCHMGR.PATCHARG.*
-preview 0 PRIMARY.PATCHMGR.ZSERVICE.MS05-015(SYNC)
```

RCS

This verb communicates with the Configuration Server’s database and allows for:

- Querying of the Configuration Server’s database **lock status**.

- Unlocking of the Configuration Server's database.

Syntax

The syntax of the RCS verb is shown below. For examples of the syntax, see the section, Examples.

```
raddbutil rcs status
```

```
raddbutil rcs unlock
```

RCS Keywords

Table 6 below lists and defines the keywords for the verb, RCS.

Table 6 RadDBUtil RCS Keywords

Keyword	Explanation
STATUS	Displays the lock status of the Configuration Server status.
UNLOCK	Unconditionally unlocks the Configuration Server.

Return Codes

Table 7 below shows the return codes that are associated with the RadDBUtil executable.

Table 7 RadDBUtil Return Codes

Return Code	Meaning
0	SUCCESS
4	WARNING
8 or 16	FAILURE <ul style="list-style-type: none"> • No database update occurred, or <ul style="list-style-type: none"> • A database update was started but not completed. Note: If the latter, the database might be in an error state.

Examples

This section presents a few examples of the simpler and more direct RADDBUTIL . EXE syntax.



As previously stated, the RADDBUTIL . EXE utility supports the following two syntax formats:

```
-keyword value
```

and

```
keyword=value
```

The examples in this section are presented in the **-keyword value** format.

IMPORT Examples

Example 1. Performing a simple import

Run a routine, daily import of the file, `sample.xpi` in order to add instances to the Radia database.

```
raddbutil import -input sample -commit yes
```

Abort if there are updates and/or deletes.

```
raddbutil import -input sample -accept A -reject U+D -commit yes
```

Ignore any updates and deletes.

```
raddbutil import -input sample -accept A -ignore U+D -commit yes
```

Results:

- The instances and resource data from the files `sample.xpi` and `sample.xpr` are imported directly into the domain and class that are specified in the input deck.
- No domain mapping is performed

Example 2. Performing a simple import and replacing the old instances

```
raddbutil import -input sample -accept A+U+D -commit yes
```

Example 3. Importing to a domain other than SOFTWARE

Import all instances from the input deck into the `SOFT0002` domain. Re-use any database elements that are identical to elements of the deck, overrule the inconsistent MSI file import error, and reject any updates.

Example 3a



The syntax of the following example is fully supported by RadDBUtil.

```
Raddbutil Import -Input Sample -Domain SOFT0002:Reuse -Commit Yes -  
Ignore Badmsi
```

Example 3b



The following example is the same as Example 2a, but includes the fully specified drive and path (with special characters) of the input material, and the name of the output decks.

```
RADDBUTIL IMPORT -INPUT "G:\CONTAINS BLANKS\SAMPLE" -OUTPUT  
Sample_Soft_2 -DOMAIN SOFT0002:REUSE -COMMIT YES -IGNORE BADMSI
```

EXPORT Examples

Example 4. An easy method by which to create export files

The following command will create `AMORTIZE.XPC` and `AMORTIZE.XPI` containing just the specified class and instance.

```
raddbutil export -output amortize PRIMARY.SOFTWARE.ZSERVICE.AMORTIZE
```

Example 5. Using the `-walk` command

The following command will create `AMORTIZE.XPC` and `AMORTIZE.XPI` containing the classes and instances. This command will resolve the package, and include any other required packages because `-walk` is specified.

```
raddbutil export -output amortize -walk 1 PRIMARY.SOFTWARE.ZSERVICE
.AMORTIZE
```

 No resources will be exported.

Example 6. Using the `-walk` and `-data` commands

Example 6a

The following command will create `AMORTIZE.XPC`, `AMORTIZE.XPI`, and `AMORTIZE.XPR` which contain the classes, instances, and resources (because `-data` is specified), and the package will be resolved because `-walk` is specified.

```
raddbutil export -output AMORTIZE -walk 1 -data 1 PRIMARY.SOFTWARE
.ZSERVICE.AMORTIZE
```

Example 6b

The following command will create `ALL.XPC`, `ALL.XPI`, and `ALL.XPR` containing the classes, instances, and resources of all services in the `SOFTWARE` domain.

```
raddbutil export -output ALL -walk 1 -data 1
PRIMARY.SOFTWARE.ZSERVICE.*
```

Example 7. Exporting and importing the PRIMARY File (ZEDMAMS used to import the class)

The following series of commands will export and then import the entire PRIMARY file of the Radia database.

```
raddbutil export -output ALL -walk 1 -data 1 PRIMARY.*.*.*
zedmams verb=import_class,file=ALL.xpc,preview=no,
logfile=ALL_PRIMARY.log
raddbutil import -input ALL -commit yes
```

Example 8. Exporting, deleting, and importing a bulletin

The following commands will export, delete, and import (respectively) the HP OpenView Patch Manager Using Radia bulletin, MS04-028.

 In the following examples, the value of `INPUT` should be enclosed in quotation marks on UNIX platforms.

Export

```
./raddbutil export -walk 1 -data 1 -output ms04-028
"PRIMARY.PATCHMGR.ZSERVICE.MS04-028(SYNC)"
```

Delete

```
raddbutil delete -walk 1 -ignore
PRIMARY.PATCHMGR.CMETHOD.*+PRIMARY.SYSTEM.ZMETHOD.*+
PRIMARY.SYSTEM.PROCESS.*+PRIMARY.PATCHMGR.METADATA.*+
PRIMARY.PATCHMGR.OPTIONS.*PRIMARY.PATCHMGR.ZSERVICE.
MS04-028(SYNC)
```

Import

```
raddbutil import -input ms04-028 -commit yes
```



In the import example, the **-domain** switch was not used because the instances are coming from multiple domains.

See the warning under DOMAIN in Table 3 on page 10.

Example 9. The DELETE verb

This series of example focuses on the DELETE verb commands.

- Delete the service PRIMARY.SOFTWARE.ZSERVICE.MSOFFICE.

```
raddbutil delete PRIMARY.SOFTWARE.ZSERVICE.MSOFFICE
```
- Delete the service PRIMARY.SOFTWARE.ZSERVICE.MSOFFICE; do not delete the methods.

```
raddbutil delete -walk 1 -ignore PRIMARY.SYSTEM.ZMETHOD.* PRIMARY  
.SOFTWARE.ZSERVICE.MSOFFICE
```
- Delete the instance PRIMARY.SOFTWARE.ZSERVICE.MSOFFICE.

```
raddbutil delete -walk 0 PRIMARY.SOFTWARE.ZSERVICE.MSOFFICE
```
- Delete the contents of foo.xpi.

```
raddbutil delete -file foo.xpi
```

Example 10. The Configuration Server's database

The following commands pertain to the Configuration Server's database.

- Show the current status of the Configuration Server's database.

```
raddbutil rcs status
```
- Unlock the Configuration Server's database.

```
raddbutil rcs unlock
```