Database Utilities

15 ADRCI: ADR Command Interpreter

The ADR Command Interpreter (ADRCI) is a command-line tool that you use to manage Oracle Database diagnostic data.

This chapter contains the following sections:

> **See Also:**
>
> *Oracle Database Administrator's Guide* (../../server.111/b28310/diag.htm#ADMIN11007) for more information on managing diagnostic data.

About the ADR Command Interpreter (ADRCI)

ADRCI is a command-line tool that is part of the fault diagnosability infrastructure introduced in Oracle Database Release 11*g*. ADRCI enables you to:

- View diagnostic data within the Automatic Diagnostic Repository (ADR).
- View Health Monitor reports.
- Package incident and problem information into a zip file for transmission to Oracle Support.

Diagnostic data includes incident and problem descriptions, trace files, dumps, health monitor reports, alert log entries, and more.

ADRCI has a rich command set, and can be used in interactive mode or within scripts. In addition, ADRCI can execute scripts of ADRCI commands in the same way that SQL*Plus executes scripts of SQL and PL/SQL commands.

ADR data is secured by operating system permissions on the ADR directories, hence there is no need to log in to ADRCI.

> **Note:**
>
> The easiest and recommended way to manage diagnostic data is with Oracle Enterprise Manager Support Workbench. ADRCI provides a command-line alternative to most of the functionality of Support Workbench, and adds capabilities such as listing and querying trace files.
> See *Oracle Database Administrator's Guide* (../../server.111/b28310/diag.htm#ADMIN11007) for complete information about the Support Workbench.

Definitions

The following are definitions of terms used in the context of ADRCI and the Oracle Database fault diagnosability infrastructure:

Automatic Diagnostic Repository (ADR)

The ADR is a file-based repository for database diagnostic data such as traces, dumps, the alert log, health monitor reports, and more. It has a unified directory structure across multiple instances and multiple products. Beginning with Release 11*g*, the database, Automatic Storage Management (ASM), and other Oracle products or components store all diagnostic data in the ADR. Each instance of each product stores diagnostic data underneath its own ADR home directory (see "ADR Home" (#BGBBEEHG)). For example, in an Oracle Real Application Clusters environment with shared storage and ASM, each database instance and each ASM instance has a home directory within the ADR. The ADR's unified directory structure enables customers and Oracle Support to correlate and analyze diagnostic data across multiple instances and multiple products.

Problem

A problem is a critical error in the database. Critical errors include internal errors such as `ORA-00600` and other severe errors such as `ORA-07445` (operating system exception) or `ORA-04031` (out of memory in the shared pool). Problems are tracked in the ADR. Each problem has a problem key and a unique problem ID. (See "Problem Key" (#BGBFBHAF) .)

Incident

An incident is a single occurrence of a problem. When a problem occurs multiple times, an incident is created for each occurrence. Incidents are tracked in the ADR. Each incident is identified by a numeric incident ID, which is unique within the ADR. When an incident occurs, the database makes an entry in the alert log, sends an *incident alert* to Oracle Enterprise Manager, gathers diagnostic data about the incident in the form of dump files (incident dumps), tags the incident dumps with the incident ID, and stores the incident dumps in an ADR subdirectory created for that incident.

Diagnosis and resolution of a critical error usually starts with an incident alert. You can obtain a list of all incidents in the ADR with an ADRCI command. Each incident is mapped to a single problem only.

Incidents are *flood-controlled* so that a single problem does not generate too many incidents and incident dumps. See *Oracle Database Administrator's Guide* (../../server.111/b28310/diag001.htm#ADMIN11259) for more information on incident flood control.

Problem Key

Every problem has a problem key, which is a text string that includes an error code (such as `ORA 600`) and in some cases, one or more error parameters. Two incidents are considered to have the same root cause if their problem keys match.

Incident Package

An incident package (package) is a collection of data about incidents for one or more problems. Before sending incident data to Oracle Support it must be collected into a package using the Incident Packaging Service (IPS). After a package is created, you can add external files to the package, remove selected files from the package, or *scrub* (edit) selected files in the package to remove sensitive data.

A package is a logical construct only, until you create a physical file from the package contents. That is, an incident package starts out as a collection of metadata in the Automatic Diagnostic Repository (ADR). As you add and remove package contents, only the metadata is modified. When you are ready to upload the data to Oracle Support, you create a physical package using ADRCI, which saves the data into a zip file.

Finalizing

Before ADRCI can generate a physical package from a logical package, the package must be finalized. This means that other components are called to add any correlated diagnostic data files to the incidents already in this package. Finalizing also adds recent trace files, alert log entries, Health Monitor reports, SQL test cases, and configuration information. This step is run automatically when a physical package is generated, and can also be run manually using the ADRCI utility. After manually finalizing a package, you can review the files that were added and then remove or edit any that contain sensitive information.

> **See Also:**
>
> *Oracle Database Administrator's Guide* (../../server.111/b28310/diag010.htm#ADMIN12304) for more information about correlated diagnostic data

ADR Home

An ADR home is the root directory for all diagnostic data—traces, dumps, alert log, and so on—for a particular instance of a particular Oracle product or component. For example, in a Real Application Clusters environment with ASM, each database instance and each ASM instance has an ADR home. All ADR homes share the same hierarchical directory structure. Some of the standard subdirectories in each ADR home include alert (for the alert log), trace (for trace files), and incident (for incident information). All ADR homes are located within the ADR base directory. (See "ADR Base" (#BGBBJDGF) .)

Some ADRCI commands can work with multiple ADR homes simultaneously. The current ADRCI *homepath* determines the ADR homes that are searched for diagnostic data when an ADRCI command is issued. See "Homepath" (#BGBEHACJ) for more information.

ADR Base

To permit correlation of diagnostic data across multiple ADR homes, ADR homes are grouped together under the same root directory called the ADR base. For example, in an Oracle Real Application Clusters (RAC) environment, the ADR base could be on a shared disk, and the ADR home for each Oracle RAC instance could be located under this ADR base.

The location of the ADR base for a database instance is set by the `DIAGNOSTIC_DEST` initialization parameter. If this parameter is omitted or is null, the database sets it to a default value. See *Oracle Database Administrator's Guide* (../../server.111/b28310/diag001.htm#ADMIN11008) for details.

When multiple database instances share an Oracle home, whether they are multiple single instances or the instances of an Oracle Real Application Clusters database, and when one or more of these instances set ADR base in different locations, the last instance to start up determines the default ADR base for ADRCI.

Homepath

All ADRCI commands operate on diagnostic data in the "current" ADR homes. More than one ADR home can be current at any one time. Some ADRCI commands (such as `SHOW INCIDENT`) search for and display diagnostic data from all current ADR homes, while other commands require that only one ADR home be current, and display an error message if more than one is current.

The ADRCI homepath determines the ADR homes that are current. It does so by pointing to a directory within the ADR base hierarchy. If it points to a single ADR home directory, that ADR home is the only current ADR home. If the homepath points to a directory that is above the ADR home directory level in the hierarchy, all ADR homes that are below the directory that is pointed to become current.

The homepath is null by default when ADRCI starts. This means that all ADR homes under ADR base are current.

The `SHOW HOME` and `SHOW HOMEPATH` commands display a list of the ADR homes that are current, and the `SET HOMEPATH` command sets the homepath.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* (../../server.111/b28310/diag001.htm#ADMIN11008) for more information on the structure and location of the ADR and its directories.
> - "Setting the ADRCI Homepath Before Using ADRCI Commands" (#BGBCAABI)
> - "SET HOMEPATH" (#BGBCFHCG)
> - "SHOW HOMES" (#BGBDAFDB)

Starting ADRCI and Getting Help

You can use ADRCI in interactive mode or batch mode. Details are provided in the following sections:

- Using ADRCI in Interactive Mode (#BGBBADAB)
- Getting Help (#BGBJJFFC)
- Using ADRCI in Batch Mode (#BGBGCJDC)

Using ADRCI in Interactive Mode

Interactive mode prompts you to enter individual commands one at a time. Each entered command is followed by its output.

To use ADRCI in interactive mode:

1. Ensure that the `ORACLE_HOME` and `PATH` environment variables are set properly.

   On the Windows platform, these environment variables are set in the Windows registry automatically upon installation. On other platforms, you must set and check environment variables with operating system commands.

   The `PATH` environment variable must include *ORACLE_HOME*/bin.

2. Enter the following command at the operating system command prompt:

   ```
   ADRCI
   ```

   The utility starts and displays the following prompt:

   ```
   adrci>
   ```

3. Enter ADRCI commands, following each with the Enter key.

4. Enter one of the following commands to exit ADRCI:

   ```
   EXIT QUIT
   ```

Getting Help

With the ADRCI help system, you can:

- View a list of ADR commands.
- View help for an individual command.
- View a list of ADRCI command line options.

To view a list of ADRCI commands:

1. Start ADRCI in interactive mode.

   See "Using ADRCI in Interactive Mode" (#BGBBADAB) for instructions.

2. At the ADRCI prompt, enter the following command:

   ```
   HELP
   ```

To get help for a specific ADRCI command:

1. Start ADRCI in interactive mode.

See "Using ADRCI in Interactive Mode" (#BGBBADAB) for instructions.

2. At the ADRCI prompt, enter the following command:

```
HELP command
```

For example, to get help on the `SHOW TRACEFILE` command, enter the following:

```
HELP SHOW TRACEFILE
```

To view a list of command line options:

- Enter the following command at the operating system command prompt:

```
ADRCI -HELP
```

The utility displays output similar to the following:

```
Syntax: adrci [-help] [script=script_filename] [exec="command [;command;...]"]
Options Description (Default) ------------------------------------------------------
----------- script script file name (None) help help on the command options (None)
exec exec a set of commands (None) ----------------------------------------------
---------------
```

Using ADRCI in Batch Mode

Batch mode enables you to run a series of ADRCI commands at once, without being prompted for input. To use batch mode, you add a command line parameter to the `ADRCI` command when you start ADRCI. Batch mode enables you to include ADRCI commands in shell scripts or Windows batch files. Like interactive mode, the `ORACLE_HOME` and `PATH` environment variables must be set before starting ADRCI.

The following command line parameters are available for batch operation:

Table 15-1 ADRCI Command Line Parameters for Batch Operation

Parameter	Description
`EXEC`	Enables you to submit one or more ADRCI commands on the operating system command line that starts ADRCI. Commands are separated by semicolons (;).
`SCRIPT`	Enables you to run a script containing ADRCI commands.

