

# CA JCLCheck™ Workload Automation

## Command Reference Guide

Version 12.0.00 Second Edition



This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, 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 LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2014 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

## CA Technologies Product References

This document references the following CA Technologies products:

- CA Workload Automation CA 7® Edition (CA WA CA 7 Edition)
- CA Workload Automation Restart Option for z/OS Schedulers (CA WA Restart Option for z/OS Schedulers)
- CA Scheduler® Job Management (CA Scheduler)
- CA Workload Automation ESP Edition (CA WA ESP Edition)
- CA APCDOC™ Automated Job Documentation (CA APCDOC)
- CA Dispatch™ (CA Dispatch)
- CA ACF2™ (CA ACF2)
- CA Top Secret® (CA Top Secret)
- CA ASM2® Backup and Restore (CA ASM2)
- CA 1® Tape Management (CA 1)
- CA TLMS® Tape Management (CA TLMS)
- CA Roscoe® Interactive Environment (CA Roscoe)
- CA Librarian® (CA Librarian)
- CA Panvalet® (CA Panvalet)

# Contact CA Technologies

## Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At <http://ca.com/support>, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

## Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to [techpubs@ca.com](mailto:techpubs@ca.com).

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at <http://ca.com/docs>.

# Documentation Changes

The following documentation updates have been made since the last release of this documentation:

## Version 12.0 Second Edition

- [Required Execution JCL](#) (see page 31)—Updated the PROCnn exec statement.
- [CA7](#) (see page 133)—Changed the default value of the schedule-id operand from zero (0) to one (1).

## Version 12.0

### Resolve External Variables

- [RESOLVE](#) (see page 188)—New.
- [The EDCHEK Options Menu \(JCK21\)](#) (see page 78)—Updated to include the new RESOLVE option.
- [Product Support Options](#) (see page 117)—Updated to include the new RESOLVE option.
- [Flowchart of JCKSPF Panels for CA JCLCheck Dialog](#) (see page 38)—Updated to include the new variable resolution dialogs for the product schedulers.
- [CA JCLCheck/SPF Menu \(JCK0200\)](#) (see page 40)—Updated to include the new RESOLVE option.
- [Additional Processing Options Panel \(JCK0203\)](#) (see page 48)—Updated to include the new RESOLVE option.

### DFSORT and ADRDSSU Utility Control Card Support

- [Process Control Options Panel \(JCK0202\)](#) (see page 45)—Added DFSORT and ADRDSSU to the list of IBM utility programs.
- [Process Control Options \(JCKOPN1\)](#) (see page 89)—Added DFSORT and ADRDSSU to the list of IBM utility programs.
- [CTLSCAN](#) (see page 142)—Added DFSORT and ADRDSSU to the list of IBM utility programs.

### REXX Enhancements

- [Optional Parameters for Default Mode \(JCK0205\)](#) (see page 269)—Updated the FORMAT options.
- [Optional Parameters for Submit Mode \(JCK0205\)](#) (see page 273)—Updated the FORMAT options.

- [Output Options](#) (see page 289)—Updated the FORMAT option description.
- [FORMAT](#) (see page 295)—Updated the syntax and usage notes.
- [REXXMEM](#) (see page 303)—Updated the usage notes.
- [JCLNeat Options Syntax Summary](#) (see page 306)—Updated the FORMAT options.

#### ISPF List of REXX Rules

- [Additional Processing Options Panel \(JCK0203\)](#) (see page 48)—Updated the STDREXX description.
- [The EDCHEK Options Menu \(JCK21\)](#) (see page 78)—Updated the STDREXX description.
- [Optional Parameters for Default Mode \(JCK0205\)](#) (see page 269)—Updated the REXXMEM option.
- [Optional Parameters for Submit Mode \(JCK0205\)](#) (see page 273)—Updated the REXXMEM option.
- [JCL Statement Reformatting Options](#) (see page 287)—Updated the REXXMEM description.
- [REXXMEM](#) (see page 303)—Added a Usage note about specifying REXXMEM.

#### JCLNeat PACK Option—Maximum Keywords per Line

- [PACK](#) (see page 301)—Updated the PACK description and options.

#### JCLNeat—Options by Statement Type

- [Batch Submission Panel Sequence](#) (see page 265):
  - Updated the Default and Submit processing descriptions.
  - Added panel JCK0205A.
- [Statement Level Options for Default Mode \(JCK0205A\)](#) (see page 270)—New.
- [JOB CARD for Default Mode \(JCK0214\)](#) (see page 272)—Updated the panel with new parameters.
- [Statement Level Options for Submit Mode \(JCK0205A\)](#) (see page 274)—New.
- [Optional Data Sets \(JCK0206\)](#) (see page 275)—Updated the panel.
- [JOB CARD for Submit Mode \(JCK0214\)](#) (see page 280)—Updated the panel.
- [JCK0POPR Pop-Up Verification](#) (see page 282)—Updated the panel.
- [Comment Options](#) (see page 286)—Added statement level options for the ICSC option.
- [JCL Statement Reformatting Options](#) (see page 287):
  - Added statement level options for the CEC option.

- Added statement level options for the CSC option.
- Added statement level options for the OPSC option.
- Added statement level options for the PACK option.
- [CEC](#) (see page 292)—Added statement level options.
- [CSC](#) (see page 293)—Added statement level options.
- [ICSC](#) (see page 296)—Added statement level options.
- [OPSC](#) (see page 299)—Added statement level options.
- [PACK](#) (see page 301)—Added statement level options.
- [JCLNeat Options Syntax Summary](#) (see page 306)—Added statement level options for CEC, CSC, ICSC, OPSC, and PACK.

#### **JCLNeat—Global String Replacement by Statement Type**

- [Global String Replacement](#) (see page 259)—Added the new JCL statement identifier *JCL-type*.
- [GSRCNTL \(JCK0213\)](#) (see page 279)—Updated the panel.

#### **Missing DSN Bypass and Selective RUNTIME Checks**

- [OPROCLIB](#) (see page 175)—New.
- [CNTLOVER](#) (see page 137)—New.
- [BYPRUNT](#) (see page 130)—New.
- [Control Card Override Panel \(JCK12\)](#) (see page 67)—New.
- [OPROC and OPROCLIB Panel \(JCK13\)](#) (see page 68)—New.
- [Runtime Options Panel \(JCK14\)](#) (see page 69)—New.
- [Additional Processing Options Panel \(JCK0203\)](#) (see page 48)—Added information about the OPROC and CNTLOVER options.

#### **Control Card Override**

- [CNTLOVDD](#) (see page 136)—New.
- [CNTLOVER](#) (see page 137)—Added usage notes.
- [Control Card Override Panel \(JCK12\)](#) (see page 67)—Added a usage note for CNTLOVDD.

#### **Add Inactive Option Counterpart to Runtime Options**

- [ALLOWMNF](#) (see page 127)—New.

- [Runtime Options Syntax Summary](#) (see page 214)—Updated the table with runtime option information.

Updated the following runtime options with an inactive option counterpart:

- [ALLOWMNF](#) (see page 127)
- [ASM2](#) (see page 128)
- [BLKMSG](#) (see page 130)
- [BYPROUT](#) (see page 130)
- [CA11](#) (see page 132)
- [CA7CTL](#) (see page 133)
- [DATEFMT](#) (see page 144)
- [DBGJCL](#) (see page 145)
- [DCBCHK](#) (see page 145)
- [DFLTJOB](#) (see page 149)
- [DICTDD](#) (see page 150)
- [DISPATCH](#) (see page 150)
- [DSNERR](#) (see page 151)
- [EDCHKLB](#) (see page 154)
- [EDCHKSL](#) (see page 154)
- [EXCLUDE](#) (see page 156)
- [FLUSHRC](#) (see page 156)
- [GRAPH](#) (see page 157)
- [IEBMSG](#) (see page 159)
- [INCLUDE](#) (see page 159)
- [INDEX](#) (see page 160)
- [INPUT](#) (see page 160)
- [INTUSR](#) (see page 161)
- [JESCRUNT](#) (see page 162)
- [JOB CARD](#) (see page 163)
- [JOB BRAM](#) (see page 163)
- [JOBTYPE](#) (see page 164)

- [KEEP7380](#) (see page 164)
- [KILLSEP](#) (see page 165)
- [LIBLIST](#) (see page 165)
- [LINECNT](#) (see page 166)
- [MIGVOL](#) (see page 171)
- [NJBCHK](#) (see page 173)
- [ONEMEM](#) (see page 174)
- [OPROCLIB](#) (see page 175)
- [PASSWORD](#) (see page 179)
- [PNFMSG](#) (see page 181)
- [PREFIX](#) (see page 182)
- [PROC](#) (see page 183)
- [REVINDEX](#) (see page 190)
- [ROSLIMIT](#) (see page 192)
- [SCHEDLER](#) (see page 194)
- [SEV](#) (see page 197)
- [SRCH](#) (see page 201)
- [STANDARD](#) (see page 201)
- [STEPMSG](#) (see page 202)
- [STPCAT](#) (see page 203)
- [SUBLIB](#) (see page 204)
- [SUPCAT](#) (see page 205)
- [SYSTEM](#) (see page 208)
- [SYSUID](#) (see page 208)
- [TSOE](#) (see page 210)
- [USRMSG](#) (see page 211)
- [UTLSEV](#) (see page 211)
- [XREFAL](#) (see page 214)

#### **Documentation Fixes**

- [SXREF](#) (see page 205)—Added the NOMEM option to the syntax.
- [XREF](#) (see page 213)—Added the NOMEM option to the syntax.

- [Runtime Options Syntax Summary](#) (see page 214):
  - Added the NOMEM option to the SXREF and XREF runtime options.
  - Added the CSI runtime option.
  - Added the USERDATA runtime option.
  - Added the VERIFY and RUNTIME options to the PREVEX runtime option.
  - Added the SIGNON runtime option.
- [Validation Options](#) (see page 113):
  - Added the CSI runtime option
  - Added the USERDATA runtime option.
- [Security Validation Options](#) (see page 120)—Added the SIGNON runtime option.
- [SIGNON](#) (see page 198)—New.
- [USERDATA](#) (see page 211)—New.
- [PREVEX](#) (see page 182)—Added the VERIFY and RUNTIME option.
- [DYNEXITS](#) (see page 152)—Added the XGET suffix name for the CAZ1XGET assembler exit.
- [TMS](#) (see page 210)—Updated the CA 1 versions.
- [AUTOPROC](#) (see page 128)—Added a bullet regarding duplicate allocations.
- [Invoking from a CLIST](#) (see page 83)—Updated the CAZ1ALDS example.
- [ISPF Edit of the Job Submission JCL](#) (see page 281)—Updated the screen example that shows the job submission JCL.
- Specification of JCLNeat Options Using the JCLNOPTS Statement—New.
- [CSI](#) (see page 141)—New.
- [Process Control Options Panel \(JCK0202\)](#) (see page 45)—Updated the screen example and text to include CSI.

Updated the following runtime options with CSI mutually exclusive information:

- [INDEX](#) (see page 160)
- [LOCATE](#) (see page 167)
- [PVT CAT](#) (see page 185)
- [REVINDEX](#) (see page 190)
- [STPCAT](#) (see page 203)
- [SUBALLOC](#) (see page 203)
- [VSAM](#) (see page 212)

# Contents

---

## Chapter 1: Introduction 19

A CA JCLCheck Sample Scenario .....	19
Product Features .....	20
Application Support.....	23
JCLNeat Component.....	26

## Chapter 2: Using CA JCLCheck 29

Input .....	30
Invoking with the Batch Facility .....	30
Required Execution JCL .....	31
Cataloged Procedure .....	36
Using the ISPF Online Panels .....	36
Flowchart of JCKSPF Panels for CA JCLCheck Dialog .....	38
CA JCLCheck/SPF Menu (JCK0200) .....	40
Control Options Menu (JCK0210).....	42
Invocation Panel (JCK0201) .....	43
Process Control Options Panel (JCK0202) .....	45
Additional Processing Options Panel (JCK0203).....	48
Output Control Options Panel (JCK03) .....	51
Security Options Panel (JCK0204) .....	53
Definition of Alternate PROCLIBS Panel (JCK05) .....	55
Specification of JOB CARD Panel (JCK07) .....	55
Define JCLCheck/APCDOC Database Panel (JCK08) .....	56
Member Select for PDS (CAZ1SLEP/CAZ1SLRP) Update and Browse .....	57
Member Select for CA Librarian (CAZ1SLLP/CAZ1DTPP) Update and Browse .....	59
Member Select for CA Panvalet (CAZ1SLNP/CAZ1DTPP) Update and Browse .....	61
Job Scheduling Interface for CA 7 Panel (JCK10) .....	63
Job Scheduling Interface for CA Scheduler Panel (JCK09).....	65
Control Card Override Panel (JCK12) .....	67
OPROC and OPROCLIB Panel (JCK13) .....	68
Runtime Options Panel (JCK14).....	69
Printoff Panel (JCK04).....	70
Alternate Print Panel (JCK04I) .....	72
Invoking with the EDCHEK Facility.....	73
EDCHEK Screen Without Error Messages.....	74
EDCHEK Screen with Error Messages.....	75

---

EDCHEK Screen Using EDCHKLB .....	75
EDCHEK Screen Using EDCHKSL .....	77
EDCHEK Screen Using EDCHKSL and EDCHKLB.....	78
The EDCHEK Options Menu (JCK21).....	78
Invoking with the SUBCHEK Facility .....	82
Invoking remote validation using ChekPlex .....	82
Invoking from a CLIST.....	83
Invoking Under CA Roscoe .....	86
The JCKRPF RPF .....	87
CA JCLCheck/Roscoe Menu (JCKMAINR).....	87
CA JCLCheck Control Options Menu (JCKOPCTL) .....	88
Process Control Options (JCKOPN1).....	89
Additional Processing Options Panel (JCKOPN1A) .....	91
Output Control Options Panel (JCKOPN2).....	92
Security Options Panel (JCKOSEC).....	95
CA JCLCheck/Invocation Menu .....	96
Job Scheduling Interface for CA 7 (JCKCA7) .....	98
Job Scheduling Interface for CA Scheduler (JCKSSCD) .....	99
Producing JCL Flow Diagrams.....	101
Operational Notes.....	102
Printer Support.....	102
Line/Nonimpact Printers (Line Printer Mode) .....	102
3800-1 Printers.....	102
3800-3 Printers.....	102
3820 Printers.....	103

## **Chapter 3: CA JCLCheck Runtime Options 107**

Specifying Options.....	107
Active and Inactive Options (or the NO Option Prefix) .....	108
Processing/Evaluation Rules .....	109
Summary of Options by Function .....	110
Input Specification Options.....	110
Supplementary Input Options.....	111
Output Options .....	112
Validation Options .....	113
Environment Specification Options.....	115
System/Related Job Analysis Options .....	117
Product Support Options .....	117
Security Validation Options.....	120
Diagnostic Options .....	121
Report File Options .....	121

---

Message Control Options .....	122
Report Format Options .....	123
Standard Report Options .....	124
Supplementary Report Options .....	124
Description of Runtime Options .....	126
ACF2 .....	126
ACTION .....	126
ALLOWMNF .....	127
APPEND .....	127
ASM2 .....	128
AUDIT .....	128
AUTOPROC .....	128
BLKMSG .....	130
BYPROUT .....	130
BYPRUNT .....	130
CA11 .....	132
CA7 .....	133
CA7CTL .....	133
CADRIVER .....	134
CCLIST .....	134
CHANGE .....	134
CNTL .....	135
CNTLOVDD .....	136
CNTLOVER .....	137
COBEXIT .....	139
COND .....	140
CSI .....	141
CTLSCAN .....	142
DATEFMT .....	144
DB2 .....	144
DBCS .....	145
DBGJCL .....	145
DBHLQ .....	145
DCBCHK .....	145
DEBUG .....	146
DELTA .....	147
DEQUE .....	147
DESTCHK .....	148
DFLTJOB .....	149
DICT .....	149
DICTDD .....	150
DIRUPD .....	150

---

DISPATCH .....	150
DSNERR .....	151
DUMPUCB .....	151
DYNEXITS.....	152
EASYPROC .....	152
EASYRDR.....	153
EDCHEK .....	153
EDCHEL.....	153
EDCHKLB.....	154
EDCHKSL.....	154
ERROR .....	154
EXCLUDE.....	156
FACOM .....	156
FLUSHRC.....	156
FULLLIST .....	157
GRAPH .....	157
HCD .....	158
IAM.....	158
IEBMSG.....	159
IMS .....	159
INCLUDE .....	159
INDEX .....	160
INPUT .....	160
INTUSR .....	161
JCLLIB.....	161
JES3 .....	162
JESCHEK.....	162
JESCRUNT .....	162
JOB .....	163
JOBCARD .....	163
JOBJBRAM .....	163
JOBSEV .....	164
JOBTYPE.....	164
KEEP7380 .....	164
KILLSEP .....	165
LIBLIST .....	165
LIBRARIAN .....	165
LINECNT.....	166
LIST .....	167
LOCATE .....	167
MCOSYS.....	168
MCOUSR.....	170

---

MIGVOL .....	171
MSGLEVEL .....	172
MSS .....	172
NETWORK.....	173
NJBCHK.....	173
NOSPIE .....	173
ONEMEM.....	174
OPROC.....	174
OPROCLIB.....	175
OPTIONS.....	176
OUTFMT .....	177
PANEXEC .....	178
PANVALET .....	178
PASSWORD.....	179
PDS .....	180
PDSRELATE .....	181
PNFMSG .....	181
PRCDELTA.....	181
PREDNO.....	182
PREFIX .....	182
PREVEX .....	182
PROC .....	183
PROCDLFT.....	184
PROCXREF.....	185
PULL .....	185
PVTCAT .....	185
PXREF .....	186
RDR.....	187
REMOTE .....	187
REPORT.....	188
RESOLVE .....	188
REVINDEX.....	190
REXXMSG .....	190
ROSCOE .....	191
ROSLIMIT.....	192
RPTRPT .....	192
RUNCPU .....	193
RUNDATE.....	193
RUNTIME .....	193
RV .....	194
SCHEDLER.....	194
SCHENV .....	194

---

SECURITY .....	195
SEV .....	197
SIGNON .....	198
SMS .....	199
SPACE .....	199
SRCH .....	201
STACK .....	201
STANDARD .....	201
STDREXX .....	202
STEPMSG .....	202
STPCAT .....	203
SUBALLOC .....	203
SUBCHEK .....	203
SUBLIB .....	204
SUBMIT .....	204
SUPCAT .....	205
SUPRPT .....	205
SXREF .....	205
SYNTAX .....	207
SYSCHEK .....	207
SYSID .....	207
SYSTEM .....	208
SYSUID .....	208
TERM .....	209
TLMS .....	209
TMS .....	210
TSOE .....	210
USER .....	211
USERDATA .....	211
USRMSG .....	211
UTLSEV .....	211
VIOPUS .....	212
VSAM .....	212
XEROX .....	213
XREF .....	213
XREFAL .....	214
Runtime Options Syntax Summary .....	214

<b>Chapter 4: CA JCLCheck Reports</b>	<b>225</b>
Report 1 - Listing of Jobstream JCL .....	226
Report 2 - Listing of Merged JCL .....	226

---

Report 3 - Data Set Cross-Reference .....	229
Report 4 - Program Cross-Reference.....	232
Report 5 - Report Listing .....	233
Report 6 - Error Messages.....	234
Report 7 - Summary Data Set Cross-Reference.....	236
Report 8 - Summary Program Cross-Reference .....	238
Report 9 - Summary Report Listing .....	239
Report 10 - Flow Diagram .....	240
Operational notes .....	241
Report 11 - Procedure Cross-Reference.....	244
Audit Report.....	245

## **Chapter 5: Using JCLNeat** **247**

Manual Control of JCLNeat .....	248
Including/Excluding Members .....	248
Required Execution JCL for JCLNeat.....	249
Specifying Options.....	251
Keyword Reordering .....	252
Global String Replacement.....	259
Global Editing.....	262
Symbolic Replacement.....	263
ISPF Panels Control of JCLNeat.....	264
Batch Submission Panel Sequence.....	265
CA JCLCheck/SPF Menu (JCK0200).....	267
Defaults Mode Panel Descriptions.....	267
Submit Mode Panel Descriptions.....	273
JCLNeat Output .....	282
Using JCLNeat under CA Roscoe.....	283

## **Chapter 6: JCLNeat Options** **285**

Summary of JCLNeat Options by Function .....	285
Input Options .....	285
Comment Options.....	286
Debugging Options.....	286
JCL Statement Reformatting Options.....	287
Output Options .....	289
Description of JCLNeat Options.....	290
BCI .....	290
BCS .....	291
CDEL .....	291
CEC .....	292

---

CSC .....	293
DCBS .....	294
FCC .....	294
FORMAT .....	295
ICSC .....	296
INCR .....	297
INLIB .....	297
JOBF .....	298
LINES .....	298
MEMBERS .....	298
MKW .....	299
MQKW .....	299
OPSC .....	299
OPTDESC .....	300
OUTLIB .....	301
PACK .....	301
RECOVER .....	302
REXXMEM .....	303
REPORT .....	304
RSYM .....	305
SECC .....	305
SEQ1 .....	305
SIM .....	306
JCLNeat Options Syntax Summary .....	306

## **Chapter 7: JCLNeat Reports** **311**

Options Report .....	312
Specifying GSRCNTL DD Control Cards .....	313
Original JCL Report .....	314
Specifying GSRCNTL DD Control Cards .....	315
Reformatted JCL Report .....	316
Changes Report .....	317
JCLNeat Options Report Using REXX .....	318
JCLNeat Original JCL Report Using REXX .....	319

## **Glossary** **321**

# Chapter 1: Introduction

---

This guide is intended for all users of CA JCLCheck. It contains descriptions of commands, control options, reports, and utilities that are part of the product.

## A CA JCLCheck Sample Scenario

A customer of CA Technologies discovered that JCL errors were costing them time and money. Common errors went unnoticed, causing expensive reruns. A recurring common error, such as 'DATA SET NOT FOUND,' caused jobs to ABEND after execution of the first several steps. The systems programmer had to analyze the JCL to locate the source of the problem and correct it, and the job had to be rerun.

The customer purchased CA JCLCheck and established a standard to scan their production JCL in batch, using CA JCLCheck to flag errors and check for nonzero return codes. CA JCLCheck improved efficiency right away, here's how.

The night operator, Joe, discovered that the initial production of JOBR was going to ABEND, as indicated by the error messages listed on [CA JCLCheck Report 6 - Error Messages](#) (see page 234).

The error messages read:

```
CAY6186E MISSING GDG NUMBER
CAY6072E CONTINUATION STATEMENT REFERENCES A NONEXISTENT DATA SET
CAY6033W OVERRIDES OUT OF ORDER
```

Joe compared other reports generated by CA JCLCheck to the problems indicated by the error messages to find the solution.

He read Report 1 - Listing of Jobstream JCL to see the JCL exactly as it was submitted for the job. He read Report 2 - Listing of Merged JCL to see the entire JCL for the job including procedure library statements. Joe compared these reports and determined that the submitted JCL was incomplete without the GDG reference. He compared the cataloged GDG with the GDG being referenced in the job stream JCL. If the submitted JCL referenced the cataloged data set, the GDG reference was valid.

Next, Joe determined from Report 2 - Listing of Merged JCL that the data set existed in a PDS that was entered in the JCL. Using one of the CA JCLCheck ISPF panels, Joe added the PDS that contained the member called by JOBR.

Finally, he discovered that the overrides were out of order because one had been deleted for a previous run and had not been reinstated for future runs. Joe inserted the override into the job stream JCL for JOBR and resubmitted the JCL.

The evening's production continued without ABENDs. When the shift ended, the correction to the JCL was noted and production ran as if a problem never occurred.

## Product Features

The following topics describe CA JCLCheck features:

- **Online Operation**

You can run CA JCLCheck online in a variety of time-sharing environments. Interfaces are supplied for TSO/ISPF and CA Roscoe. The dynamic invocation feature accommodates other time-sharing systems.

**Note:** For more information, see Special Usage Considerations in the *Programming Guide*.

- **Batch Operation**

CA JCLCheck runs in batch mode. It reads one or several job streams, and produces up to 11 reports that document the job stream and diagnose its errors. For a complete description of the CA JCLCheck reporting features, see [CA JCLCheck Reports](#) (see page 225).

Batch execution is fast (typically under one second on a 3081 for a 600 statement job stream), so CA JCLCheck runs in a quick turnaround job class.

- **Utility Control Statements**

CA JCLCheck recognizes certain IBM and user applications as utilities, capable of performing JCL-like functions, directed by control statements. CA JCLCheck checks for required DD statements and optionally reads and interprets control statements associated with these utility program functions.

**Note:** For more information, see Utility Control Statements in the *Programming Guide*.

- **SUBCHEK**

SUBCHEK is a feature that allows CA JCLCheck to run as a front-end processor for the TSO SUBMIT command. CA JCLCheck automatically validates any JCL that is submitted using TSO before the JCL is sent to the internal reader. CA JCLCheck voids the submit process if it encounters any errors and returns an error list to the submitter's terminal.

The automatic validation of JCL by CA JCLCheck whenever you submit a job from TSO EDIT is the major benefit of SUBCHEK.

- **EDCHEK**

EDCHEK is a feature that provides the ultimate online usage mode for CA JCLCheck. This allows CA JCLCheck to run as an edit macro under ISPF, permitting you to invoke CA JCLCheck by pressing a PF key while working within the ISPF editor.

- **Application Support**

CA JCLCheck provides interfaces to several popular applications. A complete list of supported applications is shown in the following section.

- **Extensive Error Detection**

CA JCLCheck recognizes more than 100 error situations, giving you the opportunity to greatly decrease the chances for ABENDs and reruns. CA JCLCheck flags and reports all JCL errors that can cause execution problems before step initiation. It also detects job ABEND errors and operating system errors.

- **Reports**

The CA JCLCheck comprehensive reporting system documents submitted JCL for a single job, complete input production JCL, data set and program cross-references, and error message listings. This reduces production problems by having factual, easily-obtainable JCL listings prior to job submission, and allows the systems operator the chance to correct error situations before jobs run.

- **CA Roscoe Options**

You can invoke CA JCLCheck as a service under the control of a number of CA Roscoe features. You can invoke it as a monitor, as a called ETSO program, by a CA Technologies supplied RPF, or even as an ISPF dialog under the ETSO-ISPF CA Roscoe interface.

- **Security**

The CA JCLCheck pre-validation of the security environment helps verify that the JCL has correct access to the appropriate resources. This helps to reduce the incidence of system 913 ABENDs and job failures because of insufficient access authority to DATA SET, DASDVOL, PROGRAM, STORCLAS, and MGMTCLAS resources.

- **DF/SMS**

If DF/SMS is active, CA JCLCheck interfaces with it to obtain the actual data, storage, and management classes that are assigned by the DF/SMS ACS routines. This information is reported and helps to verify that when a job runs, the new data sets are assigned to the correct storage class. CA JCLCheck also performs override checking of data, management, and storage class parameters set up by the user and defined by SMS.

- **Job Control Standards**

The standards enforcement option of CA JCLCheck allows you to maintain and view Job Control Standards (JCS). JCS panels allow you to create, edit, copy, rename, and delete JCS for your site by defining the actions that CA JCLCheck performs according to the criteria that the site's Standards Administrator specifies.

- **REXX Interface**

The REXX interface allows you to define site standards. You can use this interface with, or independent of Job Control Standards. The Standards Administrator can code your site's standards rules using flexible, sophisticated "AND/OR" logic combinations within a REXX EXEC.

- **DB2 Interface**

The DB2 interface provides validation for Existence and Cross-Reference reports on DB2 plans, table names, and subsystems.

- **JCLNeat Component**

JCLNeat reformats z/OS JCL to user-designed specifications. This component provides a methodology to standardize JCL appearance without affecting the efficiency and creativity of the original coder.

- Standardization of JCL saves valuable time and reduces errors in JCL.
- JCLNeat produces its own set of reports.
- JCLNeat works with the REXX programming language to allow tailoring of JCL to accommodate site standards.
- JCLNeat contains a global string replacement facility which allows literal string manipulation of JCL statements.

- **ChekPlex**

The ChekPlex feature of CA JCLCheck enables you to issue a JCL validation request between systems that are connected by XCF or TCP/IP. ChekPlex uses the General Transaction Server common component to route JCL from a local system to a remote system for validation. The results of the validation request are automatically transferred back to the original local system. You can view the results in ISPF edit or in a batch job.

**Note:** For more information, see The ChekPlex Feature in the *Programming Guide*.

## Application Support

CA JCLCheck provides an interface to numerous CA Technologies products and other vendor products. Current support includes the following products:

### **CA Roscoe**

CA Roscoe is an online program development system.

To validate JCL in the Active Work Space (AWS) or using the JCKRPF and JCKNRPF RPFs, use CA JCLCheck from CA Roscoe.

### **CA Librarian**

CA Librarian is a change control and source management system.

To read input from all or part of a CA Librarian library, use CA JCLCheck with CA Librarian. You can also perform automatic procedural checking using the AUTOPROC option.

### **CA Panexec**

CA Panexec is a load library management system.

To locate programs in CA Panexec using the PECNTL file, use CA JCLCheck with CA Panexec.

### **CA Panvalet**

CA Panvalet is a source library maintenance product.

To read input from all or part of a CA Panvalet library and perform automatic-procedure checking using the AUTOPROC option, use CA JCLCheck with CA Panvalet.

### **CA ASM2**

CA ASM2 is a DASD storage management system.

To obtain information for archived data sets, use CA JCLCheck with CA ASM2.

### **CA TLMS**

CA TLMS is a tape library management product.

To retrieve information from the Volume Master File (VMF) and to validate how a job is using the tapes, use CA JCLCheck with CA TLMS.

### **CA 1**

CA 1 is a product that controls a tape library.

To retrieve information from the tape management catalog (TMC) and to validate how a job is using the tapes, use CA JCLCheck with CA 1.

## **DB2**

DB2 is the IBM database product.

Use CA JCLCheck to validate and report on DB2 plans, table names, and subsystems.

## **CA ACF2**

The CA ACF2 product is a data security support product.

To pre-validate JCL by verifying numerous resource checks, use CA JCLCheck with CA ACF2.

## **CA Top Secret**

The CA Top Secret product provides security support.

To call CA Top Secret to pre-validate the security environment, use CA JCLCheck with CA Top Secret.

## **IAM**

Innovation Access Method (IAM) is a product of the Innovation Data Processing Company that provides an alternative access method to VSAM.

