

# Keep DEDB database online while restructuring it

Jiří Vandas – May 30 – 2.17



## Prague Technology Days

May 30 - June 1, 2018

# For Informational Purposes Only

This presentation was based on current information and resource allocations as of **May 2018** and is subject to change or withdrawal by CA at any time without notice. Notwithstanding anything in this presentation to the contrary, this presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion. Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA will make such release available (i) for sale to new licensees of such product; and (ii) to existing licensees of such product on a when and if-available basis as part of CA maintenance and support, and in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis. In the event of a conflict between the terms of this paragraph and any other information contained in this presentation, the terms of this paragraph shall govern.

Certain information in this presentation may outline CA's general product direction. All information in this presentation is for your informational purposes only and may not be incorporated into any contract. CA assumes no responsibility for the accuracy or completeness of the information. To the extent permitted by applicable law, CA provides this presentation "as is" without warranty of any kind, including without limitation, any implied warranties or merchantability, fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised in advance of the possibility of such damages. CA confidential and proprietary. No unauthorized copying or distribution permitted.

# Agenda

1 DEDB INTERNAL STRUCTURE DESCRIPTION

2 WHICH ATTRIBUTES CAN BE CHANGED

3 PREREQUISITES

4 DBFUDA00 UTILITY

# DEDB Alter

- Data Entry Database (DEDB) from Fast Path database type family
- Utility included with IMS installation
- Allows to change DEDB Fast Path database layout and parameters while remaining database online and available for applications
- Runs as a standard Fast Path IFP utility

# Database Description for DEDB

```
DBD      NAME=DEALT, ACCESS=DEDB, RMNAME=(DBFHDC20, 2)
*
AREA1    AREA  DD1=DEALTDD1, SIZE=1024, UOW=(50, 10), ROOT=(100, 30)
AREA2    AREA  DD1=DEALTDD2, SIZE=1024, UOW=(50, 10), ROOT=(100, 30)
*
SEGM     NAME=SEGROOT, BYTES=300, PARENT=0
```

## RMNAME

Randomizer name and flag for two stage randomizer capability

## SIZE

Control Interval (CI) size of VSAM dataset

## UOW (Unit Of Work)

Number of CIs per one UOW

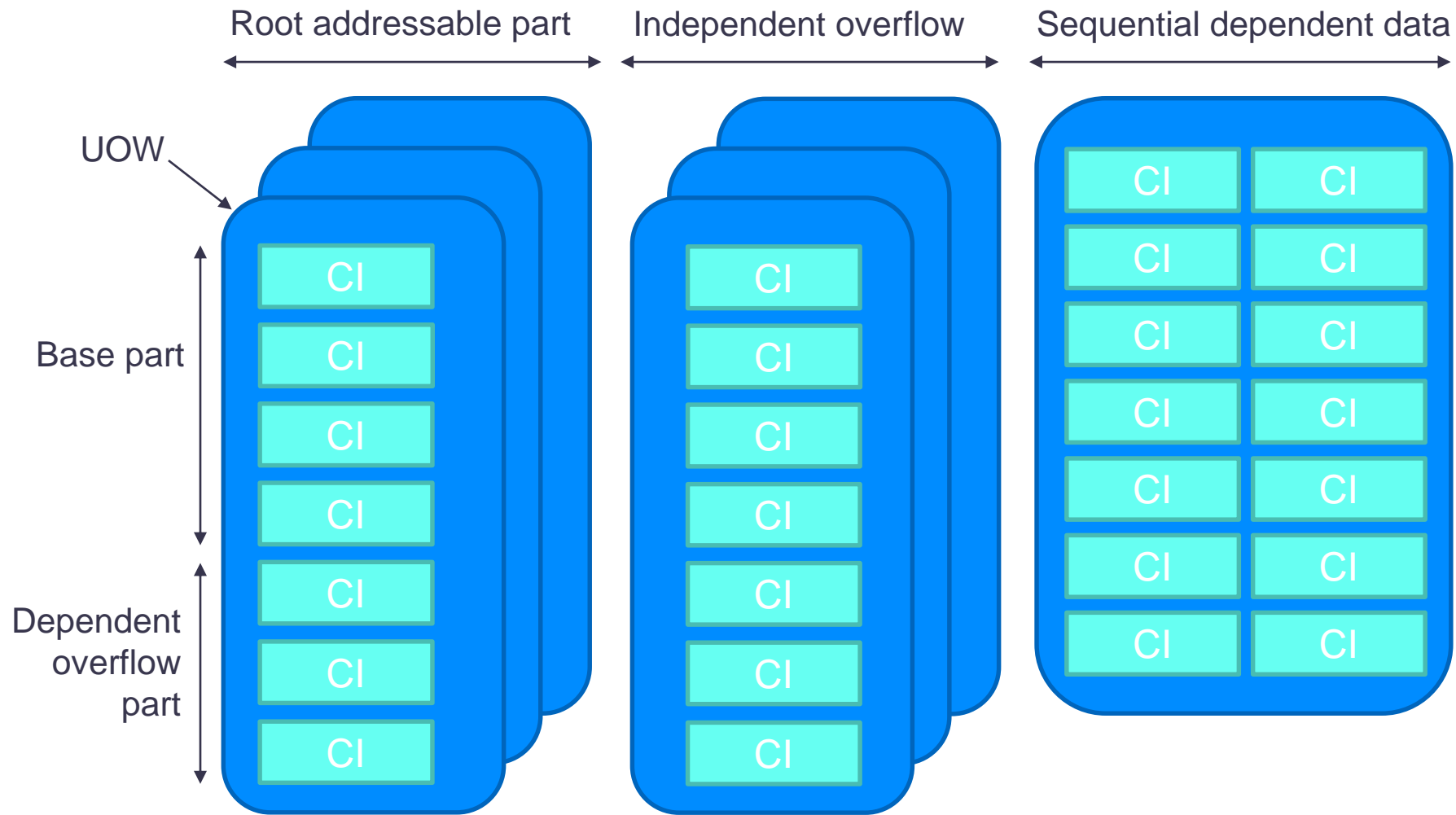
Subset of CIs dedicated to Dependent overflow part

