



# **z/OS Operating System V1R11**

## ***DISPLAY command options: ALLOC and SETALLOC***

@business on demand software

© 2009 IBM Corporation

This presentation will review the new ALLOC and SETALLOC options of the DISPLAY command.

## Agenda

- Session objectives
- Overview
- Usage and invocation
- Interactions and dependencies
- Migration and coexistence considerations
- Installation
- Session summary
- Appendix

This presentation will consist of; The Objectives of the presentation, an overview of the Problem to be solved and associated solution, how this function is invoked and used, Migration and Coexistence considerations, installation considerations, and finally, a session summary.

## Session objectives

- After this session, you should be able to:
  - ▶ Explain the purpose of the Display Alloc and Setalloc commands
  - ▶ Identify new/changed installation procedures
  - ▶ Define the functional content and benefit
  - ▶ Explain any migration issues or concerns
  - ▶ Explain how Display Alloc and Setalloc are invoked
  - ▶ Identify related Publications and References

After reviewing this presentation, you should be familiar with the Display Alloc and Setalloc commands, as well as understand how the commands are invoked and used, as well as understanding any installation and Migration considerations associated with this function, and related documentation.

## Overview

- **Problem statement / need addressed:**
  - ▶ Allow dynamic updates to the ALLOCxx parmlib options
  - ▶ Changes not needed that often, but costly when needed
  - ▶ Many Marketing Requirements for this item
  - ▶ Better diagnostic tools
- **Solution:**
  - ▶ Create new commands
    - One to display current Allocation group lock information
    - Another to display current ALLOCxx parmlib options
    - A third to update the options as needed
- **Benefit:**
  - ▶ Being able to diagnose group locking problems as it occurs instead of ex-post-facto
  - ▶ No IPL necessary to change important options

Over the years, MVS Device Allocation has been updated to allow many defaults and options to be set at IPL using the ALLOCxx parmlib member. The only problem with the above function is that it is not dynamic: customers must reIPL to change the settings.

The lack of command support for parameters in ALLOCxx compounds the aggravation already felt when trying to get “out of trouble.”

In addition, another type of display command had been requested to get a better snapshot of internal group-locks when a job hang situation arises. It is much easier to get a snapshot as it happens rather than looking through a dump.

This functional enhancement addresses customer requirements which have been received for this support.

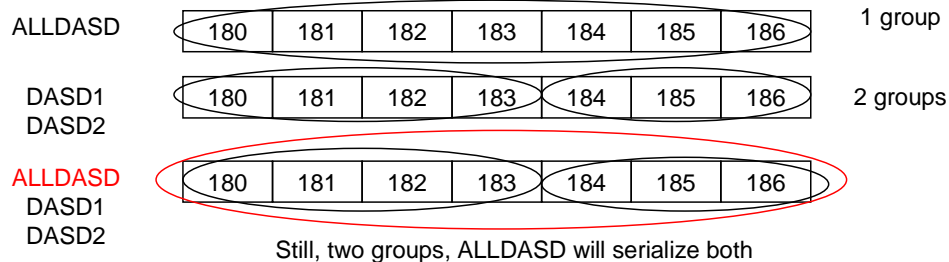
A design alternative would be to support a SET ALLOC=xx command, where all settings from a specified ALLOCxx parmlib member would be updated. However, from studying the parameters, PMRs and FITS requirements, the need is to change one or two errant parameters, not to swap in many parameter changes at one time.

For the display command, any customer may need to check what settings are active now, especially after the SETALLOC command has been used to change the IPL settings. Only those applicable settings are displayed.

The diagnostic capabilities for the D ALLOC,GRPLOCKS variant will allow new diagnostic capabilities and control over what has always been a difficult debug situation.

## Overview

- What are group locks?
  - ▶ Internal serialization mechanism
  - ▶ Serializes device selection during Common Allocation processing
  - ▶ Groups devices with similar attributes
    - Serializes many devices as one group



Allocation device groups locks are an internal serialization mechanism, used by allocation to serialize devices from other allocations going on in parallel. Allocation groups devices with similar attributes and esoteric requirements, allowing you to serialize the one group rather than many devices.