To recognize (dynamically) data sets defined as using the Innovation Access Method, use CA JCLCheck.

## **IMS**

IMS (Information Management System) is a product of IBM that provides access to IMS-database information.

## **SAF Compatible Security Products**

SAF (System Authorization Facility) are products, for example, IBM RACF.

To pre-validate the security environment, use CA JCLCheck with any SAF compatible product.

## **SECURE**

A data security product that uses standard IBM password-protect bits in the Data Set Control Block.

To read from "read" password protected data sets in this environment, use CA JCLCheck with SECURE.

## **CA APCDOC**

CA APCDOC is a comprehensive online documentation product, with complete batch support.

To update the CA APCDOC Cross-Reference database, use CA JCLCheck with CA APCDOC.

**CA Dispatch**

CA Dispatch is a report distribution management system that controls the report production and distribution. CA Dispatch controls from the moment the report is planned to the time the printed report is distributed in the end-user mailbox.

Use CA JCLCheck to report on the SYSOUT data sets CA Dispatch controls.

**CA Easyproclib**

CA Easyproclib is a product allowing you to specify private PROCLIBs in the JCL.

To check for the explicit PROCLIB DD statement to access procedure libraries that the JCL calls, use CA JCLCheck with CA Easyproclib.

**CA Scheduler**

CA Scheduler is a data center scheduling system. CA Scheduler is designed to automate production-control and oversee operations at z/OS sites.

To have CA JCLCheck perform the JCL validation on job forecasting commands that are presented to CA Scheduler, use the CA JCLCheck/CA Scheduler Interface.

**CA WA Restart Option for z/OS Schedulers**

CA WA Restart Option for z/OS Schedulers is an automated rerun and tracking system.

To validate the rerun of a job, use CA JCLCheck with CA WA Restart Option for z/OS Schedulers.

**CA WA CA 7 Edition**

CA WA CA 7 Edition is a data center scheduling system.

To analyze and simulate processing of CA WA CA 7 Edition scheduled override control statements (#JI, #JO, #JEND) in a job stream, use CA JCLCheck with CA WA CA 7 Edition.

**VIOPLUS**

VIOPLUS is an enhanced VSAM product.

To check for the VIOCTL DD statements (to validate the first qualifier of the data set name as either ACTIVE or SUPPRESS), use CA JCLCheck with VIOPLUS.

**XPAF**

XPAF is the Xerox Print Access Facility.

To perform syntax checking on XPAF output statements, use CA JCLCheck.

CA JCLCheck also supports the IBM operating system products: JES2, JES3, z/OS, and VS1.

## JCLNeat Component

The JCLNeat component of CA JCLCheck standardizes JCL formatting according to your installation's specifications. This standardization facilitates the work of both programmers and support personnel.

The JCLNeat component supports the following features:

### Statement Reformatting

You can provide a model of how JCLNeat should reformat the input JCL by using runtime options. In addition to column alignment information, JCL keyword parameters and their associated data are reordered in a user-specified sequence.

**Note:** CA Technologies recommends that the JCL is error-free before JCLNeat reformats it.

### Input/Output Sources

You can input JCL from numerous sources, for example: a sequential input stream, a partitioned data set with many members, or a CA Librarian or CA Panvalet source management library.

### JCL Comments

You can do the following tasks with the JCL comments: box in stand-alone comments, delete type-based comments, insert blank comments, specify a new fill character, or rearrange the comment starting column.

### Symbolic Parameter Replacement

When processing a PROCLIB member or an inline procedure, JCLNeat optionally replaces the symbolics (that are within the procedure) with the defaults specified on the PROC statement. This feature lets you restructure procedure libraries by eliminating unused symbolics.

### Global Editing

You can optionally make global editing changes to the data portion of the original JCL by including these requests in the CHANGES DD data set of the keywords. Specify the requests as from-character-string and to-character-string.

### Global String Replacement

You can optionally make global string changes to the data portion for any of the original data of the JCL by including these requests in the GSRCNTL DD data set. Specify the actions such as Add, Delete, Replace, and Comment.

### REXX

You can optionally change JCL parameters and sub-parameters by coding a REXX EXEC to modify various REXX JCL variables.

**Hardcopy Reporting**

You can request numerous hardcopy reports from JCLNeat. CA JCLCheck always prints the features and options that are in effect during an execution of JCLNeat. You can optionally request listings of the original JCL, the reformatted JCL, and from and to JCL statements created by each change request.

**User Specifications**

JCLNeat provides a great deal of flexibility by allowing you to tailor its capabilities to the installation's documentation and standardization needs. The component is distributed with default options and keyword ordering tables. You can change these defaults after JCLNeat is installed or at execution time.

**Important!** JCLNeat is not intended for JCL validation. Although it detects certain incorrect or illogical JCL coding conditions, JCLNeat expects the JCL to be clean at input. When the JCL is not clean at input, unpredictable results can occur. Validate the JCL with CA JCLCheck before reformatting with JCLNeat.



# Chapter 2: Using CA JCLCheck

---

For CA JCLCheck to check your JCL successfully, you must complete the following:

- Install the product
- Set up the CA JCLCheck execution JCL or environment
- Set up the CA JCLCheck runtime options
- Invoke CA JCLCheck against some JCL using one of the following execution methods: batch, online, or EDCHEK.

**Note:** You can perform these functions using various methods.

Running CA JCLCheck against a JCL stream produces either messages or reports about the condition of your JCL, and job submission, if your execution method and options allow.

To install the product, follow the instructions in the *Installation Guide*. At installation time, a set of default options is established. These options define checking capabilities for things such as the security environment, the source of the JCL, alternate processing activities, or the report types you want CA JCLCheck to issue after it checks your JCL.

Runtime options are an integral part of CA JCLCheck and are covered in detail in [CA JCLCheck Runtime Options](#) (see page 107). You can change these options using the following methods:

- ISPF online panels
- Supplementary OPTIONS file (OPTS=), if one is available
- PARM= field in the CA JCLCheck execution JCL EXEC statement

When running CA JCLCheck in batch, you can set up the options you want to use in a supplementary options file (OPTS=) or you can specify the options you want to use in the OPTS DD statement.

Once you install CA JCLCheck and set up your execution environment and your options, you can scan JCL by invoking CA JCLCheck in one of the following ways:

- From a batch job
- From the ISPF online panels
- Using the EDCHEK Facility (online checking)
- Using the SUBCHEK Facility (online submit to batch)
- From a CLIST
- From CA Roscoe

These are only some of the methods covered in this chapter. Other methods you can use to invoke CA JCLCheck are covered in the *Programming Guide*.

This chapter begins with the type of input CA JCLCheck expects to check, an overview of the execution JCL, and the completion and ABEND codes that are a result of the output. After that, the ISPF panels are explained, and a description of how to use EDCHEK, SUBCHEK, a CLIST, and CA Roscoe to invoke CA JCLCheck.

The final section of the chapter explains how graphic flow diagram reports track your JCL. This section also informs you how to set up printer support using the SYSGRAPH DD statement, and where you can display CA JCLCheck sample JCL for setting up your own flow diagrams.

## Input

To run CA JCLCheck, you must enter a job stream containing one or more jobs, exactly as you would submit the job streams to z/OS. The job stream should begin with a JOB statement to identify the first job and provide headings for reports. If a JOB statement is not present, CA JCLCheck substitutes a "dummy" JOB statement.

CA JCLCheck does not require separators between jobs, the JOB statement itself serves as a delimiter. CA JCLCheck considers multiple jobs in the input as completely independent, unless you specify the SXREF or PDSRELATE runtime options. Using either of these options causes CA JCLCheck to remember catalog and VTOC updates between jobs.

CA JCLCheck can also use all or part of a partitioned data set (PDS), cataloged procedure library, CA Panvalet library, or CA Librarian library as its input. In this case, it considers each member a separate job, whether it contains a JOB statement or not. CA JCLCheck determines the type of input by using the PDS, PANVALET, and LIBRARIAN options. In addition, the PDSRELATE option causes CA JCLCheck to treat library members as a series of related jobs.

For more information about input functions, see [CA JCLCheck Runtime Options](#) (see page 107).

## Invoking with the Batch Facility

The following topics describe invocation using the batch facility.

## Required Execution JCL

To invoke CA JCLCheck, member AZ1JCHK in CAZ2JCL is supplied. This member executes the JCLCHECK procedure that is located in CAZ2PROC or CAZ2OPTN CAZ1JCHK. AZ1JCHK also submits the JCL you want CA JCLCheck to analyze. Enter the JCL you want CA JCLCheck to analyze between the DLM=\$\$ and \$\$ as shown in the following example:

```
//AZ1JCHK JOB ...
//Z1EXJCHK EXEC CAZ1JCHK,
//          DSNQUAL='CAI',
//          DSNLOAD='CAILOAD',
//          OPTION='O(OPTS)'
//OPTS DD *
/*
//SYSIN DD DATA,DLM=$$
        (JCL to analyze)
$$
```

**Note:** Within this member, you see comments explaining how to process partitioned data sets and library files.

CAZ1JCHK looks like the following example (without the commented statements):

```
//CAZ1JCHK PROC DSNQUAL='CAI',
//          DSNLOAD='CAILOAD',
//          GOUT='*',
//          OPTION=''
//JCLCHECK EXEC PGM=JCLCHECK, PARM='&OPTION', REGION=0K
//STEPLIB DD DSN=&DSNQUAL., &DSNLOAD., JCLCHECK LOAD LIBRARY
//          DISP=SHR
//SYSTEM DD SYSOUT=*                TERMINAL ERROR REPORT
//SYSPRINT DD SYSOUT=*              REPORTS 1-9 AND 11
//SYSPRT DD SYSOUT=*                REXX OUTPUT
//SYSGRAPH DD SYSOUT=&GOUT., DCB=RECFM=FBA GRAPH REPORT
//SYSMDUMP DD SYSOUT=*              DIAGNOSTICS DATA
//SYSPROC DD DSN=SYS1.PROCLIB, DISP=SHR <=====*//
//SYSRDR DD SYSOUT=(A, INTRDR), DCB=BUFNO=1 <=====*//
```

You can edit this member based on the following DD statement descriptions. In addition, you must customize the CA JCLCheck procedure JCL for your site.

**Note:** If you use the AUTOPROC option, it is not necessary to maintain PROCLIB allocation information in the CA JCLCheck cataloged procedures, CLIST, or SUBCHEK/EDCHEK control files. For more information, see [CA JCLCheck Runtime Options](#) (see page 107).

To invoke CA JCLCheck, use the following EXEC statements:

**EXEC**

Statement that calls the JCLCHECK program and supplies all the PARM requirements.

For a complete description of all the PARM options available, see [CA JCLCheck Runtime Options](#) (see page 107).

Required.

**STEPLIB**

Statement that points to the JCLCHECK load library to call the JCLCHECK procedure.

Required if it is not set up in the linklist.

Use the following list of required and optional DD statements when you invoke CA JCLCheck:

**SYSPRINT**

File that receives the CA JCLCheck reports.

The files logical record length is 133 and it can be blocked. To change the ddname of this file, use the LIST option.

Required unless you specify PARM=NOLIST, or the NOLIST option in the options file.

**SYSPROC**

Statement that points to the procedure library that CA JCLCheck uses to resolve references to cataloged procedures.

If you have more than one procedure library, you can concatenate other data sets on DD statements to this one. To change the ddname of this file, use the PROC option. Specify the same procedure libraries for this file as you have specified in your JES procedure. Generally, the ddname is PROC00 for JES2 and IATPLBST for JES3.

Required unless you are using AUTOPROC.

**CAIVMFI**

CA TLMS Volume Master File (VMF).

Required if CA TLMS support is installed and you select the CA TLMS support option.

**PROCNn**

Provides CA JCLCheck support for the PROCLIB parameter of the JES2 /\*JOBPARM control statement and JCLLIB JCL statement.

Required unless you are using AUTOPROC.

You should add the PROCLIB allocation statements from the JES2 procedure to the JCL which executes CA JCLCheck.

Do not use with AUTOPROC.

**IATPLBnn**

Provides CA JCLCheck support for the PROC= parameter of the JES3 `//*MAIN` control statement.

If you want support, add the alternate PROCLIB allocation statements from your JES3 procedure to the JCL which executes CA JCLCheck.

Optional.

**JCHKFREE**

Marks the beginning of the sequence of DD statements that describe the data sets that you want CA JCLCheck to dequeue under CA Roscoe.

Optional unless you specify PARM=DEQUE.

**JENDFREE**

Specifies the end of the sequence of DD statements that you want CA JCLCheck to free under CA Roscoe.

Optional unless you specify PARM=DEQUE.

**SYSGRAPH**

Receives the Flow Diagram Report (Report #10).

The default ddname is SYSGRAPH. You can specify another ddname by using the [GRAPH](#) (see page 157) option. The default and minimum possible value for LRECL is 133.

Optional unless you specify PARM=GRAPH.

**COBLIB**

Describes a load library that can contain a CA JCLCheck COBOL exit routine; use with the COBEXIT option.

When you include this statement, CA JCLCheck loads the COBOL exit program named in the COBEXIT option from this library. Use this statement to facilitate development and testing of COBOL exit programs.

**Note:** For information about the COBOL exits, see the *Programming Guide*.

Optional.

**SYSDICT**

Accesses the CA APCDOC Cross-Reference database.

The SYSDICT DD statement specifies the cluster data set name for this database.

Required if you install the CA APCDOC interface at Version 1.0. For Version 1.1 and above, use the DBHLQ runtime option to dynamically allocate the database.

### **SYSDICT1**

Accesses the CA APCDOC Cross-Reference database.

The SYSDICT1 DD statement specifies the path name for the Cross-Reference database.

Required if you install the CA APCDOC interface at Version 1.0. For Version 1.1 and above, use the DBHLQ runtime option to dynamically allocate the database.

### **SYSTEM**

Contains the terminal report file.

The logical record length of this file is 80 and the file can be blocked. You can change the ddname of this file with the Term option discussed in [CA JCLCheck Runtime Options](#) (see page 107).

Optional unless you specify PARM=TERM.

### **SYSRDR**

Defines the destination for error-free job streams.

JES2 automatically processes these jobs if the statement is coded as shown on the previous page. CA JCLCheck writes the JCL for jobs containing errors to the point of the error, then terminates with a JES2 /\*PURGE (or HASP /\*DEL) control statement.

Optional unless you specify PARM=SUBMIT.

### **SYSDDxxx**

Performs for the dynamic allocation of data sets required by the CTLSCAN, EASYPROC, or PXREF options in non-z/OS systems.

When using any of these options, you should add a SYSDDxxx DD statement for every volume that contains a control statement data set (for CTLSCAN), a private procedure library (for EASYPROC), or a program library (PXREF). In practice, it is probably best to include a volume allocation DD statement for every resident or reserved volume in the installation, as part of a standard CA JCLCheck procedure. You can change the ddname prefix on these DD statements with the PREFIX option.

Optional in non-z/OS and not used in z/OS.

**SYSIN**

Points to the JCL stream that you want CA JCLCheck to analyze.

CA JCLCheck also requires the DLM parameter if an input stream data set contains any statements beginning with "/"\* (such as JES2 control statements). This data set is one of the following: a card image file, a partitioned data set (with the PDS option), a CA Panvalet library (with the Panvalet option), or a CA Librarian library (with the Librarian option).

You can change the ddname of this file with the Input option discussed in [CA JCLCheck Runtime Options](#) (see page 107).

**Note:** SYSIN requires an LRECL of 80.

Required.

**STDRULE**

Accesses the CA JCLCheck Standards Rules database.

This DD statement specifies the data set name for the database.

Optional.

**SYSTSPRT**

Receives the REXX error messages and output from REXX commands, use with the STDREXX option.

Its logical record length is 133 and it can be blocked. The ddname may be different if your site has changed the REXX installation defaults.

Optional.

**SYSEXEC**

Points to the REXX EXEC library used with the STDREXX option.

Optional.

**SYSMDUMP**

Provides a SYSOUT data set to contain any diagnostic information due to abnormal termination or use of the CA JCLCheck DEBUG option.

Optional.

## Cataloged Procedure

CA JCLCheck supplies the following cataloged procedure. This procedure performs the same function as the previous execution JCL using a library procedure to contain the JCL that CA JCLCheck checks. This procedure contains the key JCL with the exception of the SYSIN statement. Once you install CA JCLCheck at your site, use the cataloged procedure as follows:

```
//          EXEC CAZ1JCHK, OPTION='options'  
//SYSIN DD DATA,DLM='ZZ'  
        (JCL to analyze)  
ZZ  
  
or  
  
//          EXEC CAZ1JCHK  OPTION='options'  
//SYSIN DD DSN=library, DISP=SHR
```

## Using the ISPF Online Panels

CA distributes an ISPF Dialog Manager interface with CA JCLCheck. This dialog can run CA JCLCheck in the foreground, or it can submit a job to execute CA JCLCheck in the background. If you execute CA JCLCheck in the foreground, you can:

- Request that error-free jobs be submitted for execution
- Review the CA JCLCheck output using BROWSE
- Generate hardcopy output using the PRINTOFF command

Depending on how the ISPF interface is installed at your site, CA JCLCheck can appear as a part of your ISPF Option Menu, or you can call it from a command line by entering **TSO %JCKSPF** and pressing Enter.

CA JCLCheck has a menu facility and a help facility. You use the menus described in this chapter to: select processing and runtime options, to invoke CA JCLCheck, to define the CA JCLCheck database, define PROCLIBs, schedule jobs, select JCL members, and print hardcopy output. After you define CA JCLCheck options, save them in the user profile for subsequent requests.

CA JCLCheck uses the PF key settings defined by your ISPF parameters. If you need help using any panel, press the help key to display the CA JCLCheck help tutorial for that panel. If you display the help tutorial main menu, you can select topic-specific information.

To display the CA JCLCheck ISPF panels from the ISPF Primary Option Menu, select the CA JCLCheck option. This displays the CA JCLCheck/SPF Menu (JCK0200). CA JCLCheck saves the default options to your user profile during installation; however, you can enter the Options panels at any time to customize the CA JCLCheck options you want saved to your user profile.

**Note:** You must set up your JOB statement using the Specification of JOB CARD panel if you want to invoke CA JCLCheck in batch mode.

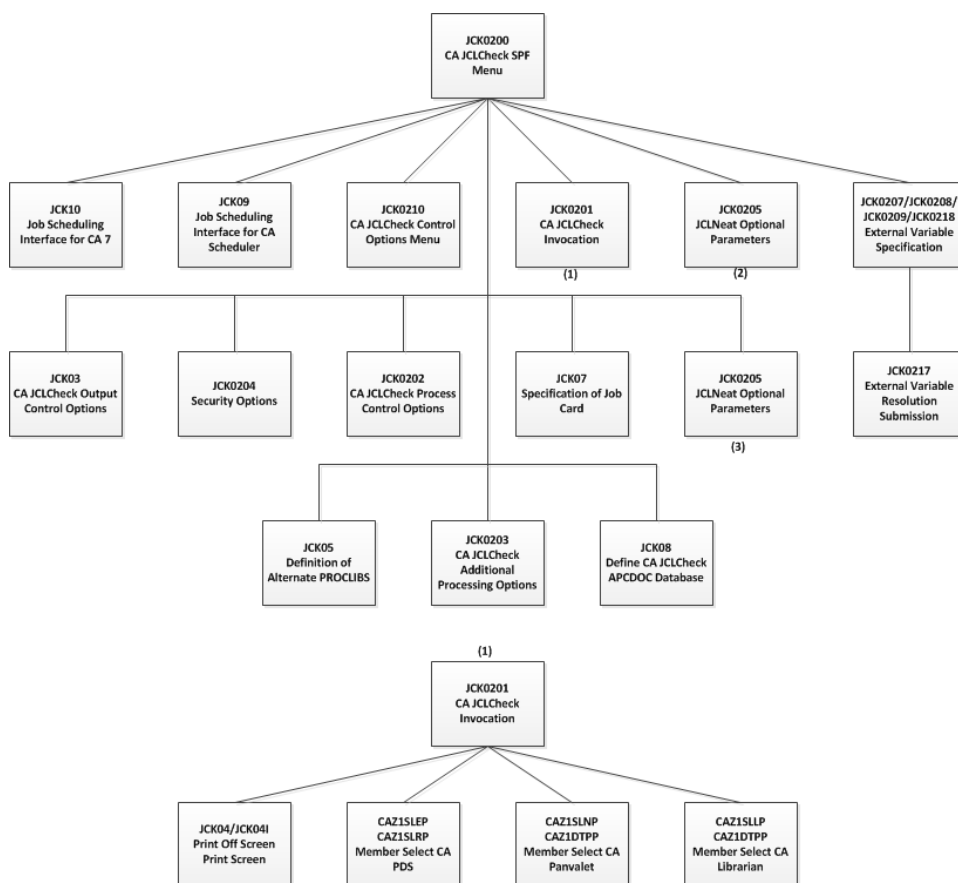
You can change your options as needed for individual runs.

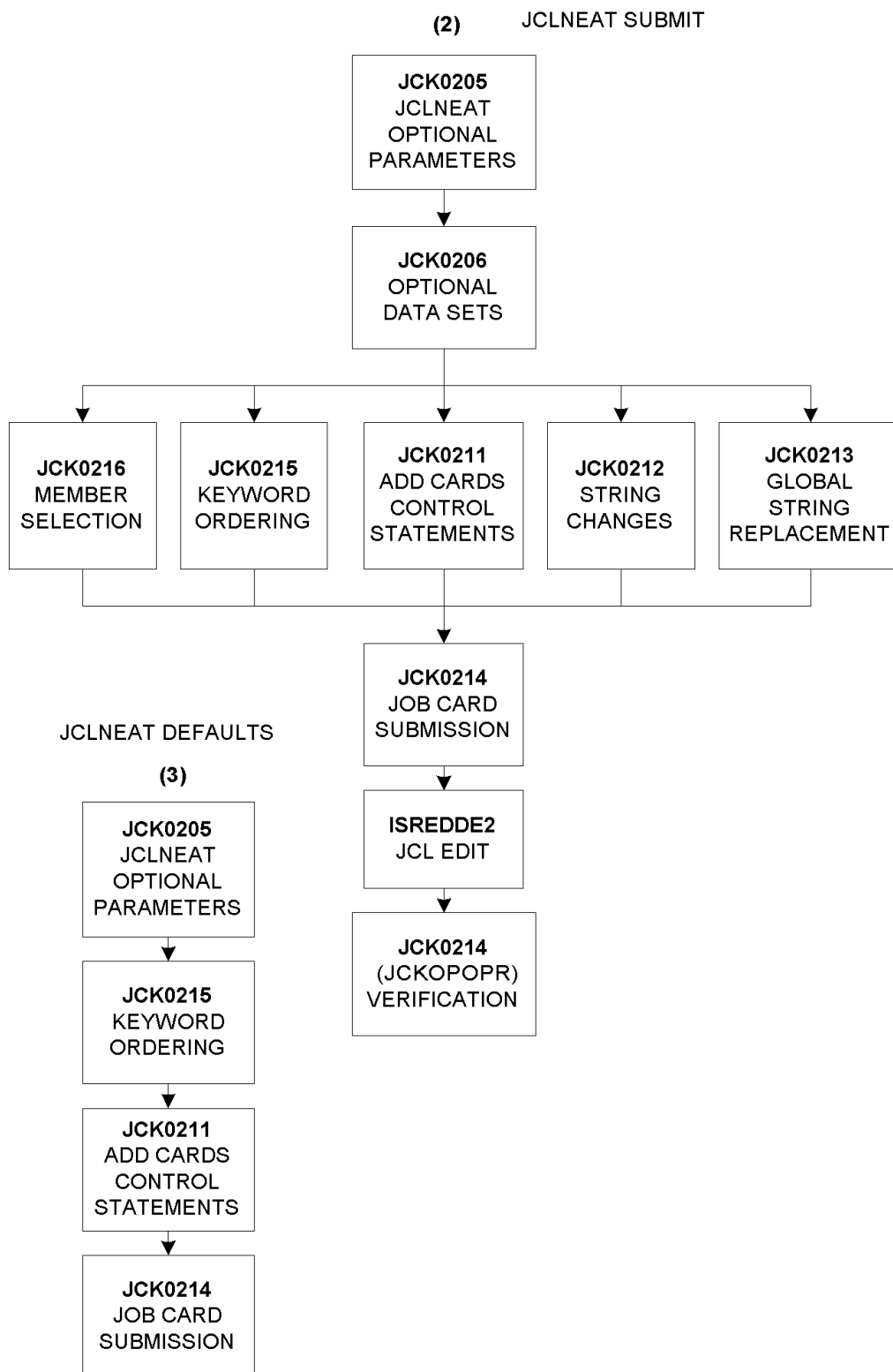
To exit these panels or return to a previous menu panel, press End.

## Flowchart of JCKSPF Panels for CA JCLCheck Dialog

This flowchart shows the CA JCLCheck ISPF online panels that you use to set up processing and invocation options, and run CA JCLCheck. Each panel contains the panel ID and name. The first panel you display when logging on to CA JCLCheck from the Primary Option menu is the CA JCLCheck/SPF Menu (JCK0200). From this panel, you can display any CA JCLCheck option or processing environment panel.

To display the panel ID, enter PANELID on the TSO command line.





## CA JCLCheck/SPF Menu (JCK0200)

Use panel JCK0200 to display the CA JCLCheck: control options panels, various types of invocation panels, and panels that allow you to establish interaction with CA WA CA 7 Edition or CA Scheduler when running CA JCLCheck.

**Note:** Upon entering JCK0200, CA JCLCheck displays message CAY6622I. This message informs you that CA JCLCheck is set up to use default options and that you can change these options using Option 0 (Option Specifications).

Do one of the following bullets to display JCK0200:

- Select CA JCLCheck from the ISPF Primary Option menu.
- Enter **TSO %JCKSPF** on the ISPF command line, if your installation allows.

To exit this panel, press End.

```
JCK0200----- CA JCLCHECK/SPF MENU -----
OPTION ==>

Date: yyyy/mm/dd   User: USERID1   Level: xx.x

  0 - Option Specifications
  1 - CA JCLCHECK - Foreground Invocation
  2 - CA JCLCHECK - Foreground Invocation / Submit
  3 - CA JCLCHECK - Batch Submit
  4 - CA 7 Interface Invocation
  5 - CA Scheduler Interface Invocation
  6 - JCLNeat - Batch Submit
  7 - RESOLVE - External Variable Resolution
  X - Exit
```

Enter the number of the option for the panels you want to display in the OPTION field, and press Enter.

### 0 - Option Specifications

Displays the CA JCLCheck Control Options Menu (JCK0210). This menu gives you a choice of four different types of control option panels that you use to view or update various types of runtime options and JOB statement specification parameters. You also select Option Specifications to create and maintain JCLNeat defaults. These defaults are in effect during the submit process unless changed either in the maintenance of the defaults or during the submit process.

### **1 - CA JCLCheck - Foreground Invocation**

Displays the CA JCLCheck Foreground Invocation panel (JCK0201). This panel enables you to modify your default options and submit JCL to CA JCLCheck checking in the foreground.

### **2 - CA JCLCheck - Foreground Invocation/Submit**

Displays the CA JCLCheck Foreground/Submit Invocation panel (JCK0201). This panel enables you to modify your default options and submit JCL to CA JCLCheck checking in the foreground.

If the JCL is error free, CA JCLCheck submits the JCL for processing.

### **3 - CA JCLCheck - Batch Submit**

Displays the CA JCLCheck Batch Invocation panel (JCK0201). This panel submits the submission JCL to CA JCLCheck checking in the background.

If the submission JCL is error free, CA JCLCheck submits the JCL for processing.

### **4 - CA 7 Interface Invocation**

Displays the Job Scheduling Interface for CA 7 panel (JCK10). This panel enables you to enter applicable forecast information.

Press Enter to submit execution of the CA 7 BTI procedure.

### **5 - CA Scheduler Interface Invocation**

Displays the Job Scheduling Interface for the CA Scheduler panel (JCK09). This panel enables you to enter applicable forecast information.

Press Enter to submit execution of the CA Scheduler CAJUTIL0 procedure.

### **6 - JCLNeat - Batch Submit**

Displays the JCLNeat Optional Parameters panel (JCK0205). This panel enables the system user to format and submit a job to perform the JCLNeat process on selected JCL members.

### **7 - RESOLVE - External Variable Resolution**

Displays the External Variable Specification panel (JCK0207, JCK0208, JCK0209, or JCK0218) for the product selected in the RESOLVE option on the Additional Processing Options panel. This panel enables you to specify product-specific parameters for external variable resolution. If no RESOLVE option is set, this selection is unavailable.

## Control Options Menu (JCK0210)

Use this panel to view or change your processing control options, output control options, security options, or JOB statement parameters.

To display this panel, select Option 0 (Option Specifications) on the CA JCLCheck/SPF Menu (JCK0200).

To exit this panel and return to the previous panel, press End.

```
----- CA JCLCheck CONTROL OPTIONS MENU -----JCK0210
OPTION ==>
      Date:           User:           Time:
                1 - Process Control Options
                2 - Output Control Options
                3 - Security Options
                4 - JCLNeat Options
                5 - Job Card Specifications
                X - Exit
```

Enter the desired option in the OPTION field, and press ENTER.

### 1 - Process Control Options

Displays the CA JCLCheck Process Control Options panel (JCK0202). Use this panel to view or change process control options.

Optionally, you can go to the Additional Processing Options panel (JCK0203) from this panel.

### 2 - Output Control Options

Displays the CA JCLCheck Output Control Options panel (JCK03). This allows you to specify options that control the output produced by CA JCLCheck.

### 3 - Security Options

Displays the CA JCLCheck Security Options panel (JCK0204). This allows you to specify options that request support for pre-validation of the security environment.

### 4 - JCLNeat Options

Displays the JCLNeat Optional Parameters panel (JCK0205). This is used to support the creation and maintenance of the defaults that will be in effect during the submit mode process, unless temporarily overridden while in the submit mode process.

## 5 - Job Card Specifications

Displays the CA JCLCheck Specification of Job Card panel (JCK07). This allows you to specify the JOB statement parameters for batch submit purposes.

## Invocation Panel (JCK0201)

This panel displays the option you selected on the CA JCLCheck/SPF Menu panel (JCK0200). The Options panels contain the default options that are set up for you to use with CA JCLCheck. You can use the Options panels to change the default options and have CA JCLCheck save them in your ISPF user profile. Use this panel to invoke CA JCLCheck with the criteria you enter in the following fields. The information in these fields determines the JCL that CA JCLCheck analyzes, specifies control options for individual runs of CA JCLCheck, and specifies whether to produce and print messages and reports.