To submit ADRCI commands on the command line:

- Enter the following command at the operating system command prompt:

```
ADRCI EXEC="COMMAND[; COMMAND]..."
```

For example, to run the `SHOW HOMES` command in batch mode, enter the following command at the operating system command prompt:

```
ADRCI EXEC="SHOW HOMES"
```

To run the `SHOW HOMES` command followed by the `SHOW INCIDENT` command, enter the following:

```
ADRCI EXEC="SHOW HOMES; SHOW INCIDENT"
```

To run ADRCI scripts:

- Enter the following command at the operating system command prompt:

```
ADRCI SCRIPT=SCRIPT_FILE_NAME
```

For example, to run a script file named `adrci_script.txt`, enter the following command at the operating system command prompt:

```
ADRCI SCRIPT=adrci_script.txt
```

A script file contains a series of commands separated by semicolons (;) or line breaks, such as:

```
SET HOMEPATH diag/rdbms/orcl/orcl; SHOW ALERT -term
```

Setting the ADRCI Homepath Before Using ADRCI Commands

When diagnosing a problem, you may want to work with diagnostic data from multiple database instances or components, or you may want to focus on diagnostic data from one instance or component. To work with diagnostic data from multiple instances or components, you must ensure that the ADR homes for all of these instances or components are "current." To work with diagnostic data from only one instance or component, you must ensure that only the ADR home for that instance or component is current. You control the ADR homes that are current by setting the ADRCI homepath.

If multiple homes are current, this means that the homepath points to a directory in the ADR directory structure that contains multiple ADR home directories underneath it. To focus on a single ADR home, you must set the homepath to point lower in the directory hierarchy, to a single ADR home directory.

For example, if the Oracle Real Application Clusters (RAC) database with database name `orclbi` has two instances, where the instances have SIDs `orclbi1` and `orclbi2`, the following two ADR homes exist:

```
/diag/rdbms/orclbi/orclbi1/ /diag/rdbms/orclbi/orclbi2/
```

Note that in all ADRCI commands and output, ADR home directory paths (ADR homes) are always expressed relative to ADR base. So if ADR base is currently /u01/app/oracle, the absolute paths of these two ADR homes are the following:

```
/u01/app/oracle/diag/rdbms/orclbi/orclbi1/ /u01/app/oracle/diag/rdbms/orclbi/orclbi2/
```

You use the `SET HOMEPATH` command to set one or more ADR homes to be current. If ADR base is /u01/app/oracle and you want to set the homepath to /u01/app/oracle/diag/rdbms/orclbi/orclbi2/, you use this command:

```
adrci> set homepath diag/rdbms/orclbi/orclbi2
```

When ADRCI starts, the homepath is null by default, which means that all ADR homes under ADR base are current. In the previously cited example, therefore, the ADR homes for both Oracle RAC instances would be current:

```
adrci> show homes ADR Homes: diag/rdbms/orclbi/orclbi1 diag/rdbms/orclbi/orclbi2
```

In this case, any ADRCI command that you run, assuming that the command supports more than one current ADR home, works with diagnostic data from both ADR homes. If you were to set the homepath to /diag/rdbms/orclbi/orclbi2, only the ADR home for the instance with SID `orclbi2` would be current:

```
adrci> set homepath diag/rdbms/orclbi/orclbi2 adrci> show homes ADR Homes:
diag/rdbms/orclbi/orclbi2
```

In this case, any ADRCI command that you run would work with diagnostic data from this single ADR home only.

See Also:

- *Oracle Database Administrator's Guide* (../../server.111/b28310/diag001.htm#ADMIN11008) for more information on the structure of ADR homes
- "ADR Base" (#BGBBJDGF)
- "ADR Home" (#BGBBEEHG)
- "Homepath" (#BGBEHACJ)
- "SET HOMEPATH" (#BGBCFHCG)
- "SHOW HOMES" (#BGBDAFDB)

Viewing the Alert Log

Beginning with Release 11*g* of Oracle Database, the alert log is written as both an XML-formatted file and as a text file. You can view either format of the file with any text editor, or you can run an ADRCI command to view the XML-formatted alert log with the XML tags stripped. By default, ADRCI displays the alert log in your default editor. You can use the `SET EDITOR` command to change your default editor.

To view the alert log with ADRCI:

1. Start ADRCI in interactive mode.

   See "Starting ADRCI and Getting Help" (#BGBBBGEF) for instructions.

2. (Optional) Use the `SET HOMEPATH` command to select (make current) a single ADR home.

   You can use the `SHOW HOMES` command first to see a list of current ADR homes. See "Homepath" (#BGBEHACJ) and "Setting the ADRCI Homepath Before Using ADRCI Commands" (#BGBCAABI) for more information.

3. At the ADRCI prompt, enter the following command:

   ```
   SHOW ALERT
   ```

   If more than one ADR home is current, you are prompted to select a single ADR home from a list. The alert log is displayed, with XML tags stripped, in your default editor.

4. Exit the editor to return to the ADRCI command prompt.

The following are variations on the `SHOW ALERT` command:

```
SHOW ALERT -TAIL
```

This displays the last portion of the alert log (the last 10 entries) in your terminal session.

```
SHOW ALERT -TAIL 50
```

This displays the last 50 entries in the alert log in your terminal session.

```
SHOW ALERT -TAIL -F
```

This displays the last 10 entries in the alert log, and then waits for more messages to arrive in the alert log. As each message arrives, it is appended to the display. This command enables you to perform "live monitoring" of the alert log. Press `CTRL-C` to stop waiting and return to the ADRCI prompt.

```
SPOOL /home/steve/MYALERT.LOG SHOW ALERT -TERM SPOOL OFF
```

This outputs the alert log, without XML tags, to the file `/home/steve/MYALERT.LOG`.

```
SHOW ALERT -P "MESSAGE_TEXT LIKE '%ORA-600%'"
```

This displays only alert log messages that contain the string 'ORA-600'. The output looks something like this:

```
ADR Home = /u01/app/oracle/product/11.1.0/db_1/log/diag/rdbms/orclbi/orclbi:
********************************************************************************
06 09.17.44.849000000 PM -07:00 AlertMsg1: ORA-600 dbgris01, addr=0xa9876541
```

See Also:

- "SHOW ALERT" (#BGBDCIHD)
- "SET EDITOR" (#BGBCICEH)
- *Oracle Database Administrator's Guide* (../../server.111/b28310/diag005.htm#ADMIN11267) for instructions for viewing the alert log with Oracle Enterprise Manager or with a text editor

Finding Trace Files

ADRCI enables you to view the names of trace files that are currently in the automatic diagnostic repository (ADR). You can view the names of all trace files in the ADR, or you can apply filters to view a subset of names. For example, ADRCI has commands that enable you to:

- Obtain a list of trace files whose file name matches a search string.
- Obtain a list of trace files in a particular directory.
- Obtain a list of trace files that pertain to a particular incident.

You can combine filtering functions by using the proper command line parameters.

The `SHOW TRACEFILE` command displays a list of the trace files that are present in the trace directory and in all incident directories under the current ADR home. When multiple ADR homes are current, the traces file lists from all ADR homes are output one after another.

The following statement lists the names of all trace files in the current ADR homes, without any filtering:

```
SHOW TRACEFILE
```

The following statement lists the name of every trace file that has the string 'mmon' in its file name. The percent sign (%) is used as a wildcard character, and the search string is case sensitive.

```
SHOW TRACEFILE %mmon%
```

This statement lists the name of every trace file that is located in the `/home/steve/temp` directory and that has the string 'mmon' in its file name:

```
SHOW TRACEFILE %mmon% -PATH /home/steve/temp
```

This statement lists the names of trace files in reverse order of last modified time. That is, the most recently modified trace files are listed first.

```
SHOW TRACEFILE -RT
```

This statement lists the names of all trace files related to incident number 1681:

```
SHOW TRACEFILE -I 1681
```

See Also:

- "SHOW TRACEFILE" (#BGBHAIFD)
- *Oracle Database Administrator's Guide* (../../server.111/b28310/diag001.htm#ADMIN11008) for information about the directory structure of the ADR

Viewing Incidents

The ADRCI `SHOW INCIDENT` command displays information about open incidents. For each incident, the incident ID, problem key, and incident creation time are shown. If the ADRCI homepath is set so that there are multiple current ADR homes, the report includes incidents from all of them.

To view a report of all open incidents:

1. Start ADRCI in interactive mode, and ensure that the homepath points to the correct directory within the ADR base directory hierarchy.

   See "Starting ADRCI and Getting Help" (#BGBBBGEF) and "Homepath" (#BGBEHACJ) for details.

2. At the ADRCI prompt, enter the following command:

   ```
   SHOW INCIDENT
   ```

   ADRCI generates output similar to the following:

```
ADR Home = /u01/app/oracle/product/11.1.0/db_1/log/diag/rdbms/orclbi/orclbi:
*************************************************************************
INCIDENT_ID PROBLEM_KEY CREATE_TIME ----------------- ------------------------ ------
--------------------------- 3808 ORA 603 2007-06-18 21:35:49.322161 -07:00 3807 ORA
600 [4137] 2007-06-18 21:35:47.862114 -07:00 3805 ORA 600 [4136] 2007-06-18
21:35:25.012579 -07:00 3804 ORA 1578 2007-06-18 21:35:08.483156 -07:00 4 rows fetched
```

The following are variations on the `SHOW INCIDENT` command:

```
SHOW INCIDENT -MODE BRIEF SHOW INCIDENT -MODE DETAIL
```

These commands produce more detailed versions of the incident report.

```
SHOW INCIDENT -MODE DETAIL -P "INCIDENT_ID=1681"
```

This shows a detailed incident report for incident 1681 only.

> **See Also:**
>
> "ADRCI Command Reference" (#BGBCFJCI)

Packaging Incidents

You can use ADRCI commands to *package* one or more incidents for transmission to Oracle Support for analysis. Background information and instructions are presented in the following topics:

- About Packaging Incidents (#BGBDGFBH)
- Creating Incident Packages (#BGBJAJCI)

About Packaging Incidents

Packaging incidents is a three-step process:

Step 1—Create a logical incident package.

The incident package (package) is denoted as logical because it exists only as metadata in the automatic diagnostic repository (ADR). It has no content until you generate a physical package from the logical package. The logical package is assigned a package number, and you refer to it by that number in subsequent commands.

You can create the logical package as an empty package, or as a package based on an incident number, a problem number, a problem key, or a time interval. If you create the package as an empty package, you can add diagnostic information to it in step 2.

Creating a package based on an incident means including diagnostic data—dumps, health monitor reports, and so on—for that incident. Creating a package based on a problem number or problem key means including in the package diagnostic data for incidents that reference that problem number or problem key. Creating a package based on a time interval means including diagnostic data on incidents that occurred in the time interval.

Step 2—Add diagnostic information to the incident package

If you created a logical package based on an incident number, a problem number, a problem key, or a time interval, this step is optional. You can add additional incidents to the package or you can add any file within the ADR to the package. If you created an empty package, you must use ADRCI commands to add incidents or files to the package.

Step 3—Generate the physical incident package

When you submit the command to generate the physical package, ADRCI gathers all required diagnostic files and adds them to a zip file in a designated directory. You can generate a complete zip file or an incremental zip file. An incremental file contains all the diagnostic files that were added or changed since the last zip file was created for the same logical package. You can create incremental files only after you create a complete file, and you can create as many incremental files as you want. Each zip file is assigned a sequence number so that the files can be analyzed in the correct order.