In the first example above, devices 180-186 are of the same device type and are defined as being only in the ALLDASD esoteric. Therefore it is one group.

In the second example above, devices 180-186 are defined as being in two esoterics, DASD1 and DASD2. Therefore two groups are created. If there is an allocation request to DASD1, then devices 180-183 are locked when determining the device to allocate to.

In the last example, all three esoterics are defined. But ALLDASD is a superset of DASD1 and DASD2, so still only two groups are defined. For example, if there are two allocation requests, one to DASD2 and then one to ALLDASD, the DASD2 case will serialize the selection of 184-186. The allocation that requests ALLDASD must wait for the other to release those group locking resources because 184-186 are contained within ALLDASD.

## Overview

- Using Display Alloc/Setalloc, you can:
  1. Display the current status of group locks
  2. Display the current ALLOCxx parmlib options
  3. Set new values for ALLOCxx parmlib options
- Value:
  1. Help debugging contention for group locks
  2. Check the active parmlib options
  3. Change any parmlib option without an IPL

Using the new Alloc/Setalloc commands, you will now be able to:

Group lock contention is a difficult situation to debug, especially in a dump or after a job has been hanging for a while and is cancelled. With a D ALLOC,GRPLOCKS variation, a customer can display the status of jobs for these three options: all, only those in contention, only those for a specific job or a specific device. This will help with possible solutions for a hang situation.

Displaying the active settings for ALLOCxx will allow the operator to see which options are currently in use. The operator can then determine if these options are suitable for their current Allocation needs.

Allowing you to dynamically update settings saves an IPL. An IPL is expensive, so any way to avoid this is a benefit. It can also help get out of situations where an IPL is necessary for any situation that requires a change to the ALLOCxx options. Since there are more buttons and tuning knobs than there were in years past, it becomes increasingly important to be able to update these settings.

## Usage and invocation

- The support is invoked by:
  - ▶ D ALLOC,GRPLOCKS command
  - ▶ D ALLOC,OPTIONS command
  - ▶ SETALLOC command
- Called by:
  - ▶ None

Issue an operator command to invoke the various functions. The different messages/displays are on the next few slides.

## Usage and invocation

- **New External Output**
  - ▶ **IEFA001I/IEFA002I for D ALLOC,GRPLOCKS**
    - IEFA001I displays the status
    - IEFA002I indicates a command failure
  - ▶ **IEFA003I for D ALLOC,OPTIONS**
    - Only displays applicable settings
  - ▶ **IEFA010I/IEFA011I for SETALLOC**
    - IEFA010I displays successful update of settings
    - IEFA011I indicates command failure

See the z/OS® MVS System Messages Vol 8 (IEF-IGD) book for detailed descriptions of these new messages.



## Usage and invocation

### ▪ Examples: D ALLOC,GRPLOCKS

```

SY1  D ALLOC,GRPLOCKS,CONTENTION
SY1  IEFA001I 15.31.25 ALLOC GROUPLOCKS 817
DISPLAYING GROUP LOCKS FOR JOBS IN CONTENTION ONLY.
GROUP 2 HAS THE FOLLOWING DEVICE(S)
00C0-00C5,02E0-02FF,0340-034F,0370-037F,0980-098F
JOBNAME  ASID  STATUS
JOB1     002C  OWNER
JOB2     002D  WAITER
DISPLAYING GROUP LOCKS FOR JOBS IN CONTENTION ONLY.
GROUP 12 HAS THE FOLLOWING DEVICE(S)
0593
JOBNAME  ASID  STATUS
JOB1     002C  OWNER
JOB2     002D  WAITER

```

This command is used to display information about the current group locks that are being held. Filters the operator can specify with this command include:

ALL/(no filter) – Displays all groups in which grouplocks are either being held or waited on.

CONTENTION/C – Displays all groups that are in grouplock contention (groups that have both owners and waiters on their grouplocks).