To display this panel, do one of the following:

- Enter Option 1 (CA JCLCheck - Foreground Invocation) in the OPTION field on the CA JCLCheck/SPF Menu panel (JCK0200).
- Enter Option 2 (CA JCLCheck - Foreground Invocation / Submit) in the OPTION field on the CA JCLCheck/SPF Menu panel (JCK0200).
- Enter Option 3 (CA JCLCheck Batch Submit) in the OPTION field on the CA JCLCheck/SPF Menu panel (JCK0200).

To exit this panel and return to the previous panel, press END.

```

----- CA JCLCheck/INVOCATION -----JCK0201
COMMAND ==>

Option Selected: FOREGROUND

Date: mm/dd/yyyy          User: USERA01          Time: 16:28

Data Set Name ==> 'CAI.APCMTL.J670.CHQAJCL(LINDA1)'
Library Type ==> PDS          (SEQ, PDS, PAN, LIB)
Member ==>                  (Batch Submit -- Use * for all members)
Volume Serial ==>           (If Not CATALOGED)
CPU-ID ==>                  (Multi-CPU Only, blank: This CPU)
SYSTEM ==> N                (Y/N/P, Error Statement and Messages only)
SYSPRINT ==> Y              (Y/N/P, Reports 1-9 according to options )

Specify any additional options (FOR THIS RUN ONLY) below
OPTIONS ==> CC(5) CT      SP(RPT 65 NOALL) V
OPTIONS ==> J F XREF PXR(RPT) SX RP ER( END)
OPTIONS ==> SYN RUNT
OPTIONS ==> SEC(DASD,STOR,MGMT,PROGRAM(DEFINE))
OPTIONS ==>
SAVED ==>
SAVED ==>

```

Enter the field values and options you want to take effect when you invoke CA JCLCheck.

This panel contains the following fields:

**Data Set Name**

Defines the data set containing the JCL that you want CA JCLCheck to analyze.

**Library Type**

Sequential file (SEQ) or the type of library: PDS, CA Panvalet, or CA Librarian.

**Member**

Member name associated with the data set.

If you leave this field blank and press Enter, you are transferred to a Member Select panel. If you specify an asterisk (\*) in this field, CA JCLCheck processes the full PDS in batch (Option 3).

**Volume Serial**

Volume serial number associated with the data set.

**CPU-ID**

ID of the CPU on which the CA JCLCheck background job executes.

This value is only valid with a batch submit (Option 3).

**SYSTEM / SYSPRINT**

Options specifying whether CA JCLCheck should produce messages (SYSTEM) and selected reports (SYSPRINT) for the current run.

**Y**

Generates the associated reports and automatically invokes the browse feature to display the report at your terminal.

**N**

Suppresses generation of the associated reports.

**P**

(Print) Generates the associated reports and invokes the browse feature.

After you browse the reports online, press Enter to display the PRINTOFF panel (JCK04). From here you can alter options and generate hardcopy output.

**Specify any additional options (FOR THIS RUN ONLY) below**

Current CA JCLCheck profile options are always shown using their minimum keyword name abbreviations.

This area has two types of input lines:

- You can change options on the lines labeled OPTIONS, but the changes are only in effect for this run. The dialog regenerates these options from the user profile each time you display this panel.
- CA JCLCheck saves options on the lines labeled SAVE in the user profile. The dialog regenerates these options from the user profile each time you display this panel and uses them for every job you run until you remove them.

For more information about profile options, see [CA JCLCheck Runtime Options](#) (see page 107).

## Process Control Options Panel (JCK0202)

Use this panel to specify options to control how CA JCLCheck processes the input JCL file. The dialog checks the syntax of all the options that you select before accepting them.

The fields contain the default option choices that are set up during installation when you enter this panel for the first time. When you change the options, specify a Y in the "Update PROFILE with these options" field. The fields contain the new options.

**Note:** You can also use this panel to display the Additional Processing Options panel.

To display this panel, enter Option 1 (Process Control Options) in the OPTION field on the CA JCLCheck Control Options Menu panel (JCK0210).

To exit this panel and return to the previous panel, press End.

```

----- CA JCLCheck PROCESS CONTROL OPTIONS -----
COMMAND ==>
Date: yyyy/mm/dd           User: USERID1           Time: hh:mm

        Go to ADDITIONAL PROCESSING OPTIONS PANEL  => N

CCLIST - Number of program control cards to be listed      (0-999) => 100
COND   - Simulate execution according to condition codes    => N
CTLSCAN - Scan utility control statements                   => N
EASYPROC - CA EASYPROCLIB support                          => N
MSS    - Stage VTOC of MSS volumes if necessary            => N
PXREF  - Check program and PDS member existence            => N
SEV    - Minimum severity of errors to be listed           (0-16) => 0
SPACE  - Check blksize efficiency for new DASD files        => N
        (ALL) - Check blksize efficiency for existing DASD files too => N
        (NN) - Minimum % of track usage (give warning if less) (0-99) => 65
SXREF  - Check JCL across jobs                              => N
VSAM   - Process private catalogs (JOB CAT, STEP CAT)      => Y
CSI    - Use Catalog Search Interface for catalog lookup   => D
PROC   - Define alternate proclibs                          => N
        - Alternate proclib(s) are selected                => Y
DICT   - Define CA APCDOC XREF database                     => N

        Update PROFILE with these options                => Y
    
```

Enter the processing values that you want to take effect, and press Enter to invoke CA JCLCheck using these options.

**Note:** For complete details about each option, see [CA JCLCheck Runtime Options](#) (see page 107).

This panel contains the following fields:

**Go to ADDITIONAL PROCESSING OPTIONS PANEL**

Y displays the Additional Process Control Options panel (JCK0203); N deactivates this option.

**CCLIST**

A number from 0 to 999 that specifies the number of program control statements you want CA JCLCheck to list.

**COND**

Y activates condition code simulation and the IF/THEN/ELSE/ENDIF statement; N deactivates condition code simulation.

Specifying Y allows you to specify an alternate completion code for a specific step.

**CTLSCAN**

Y requests that CA JCLCheck read and interpret control statements for the following IBM utility programs IEHPRGM, IDCAMS, DFSORT, ADDRSSU, IEBCOPY, and IEBGENER. Y also requests CA JCLCheck read and interpret control statements for the CA Technologies program CA XCOM Data transport. N deactivates these control statements.

### **EASYPROC**

Y activates support of CA Easyproclib; N deactivates this option.

If you enter Y and CA JCLCheck encounters an EASYPROC PROCLIB DD statement before the first EXEC statement of a job, it opens and uses the specified PROCLIB in addition to the one specified on its own SYSPROC DD.

### **MSS**

Y requests that CA JCLCheck stage the VTOC of mass storage volumes when necessary to verify the existence of a data set on MSS; N ignores this option.

### **PXREF**

Y requests activation of the CA JCLCheck program and member existence checking and reporting feature; N ignores this option.

### **SEV**

A value from 0 to 16 that defines the minimum severity of errors that you want CA JCLCheck to list.

### **SPACE**

Y activates the direct-display space usage analysis, and N deactivates this option and reporting feature in the SPACE field.

Enter Y or N in the ALL field depending on whether you want CA JCLCheck to perform the space efficiency check on both new and existing data sets.

Enter a percentage from 0 to 99 in the NN field to check the space efficiency on the new data sets created by the input job stream, or the minimum percentage of track usage. The suggested values are 75 - 85 percent.

### **SXREF**

Y activates related job data set existence checking; N deactivates this option and reporting feature (checking JCL across jobs).

This option specifies that the job streams you want CA JCLCheck to analyze are related jobs that run in succession.

### **VSAM**

Y requests that CA JCLCheck support VSAM catalogs; N deactivates this option.

Specifying Y invokes the IBM utility IDCAMS when it encounters a JOBCAT or STEPCAT statement defining a VSAM catalog. If IDCAMS accepts the catalog, the utility is called in lieu of standard catalog management during the scope of the job or step catalog.

### **CSI**

Y requests that CA JCLCheck use the IBM Catalog Search Interface when doing catalog lookup and data set existence checking. N deactivates this option.

Specifying Y invokes the IBM Catalog Search Interface when searching for a data set in the system catalog. Oftentimes, there are performance benefits when utilizing this option. The CSI option is mutually exclusive and cannot be used together with the VSAM option.

### **PROC**

Y specifies an alternate ddname for the statement defining the procedure library that you want CA JCLCheck to use; N ignores this option.

PROC lets you use alternate libraries where CA JCLCheck searches for cataloged procedures. The alternate library allows CA JCLCheck to test modified procedures without the need to store them in system PROCLIBs prior. When CA JCLCheck activates these alternate PROCLIBs and you use Option 2 (CA JCLCheck - Foreground Invocation / Submit) on the JCK0200 main menu, CA JCLCheck does not submit the JCL. It displays a message indicating that nonstandard PROCLIBs were used and that actual execution may not be possible. The PROC option is mutually exclusive with the AUTOPROC option. If you specify both options, CA JCLCheck ignores the PROC option.

### **DICT**

Y defines terms to the dictionary when your site has CA APCDOC installed; N ignores this option.

### **Update PROFILE with these options**

Y updates your current JCL profile with a new profile containing the option changes entered on this panel; N ignores these changes.

## **Additional Processing Options Panel (JCK0203)**

Use this panel to specify additional options that control how CA JCLCheck processes the input JCL file. The dialog checks the syntax of all options that are selected before accepting them. From this panel, you can specify additional options that control how CA JCLCheck processes the input JCL file.

To display this panel, enter Y in the GO TO ADDITIONAL PROCESSING OPTIONS field on the CA JCLCheck Process Control Options panel (JCK0202).

```

JCK0203 ----- CA JCLCheck ADDITIONAL PROCESSING OPTIONS -----
COMMAND ==>
DATE: yyyy/mm/dd                USER: USERID1                TIME: hh:mm

STANDARD - Name of STANDARD(S) for compliance checking or blank =>
=>                                =>                                =>                                =>
=>                                =>                                =>                                =>

STDREXX - Name of REXX STANDARD for compliance check or blank =>
REXXMSG - Allow changes to message severity via REXX EXEC      => N
SYNTAX  - Do syntax checking of JCL                            => Y
RUNTIME - Do runtime checking of JCL                           => Y
REMOTE  - Do remote validation of JCL                          => N
=> B => USERID1.JCK.CNTL(REMVAL)

CHANGE  - Do pre-validation variable substitution              => N
        - Substitute JCL, control cards, or both (JCL/CTL/BOTH) => BOTH
=> B DSN=> USERID1.JCK.CNTL MEM=> CHANGE

OPROC   - Define OPROC DD name or OPROCLIB libraries          => N
CNTLOVER - Define control card override DD name or libraries  => N
RESOLVE - Resolve external variables by calling product. . . . => CA7
SYSUID  - Specify value for system parameter SYSUID          =>

Update PROFILE with these options => Y

```

Enter the options that you want to take effect when you invoke CA JCLCheck.

#### STANDARD

Standard names (from the Standards Rules database) that you want CA JCLCheck to use for compliance checking.

If you do not want to invoke compliance checking, you can specify one to eight standard names or you can leave this field blank.

#### STDREXX

Specify the REXX member name of a CA JCLCheck REXX program that is used to enforce site standards. You can use standard IBM member pattern characters to display a member list for the data sets that are allocated to your SYSEXEC DD. Use the asterisk (\*) for multiple characters and the per cent sign (%) for a single character. You can select the member that you want to use from the list. Press Enter to have the selected member name filled in as the STDREXX value.

#### REXXMSG

Y enables you to process messages and change message severity by using the REXX EXEC specified in the STDREXX panel option. When Y is omitted, the REXX EXEC is not called to process messages.

#### SYNTAX

Y activates syntax checking of the JCL; N displays options only.

#### RUNTIME

Y causes CA JCLCheck to perform runtime checking on the JCL; N deactivates runtime checking.

### REMOTE

Y activates remote validation; N deactivates remote validation. If you activate remote validation, provide the name of the REMVAL file. The REMVAL file is a sequential data set or PDS member that defines remote validation options. Immediately before the REMVAL file field on the panel, there is a single-character field that can be set to 'B' or 'E'. This field enables you to browse or edit the REMVAL file before invoking JCL validation.

### CHANGE

Y informs CA JCLCheck to substitute variables on a temporary basis before JCL validation takes place that is based on a user-defined conversion table. CA JCLCheck examines and substitutes JCL statements, inline control statements, or both depending on the JCL/CTL/BOTH sub-option. Immediately before the CHANGE file field on the panel, there is a single-character field that can be set to 'B' or 'E'. This field enables you to browse or edit the CHANGE file before invoking JCL validation.

The CHANGE file is required. This file is an 80-byte partitioned data set member or sequential file containing the user-defined conversion table of 'from' and 'to' data strings.

### OPROC

When the OPROC field is set to Y, the JCK13 panel displays. This panel is where you can specify the OPROCLIB libraries or the OPROC ddname (if the OPROC libraries were pre-allocated).

### CNTLOVER

When the CNTLOVER field is set to Y, the JCK12 panel displays. This panel is where you can specify control card override libraries or the CNTLOVDD ddname (if the control card override libraries were pre-allocated).

### RESOLVE

Specifies the product name CA JCLCheck is to call to resolve product-specific variables imbedded in the JCL. The variable resolution occurs before CA JCLCheck validates the JCL. CA JCLCheck can resolve external variables in the following scheduling products: CA WA CA 7 Edition (CA7), CA WA ESP Edition (ESP), Tivoli Workload Scheduler (TWS), and Control-M (CTM).

### SYSUID

User ID that you want CA JCLCheck to use in replacement of the system symbolic parameter, SYSUID.

### Update PROFILE with these options

Y updates the current JCL profile options with the options specified on this panel; N ignores these changes.

**Note:** For details of these options, see [CA JCLCheck Runtime Options](#) (see page 107).

## Output Control Options Panel (JCK03)

Use this panel to specify options that control the output which CA JCLCheck produces and how it is formatted.

To display this panel, select Option 2 on the CA JCLCheck Control Options Selection panel (JCK0210).

**Note:** When you press End from this panel, you return directly to the CA JCLCheck/SPF Menu (JCK0200).

```

----- JCLCHECK OUTPUT CONTROL OPTIONS ----- JCK03
COMMAND ==>

Date: yyyy/mm/dd           User: USERA01           Time: 16:26

AUDIT   - Audit Report producing totals and averages      => Y
JOB     - List input JCL                                  (RPT 1) => Y
FULLLIST - List complete merged JCL and procedures        (RPT 2) => Y
XREF    - Produce Data Set Cross-Reference                (RPT 3) => Y
PXREF   - Produce Program Cross-Reference                (RPT 4) => Y
RPTRPT  - List SYSOUT Reports                             (RPT 5) => Y
ERROR(END) - Place Error Report last                      (RPT 6) => Y
  (START) - Place Error Report first                      => N
  (INLINE) - Place Error Messages INLINE in Report 2     => N

SPACE(RPT) - List DASD Space in Dataset Reports 3,7      => Y

SX(RPT)  - Summary DSN, PGM, RPT Cross-Reference        (RPT 7,8,9) => Y
SX(ONLY) - Summary RPTS 7,8,9 only                      (SUPPRESS RPT 3,4,5) => N
PROCXREF - Symbolic Reference Report                    (RPT 11) => N
GRAPH    - Produce JOB Flowcharts                       (RPT 10) => N
          - Print the JOB Flowcharts                     => N

Update Profile With These Options => Y

```

Supply the values you want to take effect when you invoke CA JCLCheck.

### AUDIT

Y prints the Audit Report; N deactivates this option.

### JOB

Y prints the Listing of Jobstream JCL Report (Report 1); N deactivates this option.

### FULLLIST

Y prints the Listing of Merged JCL Report (Report 2); N deactivates this option.

### XREF

Y prints the Data Set Cross-Reference Report (Report 3); N deactivates this option.

### PXREF

Y prints the Program Cross-Reference Report (Report 4); N deactivates this option.

**RPTRPT**

Y prints Report Listing (Report 5); N deactivates this option.

The Report Listing is a list of all SYSOUT reports produced by the job.

**ERROR(END)**

Y prints the error messages at the end of the listing of the Error Messages Report (Report 6); N deactivates this option.

END is the default for this option.

**ERROR(START)**

Y prints the Error Messages Report (Report 6) prior to the Listing of Jobstream JCL Report (Report 1); N deactivates this option.

**ERROR(INLINE)**

Y prints error messages within the expanded JCL listing immediately following the statement flagged as an error; N deactivates this option.

**SPACE(RPT)**

Y reports space usage for all files without any track efficiency analysis; N deactivates this option.

The DASD space usage is listed on the Data Set Cross-Reference Report (Report 3) and the Summary Data Set Cross-Reference Report (Report 7).

**SX(RPT)**

Y activates the related job data set validation feature and generates summary reports (7-9) at both the individual job and system summary level; N deactivates this option.

**SX(ONLY)**

Y requests activation of the related job data set validation feature and generates the summary reports (7-9) at only the system summary level; N deactivates this option.

CA JCLCheck suppresses reports 3, 4, and 5 when you use this option.

**PROCXREF**

Y prints the symbolic Reference Report (11); N deactivates this option.

**GRAPH**

Y produces the Flow Diagram Report (Report 10) and optionally provides the ddname of the SYSOUT file; N deactivates this option.

**Print the JOB Flowcharts**

Y prints the job flowcharts; N deactivates this option.

**Update PROFILE with these options**

Y updates the current JCL profile with the options entered on this panel; N ignores these changes.

**Security Options Panel (JCK0204)**

Use this panel to specify options that control the support of the pre-validation of the security environment. This dialog checks the syntax of all the options currently selected.

To display this panel, select Option 3 on the CA JCLCheck Control Options Menu (JCK0210). Press End to return to that menu.

```

----- CA JCLCheck SECURITY OPTIONS -----JCK0204
COMMAND ==>

Date: yyyy/mm/dd           User: USERA01           Time: 16:27

SECURITY - Do Security Checking                => Y
USER      - Alternate userid for security checking or blank =>
DASDVOL  - Perform DASDVOL security checks     => Y
MGMTCLAS - Perform MGMTCLAS security checks    => Y
STORCLAS - Perform STORCLAS security checks    => Y
PROGRAM  - Perform PROGRAM level security checks => Y
          - Check if executed program is defined to security => Y
RUNCPU   - Alternate CPU ID for security validation or blank =>
RUNDATE  - Future run date and time for security validation
          - Specify date in mm/dd/yy format or blank           =>
          - Specify time in hh:mm format or blank              =>
SHRUPD   - Datasets with DISP=SHR are to be checked for UPDATE =>

Update PROFILE with these options    => Y

```

Supply the options you want to take effect when you invoke CA JCLCheck.

**SECURITY**

Y allows you to use any of the options on this panel; N deactivates these options.

**USER**

User ID you want CA JCLCheck to use to perform security checking.

If you leave this field blank, CA JCLCheck uses the user ID in the JCL or the user ID of the person using CA JCLCheck.

**DASDVOL**

Y causes CA JCLCheck to perform DASDVOL security checking; N deactivates this option.

**MGMTCLAS**

Y causes CA JCLCheck to perform MGMTCLAS security checking; N deactivates this option.

**STORCLAS**

Y causes CA JCLCheck to perform STORCLAS security checks; N deactivates this option.

**PROGRAM**

Y causes CA JCLCheck to perform program level security checking or to check if executed programs are defined to security; N deactivates this option.

**RUNCPU**

Alternate CPU ID for security validation.

Leave this field blank if you want CA JCLCheck to ignore this option.

**RUNDATE**

Run date and run time, or a run date only for CA JCLCheck to perform future security validation; N deactivates this option.

Do not specify run time only. Leave both fields blank if you want CA JCLCheck to ignore this option.

**Update PROFILE with these options**

Y updates the current JCL profile options with the options on this panel; N deactivates these changes.

## Definition of Alternate PROCLIBS Panel (JCK05)

Use this panel to specify alternate libraries for CA JCLCheck to search for cataloged procedures.

To display this panel, enter **Y** (yes) in the PROC option field on the CA JCLCheck Process Control Options panel (JCK0202).

```

----- DEFINITION OF ALTERNATE PROCLIBS -----JCK05
COMMAND==>
DATE - yyyy/mm/dd          USERID - USER01          TIME -12:30

PROCLIB1: 'systems.oidid.cicsvs.proclib'
PROCLIB2: 'systems.oidid.imsvs.proclib'
PROCLIB3:
PROCLIB4:
PROCLIB5:
PROCLIB6:
PROCLIB7:

```

Enter the library data set names in the order in which you want CA JCLCheck to use them. When you press Enter, CA JCLCheck checks for the existence of these libraries.

## Specification of JOB CARD Panel (JCK07)

Use this panel to specify the JOB CARD parameters that CA JCLCheck uses for a batch submit.

To display this panel, enter Option 5 (Job Card Specification) on the CA JCLCheck Control Options panel (JCK0210). Press End to return to that panel.

```

----- SPECIFICATION OF JOB CARD -----
COMMAND==>
Date:                      User:                      Time:

Job Statement Information: (Verify before submitting)

==> //JOBNAME JOB (4303,D000),RGS,MSGCLASS=2,CLASS=A
==> /*
==> /*
==> /*

Update PROFILE with these options =>

```

Supply the Job Statement option information and the values for the remaining fields.

This panel contains the following fields:

**Job Statement Information**

Required JOB statement information.

This information should include a complete JOB statement including quotation marks (").

**Update PROFILE with these options**

Y updates the current JCL profile with this JOB statement information. N cancels the update.

## Define JCLCheck/APCDOC Database Panel (JCK08)

Use this panel to update JCL on the CA APCDOC Cross-Reference database.

To display this panel, enter Y in the DICT option field on the CA JCLCheck Process Control Options panel (JCK0202).

```
-----DEFINE JCLCheck/APCDOC DATA BASE ----- JCK08
COMMAND ==>

DATE - yyyy/mm/dd          USER - USERID          TIME - hh:mm

DICT   - Production XREF Database Option
UPDATE - Update the XREF Database from this run      => N

SYSTEM - System Name =>

      For CA APCDOC Release 1.1 and above, DBHLQ is required:
APCDOC Database high-level qualifier =>

-----
      For CA APCDOC Release 1.0, database names are required:
APCDOC Database Name =>
APCDOC Database Path =>

      Important note:
      Enter either high-level qualifier or database names, but not both.
```

Enter the appropriate values for the following options to take effect when you invoke CA JCLCheck.

This panel contains the following fields:

**UPDATE**

Y updates the CA APCDOC Cross-Reference database after CA JCLCheck checks the JCL; N ignores the update.

**SYSTEM**

System name (up to eight characters) for CA JCLCheck to associate the JCL with a particular system (that is, a multi-CPU partition environment).

**APCDOC Database High-level Qualifier**

The high-level qualifier of the CA APCDOC PARMFILE associated with the Cross-Reference database.

Use this field when running CA APCDOC Version 1.1 and above.

**APCDOC Database Name**

Data set name of the CA APCDOC Cross-Reference database.

Use this field when running CA APCDOC Version 1.0.

**APCDOC Database Path**

Data set name of the alternate path for the CA APCDOC Cross-Reference database.

Use this field when running CA APCDOC Version 1.0.

**Member Select for PDS (CAZ1SLEP/CAZ1SLRP) Update and Browse**

Use this panel to select members from a partitioned data set (PDS) for CA JCLCheck to process.

To display this panel:

1. Enter Option 1, 2, 3, or 4 in the OPTION field on the CA JCLCheck/SPF Menu panel (JCK0200) and press Enter.
2. Enter a complete data set name and data set type of PDS in the appropriate fields on the CA JCLCheck Invocation panel (JCK0201), and press Enter. Do not enter a member name.

```

----- CA JCLCHECK MEMBER SELECT ----- ROW 1 OF 0
COMMAND ==>                                SCROLL ==> PAGE

Data Set Name: CAI.CHN6.R880200.PROD.CNTL      Type .: PDS

CMD  MEMBER      ORDER  CREATED      LAST MODIFIED  SIZE  USERID
-----
PRODINIT      1  2000/02/01  2000/02/15 12:17    40  PROD001
PRODLOAD      2  2000/02/02  2000/02/15 12:19    72  PROD001
PROD0101      3  2000/02/09  2000/03/01 09:11     7  PROD002
PRODRUN       4  2000/03/01  2000/03/01 09:45   119  PROD001
PRODBKUP      5  2000/02/01  2000/02/17 15:32   2000  PROD001
=====
ACCTVER              2000/01/23  2000/03/24 00:13    254  ABCXY01
DOCSCAN              2000/12/25  2000/12/27 08:37     55  ABCXY01
SUBALL               2000/07/12  2000/09/10 16:17    175  PROD001
TRANBKUP             2000/07/04  2000/10/15 12:20     92  PROD001

```

Select one or more members by placing a number in the CMD field preceding the desired member, and press Enter.

CA JCLCheck displays those members selected for processing above the divider line (=====) and each has an ordinal number indicating its order of processing.

The order of the display is the order chosen. For example, PRODINIT, the member on the top line of the preceding display, is the first member that CA JCLCheck processes.

To exit the panel without making any changes, enter **CANCEL** in the COMMAND field, and press Enter.

Enter **End** to pass the selected members to CA JCLCheck for processing.

### Changing the Order of Members

To change the order of a member, enter the new order number in the CMD field next to the member name, and press Enter.

Entering an order number X next to a member name causes all members with order numbers greater than or equal to X to shift down with their order numbers incremented by 1. For example, entering 4 next to PRODLLOAD results in the following display:

CMD	MEMBER	ORDER	CREATED	LAST MODIFIED	SIZE	USERID
	PRODINIT	1	2000/02/01	2000/02/15 12:17	40	PROD001
	PROD0101	2	2000/02/09	2000/03/01 09:11	7	PROD002
	PRODLLOAD	4	2000/02/02	2000/02/15 12:19	72	PROD001
	PRODRUN	5	2000/03/01	2000/03/01 09:45	119	PROD001
	PRODBKUP	6	2000/02/01	2000/02/17 15:32	2000	PROD001
=====	==	=====	=====	=====	=====	=====

### Adding a Member

If you enter an order number next to a member below the divider line, CA JCLCheck moves the member above the line according to the order number. Use an order number greater than the last order number + 1 to leave room for insertion. You can use the REORDER command to assign consecutive order numbers. Otherwise, CA JCLCheck reorders the order numbers when you exit this panel.

### Removing a Member

To remove a member from the selected list, enter **0** in the CMD field next to the member name, and press Enter.

### Browsing a Member

To help you decide whether to process a member, you can browse it. To browse a member, enter **B** in the CMD field next to a member name, and press Enter. This causes the display of an ISPF browse panel for the member. Press End to return to the Member Select panel.

### Assigning the Next Available Order Number

To assign the next available order number, enter **N** (Next) beside the name of one or more members (either above or below the divider line). This is especially useful if you want to assign sequential order numbers to several members. Instead of entering each sequential number, enter **N** on each line.

## Member Select for CA Librarian (CAZ1SLLP/CAZ1DTPP) Update and Browse

Use this panel to select members from a CA Librarian library for CA JCLCheck to process.

To display this panel:

1. Enter Option 1, 2, 3, or 4 in the OPTION field on the CA JCLCheck/SPF Menu panel (JCK0200), and press Enter.
2. Enter a complete data set name and data set type of LIB in the appropriate fields on the CA JCLCheck Invocation panel (JCK0201). Press Enter. Do not enter a member name.

----- CA JCLCHECK MEMBER SELECT -----						
COMMAND ==>			SCROLL ==> PAGE			
Data Set Name: CAI.LIBR37.MASTER			Type .: LIBRARIAN			
Last Updated : mm/dd/yyyy 10:40			USERID: APCD004			
CMD	MODULE	ORDER	DESCRIPTION	STATUS	SIZE	PROGRAMMER
	JCLCPR01	1	FIRST SHIFT PROD	PROD	421	M. PARIS
	JCLCPR02	2	SECOND SHIFT PROD	PROD	552	B. JONES
	JCLCPR03	3	THIRD SHIFT PROD	PROD	754	K. NORMAN
	PRODSUM1	4	FIRST SHIFT TOTALS	PROD	51	H. JAMES
	PRODSUM2	5	SECOND SHIFT TOTALS	PROD	72	T.J.BAKER
=====						
	PRODBKP1		FIRST SHIFT BACKUP	PROD	429	M. PARIS

Select one or more modules by placing an order number in the CMD field preceding the desired module, and pressing Enter. CA JCLCheck places each selected module above a divider line, separating it from those modules not selected.

CA JCLCheck displays those modules selected for processing above the divider line, (=====), and each has an ordinal number indicating the order in which CA JCLCheck should process it.

The order of the display is the order for processing. For example, JCLCPR01, the module on the top line of the preceding display, is the first module that CA JCLCheck processes.

To exit the panel without making any changes, enter **CANCEL** in the COMMAND field, and press Enter.

Press End to pass the selected members to CA JCLCheck for processing.

To view additional information about the data set, press Enter. This displays the alternate Member Select panel for CA Librarian (CAZ1SDTPP).

```

----- CA JCLCHECK MEMBER SELECT -----
COMMAND ==>                                SCROLL ==> PAGE
Data Set Name: CAI.LIBR37.MASTER              Type .: LIBRARIAN

CMD MODULE  ORDER  DESCRIPTION                STATUS LANG  LAST UPDATED
JCLCPR01    1     FIRST SHIFT PROD          PROD  JCL   2000/11/22
JCLCPR02    2     SECOND SHIFT PROD         PROD  JCL   2000/10/23
JCLCPR03    3     THIRD SHIFT PROD          PROD  JCL   2000/06/15
PRODSUM1    4     FIRST SHIFT TOTAL         PROD  JCL   2000/04/18
PRODSUM2    5     SECOND SHIFT TOTAL         PROD  JCL   2000/07/21
=====
PRODBKP1    ==     FIRST SHIFT BACKUP        PROD  JCL   2000/11/22
    
```

Press Enter to toggle between the two displays.

### Changing the Order of Modules

To change the order of a module, enter the new order number in the CMD field next to the module name, and then press Enter.

