

# Report Syntax

```

ADAREP [REPORT]
      [ {CPEXLIST [, OFFSET = {column-number | 63} ] | CPLIST }
        [FROMDATE = yyyyymmdd] [TODATE = yyyyymmdd]
        [FROMSESSION = session] [TOSESSION = session] ]
      [ {FILE = file-list [, LAYOUT = 1 ] | NOFILE } ]
      [LIMCOUNT | NOCOUNT]
      [NOFDT]
      [NOPPT]
      [ { NOLGLIST | NOPHLIST | NOSTD } ]
      [NOUSERABEND] }
      [SAVETAPE
        [PLOGNUM = protection-log-number [ { SYN1 | SYN4 } = PLOG-block-number ] ] ]
      [OUTPUT='XML[,UTF8]']
      [ACODE= '{ GCB | dbcodepg } [ , { GCB | FCB | fncodepg } ] [ , { GCB | cpcodepg } ] ' ]

```

This chapter describes the syntax and parameters of the ADAREP utility.

- Optional Parameters
- Examples

## Optional Parameters

ADAREP can be specified alone to retrieve a database status report. You can optionally customize the report by added parameter values.

### ACODE: Identify Code Page for XML UTF-8 Output

Use the ACODE parameter only if the UTF8 option is specified on the OUTPUT parameter. This parameter identifies the code page to which the database name, file names, and checkpoint names should be converted. This parameter is specified in three parts, in a specific order, and with each part separated by a comma from the others:

```
ACODE='database-name-codepage , file-name-codepage , checkpoint-name-codepage'
```

The *database-name-codepage* can either be the literal "GCB" or a valid code page number; the *file-name-codepage* can either be the literal "GCB", the literal "FCB", or a valid code page number; the *checkpoint-name-codepage* can either be the literal "GCB" or a valid code page number:

- A setting of "GCB" for any part indicates that the code page for that part should be acquired from the code page in the GCB (the GCB ACODE setting).

- A setting of "FCB" for the *file-name-codepage* indicates that the code page for converting file names in the XML document to UTF-8 should be acquired from the code page setting in the FCB (the FCB ACODE setting).

If code pages for all three parts of the ACODE parameter are being specified, they must be specified in the order shown above. However, if you only need to specify the code page of the database name conversion, you only need to specify the first part. If you only need to specify the code page of the database name and file name conversions, you only need to specify the first two parts. If you don't need to specify the code page for an earlier part in the sequence, simply specify a comma for that part instead. For example, if you only need to specify the code page setting for the checkpoint name UTF-8 conversion, you might specify:

```
ACODE=' , , 37' .
```

Note the two commas specified prior to the checkpoint code page number, 37. These identify the placement of the blank database name and file name code page specifications.

If the ACODE parameter is specified but the OUTPUT parameter is omitted, a setting of OUTPUT='XML,UTF8' is assumed.

### **CPEXLIST: Print Checkpoint List in Extended Format**

### **CPLIST : Print Checkpoint List in Normal Format**

These parameters are used to print the checkpoint list in normal (CPLIST) or extended (CPEXLIST) format. Either CPEXLIST or CPLIST must be specified to display checkpoint information. CPEXLIST adds the following information to the normal CPLIST information, depending on the checkpoint origin:

- (Utility or Adabas Online System/Basic Services) function name;
- Checkpoint-specific data.

If the CPEXLIST report is to be displayed, OFFSET can also be specified for a more readable display.

The FROMDATE, TODATE, FROMSESSION, and TOSESSION parameters may be used to indicate the range of checkpoints to be printed.

### **FILE or NOFILE: File Information to be Displayed or Suppressed**

FILE defines the list of files for which status information is to be printed or displayed. If this parameter is omitted, status information for all files will be included.

If NOFILE is specified, the printing of all file and field description information is suppressed.

### **FROMDATE/ TODATE: Start/ End Checkpoint Dates for Report**

When CPLIST or CPEXLIST is specified, specific start and/or end dates for checkpoint information can be specified. Examples of valid *yyyymmdd* date specification are:

```

ADAREP FROMDATE=19960101,TODATE=19960228   January 1-February 28, 1996
ADAREP FROMDATE=19951111                   November 11, 1995 to checkpoint file end
ADAREP TODATE=19951223                     From checkpoint file begin to (and including)
                                           December 23, 1995

```

If FROMDATE is not specified, the report begins with the earliest checkpoint information in the system (or with the first on the FROMSESSION session, if later); if TODATE is not specified, the report continues up to the most recent checkpoint (or ends with the last on the TOSESSION session, if earlier).