Zip files are named according to the following scheme:

```
packageName_mode_sequence.zip
```

where:

- *packageName* consists of a portion of the problem key followed by a timestamp
- *mode* is either 'COM' or 'INC', for complete or incremental
- *sequence* is an integer

For example, if you generate a complete zip file for a logical package that was created on September 6, 2006 at 4:53 p.m., and then generate an incremental zip file for the same logical package, you would create files with names similar to the following:

```
ORA603_20060906165316_COM_1.zip ORA603_20060906165316_INC_2.zip
```

Creating Incident Packages

The following sections present the ADRCI commands that you use to create a logical incident package (package) and generate a physical package:

- Creating a Logical Incident Package (#BGBFCJIF)
- Adding Diagnostic Information to a Logical Incident Package (#BGBGAGDG)
- Generating a Physical Incident Package (#BGBCAJAI)

> **See Also:**
>
> "About Packaging Incidents" (#BGBDGFBH)

Creating a Logical Incident Package

You use variants of the `IPS CREATE PACKAGE` command to create a logical package (package).

To create a package based on an incident:

1. Start ADRCI in interactive mode, and ensure that the homepath points to the correct directory within the ADR base directory hierarchy.

   See "Starting ADRCI and Getting Help" (#BGBBBGEF) and "Homepath" (#BGBEHACJ) for details.

2. At the ADRCI prompt, enter the following command:

   ```
   IPS CREATE PACKAGE INCIDENT incident_number
   ```

   For example, the following command creates a package based on incident 3:

   ```
   IPS CREATE PACKAGE INCIDENT 3
   ```

   ADRCI generates output similar to the following:

   ```
   Created package 10 based on incident id 3, correlation level typical
   ```

   The package number assigned to this logical package is 10.

The following are variations on the `IPS CREATE PACKAGE` command:

```
IPS CREATE PACKAGE
```

This creates an empty package. You must use the `IPS ADD INCIDENT` or `IPS ADD FILE` commands to add diagnostic data to the package before generating it.

```
IPS CREATE PACKAGE PROBLEM problem_ID
```

This creates a package and includes diagnostic information for incidents that reference the specified problem ID. (Problem IDs are integers.) You can obtain the problem ID for an incident from the report displayed by the `SHOW INCIDENT -MODE BRIEF` command. Because there can be many incidents with the same problem ID, ADRCI adds to the package the diagnostic information for the first three incidents ("early incidents") that occurred and last three incidents ("late incidents") that occurred with this problem ID, excluding any incidents that are older than 90 days.

> **Note:**
>
> The number of early and late incidents, and the 90-day age limit are defaults that can be changed. See "IPS SET CONFIGURATION" (#BGBFCDGF) .

ADRCI may also add other incidents that correlate closely in time or in other criteria with the already added incidents.

```
IPS CREATE PACKAGE PROBLEMKEY "problem_key"
```

This creates a package and includes diagnostic information for incidents that reference the specified problem key. You can obtain problem keys from the report displayed by the `SHOW INCIDENT` command. Because there can be many incidents with the same problem key, ADRCI adds to the package only the diagnostic information for the first three early incidents and last three late incidents with this problem key, excluding incidents that are older than 90 days.

> **Note:**
>
> The number of early and late incidents, and the 90-day age limit are defaults that can be changed. See "IPS SET CONFIGURATION" (#BGBFCDGF) .

ADRCI may also add other incidents that correlate closely in time or in other criteria with the already added incidents.

The problem key must be enclosed in single quotes (') or double quotes (") if it contains spaces or quotes.

```
IPS CREATE PACKAGE SECONDS sec
```

This creates a package and includes diagnostic information for all incidents that occurred from *sec* seconds ago until now. *sec* must be an integer.

```
IPS CREATE PACKAGE TIME 'start_time' TO 'end_time'
```

This creates a package and includes diagnostic information for all incidents that occurred within the specified time range. *start_time* and *end_time* must be in the format 'YYYY-MM-DD HH24:MI:SS.FF TZR'. This is a valid format string for the `NLS_TIMESTAMP_TZ_FORMAT` initialization parameter. The fraction (`FF`) portion of the time is optional, and the `HH24:MI:SS` delimiters can be colons or periods.

For example, the following command creates a package with incidents that occurred between July 24th and July 30th of 2007:

```
IPS CREATE PACKAGE TIME '2007-07-24 00:00:00 -07:00' to '2007-07-30 23.59.59 -07:00'
```

> **See Also:**
>
> "IPS CREATE PACKAGE" (#BGBIDEFH)

Adding Diagnostic Information to a Logical Incident Package

You can add the following diagnostic information to an existing logical package (package):

- All diagnostic information for a particular incident
- A named file within the ADR

To add an incident to an existing package:

1. Start ADRCI in interactive mode, and ensure that the homepath points to the correct directory within the ADR base directory hierarchy.

   See "Starting ADRCI and Getting Help" (#BGBBBGEF) and "Homepath" (#BGBEHACJ) for details.

2. At the ADRCI prompt, enter the following command:

   ```
   IPS ADD INCIDENT incident_number PACKAGE package_number
   ```

To add a file in the ADR to an existing package:

- At the ADRCI prompt, enter the following command:

  ```
  IPS ADD FILE filespec PACKAGE package_number
  ```

  filespec must be a fully qualified file name (with path). Only files that are within the ADR base directory hierarchy may be added.

> ## See Also:
> "ADRCI Command Reference" (#BGBCFJCI)

Generating a Physical Incident Package

When you generate a package, you create a physical package (a zip file) for an existing logical package.

To generate a physical incident package:

1. Start ADRCI in interactive mode, and ensure that the homepath points to the correct directory within the ADR base directory hierarchy.

   See "Starting ADRCI and Getting Help" (#BGBBBGEF) and "Homepath" (#BGBEHACJ) for details.

2. At the ADRCI prompt, enter the following command:

   ```
   IPS GENERATE PACKAGE package_number IN path
   ```

   This generates a complete physical package (zip file) in the designated path. For example, the following command creates a complete physical package in the directory /home/steve/diagnostics from logical package number 2:

   ```
   IPS GENERATE PACKAGE 2 IN /home/steve/diagnostics
   ```

You can also generate an incremental package containing only the incidents that have occurred since the last package generation.

To generate an incremental physical incident package:

- At the ADRCI prompt, enter the following command:

  ```
  IPS GENERATE PACKAGE package_number IN path INCREMENTAL
  ```

See Also:

- "ADRCI Command Reference" (#BGBCFJCI)
- "About Packaging Incidents" (#BGBDGFBH)

ADRCI Command Reference

There are four command types in ADRCI:

- Commands that work with one or more current ADR homes
- Commands that work with only one current ADR home, and that issue an error message if there is more than one current ADR home
- Commands that prompt you to select an ADR home when there are multiple current ADR homes
- Commands that do not need a current ADR home

All ADRCI commands support the case where there is a single current ADR home.

Table 15-2 (#BGBFBECD) lists the set of ADRCI commands.

Table 15-2 List of ADRCI commands

Command	Description
CREATE REPORT (#BGBDCAEG)	Creates a report for the specified report type and ID.
ECHO (#BGBJJJDC)	Echoes the input string.
EXIT (#BGBDJDBA)	Exits the current ADRCI session.
HOST (#BGBJDEIH)	Executes operating system commands from ADRCI.
IPS (#BGBBGFFE)	Invokes the IPS utility. See Table 15-3 (#BGBJIGDJ) for the IPS commands available within ADRCI.
QUIT (#BGBGJCFH)	Exits the current ADRCI session.
RUN (#BGBIIFEH)	Runs an ADRCI script.
SET BASE (#BGBHGEDF)	Sets the ADR base for the current ADRCI session.
SET BROWSER (#BGBBGIHF)	Reserved for future use.

Command	Description
SET CONTROL (#BGBEGAEB)	Set purging policies for ADR contents.
SET ECHO (#BGBIBIII)	Toggles command output.
SET EDITOR (#BGBCICEH)	Sets the default editor for displaying trace and alert log contents.
SET HOMEPATH (#BGBCFHCG)	Makes current one or more ADR homes.
SET TERMOUT (#BGBFAGEG)	Toggles terminal output.
SHOW ALERT (#BGBDCIHD)	Shows alert log messages.
SHOW BASE (#BGBIGAIJ)	Shows the current ADR base.
SHOW CONTROL (#BGBGFEJE)	Shows ADR information, including the current purging policy.
SHOW HM_RUN (#BGBBBDDC)	Shows Health Monitor run information.
SHOW HOMEPATH (#BGBJCFEB)	Shows the current homepath.
SHOW HOMES (#BGBDAFDB)	Lists the current ADR homes.
SHOW INCDIR (#BGBEIFDH)	Lists the trace files created for the specified incidents.
SHOW INCIDENT (#BGBFIAAE)	Outputs a list of incidents.
SHOW PROBLEM (#BGBGBDGG)	Outputs a list of problems.
SHOW REPORT (#BGBFADBA)	Shows a report for the specified report type and ID.
SHOW TRACEFILE (#BGBHAIFD)	Lists qualified trace filenames.
SPOOL (#BGBFEEFJ)	Directs output to a file.

> **Note:**
>
> Unless otherwise specified, the commands listed below work with multiple current ADR homes.

CREATE REPORT

Purpose

Creates a report for the specified report type and run ID and stores the report in the ADR. Currently, only the `hm_run` (Health Monitor) report type is supported.

> **Note:**
>
> Results of Health Monitor runs are stored in the ADR in an internal format. To view these results, you must create a Health Monitor report from them and then view the report. You need create the report only once. You can then view it multiple times.

Syntax and Description

`create report` *`report_type run_name`*

`report_type` must be `hm_run`. *`run_name`* is a Health Monitor run name. Obtain run names with the SHOW HM_RUN (#BGBBBDDC) command.

If the report already exists it is overwritten. Use the SHOW REPORT (#BGBFADBA) command to view the report.

Example

This example creates a report for the Health Monitor run with run name `hm_run_1421`:

```
create report hm_run hm_run_1421
```

> **Note:**
>
> `CREATE REPORT` does not work when multiple ADR homes are set. For information on setting a single ADR home, see "Setting the ADRCI Homepath Before Using ADRCI Commands" (#BGBCAABI)

ECHO

Purpose

Prints the input string. You can use this command to print custom text from ADRCI scripts.

Syntax and Description

`echo` *`quoted_string`*

The string must be enclosed in single or double quotes.

This command does not require an ADR home to be set before you can use it.

Example

These examples print the string "Hello, world!":

```
echo "Hello, world!"
```

```
echo 'Hello, world!'
```

EXIT

Purpose

Exits the ADRCI utility.

Syntax and Description

```
exit
```

`EXIT` is a synonym for the `QUIT` command.

This command does not require an ADR home to be set before you can use it.

HOST

Purpose

Execute operating system commands without leaving ADRCI.

Syntax and Description

```
host ["host_command_string"]
```

Use `host` by itself to enter an operating system shell, which allows you to enter multiple operating system commands. Enter `EXIT` to leave the shell and return to ADRCI.