Entering an order number X next to a module name causes all modules with order numbers greater than or equal to X to shift down with their order numbers incremented by 1. For example, entering 4 next to JCLCPR02 results in the following display:

```

MODULE  ORDER  DESCRIPTION                STATUS  SIZE  PROGRAMMER
JCLCPR01  1  FIRST SHIFT PROD          PROD   421  M. PARIS
JCLCPR03  3  THIRD SHIFT PROD          PROD   754  K. NORMAN
JCLCPR02  4  SECOND SHIFT PROD         PROD   552  B. JONES
PRODSUM1  5  FIRST SHIFT TOTAL         PROD    51  H. JAMES
PRODSUM2  6  SECOND SHIFT TOTAL         PROD    72  T.J.BAKER
=====
    
```

### **Adding a Module**

If you enter an order number next to a module below the divider line, CA JCLCheck moves the module above the line according to the order number.

Use an order number greater than the last order number + 1 to leave room for insertion. You can use the REORDER command to assign consecutive order numbers. Otherwise, CA JCLCheck reorders the order numbers when you exit this panel.

### **Removing a Module**

To remove a module from the selected list, enter **0** in the CMD field next to the module name, and press Enter.

### **Browsing a Module**

To decide whether to select a module for processing, you can browse the module. To browse a module, enter **B** in the CMD field next to a module name, and press Enter. This displays an ISPF browse panel for the module. Press End to return to the Member Select panel.

### **Assigning the Next Available Order Number**

To assign the next available order number, enter **N** (Next) beside the names of one or more modules (either above or below the divider line). This is especially useful if you want to assign sequential order numbers to several modules. Instead of entering each sequential number, enter **N** on each line.

## **Member Select for CA Panvalet (CAZ1SLNP/CAZ1DTPP) Update and Browse**

Use this panel to select data set members from a CA Panvalet library for CA JCLCheck to process.

To display this panel:

1. Enter Option 1, 2, 3, or 4 in the OPTION field on the CA JCLCheck/SPF Menu panel (JCK0200), and press Enter.
2. Enter a complete data set name and data set type of PAN in the appropriate fields on the CA JCLCheck Invocation panel (JCK0201), and press Enter. Do not enter a member name.

```

----- CA JCLCHECK MEMBER LIST -----
COMMAND ==>                                SCROLL ==> PAGE
Data Set Name: CAI.PANV37.MASTER           Type .: CA Panvalet

  CMD   NAME      ORDER  LANG  STAT  USER  LAST ACCESS  LAST MAINT  STMTS
-----
        APCDPR01    1   JCL   TAE  12SJ  01/18/2000   01/05/2000   421
        APCDPR02    2   JCL   TAE  TS22  01/05/2000   12/23/2000   522
        APCDPR03    3   JCL   TAE  1538  12/15/2000   12/10/2000   754
        PRODSUM1    4   JCL   TAE  MMS2  12/01/2000   11/28/2000   51
        PRODSUM2    5   JCL   TAE  NLD3  11/25/2000   11/21/2000   72
=====
        PRODBKP1           JCL   TAE  JHD5  11/25/2000   11/21/2000   427
    
```

Select one or more members by placing an order number in the CMD field preceding the desired module, and pressing Enter. CA JCLCheck places each selected module above a divider line, separating it from those modules not selected.

CA JCLCheck displays those modules selected for processing above the divider line, (=====), and each has an ordinal number indicating its order in the data set.

The order of the display is the order for processing. For example, PRODINIT, the module on the top line of the preceding display, is the first module that CA JCLCheck processes.

To exit the panel without making any changes, enter **CANCEL** in the COMMAND field, and press Enter.

Enter **End** to pass the selected members to CA JCLCheck for processing.

### Changing the Order of Modules

To change the order of a module, enter the new order number in the CMD field next to the module name, and press Enter.

Entering an order number X next to a module name causes all modules with order numbers greater than or equal to X to *shift down* with their order numbers incremented by 1. For example, entering 4 next to JCLCPR02 results in the following display:

```

NAME      ORDER  LANG  STAT  USER  LAST ACCESS  LAST MAINT
-----
JCLCPR01    1   JCL   TAE  12SJ  01/18/2000   01/05/2000
JCLCPR02    2   JCL   TAE  TS22  01/05/2000   12/23/2000
JCLCPR03    4   JCL   TAE  1538  12/15/2000   12/10/2000
PRODSUM1    5   JCL   TAE  MMS2  12/01/2000   11/28/2000
PRODSUM2    6   JCL   TAE  NLD3  11/25/2000   11/21/2000
=====
    
```

### **Adding a Module**

If you enter an order number next to a module below the divider line, CA JCLCheck moves the module above the line according to the order number.

You can use an order number greater than the last order number + 1 to leave room for insertion. You can use the REORDER command to assign consecutive order numbers. Otherwise, the order numbers are reordered when you exit this panel.

### **Removing a Module**

To remove a module from the selected list, enter **0** in the CMD field next to the module name, and press Enter.

### **Browsing a Module**

To decide whether to select a module for processing, you can browse the module. To browse a module, enter **B** in the CMD field next to a module name, and press Enter. This displays an ISPF browse panel for the module. Press End to return to the Member Select panel.

### **Assigning the Next Available Order Number**

To assign the next available order number, enter **N** (Next) beside the names of one or more modules (either above or below the divider line). This is especially useful if you want to assign sequential order numbers to several modules. Instead of entering each sequential number, enter **N** on each line.

## **Job Scheduling Interface for CA 7 Panel (JCK10)**

Use this panel to specify forecasting time periods and job/system names that you want CA 7 to process. When you press Enter, this dialog causes CA JCLCheck to submit a batch job to the internal reader for execution of the CA 7 BTI procedure. The CA 7/CA JCLCheck interface, SASSJCK0, and processes the jobs you select on this panel. CA 7 extracts the job's JCL and passes it to CA JCLCheck for validation, which treats them as related jobs.

**Note:** This does not forecast work under CA 7. It only validates JCL for the jobs and time periods specified.

To display this panel, select Option 4 (CA 7 Interface Invocation) on the CA JCLCheck/SPF Menu panel (JCK0200).

```
-----JOB SCHEDULING INTERFACE FOR CA 7----- JCK10
COMMAND ==>

Date - mm/dd/yyyy           User - USER01           Time - 16:30

From Date ==> 090195       From Time ==> 12:00
To Date ==> 090295       To Time ==> 12:00
CA 7 OPID ==> USER01     Password ==>

Enter Jobs and/or Systems to be selected:

Jobname ==> PAYROLL       System ==> PRD1
Jobname ==> GENLEDGE     System ==> PRD2
Jobname ==>              System ==>
Jobname ==>              System ==>
Jobname ==>              System ==>
Jobname ==>              System ==>
Jobname ==>              System ==>
Jobname ==>              System ==>
Jobname ==>              System ==>

Press Enter to Submit or End to return to main menu
```

Supply the information you want to take effect when CA JCLCheck interfaces with CA 7.

This panel contains the following fields:

**From Date and From Time**

Defines a starting date and time using *mmdyy* format for date and *hhmm* format for time of FJOB, the forecasting command.

**To Date and To Time**

Defines an ending date and time using *mmdyy* format for date and *hhmm* format for time of FJOB, the forecasting command.

**CA 7 OPID**

Defines an operator ID for CA 7.

**Password**

Defines a password that displays CA 7.

**Enter Jobs and/or Systems to be selected**

Defines exact jobs and system names that you want forecast or a generic job name and system that reside on the CA 7 database.

You can use an asterisk (\*) as a special character to complete a string that the operating system can use to match on job name and system. For example, to use this panel to check an entire shift of JCL using the CA JCLCheck and CA 7 interface, specify the To and From date and time, the CA 7 OPID, and put an asterisk (\*) in the first job name.

**Note:** CA JCLCheck formats the data you enter in this panel into CA 7 FJOB commands. For more information about these fields, see the CA 7 documentation.

## Job Scheduling Interface for CA Scheduler Panel (JCK09)

Use this panel to specify scheduling job forecasting commands to present to CA Scheduler. When you press Enter, this dialog causes CA JCLCheck to submit a batch job to the internal reader for execution of the CA Scheduler CAJUTIL0 procedure. The jobs you select with forecast commands have their JCL extracted by CAJUTIL0 and passed to CA JCLCheck for validation.

To display this panel, select Option 5 (CA Scheduler Interface Invocation) on the CA JCLCheck/SPF menu (JCK0200).

```

----- JOB SCHEDULING INTERFACE FOR CA Scheduler-----JCK09
COMMAND ==>
  DATE - mm/dd/yyyy                USER - USER001                TIME - 08:21

  Simulate From Date ==> AUTOS      Report From Time ==>
  Simulate For        ==> 1          Report To Time   ==>
  Simulate From Time  ==>             SIMWORK          ==> N
  REVISE              ==> N          SIMVOL          ==>
  Stage JCL           ==> N          Everyday        ==> N

Select Report Names to Include:
  BACKLOG  DEVUTIL  LATEJOBS  SELECTED  UNITUTIL
  DATETRAN EXPERROR RESALL    SIMEXEC

Enter Any CA Scheduler Simulate Information:
==>
==>
==>
==>
==>
==>
==>
==>

Press Enter to Submit or End to Return to Main Menu

```

Supply the information you want to take effect when CA JCLCheck interfaces with CA Scheduler. See your CA Scheduler documentation for more information about scheduling commands.

This panel contains the following fields:

### Simulate FROM Date

Defines a starting date for simulation.

### Simulate For

Defines the number of days to simulate.

### Simulate FROM Time

Defines the starting time for simulation.

**REVISE**

Determines whether to use of the current job status.

**Y**

Uses the current job status from the tracking record.

**N**

Ignores the job status.

**Stage JCL**

Determines whether to use the JCL from the staging libraries.

**Y**

Uses the JCL from the staging libraries.

**N**

Ignores the JCL.

**Report FROM Time**

Defines a starting time for reports.

**Report TO Time**

Defines an ending time for reports.

The report interval, from date - to date, can be smaller than simulation interval.

**SIMWORK**

DEVCLASS value that fast copies the Job Master database as a temporary work file for dynamic allocation to DEVCLASS prior to the simulation run; N if you do not have available work space to copy the Job Master database.

**SIMVOL**

VOLSER that specifies the DASD volume to allocate the simulation work file to if you specified a DEVCLASS value in the SIMWORK field.

Enter a volume serial number up to six characters.

**EVERYDAY**

Y controls the specification of RFT (Report From Time) and RTT (Report To Time) keywords; N ignores this request.

**Select Report Names to Include**

S preceding the report name selects the report that you want CA Scheduler to produce.

**Note:** SIMEXEC is required if you want CA JCLCheck to produce CA JCLCheck error messages.

**Enter Any CA SCHEDULER SIMULATE Information**

Enter the CA Scheduler commands. Examples are:

```
ONLY SCHEDULE NAME=A*
ONLY SCHEDULE NAME=B*
EXCLUDE SCHEDULE NAME=ABC*
EXCLUDE JOB NAME=SAMPLE
```

**Control Card Override Panel (JCK12)**

You can use this panel to specify control card override libraries or the CNTLOVDD ddname (if the control card override libraries were pre-allocated).

To display this panel, enter Y in the CNTLOVER field on the CA JCLCheck Additional Processing Options panel (JCK0203).

**Example: Specify CNTLOVER Options**

In this example, the CNTLOVER runtime options specify three control card override libraries. If the TEST.CNTLCRDS library is not found, CA JCLCheck prints an information message and validation processing continues. If the other two libraries are not found, CA JCLCheck processing terminates.

```
JCK12 ----- DEFINITION OF CONTROL CARD OVERRIDES -----
COMMAND====>
Date - yyyy/mm/dd                Userid - USERID1                Time - hh:mm

CNTLOVDD:                        (CNTLOVDD and the libraries below are mutually exclusive)
                                   Type 'C' to continue or
                                   'E' or blank for error.

LIBRARY  1: QA.CNTL.CARDS          If Not Found?
LIBRARY  2: TEST.CNTLCRDS          If Not Found? C
LIBRARY  3: DEVL.CNTL.CARDS        If Not Found?
LIBRARY  4:                        If Not Found?
LIBRARY  5:                        If Not Found?
LIBRARY  6:                        If Not Found?
LIBRARY  7:                        If Not Found?
LIBRARY  8:                        If Not Found?
LIBRARY  9:                        If Not Found?
LIBRARY 10:                        If Not Found?
LIBRARY 11:                        If Not Found?
LIBRARY 12:                        If Not Found?
LIBRARY 13:                        If Not Found?
LIBRARY 14:                        If Not Found?
LIBRARY 15:                        If Not Found?
LIBRARY 16:                        If Not Found?

                                   Press Enter to continue
```

**Usage note:** When you specify CNTLOVDD and attempt to invoke CA JCLCheck in the ISPF foreground, CA JCLCheck expects the control card override libraries to have been pre-allocated in the TSO/ISPF session. However, when CNTLOVDD is specified and you attempt to invoke CA JCLCheck in ISPF-initiated background (a submitted independent batch job), CA JCLCheck takes the names of the CNTLOVDD libraries from the JCKUSR CLIST and imbeds them as a concatenation through JCKSKEL into the CA JCLCheck batch job. Thus, you can manage the control card override libraries for CA JCLCheck background invocations using the JCKUSR CLIST. To do this, modify the JCKUSR CLIST in CAI.CAZ2LSO as follows:

```
/******  
/* SET UP CONTROL CARD OVERRIDE LIBRARIES FOR CNTLOVDD. */  
/* THESE LIBRARIES WILL BE IMBEDDED IN THE CA JCLCHECK */  
/* BATCH INVOCATION JCL VIA THE JCKSKEL JCL SKELETON. */  
/* (THIS IS ONLY USED FOR ISPF BATCH INVOCATION WHEN */  
/* CNTLOVDD IS SPECIFIED.) */  
/******  
IF &PASS EQ 3 THEN DO  
  SET &CCODS1 = &STR(QA.CNTL.CARDS)  
  SET &CCODS2 = &STR(TEST.CNTL.CARDS)  
  SET &CCODS3 = &STR(DEVL.CNTL.CARDS)  
  SET &CCODS4 = &STR()  
  SET &CCODS5 = &STR()  
  SET &CCODS6 = &STR()  
  SET &CCODS7 = &STR()  
  SET &CCODS8 = &STR()
```

## OPROC and OPROCLIB Panel (JCK13)

You can use this panel to specify the OPROCLIB libraries or the OPROC ddname (if the OPROC libraries were pre-allocated).

To display this panel, enter Y in the OPROC field on the CA JCLCheck Additional Processing Options panel (JCK0203).

**Example: Specify OPROCLIB Options**

In this example, the OPROCLIB runtime options specify three PROC override libraries. If the DEV.PROCLIB library is not found, CA JCLCheck prints an information message and validation processing continues. If the other two libraries are not found, CA JCLCheck processing terminates.

```
JCK13 ----- DEFINITION OF OPROC DD NAME OR OPROCLIB LIBRARIES -----
COMMAND====>
Date - yyyy/mm/dd                Userid - USERID1                Time - hh:mm

OPROC:                (OPROC DD and the libraries below are mutually exclusive)
                                Type 'C' to continue or
                                'E' or blank for error.

LIBRARY  1: EMERGENCY.PROCLIB                If Not Found?
LIBRARY  2: DEV.PROCLIB                      If Not Found? C
LIBRARY  3: PROD.PROCLIB                    If Not Found?
LIBRARY  4:                                If Not Found?
LIBRARY  5:                                If Not Found?
LIBRARY  6:                                If Not Found?
LIBRARY  7:                                If Not Found?
LIBRARY  8:                                If Not Found?
LIBRARY  9:                                If Not Found?
LIBRARY 10:                                If Not Found?
LIBRARY 11:                                If Not Found?
LIBRARY 12:                                If Not Found?
LIBRARY 13:                                If Not Found?
LIBRARY 14:                                If Not Found?
LIBRARY 15:                                If Not Found?
LIBRARY 16:                                If Not Found?

                                Press Enter to continue
```

**Runtime Options Panel (JCK14)**

You can use this panel to specify the runtime options.

To display this panel, enter Y in the RUNTIME field on the CA JCLCheck Additional Processing Options panel (JCK0203).

### Example: Specify RUNTIME Options

In this example, the BYPRUNT runtime option specifies that CA JCLCheck bypass DB2 and USS runtime checks while executing all other runtime checks.

```
JCK14 ----- DEFINITION OF RUNTIME OPTIONS -----  
COMMAND====>  
Date - yyyy/mm/dd                Userid - USERID1                Time - hh:mm  
  
Perform all Runtime checks: (Y or N - Default is Y)  
    (You cannot specify N above and Y anywhere below)  
  
Select which Runtime checks to bypass:  
  
Utility control card check        : (Y or N)  
DB2 check                          : Y (Y or N)  
Data set existence and related checks: (Y or N)  
IMS check                          : (Y or N)  
JES check                          : (Y or N)  
Member existence check            : (Y or N)  
Program existence check           : (Y or N)  
Region size check                 : (Y or N)  
SCHENV check                       : (Y or N)  
SMS check                          : (Y or N)  
USS check                          : Y (Y or N)  
Volume and unit check             : (Y or N)  
  
Press Enter to continue
```

### Printoff Panel (JCK04)

Use this panel to generate hardcopy output for the CA JCLCheck SYSTEMM or SYSPRINT reports. Use the options on this panel to control the physical format of the output to the printer. You can also direct output to remote printers serviced by the DSPRINT application (or its equivalent).

**Note:** If you are running under CA Roscoe ETSO-ISPF, see Alternate Print Panel.

To display this panel:

1. Enter **P** in either the SYSTEMM or SYSPRINT fields on the CA JCLCheck/Invocation panel (JCK0201), and press Enter. CA JCLCheck displays a Member Select panel.
2. Select and process the desired members, view the error messages, and browse the reports.
3. When you are finished, press End.

```

----- PRINTOFF: PANEL TO PRINT DATASETS IMMEDIATELY -----JCK04
COMMAND ==> 1

Date - mm/dd/yyyy          User - USER01          Time hh:mm

1 - Execute PRINTOFF Function    2 - Cancel PRINTOFF Function

DATASET NAME => 'USER01.JCLCHECK.SYSPRINT'
CLASS ==> A          (Output Class the print will go to)
HOLD/NOHOLD => NOHOLD ("HOLD" will hold the Job until operator release)
FOLD/NOFOLD => NOFOLD (All characters are made UPPERCASE if "FOLD")
COPIES ==> 1        (Number of Copies, 1-999)
VOLUME ==>          (If the Dataset is not Cataloged)

"DSPRINT" Destination ID ==>    If Remote Printing is desired.

```

Supply the values you want to take effect to view these reports.

This panel contains the following fields:

#### 1 - Execute PRINTOFF Function

Prints the output.

#### 2 - Cancel PRINTOFF Function

Cancels the print request.

#### DATASET NAME

Data set name of the file you want to print.

#### CLASS

SYSOUT class for the SYSOUT file.

**Default:** A

#### HOLD/NOHOLD

Defers printing of the report until it is released by an operator; Nohold prints the report immediately.

#### FOLD/NOFOLD

Fold converts all lowercase characters to uppercase characters when printing the file; Nofold ignores character conversion.

#### COPIES

Defines a number from 1 to 999, which designates the number of copies to print.

**Default:** 1

#### VOLUME

Volume serial number of the disk for the data set that you want to print.

Leave this field blank if the data set is not cataloged.

### **DSPRINT Destination ID**

Defines the ID of the printer site.

## **Alternate Print Panel (JCK04I)**

This panel is a substitute for the JCK04 panel and is displayed if you are running the JCKSPF dialog under CA Roscoe ETSO-ISPF. Use the panel options to generate hardcopy output for the CA JCLCheck SYSTEM or SYSPRINT reports. The options on this panel allow you to control the physical format of the printer output. You can also direct output to remote printers serviced by the DSPRINT application (or an equivalent application).

To display this panel:

1. Enter **P** in either the SYSTEM or SYSPRINT fields on the CA JCLCheck/Invocation panel (JCK0201), and press Enter. CA JCLCheck displays a Member Select panel.
2. Select and process the desired members, view the error messages, and browse the reports.

When you are finished, press End.

```
----- PRINT: PANEL TO PRINT DATA SETS IMMEDIATELY ----- JCK04I
COMMAND==>

Date - mm/dd/yyyy           User - CAUSER           Time hh:mm

1 - EXECUTE THE PRINT FUNCTION   2 - CANCEL THE PRINT FUNCTION

DATASET NAME => 'USERID.JCLCHECK.SYSPRINT'
CLASS =====> A           (Output Class the Print will go to)
HOLD/NOHOLD => NOHOLD      (Hold will hold The Job Until Operator Version)
COPIES =====> 1         (Number of Copies, 1-999)

Destination ID(DEST), if Remote Printing is Desired
NODE          ==>
USER ID       ==>
```

Supply the values you want to take effect when printing the SYSTEM and SYSPRINT reports.

This panel contains the following fields:

#### **1 - EXECUTE THE PRINT FUNCTION**

Prints the output.

#### **2 - CANCEL THE PRINT FUNCTION**

Cancels the print request.

**DATASET NAME**

Data set name of the file you want to print.

**CLASS**

SYSOUT class for the SYSOUT file.

**Default:** A

**HOLD/NOHOLD**

Defers printing of the report until it is started by an operator; Nohold prints the report immediately.

**COPIES**

Number from 1 to 999, which designates the number of copies you want to print.

**Default:** 1

**NODE**

Symbolic node name that directs printing to a remote printer.

The node name can be one to eight alphanumeric or national characters.

**USER ID**

For remote printing, a user ID (one through seven alphanumeric or national characters) defined to the node.

**Note:** You must define a user ID and a node.

## Invoking with the EDCHEK Facility

The optional EDCHEK feature allows you to run CA JCLCheck as an edit macro under ISPF. CA JCLCheck analyzes and reports on the JCL in the file you are currently editing.

Edit macros are called within ISPF edit by entering either a percent sign (%) or an exclamation point (!) preceding the edit macro name.

To invoke EDCHEK, enter %EJCK, !EDCHEK, or !JCK (a shorthand form of CA JCLCheck) to invoke CA JCLCheck processing against the contents of your edit file. The preferred method to invoke EDCHEK is %EJCK. This CLIST calls the EDCHEK macro. The advantage of using %EJCK is that you can specify runtime options on the command line. These options are combined with the options specified in the EDCHEK options file. You cannot specify runtime options on the command line when using !EDCHEK or !JCK.

You can assign the %EJCK, !EDCHEK, or !JCK commands to a PF Key to make CA JCLCheck analysis a simple push button operation.

**Note:** You can use EJCK, EDCHEK, or JCK without the percent sign (%); however, this lengthens processing time.

When you invoke EDCHEK, it reads the current contents of the edit file and passes the records to CA JCLCheck for analysis. Error messages returned by CA JCLCheck are inserted directly into the edit file immediately following the offending statement and are identified by the prefix ==MSG>. The position of JCL in the edit file is not changed, however, any excluded lines are redisplayed after EDCHEK has completed.

## EDCHEK Screen Without Error Messages

If there are no errors, the screen looks like this:

```
EDIT ---- BELST02.JCL.CNTL(IEFBR14) - 01.00 ----- LAST CC WAS 00
COMMAND ==>                                SCROLL ==> CSR
***** TOP OF DATA *****
==MSG>CAY6000 NO STATEMENTS FLAGGED IN JOB "BELST02A" MAXIMUM SEVERITY WAS 0
==MSG>
==MSG>
==MSG>
000100 //BELST02J JOB (40100000), 'S. BELL', CLASS=P, MSGCLASS=X
000200 //*          SAMPLE EDIT CHECK EXECUTION JCL
000300 //STEP01 EXEC PGM=IEFBR14
***** BOTTOM OF DATA *****
```

Notice the CAY6000 message below the TOP OF DATA line.

A STATEMENTS FLAGGED message is inserted for each JOB card that CA JCLCheck finds in the file. You can easily locate the MSG lines by entering the subcommand LOCATE SPECIAL. This function is especially useful when working with a large edit file.

You can continue to edit the file with the ==MSG> lines in place or clear these lines using the RESET command. The message lines are not saved with the data set and do not need to be cleared before re-executing EDCHEK. EDCHEK executes the RESET command internally.

## EDCHEK Screen with Error Messages

This is an example of an EDCHEK screen with error messages.

```

EDIT ---- BELST02.JCL.CNTL(TESTJCS) - 01.01 ----- COLUMNS 001 072
COMMAND ==>                                SCROLL ==> CSR
***** ***** TOP OF DATA *****
==MSG> CAY6000 2 STATEMENTS FLAGGED IN JOB "BELST029" MAXIMUM SEVERITY WAS 8
==MSG>
000100 //BELST029 JOB (40100000), 'S. BELL', CLASS=P, MSGCLASS=X,
==MSG> //BELST029 JOB (40100000), "S. BELL", CLASS=P, MSGCLASS=X,
000110 // USER=BELST02
==MSG> // USER=BELST02
==MSG> CAY6341E "001 -- THIS CLASS IS RESERVED FOR SCHEDULED JOBS"
==MSG>
000200 //STEP1 EXEC PGM=IEBCOPY
==MSG> CAY6341W "002 -- ONLY SYSTEM PROGRAMMERS CAN USE AMASPZAP CONTACT
==MSG> TECH-SERVICES"
000300 //SYSPRINT DD SYSOUT=*
000400 //IN DD DSN=SYS1.PARMLIB, DISP=SHR
000500 //OUT DD DSN=BELST02.BKUP.PARMLIB, DISP=(,CATLG),
000600 // UNIT=3480,
000700 // DCB=(RECFM=F, LRECL=4104, BLKSIZE=4104)
000800 //SYSUT3 DD UNIT=SYSDA, SPACE=(CYL, (1,1))
000900 //SYSUT4 DD UNIT=SYSDA, SPACE=(CYL, (1,1))
***** ***** BOTTOM OF DATA *****

```

## EDCHEK Screen Using EDCHKLB

If you activate the EDCHKLB option, you receive generated ISPF labels on all JCL statements in error. The label format is a period (.) followed by five alphabetic characters. The following example shows the output with the labeling of errors; the lines in error are marked by labels ".AABJA" and ".AADCA", on the left side.

```

EDIT          USERID1.JCK.CNTL(EJCKERRS) - 01.00          Columns 00001 00072
Command ==>                                           Scroll ==> CSR
***** ***** Top of Data *****
==MSG> CAY6000 2 STATEMENTS FLAGGED IN JOB "USERID1L" MAXIMUM SEVERITY WAS 8
==MSG>
==MSG>
000100 //USERID1L JOB (113300000), 'EJCK ERRORS', CLASS=K,

000200 //          MSGCLASS=X
000300 //*
.AABJA //IEBGENER EXEC PGM=IEBGENERX
==MSG> //IEBGENER EXEC PGM=IEBGENERX
==MSG> CAY6042E OPERAND "IEBGENERX" IS TOO LONG FOR "PGM" PARAMETER OF "EXEC"
==MSG>
000500 //SYSPRINT DD  SYSOUT=*
000600 //SYSIN    DD  DUMMY
000700 //SYSUT2   DD  SYSOUT=*
.AADCA //SYSUT1   DD  DSN=DATASET.DOES.NOT.EXIST, DISP=SHR
==MSG> //SYSUT1   DD  DSN=DATASET.DOES.NOT.EXIST, DISP=SHR
==MSG> CAY6079E DATA SET "DATASET.DOES.NOT.EXIST" SPECIFIED AS OLD OR SHR,
==MSG>          BUT CANNOT BE FOUND
==MSG>
000900 //*

```

You can locate the next line containing a label in an EDCHEK screen by entering the subcommand LOCATE LABEL on the ISPF command line.

You can continue to edit the file with the labels in place or clear them using the RESET LABEL command. The ISPF labels are not saved with the data set and do not need to be cleared before re-executing EDCHEK. EDCHEK executes the RESET command (for ==MSG> lines) and RESET LABEL command internally. Pre-existing labels are replaced if any line containing a label is in error.

## EDCHEK Screen Using EDCHKSL

If you activate the EDCHKSL option, you receive an error severity breakdown within the panel, including the number of statements by severity and their associated line numbers. The following example shows the output using EDCHKSL:

```

EDIT ---- BELST02.JCL.CNTL(TESTJCS) - 01.01 ----- COLUMNS 001 072
COMMAND ==>                                     SCROLL ==> CSR
***** ***** TOP OF DATA *****
==MSG> CAY6000 2 STATEMENTS FLAGGED IN JOB "BELST029" MAXIMUM SEVERITY WAS 8
==MSG> CAY6000 1 STATEMENT  FLAGGED WITH SEVERITY 4
==MSG> CAY6000 1 STATEMENT  FLAGGED WITH SEVERITY 8
==MSG> CAY6000 NO ERRORS WERE SUPPRESSED
==MSG>
==MSG> LINE #  SV MSG. ID.  ERROR MESSAGE TEXT
==MSG> =====
==MSG> 000110  08 CAY6341E "001 -- THIS CLASS RESERVED FOR SCHEDULED JOBS"
==MSG> 000200  04 CAY6341W "002 -- ONLY SYSTEM PROGRAMMERS CAN USE AMAS
==MSG>                PZAP CONTACT TECH-SERVICES"
000100 //BELST029 JOB (40100000), 'S. BELL', CLASS=P, MSGCLASS=X,
==MSG> //BELST029 JOB (40100000), "S. BELL", CLASS=P, MSGCLASS=X,
000110 //  USER=BELST02
==MSG> //  USER=BELST02
==MSG> CAY6341E "001 -- THIS CLASS IS RESERVED FOR SCHEDULED JOBS"
==MSG>
000200 //STEP1   EXEC PGM=IEBCOPY
==MSG> CAY6341W "002 -- ONLY SYSTEM PROGRAMMERS CAN USE AMASPZAP CONTACT
==MSG>                TECH-SERIVCES"
000300 //SYSPRINT DD  SYSOUT=*
***** ***** BOTTOM OF DATA *****

```

## EDCHEK Screen Using EDCHKSL and EDCHKLB

The following example shows the output when both EDCHKSL and EDCHKLB options are in effect.

```

EDIT          USERID1.JCK.CNTL(EJCKERRS) - 01.00          Columns 00001 00072
Command ==>                                           Scroll ==> CSR
***** ***** Top of Data *****
==MSG> CAY6000 2 STATEMENTS FLAGGED IN JOB "USERID1L" MAXIMUM SEVERITY WAS 8
==MSG> CAY6000 2 STATEMENTS FLAGGED WITH SEVERITY 8
==MSG> CAY6000 NO ERRORS WERE SUPPRESSED
==MSG>
==MSG> LINE # LABEL# SV MSG. ID. ERROR MESSAGE TEXT
==MSG> =====
==MSG> 000400 .AABJA 8 CAY6042E OPERAND "IEBGENERX" IS TOO LONG FOR "PGM"
==MSG>                                PARAMETER OF "EXEC"
==MSG> 000800 .AADCA 8 CAY6079E DATA SET "DATASET.DOES.NOT.EXIST"
==MSG>                                SPECIFIED AS OLD OR SHR, BUT CANNOT BE
==MSG>                                FOUND
==MSG>
000100 //USERID1L JOB (113300000), 'EJCK ERRORS', CLASS=K,
000200 //          MSGCLASS=X
000300 //*
.AABJA //IEBGENER EXEC PGM=IEBGENERX
==MSG> //IEBGENER EXEC PGM=IEBGENERX
==MSG> CAY6042E OPERAND "IEBGENERX" IS TOO LONG FOR "PGM" PARAMETER OF "EXEC"
==MSG>
000500 //SYSPRINT DD SYSOUT=*
000600 //SYSIN DD DUMMY
000700 //SYSUT2 DD SYSOUT=*
.AADCA //SYSUT1 DD DSN=DATASET.DOES.NOT.EXIST, DISP=SHR
==MSG> //SYSUT1 DD DSN=DATASET.DOES.NOT.EXIST, DISP=SHR
==MSG> CAY6079E DATA SET "DATASET.DOES.NOT.EXIST" SPECIFIED AS OLD OR SHR,
==MSG>                                BUT CANNOT BE FOUND
==MSG>
000900 //*

```

## The EDCHEK Options Menu (JCK21)

To change basic runtime options, use this panel before invoking EDCHEK.