JOBNAME/J=jobname – Displays all devices in which grouplocks are either held or waited on by the particular jobname.

DEVICE/D=device – Displays all groups that are either holding or waiting on grouplocks for a particular named device.

In each of those groups, the group number, the devices, the jobnames, the asids, and the status of the jobs are displayed. If no groups match the criteria, then IEFA001I will display a message indicating that there is nothing to display. If you input an invalid jobname (one that contains a non-alphanumeric/non-national character) or an invalid device number (one that is not three or four characters), then IEFA002I is issued. Simply enter the device number, the '/' is not allowed. Only one device or jobname is allowed as input, not a list.

This current slide is displaying the result when CONTENTION is the filter

## Usage and invocation

### ▪ Examples: D ALLOC,OPTIONS

```

SY1  D ALLOC,OPTIONS
SY1  IEFA003I 15.37.28 ALLOC OPTIONS 886
SPACE          PRIMARY:          100
               SECONDARY:        30
               DIRECTORY:         5
               MEASURE:           AVEBLK
               BLKLNTH:           8192
               ROUND:             ROUND
               PRIM_ORG:           CONTIG
               RLSE:              RLSE
UNIT           NAME:             SYSALLDA
               UNITAFF:           3390
               REDIRECTED_TAPE:   TAPE
TIOT           SIZE:             32 (MAX DDS: 1635)

```

This command is used to display the current MVS Device Allocation settings that are in use, as set by the ALLOCxx parmlib member at IPL, as modified by the SETALLOC operator command, or the system defaults if no ALLOCxx member has been specified or if no SETALLOC command has been processed. IEFA003I will display the current options. Certain parmlib options are only displayed when they are applicable to the settings that the system is using. For example, if the SPEC\_WAIT POLICY is CANCEL or WTOR, the MAXNWAIT and POLICYNW are not displayed.

This slide shows part of the command output.

## Usage and invocation

### ▪ Examples: D ALLOC,OPTIONS

```
SDSN_WAIT          WAITALLOC:          NO
VOLUME_ENQ         POLICY:             WTOR
VOLUME_MNT         POLICY:             WTOR
SPEC_WAIT          POLICY:             WAITNOH
                  MAXNWAIT:            5
                  POLICYNW:           WTOR
ALLC_OFFLN         POLICY:             WAITNOH
                  MAXNWAIT:            5
                  POLICYNW:           WTOR
CATLG_ERR          FAILJOB:            YES
                  ERRORMSG:          YES
2DGT_EXPDT         POLICY:             WARN
VERIFY_VOL         POLICY:             YES
SYSTEM            IEFBR14_DELMIGDS:  NORECALL
                  TAPELIB_PREF:      BYDEVICES
                  REMIND_INTV:       60
```

This slide shows the rest of the IEFA003I output for the D ALLOC,OPTIONS command. Unfortunately there is no way to determine if these values displayed are the system defaults, the defaults as specified in ALLOCxx, or a value that had been changed with the SETALLOC command.

## Usage and invocation

### ▪ Examples: SETALLOC

```

- SY1 SETALLOC ALLC_OFFLN,POLICY=WAITHOLD
- SY1 IEFA011I SETALLOC COMMAND FAILED
- POLICYNW MUST ALSO BE SET.

- SY1 SETALLOC ALLC_OFFLN,POLICY=WAITHOLD,POLICYNW=CANCEL
- SY1 IEFA010I SETALLOC COMMAND SUCCESSFUL
- ALLC_OFFLN POLICY SET TO WAITHOLD.
- ALLC_OFFLN POLICYNW SET TO CANCEL.

- SY1 SETALLOC SPEC_WAIT,MAXNWAIT=5
- SY1 IEFA011I SETALLOC COMMAND FAILED
- POLICY MUST BE SET TO WAITNOH TO SET MAXNWAIT.

- SY1 SETALLOC SPEC_WAIT,MAXNWAIT=5,POLICY=WAITNOH,POLICYNW=CANCEL
- SY1 IEFA010I SETALLOC COMMAND SUCCESSFUL
- SPEC_WAIT POLICY SET TO WAITNOH.
- SPEC_WAIT MAXNWAIT SET TO 5.
- SPEC_WAIT POLICYNW SET TO CANCEL.