## ROOT

Number of UOWs in both Root addressable and Independent overflow part

Number of UOWs in Independent overflow part

# Structure of DEDB area



# What you can change in IMS v13

```
DBD      NAME=DEALT, ACCESS=DEDB, RMNAME= (DBFHDC20, 2)
*
AREA1     AREA  DD1=DEALTDD1, SIZE=1024, UOW= (50, 10) , ROOT= (100, 30)
AREA2     AREA  DD1=DEALTDD2, SIZE=1024, UOW= (50, 10) , ROOT= (100, 30)
AREA3     AREA  DD1=DEALTDD3, SIZE=1024, UOW= (50, 10) , ROOT= (100, 30)
*
*
      SEGM  NAME=SEGROOT, BYTES=300, PARENT=0
```

- Physical DEDB area attributes like Control Interval size, sizes of various parts and overflows
- Randomizer name for the entire DEDB database

# What you can change in IMS v14

```
DBD      NAME=DEALT, ACCESS=DEDB, RMNAME= (DBFHDC20, 2)
*
AREA1     AREA  DD1=DEALTDD1, SIZE=1024, UOW= (50, 10) , ROOT= (100, 30)
AREA2     AREA  DD1=DEALTDD2, SIZE=1024, UOW= (50, 10) , ROOT= (100, 30)
AREA3     AREA  DD1=DEALTDD3, SIZE=1024, UOW= (50, 10) , ROOT= (100, 30)
AREAxX    add new area definition here...
*
SEGM      NAME=SEGROOT, BYTES=300, PARENT=0,
          COMPRTN= (IMEXPRES, DATA, INIT)
```

- Physical DEDB area attributes like Control Interval size, sizes of various parts and overflows
- Randomizer name for the entire DEDB database
- Add one or more DEDB areas
- Add segment edit/compression routine. The change or removal is not allowed.



# The process in the nutshell

## One time settings actions

- Update system configuration
- Create staging ACB library
- Update GENMAX to accommodate additional shadow image copies
- Allocate & register shadow area data set to DBRC

## For every execution

- Prepare and generate new ACB member
- Reallocate shadow area data sets and initialize them using DBFUMIN0
- Allocate and register new shadow image copy for area
- Execute DBFUDA00 utility

# New options in PROCLIB(DFSDFxxx)

- All new options are located in section <SECTION=FASTPATH>
- ACBSHR
  - Specifies whether ACB library is shared among IMS systems. Possible options are Yes or No. It allows to automatically share new ACB across IMS plex.
- ALTERGRP
  - The Name of DEDB Alter data-sharing group. It must be specified for each data sharing IMS system, which wants to participate in DEDB ALTER process.
- ALTERHLQ
  - High level qualifier for temporary data sets.