To display this panel, do one of the following commands:

- Enter the command %EJCK while in ISPF edit mode.
- If this option panel is normally suppressed (BYPASS THIS SCREEN =Y), you can force it to reappear by specifying PANEL following the %EJCK command.

```
%EJCK [PANEL]
```

```

JCK21 ----- CA JCLCheck OPTIONS -----
COMMAND ==>
  DATE: yyyy/mm/dd          USER: USERID1          TIME: hh:mm

STANDARD - Name of STANDARD for compliance checking    =>
           =>           =>           =>           =>
           =>           =>
STDREXX  - Name of REXX EXEC for compliance checking    =>
SYNTAX   - Do syntax checking of JCL                   => Y
RUNTIME  - Do runtime checking of JCL (all runtime checks) => Y
REMOTE   - Do remote validation of JCL                 => N
           =>           =>
RESOLVE  - Resolve external variables by calling product. . . . => CA7
           => E DSN=> USERID1.JCK.CNTL          MEM=> RESPARMS
EDCHKLB  - Generate ISPF labels for lines with errors    => N
EDCHKSL  - Display summary of errors                   => Y
SECURITY - Do Security Checking                       => N
USER     - Specify Userid for security checking         =>

          BYPASS THIS SCREEN NEXT TIME..... => N

```

To specify JCK options directly from the ISPF command line using %EJCK, enter:

```
%EJCK [JCK options]
```

**Note:** The number of options you can enter is limited to the length of the command line.

To browse reports EDCHEK generates directly from the ISPF command line, enter:

```
%EJCK [REPORTS]
```

The REPORTS option requires allocation of a listing file and specification of a file name using the LIST(*ddname*) option. Press End to see the statements in error, after viewing the reports.

We recommend that you preallocate a standard sequential report file of at least 5 cylinders in size (that is, RECFM=FBA,LRECL=133), which is allocated as DISP=SHR by your logon procedure to the ddname EDCPRINT. Your systems programmer could have arranged an alternate method.

You can also access the panel (JCK20), which lists the valid spelling and abbreviations for most of the frequently changed options. This panel is a sample skeleton panel that you can change to suit your needs.

To display this panel, enter the following command on the command line:

%EJCK OPTIONS

```
JCK20 ----- CA JCLCheck RUNTIME OPTIONS ----- JCK20
COMMAND ==>

                SStandard(name1, . . . ,name8)
                STDRExx(name)
NOSYNtax        SYNTAX
NORUNT          RUNTime
NOREMote        REMote
NORESolv        SEV(4,8,12)
EDCHKLB        EDCHKLb
EDCHKSL        EDCHKSL
NOSEC          SECurity
NOUSER         USER(userid)
```

**Important!** You can enter just the commands or JCK options if you have a PF key that is defined for %EJCK such as, PANEL, options, REPORTS, OPTIONS on the command line. To perform the preceding access methods, press the PF key.

Alternatively, once you enter %EJCK, you can then enter EJCK without the percent sign to initiate further action that uses that same command (%EJCK). For example, if you enter %EJCK and then decide you want to see panel JCK20, you can enter EJCK OPTIONS rather than %EJCK OPTIONS.

On the JCK21 panel, enter the field values and options you want to take effect when you invoke CA JCLCheck using the EDCHEK macro.

**STANDARD**

Standard names on the Standard Rules database that CA JCLCheck uses for compliance checking.

If you do not want to invoke compliance checking, you can specify one to eight standard names or you can leave this field blank.

**STDREXX**

Specify the REXX member name of a CA JCLCheck REXX program that is used to enforce site standards. You can use standard IBM member pattern characters to display a member list for the data sets that are allocated to your SYSEXEC DD. Use the asterisk (\*) for multiple characters and the per cent sign (%) for a single character. You can select the member that you want to use from the list. Press Enter to have the selected member name filled in as the STDREXX value.

**SYNTAX**

Y specifies JCL syntax checking; N causes CA JCLCheck to ignore this option.

**RUNTIME**

Y causes CA JCLCheck to do runtime checking on the JCL; N causes CA JCLCheck to ignore this option.

**REMOTE**

Y activates remote validation; N deactivates remote validation.

If you activate remote validation, provide the name of the REMVAL file. The REMVAL file is a sequential or PDS member that defines remote validation options.

If your remote validation request fails, you see a 'LAST CC WAS 28' message on the top right-hand corner of the ISPF EDIT screen. For details describing the particular error, see the description of return code 28 in the *Message Reference Guide* and the specific error message in your TSO user ID job log.

**RESOLVE**

Specify the name of the scheduling product that CA JCLCheck is to call to resolve the external variables imbedded in the JCL. The variable resolution occurs before CA JCLCheck validates the JCL. You can specify the character N for none. CA JCLCheck can resolve external variables in the following scheduling products: CA WA CA 7 Edition (CA7), CA WA ESP Edition (ESP), Tivoli Workload Scheduler (TWS), and Control-M (CTM).

When a scheduling product is specified, a data set name must be specified. If the data set is a PDS, a member name is also needed. The data set contains the product-specific parameters for external variable resolution.

Immediately before the data set name field on the panel, there is a single-character field that can be set to 'B' or 'E'. This field enables you to browse (B) or edit (E) the parameter data set before invoking JCL validation.

**Note:** For information about external variable resolution, see About External Variables in the *Tutorial*.

**EDCHKLB**

Y causes CA JCLCheck to display ISPF generated labels on any line containing an error when running EDCHEK.

**EDCHKSL**

Y causes CA JCLCheck to display additional error information when running EDCHEK. This information includes totals by severity and line numbers where the errors are located in the JCL.

**SECURITY**

Y allows CA JCLCheck to use STORCLAS, PROGRAM, RUNCPU, or RUNDAT; N causes CA JCLCheck to ignore this option.

**USER**

User ID to use with security checking.

Leave this field blank if you do not want to override the user ID of the user executing CA JCLCheck. If you enter a user ID, specify Y in the SECURITY option field.

**BYPASS THIS SCREEN NEXT TIME**

Y bypasses this panel the next time you invoke EDCHEK.

## Invoking with the SUBCHEK Facility

The SUBCHEK program, an optional feature of CA JCLCheck, is installed as a front end to the IBM supplied TSO SUBMIT command processor. This feature allows CA JCLCheck to analyze the JCL in the file being submitted before the JCL is sent to the internal reader. If any errors are identified, CA JCLCheck displays messages at your terminal and does not submit the JCL for background processing. If CA JCLCheck does not detect any errors, it transfers control to the IBM SUBMIT routine for normal processing, including any SUBMIT exit routine installed on your system. CA JCLCheck displays the warning messages (severity 4 or less) and allows the submission of JCL for execution.

SUBCHEK requires you to allocate a special control statement file to the TSO user's session. This control file specifies CA JCLCheck processing options and contains file allocation information for your installation's PROCLIBs. If you do not allocate the control file, CA JCLCheck does not perform any preprocessing.

You can allocate the control file in your LOGON procedures. You can also allocate or free the control file by command or CLIST. You use a sample CLIST named SUBCHK (in the CAZ2CLS0 library) to turn SUBCHEK on and off. In this way, the installation or individual users can control the invocation of SUBCHEK.

## Invoking remote validation using ChekPlex

You can use CA JCLCheck to validate JCL on a remote system or systems. Remote validation is an optional feature of ChekPlex.

To invoke remote validation, the REMOTE option must be specified in the calling options, or in the OPTIONS data set.

ChekPlex registers with the General Transaction Server (GTS) to provide services related to JCL validation; ChekPlex shares the GTS network architecture. GTS must be set up to have communication paths to all servers participating in the validation process.

The execution JCL must contain the GTS load library. The GTS load library must be present in JOBLIB/STEPLIB, or in the linklist concatenation for every job using ChekPlex remote validation, or the load library.

The REMVAL DD name must refer to a data set that defines the following remote validation options: all nodes participating in the validation process, which data sets are to be validated, what procedure libraries are to be allocated on the target server, and what data sets are to be shipped with the source JCL from the source server to the target server.



Due to the authorized environment, ALLOCATE or FREE TSO/E commands executed in the user REXX procedure fail with RC=-3. Four programs can be used instead of the TSO/E ALLOCATE and FREE commands; contrary to the IBM commands, these programs do not require any specific environment to run with.

The CAZ1ALDS module is used to allocate data sets. The complete syntax including all the available parameters when called in a REXX environment is:

```
VAR = 'dd name,dataset name,member name,disp1,disp2,disp3,recfm,  
      lrecl,blksize,space unit,primary alloc,secondary alloc,directory,  
      unitname,volume serial,SMS storclas,SMS dataclas'  
ADDRESS ATTCHMVS 'CAZ1ALDS VAR'
```

**Usage notes:**

- SPACE UNIT can be "TRK" for Tracks or "CYL" for Cylinders. "BLK" is not supported.
- *disp1* is the status sub-parameter.
- *disp2* the normal termination disposition sub-parameter.
- *disp3* the abnormal termination (conditional) disposition sub-parameter.
- The default value for *disp1* is SHR.

**Example 1: Allocate a Data Set with a DDname**

This example shows how to allocate the data set PROD.JCK.CNTL with the ddname DD1 and status SHR:

```
VAR='DD1,PROD.JCK.CNTL'
```

**Example 2: Allocate a Data Set**

This example shows how to allocate data set PROD.JCK.TEST:

```
VAR='DD1,PROD.JCK.TEST,,NEW,CATLG,CATLG,FBA,133,,TRK,1,2,,SYSDA,TEST01'
```

This is the same as allocating with the following JCL statement:

```
//DD1 DD DISP=(NEW,CATLG,CATLG),RECFM=FBA,LRECL=133,SPACE=(TRK,(1,2)),  
// UNIT=SYSDA,VOL=SER=TEST01
```

Since the parameters are positional, commas are used for any omitted parameters.

The CAZ1ALSY module is used to allocate SYSOUT files. The complete syntax including all the available parameters when called in a REXX environment is:

```
VAR = 'dd name,class'  
ADDRESS ATTCHMVS 'CAZ1ALSY VAR'
```

**Example 1: Allocate DD1**

This example shows how to allocate DD1 to SYSOUT=A:

```
VAR='DD1,A'
```

**Example 2: Allocate DD2**

This example shows how to allocate DD2 to the default class (same as SYSOUT=\*):

```
VAR='DD2,*'
```

The CAZ1CONC module is used to concatenate data sets. The complete syntax including all the available parameters when called in a REXX environment is:

```
VAR = 'dd name1,...dd name16'  
ADDRESS ATTCHMVS 'CAZ1CONC VAR'
```

Up to 16 DDs can be concatenated with one call to the program.

The CAZ1FREE module is used to unallocate data sets. The complete syntax including all the available parameters, when called in a REXX environment, is:

```
VAR = 'dd name1,...dd name16'  
ADDRESS ATTCHMVS 'CAZ1FREE VAR'
```

Up to 16 DDs can be unallocated with one call to the program.

**Return Codes:****0**

Function was successful.

**4**

Function not complete; reason could be an invalid parameter.

**8**

Function was not successful; internal (contact CA Support).

**12**

Function was not successful; invalid parameter list given.

## Invoking Under CA Roscoe

You can use CA JCLCheck as a CA Roscoe monitor in place of, or in addition to, the standard CA Roscoe JCL syntax checker. CA JCLCheck is invoked as a subtask of CA Roscoe so that use of CA JCLCheck has minimum impact on the rest of the CA Roscoe system.

To use the CA Roscoe/CA JCLCheck monitor, load the JCL that you want CA JCLCheck to analyze into the active workspace (AWS) and enter the following commands:

JCK [*options*]

**Note:** You can specify up to 64 bytes of CA JCLCheck options.

The following panel displays the results of invoking the JCK command on JCL that contains errors:

```
> APPLID(APPLR0S)  USER(STB,BELST02)
>
...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
//STEP02 EXEC PGM=NOTTHERE
CAY6093E PROGRAM "NOTTHERE" NOT FOUND

//DD1 DD UNIT=DISK,SPACE=(TRK,0),VOL=SER=XYZZY
CAY6077W NO "DISK" UNITS ARE AVAILABLE FOR "XYZZY"

CAY6000 2 STATEMENTS FLAGGED IN JOB "BELST02A" MAXIMUM SEVERITY WAS 8
```

This panel displays the results of using the JCK command on JCL that does not have any errors:

```
> APPLID(APPLR0S)  USER(STB,BELST02)
>
...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
CAY6000 NO STATEMENTS FLAGGED IN JOB "BELST02A" MAXIMUM SEVERITY WAS 0
```

The JCK command invokes CA JCLCheck to analyze the JCL and display all messages on the terminal. The data in the AWS is not disturbed.

## The JCKRPF RPF

The Roscoe Program Facility (RPF) interface is similar to the CA JCLCheck TSO/ISPF dialog. Two RPFs are furnished with CA JCLCheck: JCKRPF (for CA JCLCheck) and JCKNRPF (for JCLNeat).

Both RPFs require that files be properly allocated for SYSTERM, SYSPRINT, and SYSGRAPH. When you invoke JCKRPF, you are presented with the CA JCLCheck/Roscoe menu. When you invoke JCKNRPF, you are presented with the r11.0 JCLNeat panel.

## CA JCLCheck/Roscoe Menu (JCKMAINR)

From this menu, you can display panels to select control options, access various types of CA JCLCheck invocation panels, and panels that allow you to establish interaction with CA WA CA 7 Edition or CA Scheduler when running CA JCLCheck under CA Roscoe.

To display this panel, enter **JCKRPF** in the command input area in CA Roscoe and press Enter.

To exit this panel, press PF3.

```

===== CA JCLCheck/Roscoe MENU =====
OPTION ==>

      DATE:           USER:           TIME:

          0 - OPTION SPECIFICATIONS
          1 - CA JCLCheck - CHECK AWS (JCK)
          2 - CA JCLCheck - FOREGROUND(ETSO)
          3 - CA JCLCheck - BATCH CHECK
          4 - CA 7 INTERFACE INVOCATION
          5 - CA SCHEDULER INTERFACE INVOCATION
          X - EXIT

PF3 - EXIT
=====

```

Enter the number of the option for the panels you want to display in the OPTION field and press Enter.

### 0 - OPTION SPECIFICATIONS

Displays the CA JCLCheck Control Options Menu (JCKOPCTL). This gives you a choice of four different types of control option panels that you use to view or update various types of control options and JOB statement specification parameters.

**1 - CA JCLCheck - CHECK AWS (JCK)**

Permits scanning of JCL pulled into the Active Work Space (AWS) at your discretion.

This option uses the JCK monitor. It cannot produce reports but does flag any errors to the terminal.

**2 - CA JCLCheck - FOREGROUND(ETSO)**

Permits scanning of JCL with the addition of full reporting.

Use this RPF to selectively browse reports.

**Note:** For information regarding the available reports, see [CA JCLCheck Reports](#) (see page 225).

**3 - CA JCLCheck - BATCH CHECK**

Permits the scheduling of a batch job for scanning JCL that has been selectively inserted into the AWS.

**4 - CA 7 INTERFACE INVOCATION**

Displays the Job Scheduling Interface for CA 7 panel (JCKSCA7) which allows you to enter applicable forecast information.

Press Enter to submit execution of the CA 7 BTI procedure.

**5 - CA SCHEDULER INTERFACE INVOCATION**

Displays the Job scheduling Interface for CA Scheduler panel (JCKSSCD) which allows you to enter applicable forecast information.

Press Enter to submit execution of the CA Scheduler CAJUTIL0 procedure.

**X - EXIT**

Selecting X causes you to exit the CA JCLCheck/INVOCATION menu.

## CA JCLCheck Control Options Menu (JCKOPCTL)

Use this panel to view or change your processing control options, output control options, or security options.

To display this panel, select Option 0 (Option Specifications) on the CA JCLCheck/Roscoe Menu (JCKMAINR).

To exit this panel and return to the previous panel, press PF3.

```
-----CA JCLCHECK/CONTROL OPTIONS MENU-----
OPTION ==>
      DATE:           USER:           TIME:
                1 - PROCESS CONTROL OPTIONS
                2 - OUTPUT CONTROL OPTIONS
                3 - SECURITY OPTIONS
                X - EXIT
```

Enter the desired option in the OPTION field, and press Enter.

### 1 - PROCESS CONTROL OPTIONS

Displays the CA JCLCheck Process Control Options panel (JCKOPN1). This allows you to view or change process control options.

Optionally, you can go to the Additional Processing Options panel (JCKOPN1A) from this panel.

### 2 - OUTPUT CONTROL OPTIONS

Displays the CA JCLCheck Output Control Options panel (JCKOPN2). This allows you to specify options that control the output produced by CA JCLCheck.

### 3 - SECURITY OPTIONS

Displays the CA JCLCheck Security Options panel (JCKOSEC). This allows you to specify options that request support for pre-validation of the security environment.

## Process Control Options (JCKOPN1)

Use this panel to specify options to control how CA JCLCheck processes the input JCL file. The dialog checks the syntax of all the options that you select before accepting them.

The options that you set up are the options in effect until you change them (when you end out of this panel).

**Note:** To display additional processing options, press PF8.

To display this panel, enter Option 1 (Process Control Options) in the OPTION field on the CA JCLCHECK/ROSCOE CONTROL OPTIONS MENU (JCKOPCTL).

To exit this panel and return to the previous panel, press PF3.

```

===== JCLCHECK PROCESS CONTROL OPTIONS =====
CTLSCAN - SCAN UTILITY CONTROL STMTS          =>
DESTCHK - VALIDATE JES REMOTE DESTINATIONS    =>
EASYRDR - EASYRDR PROCLIBS ARE TO BE SEARCHED =>
INDEX   - RESTRICT CATALOG SEARCH TO A HI-LEVEL INDEX =>
LIST    - (                                     )
MSS     - STAGE VTOC OF MSS VOLUMES IF NECESSARY =>
PASSWORD - FOR READ PROTECTED VSAM CATALOGS ( )
PDSRELATE - INPUT TREATED AS CONSECUTIVE JOBS =>
PREFIX  - DDNAME PREFIX FOR NON-MVS DYNAMIC ALLOC ( )
CA 7    - PROCESS CA 7 CONTROL STMTS ( )
COND    - SIMULATES EXECUTION ACCORDING TO CONDITIONS =>
         - STEP( )PROCSTEP( )CODE( )
         - STEP( )PROCSTEP( )CODE( )
         - STEP( )PROCSTEP( )CODE( )

PF3/15 - RETURN      PF8/20 - ADDITIONAL OPTIONS
=====

```

Enter the processing values that you want to take effect, and press Enter to invoke CA JCLCheck using these options.

**Note:** For complete details about each option, see [CA JCLCheck Runtime Options](#) (see page 107).

**CTLSCAN**

Y requests that CA JCLCheck read and interpret control statements for the following IBM utility programs IEHPRGM, IDCAMS, DFSORT, ADDRSSU, IEBCOPY, and IEBCOPY. N deactivates these control statements.

**DESTCHK**

Y activates validation of remote destination names on DD statements and JES control statements.

**EASYRDR**

Y activates support for EASYRDR.

**INDEX**

Y restricts catalog look-up and data set existence checking to the specified high-level indexes.

**LIST**

The primary report file (normally SYSPRINT).

**MSS**

Y requests that CA JCLCheck stage the VTOC of mass storage volumes when necessary to verify the existence of a data set on MSS; N ignores this option.

**PASSWORD**

The password to use for accessing read protected VSAM catalogs.

**PDSRELATE**

Y causes CA JCLCheck to treat the primary input library as a series of consecutive, related job streams.

**PREFIX**

Specifies the DD prefix that is used for dynamic allocation in non-z/OS systems, and for JES3 /\*MAIN statement PROCLIB support.

**CA 7**

Specifies CA WA CA 7 Edition scheduled override statements to be processed.

**COND**

Y activates condition code simulation and the IF/THEN/ELSE/ENDIF statement; N deactivates condition code simulation.

Enter the procedure stepnames and whether condition code simulation should be in effect (Y).

## Additional Processing Options Panel (JCKOPN1A)

Use this panel to specify additional options that control how CA JCLCheck processes the input JCL file. The dialog checks the syntax of all options selected before accepting them.

To display this panel, press PF8 in the CA JCLCheck Process Control Options panel (JCKOPN1).

```

=====CA JCLCheck ADDITIONAL PROCESSING OPTIONS =====
STANDARD - NAME OF STANDARD(S) FOR COMPLIANCE CHECKING OR BLANK =>
              =>              =>              =>              =>
              =>              =>
STDREXX  - NAME OF REXX STANDARD FOR COMPLIANCE CHECKING           =>
SYNTAX   - DO SYNTAX CHECKING OF JCL                               => Y
RUNTIME  - DO RUNTIME CHECKING OF JCL                             => Y
SYSUID   - SPECIFY VALUE FOR SYSTEM PARAMETER@SYSUID+             =>

PF3/15 - RETURN
=====

```

Enter the options you want to take effect when you invoke CA JCLCheck.

**Note:** For more information about these options, see [CA JCLCheck Runtime Options](#) (see page 107).

**STANDARD**

Standard name (from the Standards Rules database) that you want CA JCLCheck to use for compliance checking.

You can specify one to eight standard names or leave this field blank if you do not want to invoke compliance checking

**STDREXX**

Y REXX EXEC name to use for job control standard checking for data center JCL standards enforcement.

**SYNTAX**

Y activates syntax checking of the JCL; N displays options only.

**RUNTIME**

Y causes CA JCLCheck to perform runtime checking on the JCL; N deactivates runtime checking.

**SYSUID**

User ID that you want CA JCLCheck to use to replace the system symbolic parameter, SYSUID.

## Output Control Options Panel (JCKOPN2)

Use this panel to specify options that control the output which CA JCLCheck produces and how it is formatted.

To display this panel, select Option 2 on the CA JCLCheck Control Options Selection panel (JCKOPCTL).

**Note:** When you press PF3 from this panel, you return directly to the CA JCLCheck/Roscoe Menu (JCKMAINR).

===== JCLCHECK REPORTING OPTIONS =====			
JOB	- LIST INPUT JCL	(RPT 1)	=> Y
FULLLIST	- LIST COMPLETE MERGED JCL AND PROCS	(RPT 2)	=> Y
XREF	- PRODUCE DATASET CROSS REFERENCE	(RPT 3)	=> Y
PXREF	- PRODUCE PROGRAM CROSS REFERENCE	(RPT 4)	=> Y
RPTRPT	- LIST SYSOUT REPORTS	(RPT 5)	=> Y
ERROR(End)	- PLACE ERROR REPORT LAST	(RPT 6)	=> Y
(START)	- PLACE ERROR REPORT FIRST		=> N
(INLINE)	- PLACE ERROR MESSAGES IN-LINE IN REPORT 2		=> Y
SPACE(RPT)	- LIST DASD SPACE IN DATASET REPORTS		=> N
(NN)	- MAXIMUM % OF TRACK USAGE		=> 85
(ALL)	- CHECK BLKSIZE FOR EXISTING DASD FILES		=> Y
CCLIST	- NUMBER OF PROGRAM CONTROL CARDS TO BE LISTED		=> 5
SEV	- MINIMUM SEVERITY OF ERRORS TO BE LISTED		=> 00
SX(RPT)	- SUMMARY DSN, PGM, RPT CROSS REFERENCE		=> Y
GRAPH	- STEP LEVEL FLOWCHARTS		=> N
TERM	- SYSTEM OUTPUT		=> N
PF3/15 - RETURN			

Supply the values you want to take effect when you invoke CA JCLCheck.

#### **JOB**

Y prints the Listing of Jobstream JCL Report (Report 1); N deactivates this option.

#### **FULLLIST**

Y prints the Listing of Merged JCL Report (Report 2); N deactivates this option.

#### **XREF**

Y prints the Data Set Cross-Reference Report (Report 3); N deactivates this option.

#### **PXREF**

Y prints the Program Cross-Reference Report (Report 4); N deactivates this option.

#### **RPTRPT**

Y prints Report Listing (Report 5); N deactivates this option.

The Report Listing is a list of all SYSOUT reports produced by the job.

#### **ERROR(End)**

Y prints the error messages at the end of the listing of the Error Messages Report (Report 6); N deactivates this option.

End is the default for this option.

#### **ERROR(START)**

Y prints the Error Messages Report (Report 6) prior to the Listing of Jobstream JCL Report (Report 1); N deactivates this option.

**ERROR(INLINE)**

Y prints error messages within the expanded JCL listing immediately following the statement flagged as an error; N deactivates this option.

**SPACE(RPT)**

Y reports space usage for all files without any track efficiency analysis; N deactivates this option.

The DASD space usage is listed on the Data Set Cross-Reference Report (Report 3) and the Summary Data Set Cross-Reference Report (Report 7).

**SPACE(NN)**

Enter a percentage from 0 to 99 in the NN field to check the space efficiency on new data sets created by the input job stream, or the minimum percentage of track usage.

The suggested values are 75 to 85 percent.

**SPACE(ALL)**

Y in the ALL field performs the space efficiency check on both new and existing data sets; N deactivates this option.

**CCLIST**

A number from 0 to 999 that specifies the number of program control statements you want CA JCLCheck to list.

**SEV**

A value from 0 to 16 that defines the minimum severity of errors that you want CA JCLCheck to list.

**SX(RPT)**

Y activates the related job data set validation feature and generates summary reports (7-9) at both the individual job and system summary level; N deactivates this option.

**GRAPH**

Y produces the Flow Diagram Report (Report 10) and optionally provides the ddname of the SYSOUT file; N deactivates this option.

**TERM**

Contains the terminal report file.

The logical record length of this file is 80 and can be blocked. You can change the ddname of this file with the Term option discussed in the chapter "CA JCLCheck Runtime Options."

Optional unless you specify PARM=TERM.

## Security Options Panel (JCKOSEC)

Use this panel to specify options to control pre-validation of the security environment. This dialog checks the syntax of all the options currently selected.

To display this panel, select Option 3 on the CA JCLCheck Control Options Menu (JCKOPCTL). Press PF3 to return to that menu.

```

===== CA JCLCheck/SECURITY OPTIONS =====
SECURITY - DO SECURITY CHECKING                => N
USER      - ALTERNATE USERID FOR SECURITY CHECKING OR BLANK =>
DASDVOL   - PERFORM DASDVOL SECURITY CHECKS      => N
MGMTCLAS  - PERFORM MGMTCLAS SECURITY CHECKS     => N
STORCLAS  - PERFORM STORCLAS SECURITY CHECKS     => N
PROGRAM   - PERFORM PROGRAM LEVEL SECURITY CHECKS => N
           - CHECK IF EXECUTED PROGRAM IS DEFINED TO SECURITY => N
RUNCPU    - ALTERNATE CPU ID FOR SECURITY VALIDATION OR BLANK =>
RUNDATE   - Future run date and time for security validation
           - SPECIFY DATE IN MM/DD/YY FORMAT OR BLANK =>
           - SPECIFY TIME IN HH:MM FORMAT OR BLANK =>

PF3/15 - RETURN
=====

```

Supply the options you want to take effect when you invoke CA JCLCheck.

### SECURITY

Y allows you to use any of the options on this panel; N deactivates these options.

### USER

User ID you want CA JCLCheck to use to perform security checking.

If you leave this field blank, CA JCLCheck uses the user ID in the JCL or the user ID of the person using CA JCLCheck.

### DASDVOL

Y causes CA JCLCheck to perform DASDVOL security checking; N deactivates this option.

### MGMTCLAS

Y causes CA JCLCheck to perform MGMTCLAS security checking; N deactivates this option.

### STORCLAS

Y causes CA JCLCheck to perform STORCLAS security checks; N deactivates this option.

**PROGRAM**

Y causes CA JCLCheck to perform program level security checking or to check if executed programs are defined to security; N deactivates this option.

**RUNCPU**

Alternate CPU ID for security validation.

Leave this field blank if you want CA JCLCheck to ignore this option.

**RUNDATE**

Run date and run time, or only a run date for CA JCLCheck to perform future security validation; N deactivates this option.

Do not specify run time only. Leave both fields blank if you want CA JCLCheck to ignore this option.

## CA JCLCheck/Invocation Menu

Use this invocation menu to run CA JCLCheck under CA Roscoe in the foreground, or in batch to scan selected JCL for syntax and runtime errors. It has three main processing options.

To display this panel, enter 1, 2, or 3 in the OPTION field on the CA JCLCheck RPF menu panel, and press Enter.

```
=====CA JCLCheck/INVOCATION MENU=====
OPTION SELECTED:
  DATE:                USER:                TIME:

DATASET NAME ==>
MEMBER          ==>                (FULL NAME, BLANK, OR *)
LIBRARY TYPE   ==> ROS            (ROS,PDS,LIB)
CPU-ID         ==>                (MULTI-CPU ONLY)
BATCH JCL      ==> Y              (Y/N; EDIT JCL FOR BATCH OPTION?)
ALLOC FILES    ==> N              (Y/N; ALLOCATE FOREGROUND FILES?)

SPECIFY ANY ADDITIONAL OPTIONS (FOR THIS RUN ONLY) BELOW
OPTIONS ==>
OPTIONS ==>
OPTIONS ==>
OPTIONS ==>
SAVED ==>
SAVED ==>

Enter - PROCESS                PF3 - RETURN
=====
```

Enter the field values and options you want to take effect when you invoke CA JCLCheck.