```

The syntax is borrowed from the ALLOCxx parmlib member syntax, with the exception of using commas and equal sign instead of spaces and parenthesis. Refer to the z/OS MVS Initialization and Tuning reference for all of the acceptable parameters, their functions and their possible values before using this command.

IEFA010I displays each option which was set in the command when all input is valid. The operator may set more than one parameter in the same category, but may not specify the same option twice. If a parameter is specified twice, then the command is rejected, no parameters are set.

SETALLOC does not support the 2DGT\_EXPDT POLICY keyword.

Only Allocation options which are applicable to the current settings are allowed to be set, and likewise, all applicable Allocation options are required when changing to such a setting that has dependent keywords.

For example, when setting the ALLC\_OFFLN POLICY to WAITHOLD, the POLICYNW must also be set in the same command or else the new value is rejected and IEFA011I is issued. The first block shows that the command will be rejected, and the second block shows the correct way for changing that particular value.

Another example is when the SPEC\_WAIT POLICY is not WAITNOH, the operator may not change the SPEC\_WAIT MAXNWAIT value since it only applies to the WAITNOH POLICY. Change the SPEC\_WAIT POLICY to WAITNOH in order to change the value of MAXNWAIT. Once the SPEC\_WAIT policy is WAITNOH, MAXNWAIT can be changed by itself, without having to specify those other options. The third block shows the command being rejected, while the fourth block shows the correct way to set that option.

If any of the values specified for an option are not valid, then the command is rejected and IEFA011I is issued.

## Interactions and dependencies

- Hardware dependencies
  - ▶ None
- Software dependencies
  - ▶ None
- Exploiters
  - ▶ None

There are no Hardware or Software Dependencies associated with this function.

## Migration and coexistence considerations

- No migration actions
- No coexistence concerns

There are no migration or coexistence considerations associated with support.

## Installation

- New Command Authority Profiles
  - ▶ MVS.DISPLAY.ALLOC
    - For D ALLOC,GRPLOCKS and D ALLOC,OPTIONS
    - Read RACF® authority
  - ▶ MVS.SETALLOC.ALLOC
    - For SETALLOC
    - Update RACF authority

For additional information, see the MVS System Commands book.

## Session summary

- The D ALLOC,GRPLOCKS command is a diagnostic aid
- The D ALLOC,OPTIONS command displays current ALLOCxx options
- The SETALLOC command allows for dynamic updates to current Allocation settings

To summarize, the DISPLAY ALLOC and SETALLOC enhancements provide a dynamic method of updating the current Allocation options. This enhancement also provides a great problem-solving tool to show the status of internal group locks at a snapshot instead of in a dump after the fact. IPLs which were previously necessary to update the ALLOCxx options are now obsolete with the new SETALLOC command. One can also display the current options and change them if need be.



## Appendix

### ▪ Publications

- ▶ z/OS MVS Initialization and Tuning Reference (SA22-7592)
- ▶ z/OS MVS System Commands (SA22-7627)
- ▶ z/OS MVS System Messages,  
▶ Vol 8 (IEF-IGD) (SA22-7638)

The following publications are provided for reference.

## Feedback

### Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

[mailto:iea@us.ibm.com?subject=Feedback\\_about\\_V1R11\\_BCP\\_DisplayCmdsl.ppt](mailto:iea@us.ibm.com?subject=Feedback_about_V1R11_BCP_DisplayCmdsl.ppt)

This module is also available in PDF format at: [..\V1R11\\_BCP\\_DisplayCmdsl.pdf](..\V1R11_BCP_DisplayCmdsl.pdf)

You can help improve the quality of IBM Education Assistant content by providing feedback.

## Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

RACF z/OS

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>

Other company, product, or service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2009. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.