### FROMSESSION/ TOSESSION: Start/ End Session for Report

Specify a start and/or end session number. Sessions before FROMSESSION and/or after TOSESSION session numbers are not included in the report information. If FROMSESSION is not specified, the report begins with the earliest checkpoint information in the system (or with the first on the FROMDATE date, if later); if TOSESSION is not specified, the report continues up to the last checkpoint (or ends with the last on the TODATE date, if earlier).

### LAYOUT: Format Output for Printing

LAYOUT=1 specifies that the "Contents of Database" table should be printed in a single 120-character column format. Normally, the "Contents of Database" and "File Space Allocations" information are presented in two separate sections in the report. However, when LAYOUT=1 is specified, they are merged together into the "Contents of Database" section. In addition, the padding factor is added as well.

The following is an example of how the "Contents of Database" section of the report would appear if LAYOUT=1 is specified. You compare this sample to the "Contents of Database" and "File Space Allocations" sections described in *Contents of the Database: General File Status* and in *File Space Allocations*.

```

*****
* Contents of Database 68 (EXAMP68 ) *          yyyy-mm-dd hh:mm:ss
*****

```

File	Name	Loaded	TOP-ISN	MAX-ISN	PADD	Blocks Allocated	(Blocks Unused)		
					A% D%	NI	UI	AC	DATA/CYL
1	EMPLOYEES	2001-12-28	1107	1695	10 10	200	15	2	150/1
1	Extents: NI	1 UI 1 AC	1 DS	1		138	2		104/0
2	MISCELLANEOUS	2001-12-28	1779	2543	10 10	200	15	3	150/1
2	Extents: NI	1 UI 1 AC	1 DS	1		174	10		97/0
3	VEHICLES	2001-12-28	773	1695	10 10	200	15	2	150/1
3	Extents: NI	1 UI 1 AC	1 DS	1		183	8		138/0

### LIMCOUNT or NOCOUNT: Counting of Number of Records Loaded

ADAREP reads the address converter to determine the value for RECORDS LOADED for a file. For very large files, this can result in a large amount of I/O activity. If LIMCOUNT is specified, ADAREP checks the value for TOPISN for the file. If TOPISN is greater than 1000, "NOT COUNTED" appears under RECORDS LOADED.

If NOCOUNT is specified, no value is printed for RECORDS LOADED for any file. If neither LIMCOUNT nor NOCOUNT are specified, ADAREP compiles the exact value for RECORDS LOADED for each file.

**NOFDT: Suppress Printing of Field Definitions**

The printing of the field definition table (FDT) information for each file is to be omitted. The FDT of the Adabas checkpoint and security files are not printed by ADAREP.

**NOLGLIST, NOPHLIST, or NOSTD: Suppress Database Layout Printing**

If NOLGLIST is specified, the logical database layout information is to be omitted. If NOPHLIST is specified, physical database layout information is omitted. Specifying NOSTD suppresses all database layout information, and is equivalent to specifying NOLGLIST and NOPHLIST.

**NOPPT: Suppress PPT Information**

Specify the NOPPT parameter if you do not want PPT information included in the report.

**NOUSERABEND: Termination without Abend**

When a parameter error or a functional error occurs while this utility function is running, the utility ordinarily prints an error message and terminates with user abend 34 (with a dump) or user abend 35 (without a dump). If NOUSERABEND is specified, the utility will *not* abend after printing the error message. Instead, the message "*utility* TERMINATED DUE TO ERROR CONDITION" is displayed and the utility terminates with condition code 20.

**Note:**

When NOUSERABEND is specified, we recommend that it be specified as the first parameter of the utility function (before all other parameters). This is necessary to ensure that its parameter error processing occurs properly.

**OFFSET: Defines Extended Report Line Alignment**

The OFFSET parameter aligns the beginning of the extended information with the end of the normal checkpoint information for printing on a single line. The default is 63. To display extended information, OFFSET must be reduced or the extended portion may not be displayable within 80 columns.