This panel contains the following fields:

**DATASET NAME**

Data set containing the JCL that you want CA JCLCheck to analyze.

**MEMBER**

Member name associated with the data set.

If you leave this field blank and press Enter, you are transferred to a Member Select panel. If you specify an asterisk (\*) in this field, CA JCLCheck processes the full PDS in batch (Option 3).

**LIBRARY TYPE**

Sequential file (SEQ) or the type of library: PDS, CA Panvalet, or CA Librarian.

**CPU-ID**

ID of the CPU on which the CA JCLCheck background job executes.

This value is only valid with a batch submit (Option 3).

**BATCH JCL**

Y edits the JCL for batch options; N deactivates this option.

**ALLOC FILES**

Y allocates foreground files; N deactivates this option.

**SPECIFY ADDITIONAL OPTIONS (FOR THIS RUN ONLY) BELOW**

Current CA JCLCheck profile options that are always shown using their minimum keyword name abbreviations.

This area has two types of input lines:

- You can change options on the lines labeled OPTIONS, but the changes are only in effect for this run. The dialog regenerates these options from the user profile each time you display this panel.
- CA JCLCheck saves options on the lines labeled SAVE in the user profile. The dialog regenerates these options from the user profile each time you display this panel and uses them for every job you run until you remove them.

For more information about profile options, see [CA JCLCheck Runtime Options](#) (see page 107).

## Job Scheduling Interface for CA 7 (JCKCA7)

Use this panel to specify forecasting time periods and job/system names that you want CA WA CA 7 Edition (CA 7) to process. When you press Enter, this dialog causes CA JCLCheck to submit a batch job to the internal reader for execution of the CA 7 BTI procedure. The CA 7 CA JCLCheck interface, SASSJCK0, then processes the jobs you select on this panel. CA 7 extracts the job's JCL and passes it to CA JCLCheck for validation, which treats them as related jobs.

**Note:** This does not forecast work under CA 7. It only validates JCL for the jobs and time periods specified.

To display this panel, select Option 4 (CA 7 INTERFACE INVOCATION) on the CA JCLCheck/Roscoe Menu (JCKMAINR).

```

-----JOB SCHEDULING INTERFACE FOR CA 7----- JCK10
COMMAND ==>

Date - mm/dd/yyyy           User - USER01           Time - hh:mm

FROM DATE ==> 090195       FROM TIME ==> 12:00
TO DATE ==> 090295        TO TIME ==> 12:00
CA 7 OPID ==> USER01      PASSWORD ==>

ENTER JOBS AND/OR SYSTEMS TO BE SELECTED:

Jobname ==> PAYROLL        System ==> PRD1
Jobname ==> GENLEDGE       System ==> PRD2
Jobname ==>                System ==>
Jobname ==>                System ==>
Jobname ==>                System ==>
Jobname ==>                System ==>
Jobname ==>                System ==>
Jobname ==>                System ==>
Jobname ==>                System ==>
Jobname ==>                System ==>

ENTER- BUILD JCL      PF3- RETURN TO MAIN MENU
    
```

Supply the information you want to take effect to interface CA JCLCheck with CA WA CA 7 Edition.

This panel contains the following fields:

### FROM DATE and FROM TIME

Starting date and time, using *mmdyy* format for date and *hhmm* format for time of FJOB, the forecasting command.

### TO DATE and TO TIME

Ending date and time using *mmdyy* format for date and *hhmm* format for time of FJOB, the forecasting command.

**CA 7 OPID**

Operator ID for CA WA CA 7 Edition.

**PASSWORD**

Password that displays CA WA CA 7 Edition.

**ENTER JOBS AND/OR SYSTEMS TO BE SELECTED**

Exact jobs and system names that you want forecast or a generic job name and system that resides on the CA 7 database.

You can use an asterisk (\*) as a special character to complete a string for the operating system to match for job name and system. For example, to use this panel to check an entire shift of JCL using the CA JCLCheck and CA 7 interface, specify the To and From date and time, the CA 7 OPID, and put an asterisk (\*) in the first job name.

**Note:** The data you enter in this panel is formatted into CA 7 FJOB commands. For more information about these fields, see the CA WA CA 7 Edition documentation.

## Job Scheduling Interface for CA Scheduler (JCKSSCD)

Use this panel to specify scheduling job forecasting commands to present to CA Scheduler. When you press Enter, this dialog causes CA JCLCheck to submit a batch job to the internal reader for execution of the CA Scheduler CAJUTIL0 procedure. The jobs you select with forecast commands have their JCL extracted by CAJUTIL0 and passed to CA JCLCheck for validation.

To display this panel, select Option 5 (CA Scheduler Interface Invocation) on the CA JCLCheck/Roscoe Menu (JCKMAINR).

```

----- JOB SCHEDULING INTERFACE FOR CA Scheduler-----JCKSSCD
COMMAND ==>
DATE - mm/dd/yyyy                USER - USER001                TIME - 08:21

SIMULATE FROM DATE ==> AUTOS      REPORT FROM TIME ==>
SIMULATE FOR ==> 1                REPORT TO TIME ==>
SIMULATE FROM TIME ==>           SIMWORK ==> N
REVISE ==> N                      SIMVOL ==>
STAGE JCL ==> N                   EVERYDAY ==> N

SELECT REPORT NAMES TO INCLUDE:
BACKLOG  DEVUTIL  LATEJOBS  SELECTED  UNITUTIL
DATETRAN EXPERROR RESALL  SIMEXEC

ENTER ANY CA SCHEDULER SIMULATE INFORMATION:
==>
==>
==>
==>
==>
==>
==>
==>

ENTER - SUBMIT                PF3 - RETURN

```

Supply the information you want to take effect when CA JCLCheck interfaces with CA Scheduler. For more information about scheduling commands, see the CA Scheduler documentation.

This panel contains the following fields:

**SIMULATE FROM DATE**

Starting date for simulation.

**SIMULATE FOR**

Number of days to simulate.

**SIMULATE FROM TIME**

Starting time for simulation.

**REVISE**

Y uses the current job status from the tracking record; N ignores the job status.

**STAGE JCL**

Y uses the JCL from the staging libraries; N ignores the JCL.

**REPORT FROM TIME**

Starting time for reports.

**REPORT TO TIME**

Ending time for reports.

The report interval, from time-to-time, can be smaller than simulation interval.

**SIMWORK**

DEVCLASS value that fast copies the Job Master database as a temporary work file for dynamic allocation to DEVCLASS prior to the simulation run; N if you do not have available work space to copy the Job Master database.

**SIMVOL**

VOLSER that specifies the DASD volume to allocate the simulation work file to if you specified a DEVCLASS value in the SIMWORK field.

Enter a volume serial number up to six characters.

**EVERYDAY**

Y controls the specification of RFT and RTT keywords; N ignores this request.

**SELECT REPORT NAMES TO INCLUDE**

S preceding the report name selects the report that you want CA Scheduler to produce.

**Note:** SIMEXEC is required if you want CA JCLCheck to produce CA JCLCheck error messages.

**ENTER ANY CA SCHEDULER SIMULATE INFORMATION**

Enter the CA Scheduler commands. Examples are:

```
ONLY SCHEDULE NAME=A*
ONLY SCHEDULE NAME=B*
EXCLUDE SCHEDULE NAME=ABC*
EXCLUDE JOB NAME=SAMPLE
```

## Producing JCL Flow Diagrams

The Flow Diagram report is a graphic representation of the flow of the JCL processed by CA JCLCheck. The first page lists the name of the JCL member being charted and any JOBLIB, JOBCAT, or PROCLIB data sets associated with the job. STEPLIB and STEPCAT data sets are listed at the beginning of each step. Input and output data sets, respectively, are listed on the left and right side of the process box (procstep/step/program).

For more information and to view a sample Report 10 - Flow Diagram, see [CA JCLCheck Reports](#) (see page 225). For information about using the GRAPH runtime option, see [CA JCLCheck Runtime Options](#) (see page 107).

Operational notes of interest to users are provided in the next topic.

## Operational Notes

- If there are errors or severe error messages during JCL checking, CA JCLCheck does not produce a graph. This is a safety measure that safeguards against the documentation of errors and other significant problems in the graph.
- When simulating condition code checking or restarting, CA JCLCheck only graphs those steps simulated to execute. It does not graph the skipped steps and issues a warning message.
- You can allocate a SYSGRAPH DD statement that allows use of the GRAPH option with a TSO CLIST or the CA JCLCheck ISPF Interface.

## Printer Support

You can produce the graph on line printers, nonimpact printers and on nonimpact printers that can operate as line printers.

### Line/Nonimpact Printers (Line Printer Mode)

A standard SYSOUT DD statement of the following type is required:

```
//SYSGRAPH DD SYSOUT=A
```

**Important!** Do not use OPTCD=J.

### 3800-1 Printers

Use a SYSOUT DD statement with the following:

```
//SYSGRAPH DD SYSOUT=A,  
//          DCB=(RECFM=FA,LRECL=133)
```

### 3800-3 Printers

Use a SYSOUT DD statement with the following:

```
//SYSGRAPH DD SYSOUT=A,OUTPUT=oname,  
//          DCB=(OPTCD=J,RECFM=FA,LRECL=133)
```

and an OUTPUT statement with the following:

```
//oname OUTPUT FORMDEF=0101,PAGEDEF=ppppp,  
//          CHARS=(xxxx,yyyy),TRC=YES
```

***oname***

Indicates the ddname of the output statement.

***ppppp***

Indicates one of the following page definitions:

06080 or 08081

***xxxx***

Indicates one of the following character parameters for pitch:

GT12, GT15 or GT20

***yyyy***

Indicates one of the following character parameters for font:

TSCC, TSCF or TSCK

Use only the following pairing of character parameters:

(GT12,TSCC) (GT15,TSCF) (GT20,TSCK)

You can use any 15-pitch font instead of GT15, any 12-pitch font instead of GT12, or any 20-pitch font instead of GT20.

**Examples:**

Prints the graph in landscape format at 12 cpi:

```
//oname OUTPUT FORMDEF=0101,PAGEDEF=06080,
//          CHARS=(GT12,TSCC),TRC=YES,DATAACK=BLOCK
```

Prints the graph in portrait format at 20 cpi:

```
//oname OUTPUT FORMDEF=0101,PAGEDEF=08081,
//          CHARS=(GT20,TSCK),TRC=YES,DATAACK=BLOCK
```

## 3820 Printers

Use a SYSOUT DD statement as follows:

```
//SYSGRAPH DD SYSOUT=A,OUTPUT=oname,
//          DCB=(OPTCD=J,RECFM=FA,LRECL=133)
```

and an OUTPUT statement as follows:

```
//oname OUTPUT FORMDEF=01011n,PAGEDEF=ppppp,
//          CHARS=(xxxx,yyyy),TRC=YES
```

***oname***

Indicates the ddname of the output statement.

***n***

Indicates one of the following:

0

Single-sided format

1

Normal double-sided format

2

Tumble duplex format

***ppppp***

Indicates one of the following page definitions:

V06481 or A08080

***xxxx***

Indicates one of the following character parameters for pitch:

GT15 or GT20

***yyyy***

Indicates one of the following character parameters for font:

TSCF or TSCK

Use only the following pairing of character parameters:

(GT12,TSCC) (GT15,TSCF) (GT20,TSCK)

You can use any 15-pitch font instead of GT15, any 12-pitch font instead of GT12, or any 20-pitch font instead of GT20.

**Examples**

Prints the graph in landscape format at 15 cpi:

```
//oname OUTPUT FORMDEF=010111,PAGEDEF=V06481,  
//          CHARS=(GT15,TSCF),TRC=YES
```

Prints the graph in landscape format at 20 cpi:

```
//oname OUTPUT FORMDEF=010111,PAGEDEF=V06481,  
//          CHARS=(GT20,TSCK),TRC=YES
```

Prints the graph in portrait format at 20 cpi:

```
//oname OUTPUT FORMDEF=010111,PAGEDEF=A08080,  
//          CHARS=(GT20,TSCK),TRC=YES
```

### **Sample JCL for Producing a JCL Flow Diagram**

Member AZ1GCHK in CAZ2JCL contains sample SYSOUT DD and OUTPUT statements for printing the graph output on any of the supported printers. Copy this member and edit it to suit your installation's needs.



# Chapter 3: CA JCLCheck Runtime Options

---

This chapter contains information about the CA JCLCheck runtime options. They affect a wide range of input, processing, and output functions. Each installation has its own set of defaults, and with each execution of CA JCLCheck you can change the defaults or options for that run. As the initiation of processing begins, CA JCLCheck evaluates the options in effect for that execution.

You can override the default options for each execution of CA JCLCheck in one of the following ways:

- Through the ISPF online panels
- Using the supplementary OPTIONS file (OPTIONS=) (if an OPTIONS file is present)
- Using the PARM field on the EXEC statement (PARM=)

You either select these options in the ISPF online panels, or identify them to CA JCLCheck as keyword parameters in the PARM field of the EXEC statement. You can specify the OPTIONS= option in this field (which allows you to supply additional options on a card-image file), as well as other options. For more information, see [OPTIONS](#) (see page 176).

The message output that precedes the standard CA JCLCheck reports display:

- The options that were supplied (using the OPTIONS file or the PARM field)
- Warning or error messages that are associated with those options
- A complete list of the options in effect

## Specifying Options

Specify requested options in list format, with each keyword separated from the next by a comma or blanks. You can list the options in any sequence. If an option includes an additional operand, that operand must immediately follow the option keyword and must be enclosed in parentheses.

Default options are established for each installation. With each execution of CA JCLCheck, you need only supply changes to the defaults or additional options for that run.

**Note:** You should never specify LIBRARIAN, PANVALET, or PDS options as default options.

CA provides a set of suggested defaults known as vendor-supplied defaults. These defaults vary, depending on which CA products are installed at your site.

If you only have CA JCLCheck, the vendor-supplied defaults are:

**CTLSCAN, FULLLIST, JOB, LIST, PXREF(RPT), XREF**

If you have CA JCLCheck and CA APCDOC, the vendor supplied defaults are:

**ACTION, CTLSCAN, FULLLIST, JOB, LIST, PULL, PXREF(RPT), XREF, XREFAL, SYNTAX, RUNTIME**

**Note:** Your systems programmer may have chosen an entirely different set of defaults.

CA JCLCheck evaluates the requested options at the beginning of its processing. Errors in option specification may result in CA JCLCheck bypassing the option request, substituting a default option, or terminating its execution immediately with a return code of 16. The options supplied, any messages associated with those options, and a complete list of the options in effect are displayed on the message output that precedes the standard CA JCLCheck reports.

## Active and Inactive Options (or the NO Option Prefix)

CA JCLCheck options can either be active or inactive. Active options produce some sort of action and several active options have an inactive option counterpart. This inactive option counterpart suppresses the action of the active option. Inactive option keywords begin with the prefix NO and may or may not have the same spelling as the active option word that opposes them. Initially, options are inactive. To become active, you must make them the default option.

**Note:** If an option is inactive, do not specify it in the defaults as an inactive option; however, if the option is initially active you can specify it as inactive to change its status.

The following are examples of active and inactive options:

USER NOUSER

SYNTAX NOSYNTAX

RUNTIME NORUNTIME

## Processing/Evaluation Rules

You can define options as active and inactive on three different levels (as a default, in the OPTIONS file, or in the PARM field). Due to this setup, CA JCLCheck *must* process and evaluate options one level at a time by applying three specific rules at each level:

- Turn on options that are defined as active (even if they have been defined as inactive at the previous level).
- Merge the options at this level with those options at previous levels.
- Turn off options that are defined as inactive (even if they have been defined as active at the current or previous level).

**Important!** The evaluation rules that are shown previously establish the option that is in effect for a specific JCL check. If an option and its inactive counterpart are specified on different levels, you want to make sure that you are applying the option that you need. You can test which options are in effect by running CA JCLCheck using the NOSYNTAX option, and examining the "Options in Effect" report.

Levels are processed in the following order:

1. Defaults
2. Option File
3. PARM field

The following display is an example of processing options:

```
Default = RUNTIME
OPTIONS = NORUNT
PARM =    RUNTIME
```

In this example, all three rules are applied to each level in the following manner:

The Default level:

- Rule #1—The option is turned on.
- Rule #2—There is no previous level to merge the option with.
- Rule #3—No inactive counterpart exists, so the option remains active.

The OPTIONS= level:

- Rule #1—The option was not turned off at the previous level.
- The Default and OPTIONS option (RUNTIME - NORUNT) are merged.
- The option is turned off, as specified by the inactive counterpart.

The PARM= level:

- The option (now inactive) is made active. (In other words, NORUNT on the OPTIONS= level is changed to RUNTIME.)
- The options from the previous levels are merged.
- The option remains active (or turned on) because it is now active on all three levels.

The rules allow you the flexibility to change runtime options from run to run, even if undesirable option settings have been established at a default level.

**Note:** To view the inactive options and their active counterparts, see the [Runtime Options Syntax Summary](#) (see page 214) or the [Description of Runtime Options](#) (see page 126).

## Summary of Options by Function

This section lists the CA JCLCheck options by function and provides a brief description of each option. For information about options that begin with the prefix NO, see [Active and Inactive Options \(or the NO Option Prefix\)](#) (see page 108).

### Input Specification Options

Input Specification options specify source JCL information.

#### **ALLOWMNF**

Informs CA JCLCheck to process the remaining members of an INCLUDE member list when one or more members on the list were not found.

#### **EXCLUDE**

Specifies members to exclude from the partitioned data set that is defined as the CA JCLCheck primary input.

Use this option with the PDS, LIBRARIAN, or PANVALET options.

#### **INCLUDE**

Specifies members to include from the partitioned data set that is defined as the CA JCLCheck primary input.

Use this option with the PDS, LIBRARIAN, or PANVALET options.

#### **INPUT**

Specifies an alternate ddname that describes the primary input to CA JCLCheck (for example, the job streams that you want CA JCLCheck to analyze).

The default ddname is SYSIN.

**LIBRARIAN**

Specifies to retrieve the primary input from a CA Librarian master file.

**PANVALET**

Specifies to retrieve the primary input from a CA Panvalet library.

**PDS**

Specifies to retrieve the primary input from a partitioned data set.

## Supplementary Input Options

Supplementary Input Options specify supplementary JCL input source information.

**CHANGE**

Requests that CA JCLCheck substitute variables on a temporary basis before JCL validation takes place.

**CNTLOVDD**

Specifies the DDNAME of a pre-allocated set of control card override libraries.

**CNTLOVER**

Specifies a control card override library.

**EDCHEK**

Activates support for calling CA JCLCheck from EDCHEK and is a front-end to the TSO/ISPF edit session.

**FLUSHRC**

Sets the return code for CA JCLCheck when JCL is flushed.

**JESCHEK**

Activates support for invoking CA JCLCheck from the IEFUJV Job Validation SMF exit point.

**JOBCARD**

Indicates that CA JCLCheck inserts a default JOBCARD as assembled in the "JOBCARD" CSECT.

**KEEP7380**

Causes EDCHEK to preserve the information in columns 73-80 for processing by user exits.

**OPROC**

Searches a user-defined library ahead of all normal IBM defaults.

**OPROCLIB**

Specifies an override procedure library.

**OPTIONS**

Specifies that additional options are input using a card-image file.

**PROC**

Specifies an alternate ddname for the statement defining the default procedure library.

The default ddname is SYSPROC.

**SUBCHEK**

Indicates that CA JCLCheck is being called from SUBCHEK, which is used when passing control to the TSO SUBMIT command.

**SYSCHEK**

Activates support for invoking CA JCLCheck from the IEFUJV Job Validation SMF exit point and is used for compatibility purposes.

## Output Options

Output Options specify additional JCL output processing information.

**EDCHKLB**

If you activate this option, all JCL statements in error contain a generated ISPF label. All labels contain a period (.) followed by five alphabetic characters identifying the line in error. You can locate the next line in error by entering the LOCATE LABEL command.

This option is only valid with the EDCHEK runtime option.

**EDCHKSL**

Produces: additional information on the number of statements in a job by severity, a message summary about each message, and the message's associated relative line number. Additionally, if the EDCHKLB option is in effect, a generated ISPF label displays in a list for the lines in error.

This option is only valid with the EDCHEK runtime option.

**RDR**

Specifies an alternate ddname to use with the output job streams generated by the SUBMIT option.

The default ddname is SYSRDR.

**SUBMIT**

Requests the automatic submission of error-free job streams.

## Validation Options

Validation Options specify how to validate the scanned JCL.

### **BYPROUT**

Bypasses all processing of the JES3 Route card.

### **BYPRUNT**

Selects runtime checks to bypass.

### **COBEXIT**

Activates COBOL User Exit facility.

### **COND**

Activates condition code and IF/THEN/ELSE/ENDIF simulation.

### **CSI**

Requests the use of the IBM Catalog Search Interface when doing catalog lookup and data set existence checking.

### **CTLSCAN**

Requests that CA JCLCheck read and interpret control statements for certain IBM, CA, and user-defined utility programs.

### **DESTCHK**

Requests validation of remote destination names on DD statements and JES control statements.

### **DYNEXITS**

Specifies the names of the assembler exits for CA JCLCheck to load dynamically.

### **JESCRUNT**

Indicates that the JESCHECK option should perform runtime checks. This option is only valid in a JESCHECK environment.

### **JCLLIB**

Controls whether CA JCLCheck uses the default PROCLIBs instead of the procedure directories defined in the JCLLIB statement.

### **LOCATE**

Causes CA JCLCheck not to look up data sets in the system catalog when you code UNIT and VOLSER data on the DD statement.

### **NJBCHK**

Affects the validation of JCL following an NJB statement that is preceded by a route XEQ statement.

**Note:** NJB is a JCL option for JES3 users only.

#### **PREVEX**

Suppresses the expansion of library procedures and includes.

#### **PROCDFLT**

Causes CA JCLCheck to search the default PROCLIB defined by the SYSPROC DD statement when CA JCLCheck does not find a cataloged procedure in the alternate PROCLIB.

**Note:** JES2 alternate PROCLIBs are defined in the job stream using a /\*JOBPARM PROCLIB=ddname statement.

#### **PVTCAT**

Causes CA JCLCheck to look up data sets in the catalog again when the step it is analyzing has a STEPCAT DD statement.

#### **PXREF**

Activates the program and member existence checking and reporting feature.

#### **RUNTIME**

Requests the runtime checking of JCL.

#### **SCHENV**

Activates validation of the scheduling environment.

#### **SPACE**

Requests DASD space efficiency analysis.

#### **STACK**

Suppresses message CAY6158W when the file sequence count for DD statements that are stacking files on the same tape do not occur in sequential order.

#### **STANDARD**

Names the standard to use for scanning JCL.

You define the standard name using Job Control Standards.

#### **STDREXX**

Specifies the REXX EXEC member name.

CA JCLCheck fetches the member from the default library of the installation and it is specific to the site. (The IBM default is SYSEXEC DD.)

#### **SUBALLOC**

Eliminates spurious CAY6066E messages that CA JCLCheck issues for VSAM data sets, which are sub-allocations of a VSAM data space.

#### **SYNTAX**

Requests syntax checking of JCL.

**USERDATA**

Specifies user-defined data that is passed to exits and the REXX interface.

## Environment Specification Options

Environment Specification Options specify alternate processing activities under different environments.

**AUTOPROC**

Extracts the cataloged procedure library (PROCLIB) definitions from the JES2 or JES3 address space.

**CNTL**

Activates support for program control statements in the input stream.

**DBCS**

Parses double-byte character sets for IBM, Hitachi, and Fujitsu systems.

**DEQUE**

Frees selected enqueued data sets under CA Roscoe.

**FACOM**

Activates support for executing CA JCLCheck on a Hitachi or Fujitsu system.

**HCD**

Obtains UCB addresses for both static and dynamically-defined devices for users utilizing HCD (Hardware Configuration Definition).

**INDEX**

Restricts catalog lookup and data set existence checking to the specified high-level indexes.

**INTUSR**

Initializes all four user words (\$USER-\$USER4) in the GBL data area from the address pointed to by parameter 4 when CA JCLCheck is called by a user program.

**JES3**

Specifies JES3 statements as valid JCL and performs both syntactical and runtime validation.

**JOBTYP**

Specifies the type of JCL to scan JOB, started task, or time-sharing user.

**MSS**

Requests the VTOC of mass storage volumes when necessary to verify the existence of a data set on MSS.

**PASSWORD**

Specifies the password to use for accessing read protected VSAM catalogs.

**PREDNO**

Scans the system without building a predecessor list (only used with CA APCDOC).

**PREFIX**

Specifies the ddname prefix to use for dynamic allocation in non-z/OS systems.

**REMOTE**

Activates validation on remote servers.

**REINDEX**

Restricts catalog lookup and data set existence checking by ignoring the specified high-level indexes that begin with the value specified.

**ROSCOE**

Invokes CA JCLCheck within the CA Roscoe address space.

**SCHEDLER**

Requests basic support when running CA JCLCheck under CA Scheduler.

**SMS**

Specifies whether to simulate an SMS environment.

**SRCH**

Searches all libraries on a JOBLIB or STEPLIB concatenation, even if one of the libraries is a temporary data set created during the job being analyzed.

**SYSUID**

Specifies the user ID value to use in replacement of the system symbolic parameter &SYSUID.

**XREFAL**

Causes CA JCLCheck to save all references to a data set in a job in the corresponding cross-reference database record.

## System/Related Job Analysis Options

System/Related Job Analysis Options specify handling of multiple jobs.

### **PDSRELATE**

Specifies that CA JCLCheck treat the primary input library as a series of consecutive, related job streams.

### **SXREF**

Activates related job data set reference checking and reporting.

## Product Support Options

Product Support options specify options for use with other program products.

### **ACF2**

Specifies ACF2 statements as valid JCL and performs both syntactical and runtime validation.

### **ACTION**

Requests a job action report.

This option is only valid when running CA JCLCheck with CA APCDOC.

### **APPEND**

Specifies the job (in CA APCDOC) where jobs that are being added can be inserted before or after.

### **ASM2**

Requests support of the CA ASM2 storage manager.

### **CA11**

Automatically activates and deactivates the SUPCAT option, CAY6081E message suppression to indicate that a job is being rerun.

### **CA7**

Requests support, and provides criteria, for processing CA WA CA 7 Edition scheduled override statements.

### **CA7CTL**

Enables support for CA WA CA 7 Edition control statements.

### **CADRIVER**

Requests toleration support for CA DRIVER commands that appear in JCL when running CA WA CA 7 Edition or CA Scheduler.

**DB2**

Automatically activates and deactivates proper validation of DB2 plans, tables, and subsystems.

**DBHLQ**

Accesses the PARMFILE associated with the CA APCDOC Cross-Reference database. This high-level qualifier is used for CA APCDOC Version 1.1 and above.

**DELTA**

Limits the scanning and updating of job streams and library members to those that have changed since the last database update on the Cross-Reference database.

**DICT**

Accesses the installed CA APCDOC documentation database to obtain information about predecessor jobs to the one being analyzed.

**DICTDD**

Defines the ddname that is used for the CA APCDOC Cross-Reference database.

**DIRUPD**

Tells CA APCDOC to create a current date and time stamp for members that do not have one and gives it the user ID APCDOC so that you know that CA APCDOC updated the member. This option is used with the DELTA option.

**DISPATCH**

Requests that CA Dispatch information in the JCL being checked is printed in Report 5 - Report Report.

**EASYPROC**

Activates CA Easyproclib.

**EASYRDR**

Activates CA Easyproclib. This option is the same as the EASYPROC option.

**IAM.**

Activates recognition of Innovation Access Method data sets.

**IMS**

Activates support for the IBM database product (Information Management System).

**JOBJBRAM**

Passes a job list name when used with CA APCDOC.

**LIBLIST**

Enables CA APCDOC to process multiple JCL libraries at one time.

**MIGVOL**

Specifies a volume name that CA JCLCheck uses to identify data sets that storage management products archive or migrate.

**NETWORK**

Limits the network name that is associated with the SYSTEM option. This option is only available when invoked from CA JCLCheck with CA APCDOC installed.

**ONEMEM**

Forces the ending member name to be the same as the starting member name. Occurs only when the starting member name is specified for PDS, CA Panvalet, or CA Librarian.

**PANEXEC**

Requests support of the CA Panexec transparent loader.

**PRCDELTA**

Limits the CA APCDOC scanning of job streams and the updating of those jobs whose cataloged procedures have changed since the last database update on the Cross-Reference database.

**PULL**

Requests a tape pull list of tapes used in jobs being checked.

This option is only valid when running CA JCLCheck with CA APCDOC.

**REPORT**

Provides the same functionality as the RPTRPT option but is only available if invoked from CA APCDOC without the full version of CA JCLCheck.

**RESOLVE**

Requests CA JCLCheck to resolve external variables of a scheduling product imbedded in the JCL.

**ROSLIMIT**

Specifies the maximum number of JCL statements to process when CA JCLCheck is run as a CA Roscoe monitor routine.

**SECURITY**

Activates the standard security interface.

This interface supplies information about a resource and its use that your security system defines. This option works with CA ACF2, CA Top Secret, or RACF.

**SUBLIB**

The SUBLIB option allows the SUBMIT option to be specified together with the PDS, PANVALET, or LIBRARIAN option so that CA JCLCheck submits jobs from a library.

#### **SYSID**

Requests that a CPU ID associated with the job is placed on the CA APCDOC Database.

This option is only valid when running CA JCLCheck with CA APCDOC.

#### **SYSTEM**

Identifies a collection of jobs that reside in the CA APCDOC database.

#### **TLMS**

Requests support of the CA TLMS tape library management system.

#### **TMS**

Activates CA 1 support for tape data set validation.

#### **TSOE**

Activates TSOLINK (a TSO Service Facility for EDCHEK), SUBCHEK, ISPF, and CA JCLCheck as a Command Processor.

You only use TSOLINK (only available under the IBM product TSO EXTENSIONS (TSO/E)) to obtain the authorization that is required for certain types of options, such as AUTOPROC.

#### **VIOPLUS**

Activates recognition of the VIOCTL DD statement.

#### **VSAM**

Requests support of VSAM catalog management.

#### **XEROX**

Activates the XEROX XPAF output statements.

