ADAREORG V2.32d
Release Notes
11 August 2005

Introduction

All ADAREORG Software from CCA Software is now supplied as Email attachments. This is to facilitate fast transmission of new releases/upgrades/fixes.

ADAREORG V2.32d has been introduced to consolidate all previous fixes and include improved support for G and W type fields.

The recent changes to this fix level include the implimentation of the translation of 1 byte alpha containing blank (X'40') to zero (X'F0') when being converted to other types.
CPUIDs are now displayed to help diagnose codeword problems should they occur.

PLEASE REFER TO PAGE TWO FOR INSTALLATION INSTRUCTIONS

Features Supported*:

1. P type fields which are used as the source of information during P to A conversions are no longer corrupted.
2. U type fields which are used as the source of information during U to A conversions are no longer corrupted.
3. The ADAVER parameter is now an optional parameter with a default value of 7. This means ADAREORG will interpret all input formats as that of ADABAS v7. To support earlier versions of ADABAS the DBA must provide the ADAVER parameter.
4. Floating point fields (type G) are now treated as real numeric fields, which may be converted to other field types, and vice versa. This means they are no longer simply treated as binary fields. See the V232 manual for a more extensive explanation.
5. Wide field support, with restrictions (see V232 manual).
6. INC parameter has been added (see manual).
7. Conversion of numeric field types to A type has changed. "A type" is assumed to be legible and when it is on the receiving end of conversion from a numeric field, the content of the A type fields will have a leading + or – sign, as appropriate. When used as the input for a numeric conversion, leading + or – signs are checked for, and taken into account.
8. The actual CODEWORD used to run Adareorg is now displayed in the joblog.
9. Support for LBI interface of ADABAS V7.4 on input files (see manual for details)
10. Support for facilites of ADABAS V7.4.x
11. Multiple ADABAS version support, see ADAVER parameter
12. OR processing with the ACCEPTO parameter
13. AND processing with the REJECTA parameter
14. Correction of processing issues reported with ACCEPT/REJECT processing
15. 3/4 Byte ISN's, 4 Byte RABN's, 2 Byte FNR's and 2 Byte DBID's.

16. Alpha fields up to 16K.
17. 191 PE groups.
    F Fields need not be fixed length (restricted to 2 or 4 bytes)
    G Fields need not be fixed length (either 4 or 8 bytes)
18. Super-DE's up to 20 fields
19. Variable length fields (ie length of zero)
20. Additional field-level user-exit invocations per execution

NOTE: If DDWAN0n (for n=1,2) is PRESENT then no attempt is made to extract FDT information from DDADA0n (matching n=1,2). When DDWAN01(2) is ABSENT, and USERISN is specified for DDWAN03, then USERISN is assumed to have been specified for DDADA01(2), and ISNs are taken from this file (DDADA01 in preference to DDADA02). This ISN may later be overwritten by an internally generated ISN, if the ISN parameter is specified in the PARM file.

Installation Notes for PC-Diskette or E-MAIL Supplied Software

This release contains three files:

  ➢ **AR232in.zip** - Source Library Members

  ➢ **AR232ddoco.zip** - documentation in PDF format and these notes.

  ➢ **AR232dob.zip** - Load Library JCL/Modules

                    ****

1.   Install the Source Library

1.1   Unzip AR232din.zip

1.2   Upload the un-zipped *.txt files into a mainframe dataset called **xxxxxx.AR232d.INSTALL** where xxxxxxx is your high-level DSN qualifier.

        Dataset Attributes:
        Organisation: **PO**
        Record Format: **FB**
        Record Length: **80**
        Blocksize: **Multiple of 80**

                    ****

2.   Install the Load Library

2.1   Unzip AR232dbob.zip

2.2 Upload the unzipped file AR232dbob.JCL to xxxxxx.AR232db.INSTALL(AR232d).

NOTE: This file is in EBCDIC format and needs to be loaded as a BINARY file it MUST be loaded WITHOUT ASCII → EBCDIC TRANSLATION. Do not [REPEAT NOT], use the ASCII CRLF option for the upload of this file.

2.2 AR232d contains JCL to produce the load library members and will need to be modified to your library name. So please modify both the SYSLIB and SYSLMOD DD statements to point to [your library name]. You will also no doubt need to review the JOB-name, and NOTIFY-id etc.

PLEASE NOTE THE JCL CONTAINS A WARNING THAT YOU SHOULD READ BEFORE EXECUTION.

Dataset Attributes:
Organisation: **PO**
Record Format: **U**
Record Length: **0**
Blocksize: **Your choice, suggest same as ADABAS**

Once the above Source and Load Libraries are in position and populated you should be in a position to run ADAREORG V2.32d.

xxxxxx.AR232d.INSTALL install library contains JCL giving examples of user-exits and ADAREORG execution jobs.

Please ensure you modify these to conform to your local standards making appropriate changes to the dataset names in these members to match the ones at your site, especially the JOBLIB/STEPLIB card to point to the V232 Load Library.

****

3. Apply Product Protection Code

ADAREORG requires our Product Authorisation Code, a 20-byte (or more) codeword will need to be supplied so that ADAREORG will run on your system.

The code is supplied to ADAREORG as the FIRST parameter in an ADAREORG Parameter dataset (or instream), eg:

CODE=**BIEJTJLJJJJLJHMKIHKH**
EXPAND 1
....
OR

The code may be permanently zapped into the ADAREORG load module, this zap must be created by CCA and takes the place of the CODE parameter. In order to be able to run ADAREORG, you will need either a codeword or a zap supplied by CCA or your local affiliate.