You can also specify the command on the same line (*`host_command_string`*) enclosed in double quotes.

This command does not require an ADR home to be set before you can use it.

Examples

```
host
```

```
host "ls -l *.pl"
```

IPS

Purpose

Invokes the Incident Packaging Service (IPS) command set. The IPS command provides options for creating logical incident packages (packages), adding diagnostic data to packages, and generating physical packages for transmission to Oracle Support.

> **See Also:**
>
> "Packaging Incidents" (#BGBJIDHD) for more information on packaging.

The IPS command set contains the following commands:

Table 15-3 IPS Command Set

Command	Description
IPS ADD (#BGBJIDBD)	Adds an incident, problem, or problem key to a package.
IPS ADD FILE (#BGBGIGDI)	Adds a file to a package.
IPS ADD NEW INCIDENTS (#BGBHCGIE)	Finds and adds new incidents for the problems in the specified package.
IPS COPY IN FILE (#BGBGHFDH)	Copies files into the ADR from the external file system.
IPS COPY OUT FILE (#BGBJIHGD)	Copies files out of the ADR to the external file system.
IPS CREATE PACKAGE (#BGBIDEFH)	Creates a new (logical) package.
IPS DELETE PACKAGE (#BGBEBEHE)	Deletes a package and its contents from the ADR.
IPS FINALIZE (#BGBICJED)	Finalizes a package before uploading.
IPS GENERATE PACKAGE (#BGBIDIDF)	Generates a ZIP file of the specified package contents in the target directory.
IPS GET MANIFEST (#BGBDEBJI)	Retrieves and displays the manifest from a package ZIP file.
IPS GET METADATA (#BGBHCGCH)	Extracts metadata from a package ZIP file and displays it.
IPS PACK (#BGBBGEGF)	Creates a physical package (ZIP file) directly from incidents, problems, or problem keys.

Command	Description
IPS REMOVE (#BGBFEBFD)	Removes incidents from an existing package.
IPS REMOVE FILE (#BGBCBAEG)	Remove a file from an existing package.
IPS SET CONFIGURATION (#BGBFCDGF)	Changes the value of an IPS configuration parameter.
IPS SHOW CONFIGURATION (#BGBFIEAH)	Displays the values of IPS configuration parameters.
IPS SHOW FILES (#BGBHAGIH)	Lists the files in a package.
IPS SHOW INCIDENTS (#BGBFICDF)	Lists the incidents in a package.
IPS UNPACK FILE (#BGBJFCBD)	Unpackages a package ZIP file into a specified path.

> **Note:**
>
> IPS commands do not work when multiple ADR homes are set. For information on setting a single ADR home, see "Setting the ADRCI Homepath Before Using ADRCI Commands" (#BGBCAABI)

Using the <ADR_HOME> and <ADR_BASE> Variables in IPS Commands

The IPS command set provides shortcuts for referencing the current ADR home and ADR base directories. To access the current ADR home directory, use the `<ADR_HOME>` variable as follows:

```
ips add file <ADR_HOME>/trace/orcl_ora_13579.trc package 12
```

Use the `<ADR_BASE>` variable to access the ADR base directory as follows:

```
ips add file <ADR_BASE>/diag/rdbms/orcl/orcl/trace/orcl_ora_13579.trc package 12
```

IPS ADD

Purpose

Adds incidents to a package.

Syntax and Description

```
ips add {incident inc_id | problem prob_id | problemkey pr_key |
         seconds secs | time start_time to end_time}
         package pkg_id
```

Table 15-4 (#BGBIHAJE) describes the arguments of IPS ADD.

Table 15-4 Arguments of IPS ADD command

Argument	Description
`incident` *`inc_id`*	Adds an incident with ID *`inc_id`* to the package
`problem` *`prob_id`*	Adds incidents with problem ID *`prob_id`* to the package. Adds only the first three early incidents and last three late incidents for the problem, excluding any older than 90 days. (Note: These limits are defaults and can be changed. See "IPS SET CONFIGURATION" (#BGBFCDGF) .)
`problemkey` *`pr_key`*	Adds incidents with problem key *`pr_key`* to the package. Adds only the first three early incidents and last three late incidents for the problem key, excluding any older than 90 days. (Note: These limits are defaults and can be changed.)
`seconds` *`secs`*	Adds all incidents that have occurred within *`secs`* seconds of the present time
`time` *`start_time`* `to` *`end_time`*	Adds all incidents between *`start_time`* and *`end_time`* to the package. Time format is '`YYYY-MM-YY HH24:MI:SS.FF TZR`'. Fractional part (`FF`) is optional.
`package` *`pkg_id`*	Specifies the package to which to add incidents.

Example

This example adds incident 22 to package 12:

```
ips add incident 22 package 12
```

This example adds the first three early incidents and the last three late incidents with problem ID 6 to package 2, exuding any incidents older than 90 days:

```
ips add problem 6 package 2
```

This example adds all incidents taking place during the last minute to package 5:

```
ips add seconds 60 package 5
```

This example adds all incidents taking place between 10:00 a.m. and 11:00 p.m. on May 1, 2007:

```
ips add time '2007-05-01 10:00:00.00 -07:00' to '2007-05-01 23:00:00.00 -07:00'
```

IPS ADD FILE

Purpose

Adds a file to an existing package.

Syntax and Description

`ips add file` *`file_name`* `package` *`pkg_id`*

`file_name` is the full path name of the file. You can use the `<ADR_HOME>` and `<ADR_BASE>` variables if desired. The file must be under the same ADR base as the package.

`pkg_id` is the package ID.

Example

This example adds a trace file to package 12:

```
ips add file <ADR_HOME>/trace/orcl_ora_13579.trc package 12
```

> **See Also:**
>
> See "Using the <ADR_HOME> and <ADR_BASE> Variables in IPS Commands" (#BGBBHBBC) for information on the `<ADR_HOME>` directory syntax

IPS ADD NEW INCIDENTS

Purpose

Find and add new incidents for all of the problems in the specified package.

Syntax and Description

`ips add new incidents package` *`package_id`*

`package_id` is the package to update. Only new incidents of the problems in the package are added.

Example

This example adds up to three of the new late incidents for the problems in package 12:

```
ips add new incidents package 12
```

> **Note:**
>
> The number of late incidents added is a default that can be changed. See "IPS SET CONFIGURATION" (#BGBFCDGF)

IPS COPY IN FILE

Purpose

Copies a file into the ADR from the external file system.

To edit a file in a package, you must copy the file out to a designated directory, edit the file, and copy it back into the package. You may want to do this to delete sensitive data in the file before sending the package to Oracle Support.

Syntax and Description

`ips copy in file` *`filename`* `[to` *`new_name`*`][overwrite] package` *`pkgid`* `[incident` *`incid`*`]`

Copies an external file, *`filename`* (specified with full pathname) into the ADR, associating it with an existing package, *`pkgid`*, and optionally an incident, *`incid`*. Use the `to` *`new_name`* option to give the copied file a new filename within the ADR. Use the *`overwrite`* option to overwrite a file that exists already.

Example

This example copies a trace file from the file system into the ADR, associating it with package 2 and incident 4:

```
ips copy in file /home/nick/trace/orcl_ora_13579.trc to
<ADR_HOME>/trace/orcl_ora_13579.trc package 2 incident 4
```

> **See Also:**
>
> - "Using the <ADR_HOME> and <ADR_BASE> Variables in IPS Commands" (#BGBBHBBC) for information on the `<ADR_HOME>` variable
> - "IPS SHOW FILES" (#BGBHAGIH) for information on listing files in a package

IPS COPY OUT FILE

Purpose

Copies a file from the ADR to the external file system.

To edit a file in a package, you must copy the file out to a designated directory, edit the file, and copy it back into the package. You may want to do this to delete sensitive data in the file before sending the package to Oracle Support.

Syntax and Description

`ips copy out file` *`source`* `to` *`target`* `[overwrite]`

Copies a file, *`source`*, to a location outside the ADR, *`target`* (specified with full pathname). Use the `overwrite` option to overwrite the file that exists already.

Example

This example copies a file in the ADR home called `/trace/ora_26201` to a local folder.

```
ips copy out file <ADR_HOME>/trace/orcl_ora_13579.trc to
/home/nick/trace/orcl_ora_13579.trc
```

See Also:

- "Using the <ADR_HOME> and <ADR_BASE> Variables in IPS Commands" (#BGBBHBBC) for information on the `<ADR_HOME>` directory syntax
- "IPS SHOW FILES" (#BGBHAGIH) for information on listing files in a package

IPS CREATE PACKAGE

Purpose

Creates a new package. ADRCI automatically assigns the package number for the new package.

Syntax and Description

`ips create package {incident` *`inc_id`* `| problem` *`prob_id`* `| problemkey` *`prob_key`* `| seconds` *`secs`* `| time` *`start_time`* `to` *`end_time`*`} [correlate basic |typical | all]`

Optionally, you can add incidents to the new package using the provided options.

Table 15-5 (#BGBEDGGG) describes the arguments for `IPS CREATE PACKAGE`.

Table 15-5 Arguments of IPS CREATE command

Argument	Description
`incident` *`inc_id`*	Adds an incident with ID *`inc_id`* to the package
`problem` *`prob_id`*	Adds all incidents with problem ID *`prob_id`* to the package. Adds only the first three early incidents and last three late incidents for the problem, excluding any older than 90 days. (Note: These limits are defaults and can be changed. See "IPS SET CONFIGURATION" (#BGBFCDGF) .)
`problemkey` *`pr_key`*	Adds all incidents with problem key *`pr_key`* to the package. Adds only the first three early incidents and last three late incidents for the problem key, excluding any older than 90 days. (Note: These limits are defaults and can be changed.)
`seconds` *`secs`*	Adds all incidents that have occurred within *`secs`* seconds of the present time

Argument	Description
`time` *`start_time`* `to` *`end_time`*	Adds all incidents taking place between *`start_time`* and *`end_time`* to the package. Time format is '`YYYY-MM-YY HH24:MI:SS.FF TZR`'. Fractional part (`FF`) is optional.
`correlate [basic |typical | all]`	Selects a method of including correlated incidents in the package. There are three options for this argument: • `correlate basic` includes incident dumps and incident process trace files. • `correlate typical` includes incident dumps and any trace files that were modified within five minutes of each incident. You can alter the time interval by modifying the `INCIDENT_TIME_WINDOW` configuration parameter. • `correlate all` includes the incident dumps, and all trace files that were modified between the time of the first selected incident and the last selected incident. The default value is `correlate typical`.
`package` *`pkg_id`*	Adds incidents to the package with ID *`pkg_id`*.

Examples

This example creates a package with no incidents:

```
ips create package
```

Output:

```
Created package 5 without any contents, correlation level typical
```

This example creates a package containing all incidents between 10 AM and 11 PM on the given day:

```
ips create package time '2007-05-01 10:00:00.00 -07:00' to '2007-05-01 23:00:00.00
-07:00'
```

Output:

```
Created package 6 based on time range 2007-05-01 10:00:00.00 -07:00 to 2007-05-01
23:00:00.00 -07:00, correlation level typical
```

This example creates a package and adds the first three early incidents and the last three late incidents with problem ID 3, excluding incidents that are older than 90 days:

```
ips create package problem 3
```

Output:

```
Created package 7 based on problem id 3, correlation level typical
```

> **Note:**
>
> The number of early and late incidents added, and the 90-day age limit are defaults that can be changed. See "IPS SET CONFIGURATION" (#BGBFCDGF) .