## **Security Validation Options**

Security Validation options specify options to use when security pre-validation is in effect.

#### **RUNCPU**

Specifies a CPU ID to use for the security validation process.

#### **RUNDATE**

Specifies a future run date and time to use to determine if access is granted at that time.

See the [DATEFMT](#) (see page 144) option for information about specifying European date format.

**Note:** RUNDATE is only valid for CA ACF2 r6.0 users.

**SIGNON**

Uses the current user ID and makes that ID the current user for the processing duration of the job. Controls the use of the alternate user ID feature.

**USER**

Specifies the user ID to use with security checking.

## Diagnostic Options

Diagnostic options specify options to use for abnormal processing.

**DBGJCL**

Turns on JCL debugging. Use at the request of CA Support.

**DEBUG**

Generates debugging information.

**DUMPUCB**

Requests a formatted dump of the portion of the system nucleus containing UCB pointers and control blocks.

**NOSPIE**

Suppresses the SPIE function, so that CA JCLCheck abnormally terminates if there is an OCx interrupt.

## Report File Options

Report File options specify files to use for reporting.

**LIST**

Writes reports to the primary report file (normally SYSPRINT).

**SUPRPT**

Conditionally suppresses report generation that is based on the highest severity specified.

**TERM**

Generates the terminal report file containing only the erroneous statements and their associated messages (normally SYSTERM).

## Message Control Options

Message control options control when various messages can be issued.

### **BLKMSG**

Eliminates the CAY6087W (BLKSIZE NOT SPECIFIED) message for DD statements within procedures.

### **DCBCHK**

Suppresses message CAY6086E for SYSOUT data sets.

### **DSNERR**

Issues error message CAY6079E DATASET <data set name> SPECIFIED AS OLD OR SHR, BUT CANNOT BE FOUND each time that <data set name> is referenced.

### **IEBMSG**

Issues the CAY6119W message when the SYSIN DD statement is omitted when executing IEBCOPY regardless of the release level of the operating system.

### **MCOSYS**

Activates System message control option list. This option allows the user or administrator to suppress messages or set message severity.

### **MCOUSR**

Activates User message control option list. This option allows the user or administrator to suppress messages or set message severity.

### **MSGLEVEL**

Permits suppression of informational messages on the SYSTERM file.

### **PNFMSG**

Forces message CAY6093E PROGRAM NOT FOUND if a program or an EXEC statement is found to exist in a library mentioned in the CA JCLCheck STEPLIB/JOBLIB.

### **REXXMSG**

Enables the user or administrator to process messages and change message severity via the REXX EXEC specified in the STDREXX option.

### **STEPMSG**

Allows the CAY6128W message to be issued for duplicate step names within a job.

### **STPCAT**

Suppresses the CAY6162W message indicating the STEPCAT DD should be the first DD statement in the step.

**SUPCAT**

Suppresses messages CAY6081E, CAY6080W, and CAY6129E when a DD has DISP=(,CATLG) for a data set that is already cataloged.

**USRMSG**

Issues the CAY6088E message only when USER is coded on a JOB statement without the PASSWORD parameter.

**UTLSEV**

Downgrades messages CAY6060 through CAY6085 to warnings (SEV=4) when issued for utility control statements.

## Report Format Options

Report format options specify alternate processing options for reports.

**CCLIST**

Lists the first few statements of every input stream data set (such as utility control statements).

**DATEFMT**

Specifies European date format (dd/mm/yy).

Specify this option before the RUNDATE option to use the date in European date format.

**EDCHEL**

Enables the expansion of Procedure and Include statements when the EJCK macro is invoked from EDCHEK.

**ERROR**

Generates Report 6 - Error Messages that specify location of error messages on the CA JCLCheck reports.

**FULLLIST**

Generates Report 2 - Listing of Merged JCL Report.

**JOBSEV**

Sets the return code to the highest error message severity of all jobs CA JCLCheck analyzes in this run.

Otherwise, the return code is set to the highest error message severity of the last job CA JCLCheck analyzes in this run.

**LINECNT**

Specifies the number of detail lines to print on each report page.

**OUTFMT**

Specifies reports are to be generated in alternative formats.

**SEV**

Specifies the minimum severity level of errors to include on reports.

**KILLSEP**

Prevents the printing of the separation lines between each JCLCheck report.

## Standard Report Options

Standard Report options specify the production of standard reports.

**AUDIT**

Produces Audit Trail Report

**JOB**

Produces Report 1 - Listing of Jobstream JCL Report

**PROCXREF**

Produces Report 11 - Symbolic Cross-Reference Report

**PXREF(RPT)**

Produces Report 4 - Program Cross-Reference Report

**RPTRPT**

Produces Report 5 - Report Listing

**XREF**

Produces Report 3 - Data Set Cross-Reference Report

## Supplementary Report Options

Supplementary Report options specify the production of special reports for use with optional analysis features.

**GRAPH**

Produces the Report 10 - Flow Diagram Report, and, optionally, provides the ddname of the SYSOUT file.

**SPACE**

Activates the direct-access space efficiency analysis and reporting feature.

**SXREF**

Activates the related job data set existence checking and reporting feature, based on other options, it can produce Reports 7, 8, and 9. For a complete description of all CA JCLCheck reports, see [CA JCLCheck Reports](#) (see page 225). The following list summarizes the correspondence of the various reports and their controlling options.

**AUDIT**

Audit Report

**JOB**

Report 1 - Listing of Jobstream JCL

**FULLLIST**

Report 2 - Listing of Merged JCL

**XREF**

Report 3 - Data Set Cross-Reference

**PXREF**

Report 4 - Program Cross-Reference

**RPTRPT**

Report 5 - Report Listing

**ERROR**

Report 6 - Error Messages

**SXREF/XREF**

Report 7 - Summary Data Set Cross-Reference

**SXREF/PXREF**

Report 8 - Summary Program Cross-Reference

**SXREF/RPTRPT**

Report 9 - Summary Report

**GRAPH**

Report 10 - Flow Diagram Report

**PROCXREF**

Report 11 - Procedure Cross-Reference

## Description of Runtime Options

This section describes CA JCLCheck runtime options in alphabetical order. Where appropriate, the list includes active and inactive option keywords. Each of the following options is available in Batch, ISPF (EDCHECK, SUBCHECK, and TSO Foreground) as well as JESCHECK unless noted otherwise.

### ACF2

The ACF2 option specifies syntactical checking and runtime validation for ACF2 JCL.

This option has the following format:

ACF2|NOACF2

**Default:** If ACF2 is active, ACF2 is the default; otherwise, NOACF2 is the default.

The ACF2 option is optional. CA JCLCheck attempts to dynamically determine the presence of ACF2 on the active system.

### ACTION

The ACTION option produces a job action report. This option is only valid when running CA JCLCheck with CA APCDOC.

This option has the following format:

ACTion|NOACTion

**Default:** NOACTion

## ALLOWMNF

The ALLOWMNF option is directly related to the INCLUDE, PDS, LIBRARIAN, and PANVALET options. The ALLOWMNF option informs CA JCLCheck to process the remaining members of an INCLUDE member list when one or more members on the list were not found. Without this option, CA JCLCheck aborts validation processing when a member is not found.

When ALLOWMNF is specified and one or more members on the INCLUDE list are not found, the CAY6180 message is changed to a warning message. Also, when ALLOWMNF and SXREF are specified and one or more members on the INCLUDE list are not found, the CAY6632W message is also displayed to warn users of unexpected validation results.

This option has the following format:

```
ALLOWmnf/NOALLmnf
```

Default: NOALLmnf

## APPEND

The APPEND option specifies the job before which or after which to put jobs that are being added.

This option is only valid when running CA JCLCheck with CA APCDOC.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

```
APPEND (AFTER, jobname) | NOAPPEND  
      (BEFORE, jobname)
```

**Default:** NOAPPEND

Using this option causes CA APCDOC to add the jobs for the current run instead of rebuilding the entire system. If the job name is omitted, the jobs are added before the first job or after the last job in the system.

## ASM2

The ASM2 option requests CA JCLCheck support CA ASM2.

This option has the following format:

ASM2|NOASm2

**Default:** If CA ASM2 is active, you need to specify the ASM2 option; otherwise, this option is not active.

This option causes CA JCLCheck to access the Intelligent Transparent Restore (IXR) to obtain information for archived data sets.

The ASM2 option requires that CA JCLCheck be properly authorized by APF.

## AUDIT

The AUDIT option produces:

- A statistical summary of jobs, steps, and DD statements
- Input records of jobs, steps, and DD statements
- Averages of jobs, steps, and DD statements

This option has the following format:

AUDit|NOAUDit

**Default:** NOAUDit

## AUTOPROC

The AUTOPROC option extracts the cataloged procedure library (PROCLIB) from the JES2 or JES3 address space. With this option activated, it is not necessary to maintain PROCLIB allocation information in the CA JCLCheck cataloged procedures, CLIST, or SUBCHEK/EDCHEK control files.

This option has the following format:

AUtoproc|NOAUtopr

**Default:** NOAUtopr

When AUTOPROC is specified in a batch execution of CA JCLCheck, the SYSPROC and other PROCLIB allocation DD statements normally required in the execution JCL **must** be omitted.

When AUTOPROC is specified in a SUBCHEK or EDCHEK control file, the ddnames used in the JES2 procedure (for example, PROC00, PROC01) or JES3 procedure (for example, IATPLBST) must not already be allocated to the TSO session.

- If the allocation of the procedure libraries is successful, these libraries are searched for the procedures. The ddnames that are supplied by the PROC option are ignored.

If the allocation of the procedure libraries is unsuccessful, the CAY6182S message is issued and the ddname that is supplied by the PROC runtime option (if allocated) is the target of the procedure library search.

Specifically, AUTOPROC is in effect unless there is an error message indicating the allocation failed.

- The AUTOPROC option requires that CA JCLCheck is APF authorized.
- The AUTOPROC option is not available when executing CA JCLCheck under VM/CMS or CA Roscoe Version 5.6 and below.
- The AUTOPROC option is not available on Fujitsu MSP operating systems.
- The AUTOPROC and NORUNTim options are mutually exclusive with the following four exceptions:
  - CA JCLCheck Common Component with CA WA CA 7 Edition
  - CA JCLCheck Common Component with CA Scheduler
  - CA JCLCheck Common Component with CA WA ESP Edition
  - CA JCLCheck Full Product
- The AUTOPROC option on JES3 is supported only on the master system that owns the PROCLIBS.
- The AUTOPROC option on JES3 supports only static defined proclibs.
- The AUTOPROC option on JES2 supports static proclibs, JOBCLASS definitions, and DYNAMICALLY added PROCLIBS that are supported by z/OS 1.2 and above.
- The AUTOPROC option uses the DDNAMES as specified in JES2, such as, PROC00, PROC01... Care must be taken so as not to duplicate allocations to the same DDNAME when using the EDCHKDD ALLOC commands. If a duplicate allocation is detected, message CAY6300 is issued with a DAIR code of 0410. When allocating procedure libraries, use either AUTOPROC option or the EDCHKDD ALLOC commands, not both.

Also, review the related option DFLTJOB (Default Job Class).

## BLKMSG

The BLKMSG option suppresses message CAY6087W message (BLKSIZE NOT SPECIFIED FOR NEW OR DUMMY DATASET) for DD statements in library and instream procedures.

This option has the following format:

BLKMsg | NOBLkmsg

## BYPROUT

The BYPROUT option bypasses all processing of the JES3 Route card.

This option has the following format:

BYPRout | NOBYprou

## BYPRUNT

The BYPRUNT runtime option lets you select runtime checks to bypass. The BYPRUNT runtime option has 12 defined sub-options that enable individual runtime checks to be bypassed, that is, not done. All other runtime checks are made.

This option has the following format:

BYPRUnt(*suboption1, suboption2, suboption3, . . . suboption12*) | NOByprun

### ***suboption***

Specifies the runtime check to bypass.

**Default:** NOByprun

You can specify the following suboptions:

- CTL—Bypass utility control card checking.
- DB2—Bypass DB2 runtime checking.
- DSE—Bypass data set existence checking, including catalog management and tape management. This sub-option also bypasses MEM and PGM runtime checking.
- IMS—Bypass IMS runtime checking.
- JES—Bypass JES runtime checking.
- MEM—Bypass member existence checking (if DSE is not specified).
- PGM—Bypass program existence checking (if DSE is not specified).
- RGN—Bypass region size checking.

- SCE—Bypass SCHENV checking.
- SMS—Bypass SMS runtime checking.
- USS—Bypass USS runtime checking.
- VOL—Bypass volume and unit runtime checking.

**Usage notes:**

- The BYPRUNT and NORUNTIM options are mutually exclusive.
- The SECURITY option handles all security-related runtime checks. The SECURITY option is not part of or related to the BYPRUNT option.
- The following BYPRUNT sub-options are mutually exclusive with these CA JCLCheck options:
  - The DSE sub-option and the TLMS and TMS options.
  - The IMS sub-option and the IMS option.
  - The JES sub-option and the DESTCHK option.
  - The SCE sub-option and the SCHENV option.
  - The SMS sub-option and the SMS option. If the SMS sub-option is specified, CA JCLCheck does not use the active SMS configuration and does not simulate SMS processing.

**Example: Bypass Runtime Checks**

In this example, the DB2 and USS runtime checks are bypassed while executing all other runtime checks:

```
BYPRUNT(DB2,USS)
```

**Example: Specify the BYPRUNT Runtime Option under EDCHEK**

When running CA JCLCheck under EDCHEK, you have the flexibility to specify the BYPRUNT runtime option in one of two places:

- You can specify %EJCK BYPRUNT(suboption) on the ISPF EDIT command line.
- You can specify the BYPRUNT runtime option in the EDCHEKDD file as follows:

```
EDCHEK PROC(PROC00)          -
      LIST(EDCPRINT)         -
      BYPRUNT(DB2,USS)
ALLOC FI(PROC00) DA(SYS1.PROCLIB)
ALLOC FI(*) DA(SYS2.PROCLIB)
ALLOC FI(*) DA(USER.PROCLIB)
GO
```

**Example: Specify the BYPRUNT Runtime Option in Batch**

This example shows how to specify the BYPRUNT runtime option in a batch job:

```
//USERID1J JOB '1234', 'DEVELOPER', MSGLEVEL=(1,1), CLASS=A,
//          MSGCLASS=X, REGION=0M, COND=(0,LT), NOTIFY=&SYSUID
//JCLCHECK EXEC PGM=JCLCHECK, PARM='OPTIONS(JCLOPTS)', COND=(0,LT)
//STEPLIB DD DSN=FID.JCK.CAILOAD, DISP=SHR
//          DD DSN=FID.JCK.CAIPLD, DISP=SHR
//SYSTEM DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPROC DD DSN=SYS1.PROCLIB, DISP=SHR
//SYSIN DD DSN=USERID1.JCK.CNTL(JCKINPUT), DISP=SHR

//JCLOPTS DD *
//          BYPRUNT(DB2,USS)
//
```

**Note:** For information about using the BYPRUNT runtime option in ISPF, see the [Runtime Options Panel \(JCK14\)](#) (see page 69).

## CA11

The CA11 option automatically activates and deactivates the SUPCAT option to indicate that a job is being rerun. This option causes CA JCLCheck to suppress error messages for error conditions that CA WA Restart Option for z/OS Schedulers prevents. The SUPCAT option is turned off at each job statement and turned on when a step that executes CA WA Restart Option for z/OS Schedulers is encountered in the JCL.

If CA WA CA 7 Edition is supplying the CA WA Restart Option for z/OS Schedulers step at submission time, activate the SUPCAT option and *do not* use this option.

This option has the following format:

```
CA11(stepname.procstep)|NOCA11
```

*stepname* and *procstep* are optional. *stepname* and *procstep* are the names of the CA WA Restart Option for z/OS Schedulers restart your installation uses. If you omit this information, CA JCLCheck does not recognize refer backs to the CA WA Restart Option for z/OS Schedulers restart procedure.

## CA7

This CA7 option requests support of and provides criteria for processing CA 7 scheduled override statements.

This option has the following format:

```
CA7([schedule-id] [date] [time])
```

The *schedule-id* operand is optional. If omitted, it defaults to one (1) and the test of the ID of all #JI, #JO, #XI, or #XO statements is considered true.

The Julian date (*yyddd*) and time (*hhmm*) operands are optional. If you omit these operands, CA JCLCheck uses the current system date and time. #JI, #JO, #XI, or #XO statements are considered true if this date falls between the begin date/time (BD,BT) and end date/time (ED,ET) parameters.

#JI, #JO, #JEnd, #XI, #XO, #XEnd, #VER, #HLD, #MNT, #MSG, #NOX, #NTR, #JCL, #RES, and #SCC, JCL statements in the input file are conditionally included or omitted based on testing of the ID, BD, BT, ED, and ET parameters of the #JI, #JO, #XI, and #XO statements.

For example, CA7(1 93214 1900) requests that CA JCLCheck process the input and include or exclude JCL based on a schedule ID of 1, as if the job were to run on Friday, August 2, 1998, at 7:00 p.m.

## CA7CTL

The CA7CTL option enables support for CA WA CA 7 Edition control statements (they start with the pound (#) sign).

This option has the following format:

```
CA7ctl|NOCA7ctl
```

If this feature is not specified, these statements are treated as in-stream data and a //SYSIN DD \* statement is generated. If this option is specified and the CA WA CA 7 Edition runtime option is not specified, the "#" statements are treated as comments.

## CADRIVER

The CADRIVER option requests basic support for CA DRIVER commands that may appear in JCL when running CA WA CA 7 Edition or CA Scheduler.

This option has the following format:

CADriver|NOCADriv

**Default:** NOCADriv

## CCLIST

This CCLIST option requests a listing of the first few cards of every input stream data set (such as utility control statements). The listing appears on the Listing of Merged JCL Report, Report #2.

This option has the following format:

CCList[(*number-of-cards*)]|NOCCList

***number-of-cards***

(Optional) Specifies the number of cards to print.

**Default:** CCList(5)

**Limits:** 999

**Example: List ten cards**

This example requests that CA JCLCheck list the first ten cards of every input stream.

CC(10)

## CHANGE

The CHANGE option requests that CA JCLCheck substitute variables on a temporary basis before JCL validation takes place.

Immediately after each input JCL or inline control statement has been read, the CHANGE runtime option causes the input line to be examined against a user-defined conversion table that contains one or more 'from' and 'to' data strings. If a match is found, the 'from' data string is converted to the 'to' data string. For a detailed description of the conversion table, see Pre-Validation Variable Substitution in the *Programming Guide*.

This option has the following format:

```
CHAnge( target , dsname[ , member ] ) | NOCHAnge
```

**Default:** NOCHAnge

**target**

The target sub-option informs CA JCLCheck where to search for and implement CHANGE conversions. This is a required sub-option.

**JCL**

Indicates JCL statements only (including comment lines).

**CTL**

Indicates inline control statements only.

**BOTH**

Indicates JCL and inline control statements.

**dsname**

The dsname sub-option is the LRECL 80 sequential file or PDS containing the user-defined conversion table. This is a required sub-option.

**member**

The member sub-option is the LRECL 80 PDS member containing the user-defined conversion table. If the dsname sub-option is a PDS, then the member sub-option is a required sub-option. If the dsname sub-option is a sequential data set, then the member sub-option must not be specified.

**Example: Using the CHANGE option**

The following example shows the CHANGE option using the target sub-option of BOTH, and specifying a PDS and member name:

```
CHANGE ( BOTH , USERID1 . JCK . CNTL , CHGMEM )
```

## CNTL

The CNTL option activates support for program control statements in the input stream.

This option has the following format:

```
CNTL | NOCNTL
```

**Default:** NOCNTL

CA JCLCheck performs syntax checking of CNTL and NOCNTL JCL statements for a subsystem (that is, the IBM Print Services facility, PSF).

## CNTLOVDD

The CNTLOVDD runtime option lets you specify the DDNAME of a pre-allocated set of control card override libraries. This runtime option is mutually exclusive with the CNTLOVER runtime option.

This option has the following format:

CNTLOVDD(*ddname*) | NOCNTLDD

### ***ddname***

Specifies the DDNAME of a pre-allocated set of control card override libraries. The control card override libraries have already been allocated under the DDNAME specified before CA JCLCheck is invoked.

### **Usage notes:**

- All data sets in the CNTLOVDD concatenation must be DSORG=PO files (that is, PDS or PDSE). You can use simulated PDS files in the concatenation, for example, CA Librarian with SUBSYS=LAM.
- The CNTLOVDD runtime option must be used with CTLSCAN.
- When the control card member is not found in the CNTLOVDD concatenation, the library member referenced in the utility CONTROL CARD DD is used.
- When the utility CONTROL CARD DD statement in the JCL being checked does not point to a PDS member, the feature is not activated for that DD statement.
- When the control card member is validated from an override library, a message informs you of the library the control cards came from.

### **Example: Specify the CNTLOVDD Runtime Option under EDCHEK**

When running CA JCLCheck under EDCHEK, you have the flexibility to specify the CNTLOVDD runtime option in one of two places:

- You can pre-allocate the override concatenation using a private CLIST, REXX, or the TSO logon procedure, and then specify %EJCK CNTLOVDD(*ddname*) on the ISPF EDIT command line.
- You can also specify the CNTLOVDD option in the EDCHEKDD file. The following example shows how to use the CNTLOVDD option in the EDCHEKDD file:

```
EDCHEK AUTOPROC          -
      LIST(EDCPRINT)      -
      CNTLOVDD(CNTLCRDS)
ALLOC FI(CNTLCRDS)  DA(MY.CNTLCRDS)
GO
```

**Example: Specify the CNTLOVDD Runtime Option in Batch**

This example shows how to specify the CNTLOVDD runtime option in a batch job. Before CA JCLCheck validates the IDCAMS SYSIN DD data set PROD.CNTL.CARDS(PRDCARDS), CA JCLCheck looks for the PRDCARDS member in each library of the override concatenation and uses the first one found for CA JCLCheck validation. If not found in any of these libraries, the PROD.CNTL.CARDS library member PRDCARDS is used for CA JCLCheck validation.

```
//USERID1J JOB '1234', 'DEVELOPER', MSGLEVEL=(1,1), CLASS=A,
//          MSGCLASS=X, REGION=0M, COND=(0,LT), NOTIFY=&SYSUID
//JCLCHECK EXEC PGM=JCLCHECK, PARM='OPTIONS(JCLOPTS)', COND=(0,LT)
//STEPLIB DD DSN=FID.JCK.CAIBLOAD, DISP=SHR
//          DD DSN=FID.JCK.CAIPLD, DISP=SHR
//SYSTEM DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPROC DD DSN=SYS1.PROCLIB, DISP=SHR
//CNTLCRDS DD DISP=SHR, DSN=QA.CNTL.CARDS
//          DD DISP=SHR, DSN=TEST.CNTL.CARDS
//          DD DISP=SHR, DSN=DEVL.CNTL.CARDS
//SYSIN DD DATA, DLM=FF
//IDCAMSTP EXEC PGM=IDCAMS
//SYSIN DD DISP=SHR, DSN=PROD.CNTL.CARDS(PRDCARDS)
//SYSPRINT DD SYSOUT=*
FF
//JCLOPTS DD *
          CNTLOVDD(CNTLCRDS)
          CTLSCAN LIST
//
```

**Note:** For information about using the CNTLOVDD runtime option in ISPF, see the [Control Card Override Panel \(JCK12\)](#) (see page 67).

**CNTLOVER**

The CNTLOVER runtime option lets you specify a control card override library. This runtime option is mutually exclusive with the CNTLOVDD runtime option.

This option has the following format:

```
CNTLOVER(library[,C])|NOCNTLER
```

***library***

Specifies the data set name of a control card override library. CA JCLCheck allocates the specified data set.

**C**

Indicates to issue an information message and continue validation processing when the library is not found.

**Usage notes:**

- You can specify multiple CNTLOVER options; the libraries are concatenated in the order they are specified.
- The libraries must be cataloged PDS or PDSE libraries.
- The CNTLOVER runtime option must be used with CTLSCAN.
- When the control card member is not found in the CNTLOVER concatenation, the library member referenced in the utility CONTROL CARD DD is used.
- When the utility CONTROL CARD DD statement in the JCL being checked does not point to a PDS member, the feature is not activated for that DD statement.
- When the control card member is validated from an override library, a message informs you of the library the control cards came from.

**Example: Bypass a Missing DSN**

In this example, CA JCLCheck attempts to locate TEST.CNTLCRDS. If the library is not found, CA JCLCheck displays an information message and validation processing continues.

```
CNTLOVER(TEST.CNTLCRDS,C)
```

**Example: Specify the CNTLOVER Runtime Option under EDCHEK**

When running CA JCLCheck under EDCHEK, you have the flexibility to specify the CNTLOVER runtime option in one of two places:

- You can specify %EJCK CNTLOVER(library) on the ISPF EDIT command line.
- You can specify the CNTLOVER runtime option in the EDCHEKDD file as follows:

```
EDCHEK PROC (PROC00)          -
      LIST (EDCPRINT)         -
      CNTLOVER (QA.CNTL.CARDS) -
      CNTLOVER (TEST.CNTLCRDS,C) -
      CNTLOVER (DEVL.CNTL.CARDS)
ALLOC FI (PROC00)    DA (SYS1.PROCLIB)
ALLOC FI (*)         DA (SYS2.PROCLIB)
ALLOC FI (*)         DA (USER.PROCLIB)
GO
```

**Example: Specify the CNTLOVER Runtime Option in Batch**

This example shows how to specify the CNTLOVER runtime option in a batch job. Before CA JCLCheck validates the IDCAMS SYSIN DD data set PROD.CNTL.CARDS(PRDCARDS), CA JCLCheck looks for the PRDCARDS member in each library of the override concatenation and uses the first one found for CA JCLCheck validation. If not found in any of these libraries, the PROD.CNTL.CARDS library member PRDCARDS is used for CA JCLCheck validation.

```
//USERID1J JOB '1234', 'DEVELOPER', MSGLEVEL=(1,1), CLASS=A,
//          MSGCLASS=X, REGION=0M, COND=(0,LT), NOTIFY=&SYSUID
//JCLCHECK EXEC PGM=JCLCHECK, PARM='OPTIONS(JCLOPTS)', COND=(0,LT)
//STEPLIB DD DSN=FID.JCK.CAIBLOAD, DISP=SHR
//          DD DSN=FID.JCK.CAIBPLD, DISP=SHR
//SYSTEM DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPROC DD DSN=SYS1.PROCLIB, DISP=SHR
//SYSIN DD DATA, DLM=FF
//IDCAMSTP EXEC PGM=IDCAMS
//SYSIN DD DISP=SHR, DSN=PROD.CNTL.CARDS(PRDCARDS)
//SYSPRINT DD SYSOUT=*
FF
//JCLOPTS DD *
CNTLOVER(QA.CNTL.CARDS)
CNTLOVER(TEST.CNTL.CARDS)
CNTLOVER(DEVL.CNTL.CARDS)
CTLSCAN LIST
//
```

**Note:** For information about using the CNTLOVER runtime option in ISPF, see the [Control Card Override Panel \(JCK12\)](#) (see page 67).

**COBEXIT**

The COBEXIT option activates the COBOL User Exit facility.

This option has the following format:

```
COBEXIT(module-name) | NOCOBOL
```

**Default:** NOCOBOL

**module-name**

Specifies the name of the COBOL load module to invoke for user exit processing.

- CA JCLCheck dynamically loads the specified COBOL exit from the specified library if there is a COBLIB DD statement present. If CA JCLCheck is running authorized, the COBOL exit and all called modules must also reside in an APF-authorized library.
- This feature is not available under CA Roscoe.

**Note:** For a complete description of the COBOL User Exit facility, see the *Programming Guide*.

**Example: Use module JCEXIT12**

The following example activates the COBOL Exit Feature and directs CA JCLCheck to use module JCEXIT12 for the exit program:

```
COB(JCEXIT12)
```

## COND

The COND option activates condition code simulation. If this option is specified without parameters, CA JCLCheck assumes that every step ends with a condition code of zero. This option also allows you to specify an alternate completion condition for a specific step.

This option has the following format:

```
COND[(step.procstep codeabendcc)]|NOCOND
```

**Default:** NOCOND

**step.procstep**

Defines the fully-qualified step name to which the completion code applies.

**code**

Defines the completion code value (0 to 4095) or one of the following keywords:

ABEnd, ABEndCC

**abendcc**

Defines either a system completion code (Sxxx) or user completion code (Uxxx). Specify S followed by three hexadecimal characters or U followed by a four-digit decimal user code.

- CA JCLCheck evaluates condition codes for IF statements and EXEC statements, including overrides from an EXEC PROC=statement. It skips any steps that have a condition satisfied.

- The default for the COND option is NOCOND. Thus, if you do not specify COND, CA JCLCheck assumes NOCOND, and every step runs regardless of condition codes specified in the JCL. If COND is specified without any parameters, CA JCLCheck assumes every step ends with a condition code of zero.
- If you specify ABEnd or ABEndCC, CA JCLCheck simulates an abnormal termination of the step and if there is any conditional disposition, CA JCLCheck uses all DD statements in the step. Execution of subsequent steps depends on the evaluation of an IF/THEN/ELSE/ENDIF construct with COND=EVEN and COND=ONLY as appropriate. Although subsequent steps can be bypassed, CA JCLCheck continues to analyze the JCL for syntactical (not logical) correctness.

This option may be specified multiple times to simulate nonzero completion codes for more than one job step.

- If more than one COND option is specified for the same step name, CA JCLCheck observes the last COND parameter for the step.
- Typically only one job is processed when using the COND option to analyze conditional execution characteristics. If the input to CA JCLCheck is a series of multiple jobs, COND option specifications only apply to the first job stream. Condition code processing remains active for remaining jobs but all completion codes default to zero.

#### Examples: Assign completion codes

The following example causes CA JCLCheck to assign a completion code of 12 to the step named STEPA.PSTP2 as well as to simulate an abnormal termination of step STEPB. Conditional execution of all other steps is appropriately evaluated.

```
CON(STEPA.PSTP2 12) CON(STEPB ABEnd)
```

The following example causes CA JCLCheck to assign a user completion code of U1000 to the step named STEPA.PSTP2 as well as to simulate an abnormal termination of step named STEPA.PSTP2. Conditional execution of all other steps is appropriately evaluated.

```
CON(STEPA.PSTP2 ABEndCC U1000)
```

## CSI

The CSI option requests that CA JCLCheck call the IBM Catalog Search Interface when doing catalog lookup and data set existence checking.