**OUTPUT: Create XML Report**

This optional parameter should be used when you want to create an XML version of the database report in addition to the normal one. The format of the XML report can be either EBCDIC or converted to UTF-8:

<b>If you specify:</b>	<b>The XML report format will be:</b>
OUTPUT= ' XML '	EBCDIC
OUTPUT= ' XML , UTF8 '	Converted to UTF-8 (including all tags). When UTF-8 format is requested, you may also use the ACODE parameter to specify the specific code page to which the database base name, file names, and checkpoint names should be converted.

When the OUTPUT parameter is specified, additional DD statements (DDXMLR1 and, optionally, DDXMLR2) must be specified in the ADAREP job. DDXMLR1 is required and identifies the data set to which the XML version of the base database report should be written. DDXMLR2 is used to store the XML version of any checkpoint information in the report; it is therefore only required if checkpoint information is requested for the report via the CPLIST or CPEXLIST parameters. The data sets specified by these DD statements must be defined with DCB parameters RECFM=VB, LRECL=512.

The resulting XML documents exactly match the information produced in the regular report, including any filtering done for the report (for example if NOFDT or NOPPT are requested).

Once the XML document is created, it can be transferred to a Windows or UNIX machine and you can use your own code or other tools to transform it.

The XML document is structured as defined by an XML schema definition file (*adabas.xsd*) provided in the online Community Discussion Forum for Adabas. A stylesheet (*adabas.xsl*) is also provided for your use at the same location. You can transfer these XML files from the Adabas Forum to any system (probably Windows or UNIX) using automated tools or user-written programs in the programming language of your choice (Natural, Java, .Net, C, Perl, or Python).

### **PLOGNUM: Protection Log Number**

PLOGNUM specifies the number of the nucleus protection log used during the ADASAV save operation. The number of the nucleus protection log is supplied on the tape, so specifying a value for the PLOGNUM parameter *overrides* the information on the tape.

### **SAVETAPE: Print Save Tape Report**

If SAVETAPE is specified, the report is printed from a save tape. The purpose of the save tape report is to determine what the save tape contains. For more information, see the section Processing Save Tape Input.

CPLIST/ CPEXLIST cannot be specified with SAVETAPE. NOCOUNT must be specified with SAVETAPE because ADAREP does not count the number of records in a file on a save tape. If CPLIST/ CPEXLIST is specified or NOCOUNT is not, ADAREP prints a warning message, changes these options internally, and continues processing.

### **SYN1/SYN4: Beginning Block Number**

The SYN1 and SYN4 parameters are mutually exclusive and specify the block number containing the SYN1 or SYN4 checkpoint at which the corresponding save operation began. These block numbers are supplied on the tape, as appropriate, so specifying a value for the SYN1 or SYN4 parameters *overrides* the information on the tape.

## **Examples**

### **Example 1:**

ADAREP REPORT
---------------

All database, file, and field information is to be printed. Checkpoint data is not to be printed.

**Example 2:**

```
ADAREP REPORT NOFDT
```

Database and file information is to be printed. Checkpoint data and field definitions for each file are not to be printed.

**Example 3:**

```
ADAREP CPEXLIST, FROMDATE=19980701, TODATE=19980715  
ADAREP NOSTD, NOFILE
```

A checkpoint list (extended format) is to be printed for all checkpoints taken between July 1, 1998 and July 15, 1998. No other information is to be printed.

**Example 4:**

```
ADAREP REPORT SAVETAPE, NOCOUNT
```

All database, file, and field information is to be printed from a save tape. NOCOUNT is required because ADAREP does not count the number of records in a file on a save tape.

**Example 5:**

```
ADAREP CPEXLIST, OUTPUT='XML'
```

All database, file, field information, and checkpoint data in extended format will be printed in the ADAREP report and a CML version of the report will be produced.

**Example 6:**

```
ADAREP CPLIST, OUTPUT='XML, UTF8'
```

All database, file, field information, and checkpoint data will be printed in the ADAREP report and an XML version of the report will be produced and converted to UTF-8.

**Example 7:**

```
ADAREP REPORT OUTPUT='XML, UTF8', ACODE=' , 37'
```

All database, file, and field information will be printed in the ADAREP report and an XML version of the report will be produced and converted to UTF-8. Code page 37 will be used for the UTF-8 conversion of file names.