> **See Also:**
>
> "Creating Incident Packages" (#BGBJAJCI)

IPS DELETE PACKAGE

Purpose

Drops a package and its contents from the ADR.

Syntax and Description

`ips delete package` *`pkg_id`*

`pkg_id` is the package to delete.

Example

```
ips delete package 12
```

IPS FINALIZE

Purpose

Finalizes a package before uploading.

Syntax and Description

`ips finalize package` *`pkg_id`*

`pkg_id` is the package ID to finalize.

Example

```
ips finalize package 12
```

> **See Also:**
>
> *Oracle Database Administrator's Guide* (../../server.111/b28310/diag010.htm#ADMIN12304) for more information on finalizing packages

IPS GENERATE PACKAGE

Purpose

Creates a physical package (a ZIP file) in target directory.

Syntax and Description

`ips generate package` *`package_id`* `[in` *`path`*`] [complete | incremental]`

`package_id` is the ID of the package to generate. Optionally, you can save the file in the directory *`path`*. Otherwise, the package is generated in the current working directory.

The `complete` option means the package forces ADRCI to include all package files. This is the default behavior.

The `incremental` option includes only files that have been added or changed since the last time that this package was generated. With the `incremental` option, the command finishes more quickly.

Example

This example generates a physical package file in path `/home/steve`:

```
ips generate package 12 in /home/steve
```

This example generates a physical package from files added or changed since the last generation:

```
ips generate package 14 incremental
```

> **See Also:**
>
> "Generating a Physical Incident Package" (#BGBCAJAI)

IPS GET MANIFEST

Purpose

Extracts the manifest from a package ZIP file and displays it.

Syntax and Description

`ips get manifest from file` *`filename`*

`filename` is a package ZIP file. The manifest is an XML-formatted set of metadata for the package file, including information on ADR configuration, correlated files, incidents, and how the package was generated.

This command does not require an ADR home to be set before you can use it.

Example

```
ips get manifest from file /home/steve/ORA603_20060906165316_COM_1.zip
```

IPS GET METADATA

Purpose

Extracts ADR-related metadata from a package file and displays it.

Syntax and Description

`ips get metadata {from file` *`filename`* `| from adr}`

`filename` is a package ZIP file. The metadata in a package file (stored in the file `metadata.xml`) contains information about the ADR home, ADR base, and product.

Use the `from adr` option to get the metadata from a package ZIP file that has been unpacked into an ADR home using `IPS UNPACK`.

The `from adr` option requires an ADR home to be set.

Example

This example displays metadata from a package file:

```
ips get metadata from file /home/steve/ORA603_20060906165316_COM_1.zip
```

This next example displays metadata from a package file that was unpacked into the directory /scratch/oracle/package1:

```
set base /scratch/oracle/package1 ips get metadata from adr
```

In this previous example, upon receiving the `SET BASE` command, ADRCI automatically adds to the homepath the ADR home that was created in /scratch/oracle/package1 by the `IPS UNPACK FILE` command.

> **See Also:**
> "IPS UNPACK FILE" (#BGBJFCBD) for more information on unpacking package files

IPS PACK

Purpose

Creates a package and generates the physical package immediately.

Syntax and Description

```
ips pack [incident inc_id | problem prob_id | problemkey prob_key |
seconds secs | time start_time to end_time]
[correlate {basic |typical | all}] [in path]
```

ADRCI automatically generates the package number for the new package. IPS PACK creates an empty package if no package contents are specified.

Table 15-6 (#BGBCEJAI) describes the arguments for `IPS PACK`.

Table 15-6 Arguments of IPS PACK command

Argument	Description
`incident` *`inc_id`*	Adds incident with ID *`inc_id`* to the package
`problem` *`prob_id`*	Adds incidents with problem ID *`prob_id`* to the package. Adds only the first three early incidents and last three late incidents for the problem, excluding any older than 90 days. (Note: These limits are defaults and can be changed. See "IPS SET CONFIGURATION" (#BGBFCDGF) .)
`problemkey` *`pr_key`*	Adds incidents with problem key *`pr_key`* to the package. Adds only the first three early incidents and last three late incidents for the problem key, excluding any older than 90 days. (Note: These limits are defaults and can be changed.)
`seconds` *`secs`*	Adds all incidents that have occurred within *`secs`* seconds of the present time
`time` *`start_time`* `to` *`end_time`*	Adds all incidents taking place between *`start_time`* and *`end_time`* to the package. Time format is '`YYYY-MM-YY HH24:MI:SS.FF TZR`'. Fractional part (`FF`) is optional.

Argument	Description
`correlate [basic |typical | all]`	Selects a method of including correlated incidents in the package. There are three options for this argument: • `correlate basic` includes incident dumps and incident process trace files. • `correlate typical` includes incident dumps and any trace files that were modified within five minutes of each incident. You can alter the time interval by modifying the `INCIDENT_TIME_WINDOW` configuration parameter. • `correlate all` includes the incident dumps, and all trace files that were modified between the time of the first selected incident and the last selected incident. The default value is `correlate typical`.
`[in path]`	Saves the physical package to directory *path*.

Example

This example creates an empty package:

```
ips pack
```

This example creates a physical package containing all information for incident 861:

```
ips pack incident 861
```

This example creates a physical package for all incidents in the last minute, fully correlated:

```
ips pack seconds 60 correlate all
```

> **See Also:**
> "IPS SET CONFIGURATION" (#BGBFCDGF) for more information on setting configuration parameters.

IPS REMOVE

Purpose

Removes incidents from an existing package.

Syntax and Description

```
ips remove {incident inc_id | problem prob_id | problemkey prob_key}        package pkg_id
```

After removing incidents from a package, the incidents continue to be tracked within the package metadata to prevent ADRCI from automatically including them later (such as with ADD NEW INCIDENTS).

Table 15-7 (#BGBFEHJI) describes the arguments of IPS REMOVE.

Table 15-7 Arguments of IPS REMOVE command

Argument	Description
`incident inc_id`	Removes the incident with ID *inc_id* from the package
`problem prob_id`	Removes all incidents with problem ID *prob_id* from the package
`problemkey pr_key`	Removes all incidents with problem key *pr_key* from the package
`package pkg_id`	Removes incidents from the package with ID *pkg_id*.

Example

This example removes incident 22 from package 12:

```
ips remove incident 22 package 12
```

> **See Also:**
> "IPS GET MANIFEST" (#BGBDEBJI) for information on package metadata.

IPS REMOVE FILE

Purpose

Removes a file from an existing package.

Syntax and Description

```
ips remove file file_name package pkg_id
```

file_name is the file to remove from package *pkg_id*. The complete path of the file must be specified. (You can use the <ADR_HOME> and <ADR_BASE> variables if desired.)

After removal, the file continues to be tracked within the package metadata to prevent ADRCI from automatically including it later (such as with `ADD NEW INCIDENTS`). Removing a file, therefore, only sets the `EXCLUDE` flag for the file to 1.

Example

This example removes a trace file from package 12:

```
ips remove file <ADR_HOME>/trace/orcl_ora_13579.trc package 12 Removed file
<ADR_HOME>/trace/orcl_ora_13579.trc from package 12 ips show files package 12 . . .
********************************************************* FILE RECORD
********************************************************* ----------------------------
------------------------------ FILE INFORMATION: FILE_LOCATION <ADR_HOME>/trace
FILE_NAME orcl_ora_13579.trc LAST_SEQUENCE 0 EXCLUDE 1 . . .
```

> ### See Also:
>
> - "IPS GET MANIFEST" (#BGBDEBJI) for information on package metadata.
> - "Using the <ADR_HOME> and <ADR_BASE> Variables in IPS Commands" (#BGBBHBBC) for information on the `<ADR_BASE>` directory syntax
> - "IPS SHOW FILES" (#BGBHAGIH)

IPS SET CONFIGURATION

Purpose

Changes the value of an IPS configuration parameter.

Syntax and Description

`ips set configuration` *`parameter_id value`*

`parameter_id` is the parameter ID to change, and *`value`* is the new value. For a list of the configuration parameters and their IDs, use "IPS SHOW CONFIGURATION" (#BGBFIEAH) .

Example

```
ips set configuration 3 10
```

IPS SHOW CONFIGURATION

Purpose

Displays a list of IPS configuration parameters and their values. These parameters control various thresholds for IPS data, such as timeouts and incident inclusion intervals.

Syntax and Description

```
ips show configuration [parameter_id]
```

IPS SHOW CONFIGURATION lists the following information for each configuration parameter:

- Parameter ID
- Name
- Description
- Unit used by parameter (such as days or hours)
- Value
- Default value

Optionally, you can get information about a specific parameter by supplying a *parameter_id*.

Example

This command describes all IPS configuration parameters:

```
ips show configuration
```

Output:

```
IPS CONFIGURATION PARAMETER ************************************************************
------------------------------------------------------------ PARAMETER INFORMATION:
PARAMETER_ID 1 NAME CUTOFF_TIME DESCRIPTION Maximum age for an incident to be
considered for inclusion UNIT Days VALUE 90 DEFAULT_VALUE 90 ------------------------
--------------------------------
********************************************************** IPS CONFIGURATION PARAMETER
********************************************************** ---------------------------
------------------------------- PARAMETER INFORMATION: PARAMETER_ID 2 NAME
NUM_EARLY_INCIDENTS DESCRIPTION How many incidents to get in the early part of the
range UNIT Number VALUE 3 DEFAULT_VALUE 3 ----------------------------------------
------------- ********************************************************** IPS
CONFIGURATION PARAMETER ************************************************************ ---
------------------------------------------------------- PARAMETER INFORMATION:
PARAMETER_ID 3 NAME NUM_LATE_INCIDENTS DESCRIPTION How many incidents to get in the
late part of the range UNIT Number VALUE 3 DEFAULT_VALUE 3 ---------------------------
------------------------------
********************************************************** IPS CONFIGURATION PARAMETER
********************************************************** ---------------------------
------------------------------ PARAMETER INFORMATION: PARAMETER_ID 4 NAME
INCIDENT_TIME_WINDOW DESCRIPTION Incidents this close to each other are considered
correlated UNIT Minutes VALUE 5 DEFAULT_VALUE 5 ------------------------------------
------------------- ********************************************************** IPS
CONFIGURATION PARAMETER ************************************************************ ---
------------------------------------------------------- PARAMETER INFORMATION:
PARAMETER_ID 5 NAME PACKAGE_TIME_WINDOW DESCRIPTION Time window for content inclusion
is from x hours before first included incident to x hours after last incident UNIT
Hours VALUE 24 DEFAULT_VALUE 24 ----------------------------------------------------
---- ********************************************************** IPS CONFIGURATION
PARAMETER ************************************************************ -----------------
---------------------------------------- PARAMETER INFORMATION: PARAMETER_ID 6 NAME
DEFAULT_CORRELATION_LEVEL DESCRIPTION Default correlation level for packages UNIT
Number VALUE 2 DEFAULT_VALUE 2 -----------------------------------------------------
---
```

Example

This command describes configuration parameter 3:

```
ips show configuration 3
```

Configuration Parameter Descriptions

Table 15-8 (#BGBBADJC) describes the IPS configuration parameters in detail.