# IMS staging ACBLIB

- Contains the new version on generated DBDs with structural changes
- Different from ACBLIBA and ACBLIBB libraries
- Can be shared/reused with the one for HALDB ALTER
- The new library must be allocated by IMS under the ACBLIB DD:
- Can be hardcoded in IMS Control Region in JCL:  

```
//IMSACB      DD  DISP=SHR,DSN=IDI.IMSJ.ACBLIB
```

  - When running ACBGEN, you are not able to get exclusive access unless shutting down IMS region

# IMS staging ACBLIB

- Dynamically allocate by IMS from DFSMDA member

```
*  
*  DYNAMIC ALLOCATION OF STAGING ACB LIBRARY  
*  
DFSMDA TYPE=INITIAL  
DFSMDA TYPE=IMSACB,DSNAME=IDI.IMSJ.ACBLIB  
DFSMDA TYPE=FINAL  
END
```

- Compile and link-edit. The output must be placed in IMSACB named load module, which needs to be in IMS control region STEPLIB concatenation

# Recon requirements

- Upgrade Recon MINVER at least to 13.1
  - Sufficient level for using both IMS v13 or IMS v14 features
  - IMS regions themselves must be on proper release v13 or v14
    - All data sharing systems must be upgraded to the same release
- DEDB database and all areas must be registered in DBRC
  - When using Multiple area data sets database is already registered
  - Use INIT.DB, INIT.DBDS and INIT.ADS

# Shadow data sets

- Need to additionally allocate Shadow database data set
  - IDCAMS allocations according to the new DBD changes
- Register data set into DBRC for particular area

```
INIT.ADS DBD(DEALT) AREA(DEALTDD1) -
```

```
ADDN(DEALTDB1) -
```

```
ADSN(IDI.VANJI01.DEALTD1) -
```

**SHADOW**

- Both ADDN and ADSN names must be unique within DBRC
- For Multiple area data sets you must allocate the same number of shadow data sets (max. 6) unless you want to change them
  - Allocate one resp. multiple if you want to disable resp. enable MADS

# Shadow data sets

- Pre-allocate Shadow database image copy
  - IDCAMS allocations according to the new DBD changes

- Register image copy into DBRC for particular area

```
INIT.ADS DBD(DEALT) AREA(DEALTDD1) -  
    ADDN(DEALTDC1) -  
    ADSN(IDI.VANJI01D.IC3.DEALT.DEALTD1) -  
    SHADOW IC
```

- Increase CHANGE.DBDS GENMAX value to accommodate new image copy taken during execution

# Shadow data sets

- To check registration use LIST.DBDS

```
ADS LIST:
-ADS DDN--ADS DSN-          -STAT-  CREATE
DEALTDA1 IDI.VANJI01.DEALTDS1 AVAIL    -RUNNING-
                                      NO

SHADOW ADS LIST:
-ADS DDN--ADS DSN-          -STAT-  CREATE
DEALTDB1 IDI.VANJI01.DEALTD1  AVAIL    -RUNNING-
DEALTDC1 IDI.VANJI01D.IC3.DEALT.DEALTD1 AVAIL    NO  IC
```

- Problems:

After running DEDB ALTER these data sets will be swapped and the shadow data set becomes active → you always need to refer to DBRC for active data set names, which makes difficult to have areas allocated in JCL directly.



# Initialize shadow area / data set

- Make sure to ACBGEN the modified DBD and PSBs
- Initialize shadow data set by standard DBFUMIN0 utility
  - Needs to run with Recon data sets
  - ACBLIB points to staging ACB library
  - Specify area and SHADOW argument: **default is active data set !**

```
//DBFUMIN0 EXEC PGM=DBFUMIN0,PARM='DBRC=Y'  
//STEPLIB DD DISP=SHR,DSN=IDI.IMSJ.SDFSRESL  
//ACBLIB DD DISP=SHR,DSN=&STAGING(DEALT)  
//SYSPRINT DD SYSOUT=*  
//RECON1 DD DISP=SHR,DSN=IDI.VANJI01.IMSJ.RECON1  
//RECON2 DD DISP=SHR,DSN=IDI.VANJI01.IMSJ.RECON2  
//RECON3 DD DISP=SHR,DSN=IDI.VANJI01.IMSJ.RECON3  
//DEALTDB1 DD DISP=SHR,DSN=IDI.VANJI01.DEALTDT1  
//DEALTD1 DD DISP=SHR,DSN=IDI.VANJI01.D1C3.DEALT.DEALTDT1  
//CONTROL DD *  
AREA=DEALTDD1  
SHADOW
```

# Initialize shadow area / data set

- Problem: Omit shadow data set and image copy allocations in JCL

```
DFS3725I  AREA=DEALTDD1 JCL DD STATEMENT DOES NOT EXIST
```

- Bit more complicated to have a prepared (automatic) job already because data set names for area are changing/switching