Table 15-8 IPS Configuration Parameters

Parameter	ID	Description
CUTOFF_TIME	1	Maximum age, in days, for an incident to be considered for inclusion

Parameter	ID	Description
NUM_EARLY_INCIDENTS	2	Number of incidents to include in the early part of the range when creating a package based on a problem. By default, ADRCI adds the three earliest incidents and three most recent incidents to the package.
NUM_LATE_INCIDENTS	3	Number of incidents to include in the late part of the range when creating a package based on a problem. By default, ADRCI adds the three earliest incidents and three most recent incidents to the package.
INCIDENT_TIME_WINDOW	4	Number of minutes between two incidents in order for them to be considered correlated.
PACKAGE_TIME_WINDOW	5	Number of hours to use as a time window for including incidents in a package. For example, a value of 5 includes incidents five hours before the earliest incident in the package, and five hours after the most recent incident in the package.

Parameter	ID	Description
DEFAULT_CORRELATION_LEVEL	6	The default correlation level to use for correlating incidents in a package. The correlation levels are: • 1 ("basic" correlation): includes incident dumps and incident process trace files. • 2 ("typical" correlation): includes incident dumps and any trace files that were modified within the time window specified by INCIDENT_TIME_WINDOW (see above). • 4 ("all"): includes the incident dumps, and all trace files that were modified between the first selected incident and the last selected incident. Additional incidents can be included automatically if they occurred in the same time range.

> **See Also:**
> "IPS SET CONFIGURATION" (#BGBFCDGF)

IPS SHOW FILES

Purpose

Lists files included in the specified package.

Syntax and Description

`ips show files package` *`pkg_id`*

`pkg_id` is the package ID to display.

Example

This example shows all files associated with package 3:

```
ips show files package 3
```

Output:

```
********************************************************** FILE RECORD
********************************************************** ----------------------------
------------------------------- FILE INFORMATION: FILE_LOCATION
<ADR_HOME>/incident/incdir_4!/nick/oracle/log/diag/rdbms/ FILE_NAME
orcl_ora_13579_i4_2.trc LAST_SEQUENCE 0 EXCLUDE 0 ------------------------------------
---------------------- ********************************************************** FILE
RECORD ********************************************************** --------------------
------------------------------------- FILE INFORMATION: FILE_LOCATION
<ADR_HOME>/incident/incdir_4!/nick/oracle/log/diag/rdbms/ FILE_NAME
orcl_ora_13579_i4.trc LAST_SEQUENCE 0 EXCLUDE 0 ---------------------------------------
-------------------- ********************************************************** FILE
RECORD ********************************************************** --------------------
------------------------------------- FILE INFORMATION: FILE_LOCATION
<ADR_HOME>/incident/incdir_4!/nick/oracle/log/diag/rdbms/ FILE_NAME
orcl_ora_13579_i4_sql_2.trc LAST_SEQUENCE 0 EXCLUDE 0 ------------------------------
--------------------------
```

IPS SHOW INCIDENTS

Purpose

Lists incidents included in the specified package.

Syntax and Description

`ips show incidents package` *`pkg_id`*

`pkg_id` is the package ID to display.

Example

This example lists the incidents in package 3:

```
ips show incidents package 3
```

Output:

```
********************************************************** Main INCIDENTS
********************************************************** ----------------------------
------------------------------
********************************************************** INCIDENT RECORD
********************************************************** ----------------------------
------------------------------ INCIDENT INFORMATION: INCIDENT_ID 3827 PROBLEM_ID 3
EXCLUDE 0 ------------------------------------------------------------
********************************************************** Correlated INCIDENTS
********************************************************** ----------------------------
------------------------------
********************************************************** INCIDENT RECORD
********************************************************** ----------------------------
------------------------------ INCIDENT INFORMATION: INCIDENT_ID 3633 PROBLEM_ID 2
EXCLUDE 0 ------------------------------------------------------------
********************************************************** INCIDENT RECORD
********************************************************** ----------------------------
------------------------------ INCIDENT INFORMATION: INCIDENT_ID 3634 PROBLEM_ID 3
EXCLUDE 0 ------------------------------------------------------------
```

IPS UNPACK FILE

Purpose

Unpackages a physical package file into the specified path.

Syntax and Description

`ips unpack file` *`file_name`* `[into` *`path`*`]`

`file_name` is the full pathname of the physical package (zip file) to unpack. Optionally, you can unpack the file into directory *`path`*, which must exist and be writable. If you omit the path, the current working directory is used. The destination directory is treated as an ADR base, and the entire ADR base directory hierarchy is created, including a valid ADR home.

This command does not require an ADR home to be set before you can use it.

Example

```
ips unpack file /tmp/ORA603_20060906165316_COM_1.zip into /tmp/newadr
```

PURGE

Purpose

Purges diagnostic data in the current ADR home, according to current purging policies. Only ADR contents that are due to be purged are purged.

Diagnostic data in the ADR has a default lifecycle. For example, information about incidents and problems is subject to purging after one year, whereas the associated dump files (dumps) are subject to purging after only 30 days.

Some Oracle products, such Oracle Database, automatically purge diagnostic data at the end of its lifecycle. Other products and components require you to purge diagnostic data manually with this command. You can also use this command to purge data that is due to be automatically purged.

The SHOW CONTROL (#BGBGFEJE) command displays the default purging policies for short-lived ADR contents and long-lived ADR contents.

Syntax and Description

```
purge [[-i {id | start_id end_id}] | [-age mins [-type {ALERT|INCIDENT|TRACE|CDUMP|HM}]]]
```

Table 15-9 (#BGBIHCJI) describes the flags for `PURGE`.

Table 15-9 Flags for the PURGE command

Flag	Description
`[[-i {id1 | start_id end_id}]`	Purges either a specific incident ID (*`id`*) or a range of incident IDs (*`start_id`* and *`end_id`*)
`[-age mins]`	Purges only data older than *`mins`* minutes.
`[-type {ALERT|INCIDENT|TRACE|CDUMP|HM}]`	Specifies the type of diagnostic data to purge (alert log messages, incident data, trace files (including dumps), core files, or Health Monitor run data and reports).

Examples

This example purges all diagnostic data in the current ADR home based on the default purging policies:

```
purge
```

This example purges all diagnostic data for all incidents between 123 and 456:

```
purge -i 123 456
```

This example purges all incident data from the last hour:

```
purge -age 60 -type incident
```

> **Note:**
>
> `PURGE` does not work when multiple ADR homes are set. For information on setting a single ADR home, see "Setting the ADRCI Homepath Before Using ADRCI Commands" (#BGBCAABI)

QUIT

See "EXIT" (#BGBDJDBA)

RUN

Purpose

Runs an ADRCI script.

Syntax and Description

`run` *`script_name`*

`@` *`script_name`*

`@@` *`script_name`*

`script_name` is the file containing the ADRCI commands to execute. ADRCI looks for the script in the current directory unless a full pathname is supplied. If the file name is given without a file extension, ADRCI uses the default extension ".adi".

The `run` and `@` commands are synonyms. The `@@` command is similar to run and `@` except that when used inside a script, `@@` uses the path of the calling script to locate *`script_name`*, rather than the current directory.

This command does not require an ADR home to be set before you can use it.

Example

```
run my_script

@my_script
```

SET BASE

Purpose

Sets the ADR base to use in the current ADRCI session.

Syntax and Description

set base *`base_str`*

`base_str` is a full path to a directory. The format for *`base_str`* depends on the operating system. If there are valid ADR homes under the base directory, these homes are added to the homepath of the current ADRCI session.

This command does not require an ADR home to be set before you can use it.

Example

```
set base /net/sttttd1/scratch/steve/view_storage/steve_v1/log
```

> **See Also:**
>
> "ADR Base" (#BGBBJDGF)

SET BROWSER

Purpose

Sets the default browser for displaying reports.

> **Note:**
>
> This command is reserved for future use. At this time ADRCI does not support HTML-formatted reports in a browser.

Syntax and Description

`set browser` *`browser_program`*

`browser_program` is the browser program name (it is assumed the browser can be started from the current ADR working directory). If no browser is set, ADRCI will display reports to the terminal or spool file.

This command does not require an ADR home to be set before you can use it.

Example

```
set browser mozilla
```

> **See Also:**
>
> - "SHOW REPORT" (#BGBFADBA) for more information on showing reports
> - "SPOOL" (#BGBFEEFJ) for more information on spooling

SET CONTROL

Purpose

Sets purging policies for ADR contents.

Syntax and Description

`set control` (*`purge_policy`* = *`value`*, `...`)

`purge_policy` is either `SHORTP_POLICY` or `LONGP_POLICY`. See "SHOW CONTROL" (#BGBGFEJE) for more information.

`value` is the number of hours after which the ADR contents become eligible for purging.

This command works with a single ADR home only.

Example

```
set control (SHORTP_POLICY = 360)
```

SET ECHO

Purpose

Turns command output on or off. This command only affects output being displayed in a script or using the spool mode.

Syntax and Description

```
set echo on|off
```

This command does not require an ADR home to be set before you can use it.

Example

```
set echo off
```

> ### See Also:
> "SPOOL" (#BGBFEEFJ) for more information on spooling

SET EDITOR

Purpose

Sets the editor for displaying the alert log and the contents of trace files.

Syntax and Description

```
set editor editor_program
```

editor_program is the editor program name. If no editor is set, ADRCI uses the editor specified by the operating system environment variable `$EDITOR`. If `$EDITOR` is not set, ADRCI uses `vi` as the default editor.

This command does not require an ADR home to be set before you can use it.

Example

```
set editor xemacs
```

> ### See Also:
> "SHOW REPORT" (#BGBFADBA)

SET HOMEPATH

Purpose

Makes one or more ADR homes current. Many ADR commands work with the current ADR homes only.

Syntax and Description

```
set homepath homepath_str1 homepath_str2 ...
```

The *homepath_strn* strings are the paths of the ADR homes *relative to the current ADR base*. The "diag" in the directory name can be omitted. If the specified path contains multiple ADR homes, all of the homes are added to the homepath.

If a desired new ADR home is not within the current ADR base, use `SET BASE` to set a new ADR base and then use `SET HOMEPATH`.

This command does not require an ADR home to be set before you can use it.

Example

```
set homepath diag/rdbms/aime3/aime3 diag/rdbms/aime3/aime32
```

> **See Also:**
> "Homepath" (#BGBEHACJ)

SET TERMOUT

Purpose

Turns output to the terminal on or off.