- Problem: Reallocate shadow data set according to new DBD changes

```
DFS2509I  DEFINED CI SIZE IN CATALOG NOT CONSISTENT WITH DBD SIZE  
PARM
```

- To fix follow/copy DBDGEN recommended VSAM define cluster parameters

# Initialize shadow area / data set

- Problem: Reallocate shadow data set according to new DBD changes

DFS4620E DEDB ALTER VALIDATION ERROR: SHADOW AREA SDEP PART IS LESS  
FOR ALTERAREA FUNCTION

- Applicable only when using Sequential Dependent Segments
- Message from DEDB ALTER utility, but needs to be fixed in this step
- The new SDEP part can not be smaller than existing SDEP part
- New primary allocation = new RA + new IOVF + old SDEP
  - New RA and IOVF size: calculate from new DBD
  - Old SDEP size: calculate from old DBD or check /DISPLAY AREA output or pointer checker tool

# Run DEDB Alter utility DBFUDA00

- Running in IFP region
- Available functions:
  - ALTERAREA area\_name
  - ADDAREA
  - REPLRAND

```
//FPU      EXEC  PGM=DFSRR00,REGION=1M,  
//          PARM=(IFP,&DBD,DBF#FPU0,&REST,00,,1,  
//          &DIRCA,&PLRD,0,,, &IMSID,&AGN,&SSM,,  
//          &ALTID)  
//STEPLIB  DD  DISP=SHR,DSN=IDI.IMSJ.SDFSRESL  
//PROCLIB  DD  DISP=SHR,DSN=IDI.IMSJ.PROCLIB  
//SYSPRINT DD  SYSOUT=*  
//SYSUDUMP DD  SYSOUT=*  
//SYSIN    DD  *  
TYPE              ALTER  
ALTERAREA DEALTDD1  
RETRY              NO  
TIMEOUT           30  
UNKEYSEG  ALL  
GO
```

# Optional control statements

- TIMEOUT, RETRY, RETRYWAIT
  - Applicable since IMS v13
  - Allows utility to suspend applications in the end of the process to synchronize shadow area data set with active area and commit structural changes of DEDB
  - Defines the length of timeout for looking of quite point for a switch (default 15 seconds)
  - If the process fails, how many times it should be repeated (with how long pause in between)

# Optional control statements

- PHYSICAL option for ALTERAREA
  - Applicable in IMS v14
  - Applicable only when Sequential dependent segments are used
  - **ADJNO:** Does not adjust the SDEP logical begin to the first SDEP CI of the SDEP part in the shadow area data set. PHYSICAL ADJNO is the default.
  - **ADJYES:** Adjusts the SDEP logical begin to the first SDEP CI of the SDEP part in the shadow area data set, dissolves the wraparound state of the SDEP part, and reduces the size of the SDEP segments processed. The cycle count is reset to 1 for the SDEP logical begin.

# Optional control statements

- UNKEYSEG option for ALTERAREA
  - Applicable since IMS v13
  - **NONE**: If any un-keyed segments are defined in the DEDB definition, the utility will terminate.
  - **ISRTFILA**: The DEDB Alter utility is allowed to run if un-keyed segments are inserted with FIRST or LAST rule. Detection of any other update leads to utility termination.
  - **ALL**: The utility is allowed to run and processes updates that involve un-keyed segments, but various rules applies.

# Limitations...

- IMS v13 limitation
  - DEDB database areas with sequential dependent (SDEP) segments that are defined cannot be altered.
- IMS v13 and IMS v14 limitations
  - Databases in Virtual Storage Option (VSO) mode or Shared Virtual Storage Option (SVSO) mode are not directly supported. → You have to unload them back to DASD using /VUNLOAD command before the execution.
  - Only one area per DEDB can be altered at the same time
    - But multiple instances of utility can run concurrently on different DEDBs



# Two stage randomizer limitation

- To use all DEDB Alter features, real two stage randomizer is necessary
  - Randomizer must always based on the key assign the root segment into the same area regardless number of areas or current sizes of overflow parts etc.
  - Typically such randomizer must be table-driven
- IBM IMS provided randomizer DBFHDC20 and DBFHDC24
  - Is not real two stage randomizer
  - But still key will not move between areas even if the total number of RAPs in the DEDB area changes.
  - Usable only for ALTERAREA to change the ROOT or UOW parameters for any particular areas in the DBD.
  - Impossible to use when doing ADDAREA !



**Thank You.**



## **Jiří Vandas**

Principal Architect

Jiri.Vandas@ca.com



@cainc



slideshare.net/CAinc



linkedin.com/company/ca-technologies

**ca.com**