Syntax and Description

```
set termout on|off
```

This setting is independent of spooling. That is, the output can be directed to both terminal and a file at the same time.

This command does not require an ADR home to be set before you can use it.

> **See Also:**
> "SPOOL" (#BGBFEEFJ) for more information on spooling.

Example

```
set termout on
```

SHOW ALERT

Purpose

Shows the contents of the alert log in the default editor.

Syntax and Description

```
show alert [-p predicate_string] [-tail [num] [-f]] [-term]

[-file alert_file_name]
```

Except when using the `-term` flag, this command works with only a single current ADR home. If more than one ADR home is set, ADRCI prompts you to choose the ADR home to use.

Table 15-10 Flags for the SHOW ALERT command

Flag	Description
`-p predicate_string`	Uses a SQL-like predicate string to show only the alert log entries for which the predicate is true. The predicate string must be enclosed in double quotes. Table 15-11 (#BGBBICGI) lists the fields that can be used in the predicate string.
`-tail [num] [-f]`	Displays the most recent entries in the alert log. Use the *num* option to display the last *num* entries in the alert log. If *num* is omitted, the last 10 entries are displayed. If the `-f` option is given, after displaying the requested messages, the command does not return. Instead, it remains active and continuously displays new alert log entries to the terminal as they arrive in the alert log. You can use this command to perform live monitoring of the alert log. To terminate the command, press CTRL-C.
`-term`	Directs results to the terminal. Outputs the entire alert logs from all current ADR homes, one after another. If this option is not given, the results are displayed in the default editor.
`-file alert_file_name`	Enables you to specify an alert file outside the ADR. *alert_file_name* must be specified with a full path name. Note that this option cannot be used with the *-tail* option.

Table 15-11 Alert Fields for SHOW ALERT

Field	Type
ORIGINATING_TIMESTAMP	timestamp
NORMALIZED_TIMESTAMP	timestamp
ORGANIZATION_ID	text(65)
COMPONENT_ID	text(65)
HOST_ID	text(65)
HOST_ADDRESS	text(17)
MESSAGE_TYPE	number
MESSAGE_LEVEL	number
MESSAGE_ID	text(65)
MESSAGE_GROUP	text(65)
CLIENT_ID	text(65)
MODULE_ID	text(65)
PROCESS_ID	text(33)
THREAD_ID	text(65)
USER_ID	text(65)
INSTANCE_ID	text(65)
DETAILED_LOCATION	text(161)

Field	Type
UPSTREAM_COMP_ID	text(101)
DOWNSTREAM_COMP_ID	text(101)
EXECUTION_CONTEXT_ID	text(101)
EXECUTION_CONTEXT_SEQUENCE	number
ERROR_INSTANCE_ID	number
ERROR_INSTANCE_SEQUENCE	number
MESSAGE_TEXT	text(2049)
MESSAGE_ARGUMENTS	text(129)
SUPPLEMENTAL_ATTRIBUTES	text(129)
SUPPLEMENTAL_DETAILS	text(129)
PROBLEM_KEY	text(65)

Example

This example shows all alert messages for the current ADR home in the default editor:

```
show alert
```

This example shows all alert messages for the current ADR home and directs the output to the terminal instead of the default editor:

```
show alert -term
```

This example shows all alert messages for the current ADR home with message text describing an incident:

```
show alert -p "message_text like '%incident%'"
```

This example shows the last twenty alert messages, and then keeps the alert log open, displaying new alert log entries as they arrive:

```
show alert -tail 20 -f
```

This example shows all alert messages for a single ADR home in the default editor when multiple ADR homes have been set:

```
show alert Choose the alert log from the following homes to view: 1: diag/rdbms/ 2:
diag/tnslsnr/sta00339/listener Q: to quit Please select option: 1
```

> **See Also:**
> "SET EDITOR" (#BGBCICEH)

SHOW BASE

Purpose

Shows the current ADR base.

Syntax and Description

```
show base [-product product_name]
```

Optionally, you can show the product's ADR base location for a specific product. The products currently supported are "CLIENT" and "ADRCI".

This command does not require an ADR home to be set before you can use it.

Example

This example shows the current ADR base:

```
show base
```

Output:

```
ADR base is "/scratch/nick/rdbms/log"
```

This example shows the current ADR base for Oracle Database clients:

```
show base -product client
```

SHOW CONTROL

Purpose

Displays information about the Automatic Diagnostic Repository (ADR), including the purging policy.

Syntax and Description

```
show control
```

Displays various attributes of the ADR, including the following purging policy attributes:

Attribute Name	Description
SHORTP_POLICY	Number of hours after which to purge ADR contents that have a short life. Default is 720 (30 days).
LONGP_POLICY	Number of hours after which to purge ADR contents that have a long life. Default is 8760 (365 days).

SHOW HM_RUN

Purpose

Shows all information for Health Monitor runs.

Syntax and Description

```
show hm_run [-p predicate_string]
```

`[-p predicate_string]` is a SQL-like predicate specifying the field names to select. Table 15-12 (#BGBEICHA) displays the list of field names you can use.

Table 15-12 Fields for Health Monitor Runs

Field	Type
RUN_ID	number
RUN_NAME	text(31)
CHECK_NAME	text(31)
NAME_ID	number
MODE	number
START_TIME	timestamp
RESUME_TIME	timestamp
END_TIME	timestamp

Field	Type
MODIFIED_TIME	timestamp
TIMEOUT	number
FLAGS	number
STATUS	number
SRC_INCIDENT_ID	number
NUM_INCIDENTS	number
ERR_NUMBER	number
REPORT_FILE	bfile

Example

This example displays data for all Health Monitor runs:

```
show hm_run
```

This example displays data for the Health Monitor run with ID 123:

```
show hm_run -p "run_id=123"
```

> **See Also:**
>
> *Oracle Database Administrator's Guide* (../../server.111/b28310/diag007.htm#ADMIN11270) for more information on Health Monitor

SHOW HOMEPATH

Purpose

Identical to the `SHOW HOMES` command.

Syntax and Description

```
show homepath | show homes | show home
```

This command does not require an ADR home to be set before you can use it.

Example

```
show homepath
```

Output:

```
ADR Homes: diag/diagtool/user_nick/host_3075434791_11 diag/rdbms/db1/db1
diag/rdbms/db2/db2
```

> **See Also:**
> "SET HOMEPATH" (#BGBCFHCG) for information on how to set the homepath

SHOW HOMES

Purpose

Show the ADR homes in the current ADRCI session.

Syntax and Description

```
show homes | show home | show homepath
```

This command does not require an ADR home to be set before you can use it.

Example

```
show homes
```

Output:

```
ADR Homes: diag/diagtool/user_nick/host_3075434791_11 diag/rdbms/db1/db1
diag/rdbms/db2/db2
```

SHOW INCDIR

Purpose

Shows trace files for the specified incident.

Syntax and Description

```
show incdir [id | id_low id_high]
```

You can provide a single incident ID (*id*) or a range of incidents (*id_low* to *id_high*). If no incident ID is given, trace files for all incidents are listed.

Example

This example shows all trace files for all incidents:

```
show incdir
```

Output:

```
ADR Home = /ade/sfogel_emdb/oracle/log/diag/rdbms/emdb/emdb:
*************************************************************************
diag/rdbms/emdb/emdb/incident/incdir_3801/emdb_ora_23604_i3801.trc
diag/rdbms/emdb/emdb/incident/incdir_3801/emdb_m000_23649_i3801_a.trc
diag/rdbms/emdb/emdb/incident/incdir_3802/emdb_ora_23604_i3802.trc
diag/rdbms/emdb/emdb/incident/incdir_3803/emdb_ora_23604_i3803.trc
diag/rdbms/emdb/emdb/incident/incdir_3804/emdb_ora_23604_i3804.trc
diag/rdbms/emdb/emdb/incident/incdir_3805/emdb_ora_23716_i3805.trc
diag/rdbms/emdb/emdb/incident/incdir_3805/emdb_m000_23767_i3805_a.trc
diag/rdbms/emdb/emdb/incident/incdir_3806/emdb_ora_23716_i3806.trc
diag/rdbms/emdb/emdb/incident/incdir_3633/emdb_pmon_28970_i3633.trc
diag/rdbms/emdb/emdb/incident/incdir_3633/emdb_m000_23778_i3633_a.trc
diag/rdbms/emdb/emdb/incident/incdir_3713/emdb_smon_28994_i3713.trc
diag/rdbms/emdb/emdb/incident/incdir_3713/emdb_m000_23797_i3713_a.trc
diag/rdbms/emdb/emdb/incident/incdir_3807/emdb_ora_23783_i3807.trc
diag/rdbms/emdb/emdb/incident/incdir_3807/emdb_m000_23803_i3807_a.trc
diag/rdbms/emdb/emdb/incident/incdir_3808/emdb_ora_23783_i3808.trc
```

This example shows all trace files for incident 3713

```
show incdir 3713
```

Output:

```
ADR Home = /ade/sfogel_emdb/oracle/log/diag/rdbms/emdb/emdb:
*************************************************************************
diag/rdbms/emdb/emdb/incident/incdir_3713/emdb_smon_28994_i3713.trc
diag/rdbms/emdb/emdb/incident/incdir_3713/emdb_m000_23797_i3713_a.trc
```

This example shows all tracefiles for incidents between 3801 and 3804:

```
show incdir 3801 3804
```

Output:

```
ADR Home = /ade/sfogel_emdb/oracle/log/diag/rdbms/emdb/emdb:
*************************************************************************
diag/rdbms/emdb/emdb/incident/incdir_3801/emdb_ora_23604_i3801.trc
diag/rdbms/emdb/emdb/incident/incdir_3801/emdb_m000_23649_i3801_a.trc
diag/rdbms/emdb/emdb/incident/incdir_3802/emdb_ora_23604_i3802.trc
diag/rdbms/emdb/emdb/incident/incdir_3803/emdb_ora_23604_i3803.trc
diag/rdbms/emdb/emdb/incident/incdir_3804/emdb_ora_23604_i3804.trc
```

SHOW INCIDENT

Purpose

Lists all of the incidents associated with the current ADR home. Includes both open and closed incidents.

Syntax and Description

```
show incident [-p predicate_string] [-mode {BASIC|BRIEF|DETAIL}]

[-orderby field1, field2, ...] [ASC|DSC]
```

Table 15-13 (#BGBGHGIC) describes the flags for `SHOW INCIDENT`.

Table 15-13 Flags for SHOW INCIDENT command

Flag	Description
`-p predicate_string`	Use a predicate string to show only the incidents for which the predicate is true. The predicate string must be enclosed in double quotes. Table 15-14 (#BGBGJFFJ) lists the fields that can be used in the predicate string.
`[-mode {BASIC|BRIEF|DETAIL}]`	Choose an output mode for incidents. `BASIC` is the default. • `BASIC` displays only basic incident information (the `INCIDENT_ID`, `PROBLEM_ID`, and `CREATE_TIME` fields). It does not display flood-controlled incidents. • `BRIEF` displays all information related to the incidents, as given by the fields in Table 15-14 (#BGBGJFFJ) . It includes flood-controlled incidents. • `DETAIL` displays all information for the incidents (as with `BRIEF` mode) as well as information about incident dumps. It includes flood-controlled incidents.
`[-orderby field1, field2, ...] [ASC|DSC]`	Show results sorted by field in the given order, as well as in ascending (`ASC`) and descending order (`DSC`). By default, results are shown in ascending order.

Table 15-14 Incident Fields for SHOW INCIDENT

Field	Type

Field	Type
INCIDENT_ID	number
PROBLEM_KEY	text(550)
PROBLEM_ID	number
CREATE_TIME	timestamp
CLOSE_TIME	timestamp
STATUS	number
FLAGS	number
FLOOD_CONTROLLED	number (decoded to a text status by ADRCI)
ERROR_FACILITY	text(10)
ERROR_NUMBER	number
ERROR_ARG1	text(64)
ERROR_ARG2	text(64)
ERROR_ARG3	text(64)
ERROR_ARG4	text(64)
ERROR_ARG5	text(64)
ERROR_ARG6	text(64)
ERROR_ARG7	text(64)
ERROR_ARG8	text(64)

Field	Type
SIGNALLING_COMPONENT	text(64)
SIGNALLING_SUBCOMPONENT	text(64)
SUSPECT_COMPONENT	text(64)
SUSPECT_SUBCOMPONENT	text(64)
ECID	text(64)
IMPACT	number

Examples

This example shows all incidents for this ADR home:

```
show incident
```

Output:

```
ADR Home = /ade/sfogel_emdb/oracle/log/diag/rdbms/emdb/emdb:
*************************************************************************** INCIDENT_ID
PROBLEM_KEY CREATE_TIME -------------------- ----------------------------------------
--- ---------------------------- 3808 ORA 603 2007-06-18 21:35:49.322161 -07:00 3807
ORA 600 [4137] 2007-06-18 21:35:47.862114 -07:00 3806 ORA 603 2007-06-18
21:35:26.666485 -07:00 3805 ORA 600 [4136] 2007-06-18 21:35:25.012579 -07:00 3804 ORA
1578 2007-06-18 21:35:08.483156 -07:00 3713 ORA 600 [4136] 2007-06-18 21:35:44.754442
-07:00 3633 ORA 600 [4136] 2007-06-18 21:35:35.776151 -07:00 7 rows fetched
```

This example shows the detail view for incident 3805:

```
adrci> show incident -mode DETAIL -p "incident_id=3805"
```

Output:

```
ADR Home = /ade/sfogel_emdb/oracle/log/diag/rdbms/emdb/emdb:
*******************************************************************************
********************************************************* INCIDENT INFO RECORD 1
********************************************************* INCIDENT_ID 3805 STATUS
closed CREATE_TIME 2007-06-18 21:35:25.012579 -07:00 PROBLEM_ID 2 CLOSE_TIME 2007-06-
18 22:26:54.143537 -07:00 FLOOD_CONTROLLED none ERROR_FACILITY ORA ERROR_NUMBER 600
ERROR_ARG1 4136 ERROR_ARG2 2 ERROR_ARG3 18.0.628 ERROR_ARG4 <NULL> ERROR_ARG5 <NULL>
ERROR_ARG6 <NULL> ERROR_ARG7 <NULL> ERROR_ARG8 <NULL> SIGNALLING_COMPONENT <NULL>
SIGNALLING_SUBCOMPONENT <NULL> SUSPECT_COMPONENT <NULL> SUSPECT_SUBCOMPONENT <NULL>
ECID <NULL> IMPACTS 0 PROBLEM_KEY ORA 600 [4136] FIRST_INCIDENT 3805 FIRSTINC_TIME
2007-06-18 21:35:25.012579 -07:00 LAST_INCIDENT 3713 LASTINC_TIME 2007-06-18
21:35:44.754442 -07:00 IMPACT1 0 IMPACT2 0 IMPACT3 0 IMPACT4 0 KEY_NAME Client ProcId
KEY_VALUE oracle@stadh43 (TNS V1-V3).23716_3083142848 KEY_NAME SID KEY_VALUE 127.52237
KEY_NAME ProcId KEY_VALUE 23.90 KEY_NAME PQ KEY_VALUE (0, 1182227717) OWNER_ID 1
INCIDENT_FILE /.../emdb/emdb/incident/incdir_3805/emdb_ora_23716_i3805.trc OWNER_ID 1
INCIDENT_FILE /.../emdb/emdb/trace/emdb_ora_23716.trc OWNER_ID 1 INCIDENT_FILE
/.../emdb/emdb/incident/incdir_3805/emdb_m000_23767_i3805_a.trc 1 rows fetched
```

SHOW PROBLEM

Purpose

Show problem information for the current ADR home.

Syntax and Description

```
show problem [-p predicate_string]

    [-last num | -all]

    [-orderby field1, field2, ...] [ASC|DSC]]
```

Table 15-15 (#BGBDGHBC) describes the flags for `SHOW PROBLEM`.

Table 15-15 Flags for SHOW PROBLEM command

Flag	Description
`[-p predicate_string]`	Use a SQL-like predicate string to show only the incidents for which the predicate is true. The predicate string must be enclosed in double quotes. Table 15-16 (#BGBGAACH) lists the fields that can be used in the predicate string.
`[-last num| -all]`	Shows the last *num* problems (`-last`), or list all the problems (*-all*). By default, SHOW PROBLEM lists the most recent 50 problems.

Flag	Description
[-orderby *field1*, *field2*, ...] [ASC|DSC]	Show results sorted by field in the given order (*field1*, *field2*, ...), as well as in ascending (ASC) and descending order (DSC). By default, results are shown in ascending order.

Table 15-16 Problem Fields for SHOW PROBLEM

Field	Type
PROBLEM_ID	number
PROBLEM_KEY	text(550)
FIRST_INCIDENT	number
FIRSTINC_TIME	timestamp
LAST_INCIDENT	number
LASTINC_TIME	timestamp
IMPACT1	number
IMPACT2	number
IMPACT3	number
IMPACT4	number
SERVICE_REQUEST	text(64)
BUG_NUMBER	text(64)

Example

This example lists all the problems in the current ADR home:

```
show problem -all
```

This example shows the problem with ID 4:

```
show problem -p "problem_id=4"
```

SHOW REPORT

Purpose

Show a report for the specified report type and run name. Currently, only the `hm_run` (Health Monitor) report type is supported, and only in XML formatting. To view HTML-formatted Health Monitor reports, use Enterprise Manager or the `DBMS_HM` PL/SQL package. See *Oracle Database Administrator's Guide* (../../server.111/b28310/diag007.htm#ADMIN11276) for more information.

Syntax and Description

```
SHOW REPORT report_type run_name
```

report_type must be `hm_run`. *run_name* is the Health Monitor run name from which you created the report. You must first create the report using the CREATE REPORT command.

This command does not require an ADR home to be set before you can use it.

Example

```
show report hm_run hm_run_1421
```

> ### See Also:
>
> - "CREATE REPORT" (#BGBDCAEG)
> - "SHOW HM_RUN" (#BGBBBDDC)

SHOW TRACEFILE

Purpose

List trace files.

Syntax and Description

```
show tracefile [file1 file2 ...] [-rt | -t]

[-i inc1 inc2 ...] [-path path1 path2 ...]
```

This command searches for one or more files under the trace directory and all incident directories of the current ADR homes, unless the `-i` or `-path` flags are given.

This command does not require an ADR home to be set unless using the `-i` option.

Table 15-18 (#BGBIDFAJ) describes the arguments of `SHOW TRACEFILE`.

Table 15-17 Arguments for SHOW TRACEFILE Command

Argument	Description
file1 file2	Filter results by file name. The % symbol is a wildcard character.

Table 15-18 Flags for SHOW TRACEFILE Command

Flag	Description
`-rt | -t`	Order the trace filenames by timestamp. `-t` sorts the filenames in ascending order by timestamp, and `-rt` sorts them in reverse order. Note that filenames are only ordered relative to their directory. Listing multiple directories of trace files applies a separate ordering to each directory. Timestamps are listed next to each filename when using this option.
`[-i` *inc1 inc2* ...]	Select only the trace files produced for the given incident IDs.
`-path` *path1 path2*	Query only the trace files under the given pathnames.

Example

This example shows all the trace files under the current ADR home:

```
show tracefile
```

This example shows all the MMON trace files, sorted by timestamp in reverse order:

```
show tracefile %mmon% -rt
```

This example shows all trace files for incidents 1 and 4, under the path `/home/steve/temp`:

```
show tracefile -i 1 4 -path /home/steve/temp
```

SPOOL

Purpose

Directs ADRCI output to a file.

Syntax and Description

```
SPOOL filename [[APPEND] | [OFF]]
```

filename is the filename where the output is to be directed. If a full pathname is not given, the file is created in the current ADRCI working directory. If no file extension is given, the default extension ".ado" is used. APPEND causes the output to be appended to the end of the file. Otherwise, the file is overwritten. Use OFF to turn off spooling.

This command does not require an ADR home to be set before you can use it.

Example

```
spool myfile

spool myfile.ado append

spool off

spool
```

Troubleshooting ADRCI

The following are some common ADRCI error messages, with their possible causes and remedies:

No ADR base is set

Cause: You may have started ADRCI with a null or invalid value for the ORACLE_HOME environment variable.

Action: Exit ADRCI, set the ORACLE_HOME environment variable, and restart ADRCI. See "ADR Base" (#BGBBJDGF) for more information.

DIA-48323: Specified pathname *string* must be inside current ADR home

Cause: A file outside of the ADR home is not allowed as an incident file for this command.

Action: Retry using an incident file inside the ADR home

DIA-48400: ADRCI initialization failed

Cause: The ADR Base directory does not exist

Action: Check the value of the DIAGNOSTIC_DEST initialization parameter, and ensure that it points to an ADR base directory that contains at least one ADR home. If DIAGNOSTIC_DEST is missing or null, check for a valid ADR base directory hierarchy in *ORACLE_HOME*/log.

DIA-48431: Must specify at least one ADR home path

Cause: The command requires at least one ADR home to be current

Action: Use the `SET HOMEPATH` command to make one or more ADR homes current.

DIA-48432: The ADR home path *string* is not valid

Cause: The supplied ADR home is not valid, possibly because the path does not exist.

Action: Check if the supplied ADR home path exists

DIA-48447: The input path [*path*] does not contain any ADR homes

Cause: When using `SET HOMEPATH` to set an ADR home, you must supply a path relative to the current ADR base.

Action: If the new desired ADR home is not within the current ADR base, first set ADR base with `SET BASE`, and then use `SHOW HOMES` to check the ADR homes under the new ADR base. Next, use `SET HOMEPATH` to set a new ADR home if necessary.

DIA-48448: This command does not support multiple ADR homes

Cause: There are multiple current ADR homes in the current ADRCI session.

Action: Use the `SET HOMEPATH` command to make a single ADR home current.