This option has the following format:

```
CSI | NOCSI
```

Default: CSI

The CSI option is mutually exclusive and cannot be used together with the following CA JCLCheck options:

- INDEX
- LOCATE
- REINDEX
- PVT CAT
- STP CAT
- SUBALLOC
- VSAM

## CTLSCAN

The CTLSCAN option reads and interprets control statements of certain IBM utility programs and the CA XCOM Data transport program. Syntax checking and validation is performed for the following utilities:

### IBM

#### IEHPROGM

ADD, BLDA, BLDG, BLDX, CATLG, CONNECT, DLTA, DLTX, DELETEP, LIST, RELEASE, RENAME, REPLACE, SCRATCH, UNCATLG

#### IDCAMS

ALLOCATE, ALTER, BLDINDEX, CANCEL, CHKLIST, CREATE, CNVTCAT, DCOLLECT, DEFINE, DELETE, EXAMINE, EXPORT, EXPORTRA, IMPORT, IMPORTRA, MPRA, LISTCAT, LISTCRA, PARM, PRINT, REPRO, RESETCAT, SET, SHCDS, VERIFY

IF/THEN/ELSE (Note 1)

DO/END (Note 1)

#### LINKAGE EDITOR

ALIAS, CHANGE, ENTRY, EXPAND, IDENTIFY, INCLUDE, INSERT, LIBRARY, MODE, NAME, ORDER, OVERLAY, PAGE, REPLACE, SETCODE, SETSSI

#### DB-2

BIND, DCLGEN, DSN, END, FREE, REBIND, RUN

#### DFSORT

DFSORT (Note 1)

ALTSEQ, DEBUG, INCLUDE, INREC, JOINKEYS, JOIN, MERGE, MODS, OMIT, OPTION, OUTFIL, OUTREC, RECORD, REFORMAT, SORT, SUM

**ADDRSSU**

ADDRSSU (Note 1)

All ADDRSSU supported options.

**IEBCOPY**

IEBCOPY (Note 1)

ALTERMOD, COPY, COPYGRP, COPYMOD, INDD=, EXCLUDE, SELECT

**IEBGENER**

IEBGENER (Note 1)

GENERATE, EXITS, LABELS, MEMBER, RECORD

**Note 1:** Syntax checking only. No runtime checking performed.

**CA****CA XCOM Data transport**

Basic syntax checking is performed for all statements.

Validation is performed for the LFILE statement. If the following statements describe attributes of the LFILE statement, they are also validated: LVOL, SPACE, TYPE, FILEOPT, BLKSIZE, LRECL, and RECFM.

This option has the following format:

CTLscan(RPT1list)|NOCTL

**Default:** CTLscan(RPT1list)

- CA JCLCheck reads control statements from the data set the control statement DD specifies (normally SYSIN for IBM and SYSIN01 for XCOM). CA JCLCheck assigns a statement number to each statement. The number displays on the reports as a suffix after the decimal point of the control statement DD. For example, if the control statement DD is statement 4, the first control statement is 4.1, the second 4.2, and the tenth 4.10.
- Specifying the CCLIST option does not affect this option. Under CTLSCAN, all control statements to IEHPROGM, IDCAMS, and XCOM steps are listed.
- Specifying this option can cause CA JCLCheck to list control statements for other utility programs. CA JCLCheck declares these programs in the JCLUPT CSECT.

**Note:** For more information, see the *Programming Guide*.

- RPT1list shows control cards on Report 1, as well as the default of Report 2.
- XCOM contains a number of parameters that are longer than eight characters; therefore, just the first eight characters are reported.

- The ADRDSSU control card processing uses ADRDSSU TYPE=SCAN to validate the control cards. During the SCAN processing, ADRDSSU may attempt to open DD names defined in the control cards causing IBM informational message 'IEC130I ddname DD STATEMENT MISSING' to be listed in the JOBLOG for the CA JCLCheck job.

These are informational messages and can be ignored. CA JCLCheck verifies if DD names that are listed in the JCL are defined in the JCL that is being checked. CA JCLCheck issues message CAY6125E DDNAME 'ddname' NOT FOUND if the DD is not in the JCL.

## DATEFMT

The DATEFMT option specifies the date in European date format (*dd/mm/yy*). You must specify this option before the RUNDATE option, and you must specify the RUNDATE option to use the date in this format.

This option has the following format:

DATEfmt (Euro) | NODAtfmt

## DB2

The DB2 option is used to activate proper validation of DB2 plans, tables, and subsystems.

This option has the following format:

DB2(*subsystem,planname*) | NODB2

**Default:** DB2 is the default if a DB2 subsystem is active; otherwise, NODB2 is the default.

### ***subsystem***

(Optional) The default subsystem is the first subsystem found in the subsystem names table associated with DB2. If a DB2 subsystem is found, DB2 plans will be validated unless the NODB2 option is coded.

### ***planname***

(Optional) The default *planname* is CAZ2JCAF. This is the PLANNAME used in the installation step Install Support for the DB2 Interface in the *Installation* guide. Specification of a unique plan name for each release of CA JCLCheck allows for multiple releases of CA JCLCheck to be bound to the same DB2 system simultaneously. This is useful for sustaining the current usage of a production release while also validating functionality of a test release of CA JCLCheck.

## DBCS

The DBCS option causes CA JCLCheck to parse double-byte character sets for Hitachi, IBM, and Fujitsu systems. This option displays and prints message text in double-byte format properly.

This option has the following format:

DBCS|NODBCS

**Default:** NODBCS

## DBGJCL

The DBGJCL option turns on JCL debugging. Use this option only at the request of Technical Support.

This option has the following format:

DBGjcl|NODBGjcl

## DBHLQ

The DBHLQ option is used with the DICT option and specifies the high-level qualifier of the CA APCDOC PARMFILE associated with the Cross-Reference database. This option is valid for CA APCDOC Version 1.1 and above.

This option has the following format:

DBHLQ(*high-level qualifier*)

## DCBCHK

The DCBCHK option suppresses message CAY6086E for SYSOUT data sets.

This option has the following format:

DCBchk|NODCbchk

CA JCLCheck flags an invalid LRECL/BLKSIZE specification regardless of the statement type. Certain versions of JES ignore invalid DCB information on spooled data sets.

## DEBUG

IBM Operating Systems – Debug Actions:

The DEBUG option causes an SVC DUMP to be taken for diagnostic analysis by CA. Use of the DEBUG option requires that CA JCLCheck be APF authorized. After the DUMP is produced, CA JCLCheck processing continues.

Non-IBM Operating Systems – Debug Actions:

The DEBUG option causes an ABEND to produce a formatted dump on the standard SYSABEND DD statement. For additional information, see ABEND codes: U1000, U2222, and U3xxx. Once the dump is produced, CA JCLCheck processing terminates.

This option has the following format:

```
DEBUg[ ([error-msg-number] [occurrence-number])] |NODEbug
```

**Default:** NODEbug

***error-msg-number***

(Optional) Defines the number of CA JCLCheck errors. These errors are prefixed by the character string CAY6 and are defined by the JCLMSG modules, or by the user of the Job Control Standards option.

***occurrence-number***

(Optional) Defines how many occurrences of the specified *error-msg-number* should occur at the time CA JCLCheck issues an ABEND.

If you do not supply any operands, CA JCLCheck DEBUG actions are taken upon completion of CA JCLCheck processing.

If you do not specify the *occurrence-number*, a default value of 1 is used.

- Typically, you use this option only at the request of CA Technical Support; however, it can be useful in debugging user exit routines.
- CA JCLCheck automatically produces the same formatted dump if it encounters a program check during operation. In this case, the user-ABEND code is set to 1000.

**Example: Cause a dump on the second occurrence of message 66**

The following example causes CA JCLCheck to take DEBUG actions on the second occurrence of message CAY6066.

```
DEBUG(66 2)
```

**Example: Cause a dump on the occurrence of message number 79**

The following example causes CA JCLCheck to take DEBUG actions on the first occurrence of message CAY6079.

```
DEBUG(79)
```

## DELTA

The DELTA option limits the scanning and updating of job streams and library members to those that have changed since the last database update.

This option is only valid when running CA JCLCheck with CA APCDOC.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

```
DELTA|NODELTA
```

**Default:** NODELTA

## DEQUE

The DEQUE option issues a DEQ macro instruction to free selected data sets that are enqueued to CA JCLCheck (or to the job that invokes CA JCLCheck under CA Roscoe).

You can optionally request a list of the DEQ macro return codes, data set names, and so forth.

This option has the following format:

```
DEQue[(List|NOList)]|NODEQue
```

**Default:** NODEQue

- Use List to obtain a list of the DEQ macro return codes, data set names, and so forth. You normally use this feature only upon request of CA to assist in problem determination.
- CA JCLCheck must be linked as authorized and executed from an authorized library to use this feature.
- An alternate method is to run program CAZ2DEQ without options, but with the same DDs.
- This option is available for use in batch or under CA Roscoe, not under TSO.

- This option requires two additional JCL statements, which CA JCLCheck uses to bracket the JCL statements that describe the data sets to dequeue:

```
//JCHKFREE DD DUMMY
```

Code this statement as shown. It marks the beginning of the sequence of DD statements that describe the data sets to dequeue.

```
/JENDFREE DD DUMMY
```

Code this statement as shown. It marks the end of the sequence of DD statements to free.

#### Example: Free selected data sets

The following example dequeues the data sets allocated to SYSPROC and PROC01 and makes them available for use by other jobs on an exclusive basis (DISP=OLD).

```
//JCLCHECK EXEC PGM=JCLCHECK,PARM=DEQ
//SYSPRINT DD SYSOUT=*
//JCHKFREE DD DUMMY marks start of DSNs to free

//SYSPROC DD DSN=SYS1.PROCLIB,DISP=SHR
// DD DSN=SYS2.PROCLIB,DISP=SHR
//PROC01 DD DSN=TEST.PROCLIB,DISP=SHR
//JENDFREE DD DUMMY marks end of dsn's to free
//SYSTEM DD SYSOUT=*
```

## DESTCHK

The DESTCHK option validates remote destination names on DD statements and JES control cards.

This option has the following format:

```
DESTchk[(Jes|NOJes)]|NODEStch
```

**Default:** NODEStch

#### Jes

(Optional) Causes CA JCLCheck to validate remote destination names using calls to JES2 or JES3.

**Note:** JES2 SP410 versions and above have generally relaxed destination validation. Validation by CA JCLCheck beyond that of JES2 is based on the user-defined DESTTBL.

**NOJes**

Turns off this option. If NOJES or DESTCHK is omitted, CA JCLCheck *does not* validate remote destination names using calls to JES2. Validity checking is directed solely against the user-defined DESTTBL.

**Note:** For more information about the DESTTBL CSECT, see the *Programming Guide*.

Use of the DESTCHK(J) option with JES2 (Version 3.0 or higher) or with JES3 requires APF to properly authorize CA JCLCheck.

**DFLTJOB**

The DFLTJOB option specifies a default job class to use when the AUTOPROC option is active and no class parameter is specified on the JOB card. If the JOB card or the DFLTJOB option does not specify a job class, class A is used. The job class is used to assign the JES procedure library for the job being checked.

This option has the following format:

DFLTJOB(x) |NODFLtjc

**Note:** x is any valid JES job class.

**DICT**

The DICT option accesses the installed CA APCDOC Cross-Reference database to obtain information about predecessor jobs to the one being analyzed.

This option is only valid when running CA JCLCheck with CA APCDOC.

This option has the following format:

DIct(Update) |NODIct

**Default:** NODict

Update requests that CA JCLCheck access the CA APCDOC Cross-Reference Database and update the database with the jobs, data sets, and programs in the input JCL.

## DICTDD

The DICTDD option supplies a ddname that is used for the CA APCDOC Cross-Reference database.

This option is only valid when running CA JCLCheck with CA APCDOC.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

DICTDd (*ddname*) |NODICTdd

## DIRUPD

The DIRUPD option, used with DELTA, directs CA APCDOC to create a current date and time stamp for members without one. The option also supplies the user ID APCDOC so that you know that CA APCDOC updated the member.

This option is only valid when running CA JCLCheck with CA APCDOC.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

DIRUPD |NODIRUPD

**Default:** NODIRUPD

## DISPATCH

The DISPATCH option prints CA Dispatch information in the JCL being checked in Report 5 - Report Listing.

This option has the following format:

DISPATCH (R,M,A) |NODISpat

R is the report class, M is the message class, and A is the archive class.

When you specify the CA Dispatch SYSOUT classes, all reports that use the class display the external writer name as a CA Dispatch controlled report.

## DSNERR

The DSNERR option issues error message CAY6079E DATASET *<data set name>* SPECIFIED AS OLD OR SHR, BUT CANNOT BE FOUND each time that *<data set name>* is referenced.

This option has the following format:

DSNErr|NODSnerr

Normally, CA JCLCheck issues this message only the first time the not found condition is encountered. Since the data set is cataloged, all subsequent references are resolved as well. Some installations want all jobs that refer to a (not cataloged data set) to be flagged with errors because they run their daily production schedule through CA JCLCheck.

## DUMPUCB

The DUMPUCB option requests a formatted dump of the portion of the system nucleus containing UCB pointers and control blocks.

This option has the following format:

DUmpucb|NODUmpuc

**Default:** NODUmpuc

Using this option causes CA JCLCheck to terminate with a user completion code 1000, unless a particular error occurrence is requested by the DEBUG option. The formatted dump appears on the Listing of Merged JCL Report, the Data Set Cross-Reference Report, the Program Cross-Reference Report, and the Report Listing (Reports 2 through 5). This option requires that the DEBUG Option also be specified.

Use this option only upon the request of Technical Support.

## DYNEXITS

The DYNEXITS option specifies the names of the assembler exits that are to be dynamically loaded by CA JCLCheck.

This option has the following format:

DYNexits(*exit1*,...*exit9*) |NODYNexi

### ***exit1 through exit9***

Define the suffix names of the standard assembler exits that are as follows:

XJOB,XEXE,XDD,XSTE,XERR,XE0J,XE0F,XCTL,XSEC,XGET

These exit suffixes correlate to the assembler exits which begin with CAZ1 and end with one of these suffixes, for example CAZ1XJOB.

### **NODYNexi**

Specifies no assembler exits are to be loaded.

**Note:** Exits linked into CAZ2CJCL or passed as a register 1 parameter are not overridden with the dynamically loaded exit.

## EASYPROC

The EASYPROC option activates support of CA Easyproclib (or EASYRDR).

This option has the following format:

EAsyproc( [ONLY] ) |NOEasyrd

### **Default:** NOEasyrd

- If you select this option and CA JCLCheck encounters an EASYPROC PROCLIB DD statement prior to the first EXEC statement of a job, CA JCLCheck opens and uses the specified PROCLIB in addition to the one specified on its own SYSPROC DD.
- In non-z/OS systems, this option requires a SYSDDxxx DD statement in the CA JCLCheck JCL allocating the volume containing the procedure library. In practice, if you use the EASYPROC option in a non-z/OS system, it is usually most convenient to add a SYSDDxxx DD statement to the CA JCLCheck cataloged procedure for every resident or reserved volume in the installation. The CTLSCAN, EASYPROC and PXREF options all use these DD statements for dynamic allocation.
- The sub-parameter ONLY is optional. If you specify this sub-parameter, CA JCLCheck only searches the PROCLIB DD statement for a procedure resolution.

## EASYRDR

The EASYRDR option has the same functionality as the EASYPROC option, which activates support of CA Easyproclib (or EASYRDR). For more information, see the EASYPROC option.

This option has the following format:

```
Easyrdr([ONLY])|NOEasyrd
```

**Default:** NOEasyrd

## EDCHEK

The EDCHEK option activates support for calling CA JCLCheck from EDCHEK, a front-end to the ISPF edit session.

This option is only valid in a TSO/ISPF session.

This option has the following format:

```
EDCHEK
```

**Default:** By default, the EDCHEK option must be coded in the file pointed to by the EDCHEK DD statement.

## EDCHEL

The EDCHEL option enables the expansion of procedure and include statements when the EJCK macro is invoked from EDCHEK. The expanded procedure and include statements will be inserted as NOTELINES instead of MSGLINES to distinguish them from error messages.

This option has the following format:

```
EDCHEL
```

## EDCHKLB

The EDCHKLB option provides generated ISPF labels on all JCL statements in error. The label format is a period (.), then five alphabetic characters, for example, ".AABJA". The label identifies the line in error. You can locate the next line in error by entering the LOCATE LABEL command.

This option is only valid with the EDCHEK runtime option.

This option has the following format:

EDCHKLB|NOEDCHKL

## EDCHKSL

The EDCHKSL option enhances the EDCHEK Edit Error report with:

- Additional information about the number of statements by their severity level.
- A message summary for each error message.
- The relative line number that is associated with each error message.
- A generated ISPF label for each line containing a JCL statement in error, if the EDCHKLB option is in effect. The labels display in a column entitled LABEL #. If the EDCHKLB option is not in use, the LABEL # column is blank.

This option is only valid with the EDCHEK runtime option.

This option has the following format:

EDCHKSL|NOEDCHKS

## ERROR

The ERROR option specifies the location of error messages on the CA JCLCheck reports.

This option has the following format:

```
ERror([Start] [Inline] End [Inline])&NOERR([Start] [Inline] End [Nor6head]])
```

**Default:** Error messages are generated in Report 6 only.

#### **Start**

Causes Report 6 - Error Messages to print at the start of the listing (Start), the (End) operand at the end of the listing (End), or within the expanded JCL listing, immediately following the statement flagged in error (Inline) operand. End is the default. Start also causes Report 6 - Error Messages to print immediately prior to Report 1 - Listing of Jobstream JCL.

#### **Inline**

Causes errors to print on Report 2 - Listing of Merged JCL Report.

#### **End**

Causes Report 6 to print immediately following Report 5 - Report Listing.

#### **Nor6head**

Suppresses the header on Report 6.

- If you do not specify the ERROR option (or you do not specify a location operand), error reports can appear in multiple places.
- Specification ER(I) should not be used with EDCHEK, SUBCHEK, or SPF SYSTEMM because it suppresses message writing (except for CAY6000) to the SYSTEMM exit; hence no error messages are displayed.
- Do not use ER(S E) with EDCHEK or SUBCHEK. It causes CA JCLCheck to write duplicate entries to the SYSTEMM exit and to display duplicate error messages.

#### **Example:**

The following example requests that CA JCLCheck include error messages in the body of the listing (Report 2) and as a separate report at the end (Report 6):

```
ER(I E)
```

## EXCLUDE

The EXCLUDE option specifies members to exclude from the partitioned data set that is defined as the CA JCLCheck primary input with the PDS, LIBRARIAN, or PANVALET options. You can specify multiple EXCLUDES when you want to specify more than ten members.

This option has the following format:

```
EXCLude(member1 . . . member10) |NOEXclud
```

This option causes CA JCLCheck to exclude data set members.

You must specify the PDS, LIBRARIAN, or PANVALET option with the EXCLUDE option. If you specify a range on the PDS, LIBRARIAN, or PANVALET option, CA JCLCheck processes all members within the range except those members listed in the EXCLUDE option.

## FACOM

The FACOM option requests support for executing on Hitachi and FUJITSU systems.

This option is only valid on Hitachi and Fujitsu systems.

This option has the following format:

```
FAcom(iefname) |NOFAcom
```

**Default:** NOFAcom

## FLUSHRC

The FLUSHRC option sets the return code for CA JCLCheck when JCL is flushed.

This option has the following format:

```
FLUSHrc(nn) |NOFLushr
```

**Default:** FLUSHrc (64)

Should JCL exist between a NULL or EOJ statement and the next JOB statement, it is flushed when submitted for execution. When this condition is detected, CA JCLCheck uses a return code of 64 upon termination.

To override the return code value (1-256), use the FLUSHRC(*nn*) option.

Using FLUSHRC(0) suppresses message CAY6408I.

**Note:** When FLUSHRC(4) is specified and CA JCLCheck issues an error message with a return code of 8, CA JCLCheck terminates using the greater value, in this example 8. In other words, using the FLUSHRC option does not cause CA JCLCheck to use a lower return code than it would have already issued.

## FULLLIST

The FULLLIST option generates Report 2 - Listing of Merged JCL. It requests that CA JCLCheck list *all* input JCL on Report 2.

This option has the following format:

```
Fulllist|NOFull
```

**Default:** Fulllist

If you omit this option (or if the default is nullified), the listing contains only statements CA JCLCheck flags with errors or warnings. In such a case, CA JCLCheck does not generate Report 2 if CA JCLCheck does not flag any statements.

## GRAPH

Requests Report 10 - Flow Diagram Report and, optionally, provides the ddname of the SYSOUT file.

This option has the following format:

```
GRaph[ (ddname, Left|Right) ] |NOGraph
```

***ddname***

Defines the name of the DD statement. Use this option to specify a DD statement that defines the SYSOUT data set for the Flow Diagram Report. The default ddname is SYSGRAPH. Use this DD statement to specify LRECL and FCB information for this report.

- This option only takes effect if the JCL being verified contains no error or severe error messages. Some warning messages still allow the GRAPH to function.

- For more information about this DD statement, see [Using CA JCL Check](#) (see page 29). For more information about Report 10 - Flow Diagram Reports, see [CA JCLCheck Reports](#) (see page 225).

#### **Left|Right**

Determines which side of the graph lists the existing data sets.

**Default:** Left

## **HCD**

The HCD option obtains UCB addresses and scans UCBs for both static and dynamic devices defined using Hardware Configuration Definition services (HCD).

This option has the following format:

HCD|NOHCD

**Default:** For IBM z/OS operating systems, HCD is the default. For Fujitsu and Hitachi operating systems, NOHCD is the default.

The HCD option requires that CA JCLCheck be properly authorized by APF.

## **IAM**

The IAM option activates CA JCLCheck recognition of data sets under IAM (Innovation Access Method) control. This option is only required if CA JCLCheck cannot dynamically determine that IAM is present on the active system.

This option has the following format:

IAM|NOIAM

**Default:** If IAM is active, IAM is the default; otherwise, NOIAM is the default.

A file is determined to be an IAM file based on one or more of the following attributes:

1. The DSNNAME contains the characters of \$IAM.
2. The OWNER value as stored in the VSAM/ICF catalog is \$IAM.
3. The SMS storage class name contains the characters \$IAM.

In order to obtain the OWNER value from the VSAM/ICF catalog, the JCLCheck VSAM option must be specified.

In order to obtain the SMS storage class from the active SMS configuration, the NOSMS and SMS options must not be specified.

## IEBMSG

With MVS Release 4.2 or higher, IEBCOPY no longer requires SYSIN DD STATEMENTS. Therefore, CAY6119W messages are not issued for missing IEBCOPY SYSIN DD statements even when marked required in JCLUPT. If you want to issue this message without regard to the operating system level, use the IEBMSG option.

This option has the following format:

```
IEBmsg|NOIEbmsg
```

## IMS

The IMS option requests support for the IBM-database product Information Management System, to validate PSB and DBD entries as coded on the parameter statement in the JCL. This option also validates DD names that are referenced within DBDs, and supports the IMS dynamic allocation feature.

This option has the following format:

```
IMS|NOIMS
```

**Default:** NOIMS

## INCLUDE

The INCLUDE option specifies members to include from the library that is defined as the CA JCLCheck primary input. Use this option with the PDS, LIBRARIAN, or PANVALET options. You can specify multiple INCLUDEs when you need to specify more than ten members.

This option has the following format:

```
INCLude(member1... member10)|NOINclud
```

Include *member1* through *member10*.

- You must specify the PDS, LIBRARIAN, or PANVALET option with the INCLUDE option. If you specify a range on the PDS, LIBRARIAN, or PANVALET option, CA JCLCheck ignores this range. It processes the members listed in the INCLUDE option.
- You must select at least one member and a maximum of ten members.
- The INCLUDE option is one of the few runtime options that you can repeat indefinitely. Although there is a limit of up to ten members per option, there is no limit on the number of repeatable INCLUDE options.

## INDEX

The INDEX option restricts catalog lookup and data set existence checking to the specified high-level indexes. This option is mutually exclusive and cannot be used together with the CSI option.

This option has the following format:

```
INDex (hi-index1[,hi-index2, . . . ,hi-index16]) | NOINDEX
```

The names of the high-level indexes prompt CA JCLCheck to process normally (for example, look up in system catalogs and VTOCs, and check for programs/members).

- You can specify up to 16 index names. At least one is required.
- You can use this option in a multi-CPU environment. The INDEX option prevents mounts for catalogs on volumes that are not available to the system where CA JCLCheck is executing.
- Under this option, any data set that starts with an index not specified on the list is assumed to be cataloged on a 3380 with a volume serial of \*XAVOL (unless the DD statement includes UNIT and VOLUME information).

### Example: Restrict Catalog Lookup and Data Set Existence Checking

This example shows how to restrict the CA JCLCheck catalog lookup and data set existence checking to data set names that begin with the high-level index of TRU or XTR:

```
IND(TRU, XTR)
```

## INPUT

The INPUT option specifies the ddname that is used to describe the primary input to CA JCLCheck.

This option has the following format:

```
INPut [ddname] | NOINPut
```

**Default:** INput(SYSIN)

*ddname* is the name of the DD statement pointing to the job streams that CA JCLCheck analyzes.

- If you omit this option (or the ddname operand for this option), CA JCLCheck uses SYSIN for its primary input ddname.
- The primary input file can be:
  - A sequential card image file.

- A series of concatenated sequential card image files.
- A partitioned data set (if you specify the PDS option).
- A CA Panvalet library (if you specify the PANVALET option).
- A CA Librarian master file (if you specify the Librarian option).

#### Example: Specify a Primary Input File

The following example directs CA JCLCheck to locate its primary input file from the DD statement named MYJOB.

```
INP(MYJOB)
```

## INTUSR

The INTUSR option initializes all four user words (\$USER-\$USER4) in the GLBL data area from the address pointed to by parameter 4. This initialization occurs when a user program calls CA JCLCheck.

This option has the following format:

```
INTusr|NOINTusr
```

If this feature is not specified, only \$USER is initialized.

## JCLLIB

The JCLLIB option requests support for the JCLLIB statement.

The NOJCLLIB option makes JCLLIB support inactive.

This option has the following format:

```
JCLlib|NOJCLlib
```

**Default:** JCLlib

## JES3

The JES3 option specifies the syntax rules that JES3 JCL is eligible for syntactical and runtime validation.

This option has the following format:

JES3|NOJES3

**Default:** If the JES3 subsystem is active, JES3 is the default; otherwise, NOJES3 is the default.

The JES3 option is optional. CA JCLCheck attempts to dynamically determine the presence of JES3 on the active system.

## JESCHEK

The JESCHEK option activates support for calling CA JCLCheck from the IEFUJV Job Validation SMF exit point.

This option is only valid in a JES2 environment.

**Note:** For more information, see the *Installation Guide*.

This option has the following format:

JESChEk

## JESCRUNT

The JESCRUNT option indicates that the JESCheck option should perform runtime checks.

This option is only valid in JESCHEK environments.

This option has the following format:

JESCRunt|NOJESCRu

## JOB

The JOB option produces Report 1 - Listing of Jobstream JCL.

This option has the following format:

Job|NOJob

**Default:** Job

This is a standard default option.

## JOBCARD

When checking PROCS, CA JCLCheck inserts a default JOBCARD. The JOBCARD option causes CA JCLCheck to use the *nn*TH default JOBCARD as assembled in the "JOBCARD" CSECT using the USERMOD MZ2C020. The *nn* value can be specified within a range of 1-32.

This option has the following format:

JOBCARD(*nn*)|NOJOBCar

**Default:** JOBCARD(1)

When the value of *nn* represents a value higher than the amount of default JOBCARDS assembled in the "JOBCARD" CSECT, JOBCARD(1) is used.

## JOBIBRAM

The JOBIBRAM option is used with CA APCDOC to pass a job list name.

This option is only valid with the CA APCDOC product.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

JOBIBRAM(*jobList*)|NOJOBIBr

## JOBSEV

The JOBSEV option sets the return code to the highest error message severity of all jobs CA JCLCheck analyzes in this run. Otherwise, the return code is set to the highest error message severity of the last job CA JCLCheck analyzes in this run.

This option has the following format:

`JOBSEV | NOJOBSEV`

**Default:** JOBSEV

## JOBTYPE

The JOBTYPE option specifies the type of JCL to scan: JOB, started task, or time-sharing user.

This option has the following format:

`JOBType(type) | NOJOBType`

*type* is one of the following values:

J—(batch job) - System Symbolics are disabled

S—(started task) - System Symbolics are enabled

T—(time-sharing user) - System Symbolics are enabled

**Default:** JOBType(J)

## KEEP7380

The KEEP7380 option causes EDCHEK to preserve the information in columns 73-80 for processing by user exits.

Normally EDCHEK uses columns 73-80 of the JCL image to track the sequence of the statement in the ISPF member being checked.

This option is only valid with EDCHEK.

Use of this option alters the functionality of EDCHEK as follows:

- The statement in error and its messages appear at the top of the screen (rather than at the location of the line within the file).
- The line numbers are missing from the EDCHKSL report when using the EDCHKSL option.

This option has the following format:

```
KEEP7380|NOKEp73
```

## KILLSEP

The KILLsep option prevents the printing of the separation lines between each CA JCLCheck report. For example, KILLsep prevents the printing of the separation lines when running CA JCLCheck with reports 1 through 6 turned OFF because only the summary reports are desired.

This option has the following format:

```
KILLsep|NOKILLse
```

## LIBLIST

The LIBLIST option enables CA APCDOC to process multiple JCL libraries at one time.

This option is only valid when running CA JCLCheck with CA APCDOC.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

```
LIBLIST(liblst)|NOLIBlis
```

## LIBRARIAN

The LIBRARIAN option specifies that the primary input is retrieved from a CA Librarian master file.

This option has the following format:

```
LIBrarian([member1]...membern)|NOLIBrar
```

**Default:** NOLIBrar

*member1* is the name of the starting member CA JCLCheck processes and *membern* is the name of the ending member CA JCLCheck processes.

- Do not specify the LIBRARIAN option in your default options.
- CA JCLCheck processes members one by one, in alphabetical member name order; it treats each as a separate job.
- The member name operands are optional. You can use one or both to restrict processing to a subset of the library. If you do not specify a member name, CA JCLCheck uses the entire master file as input. If you omit *membern*, it defaults to 99999999 (implying the last member of the CA Librarian master file).
- If a member does not begin with a JOB statement, CA JCLCheck does not use the default JOB statement in the JOBCARD CSECT, but uses the actual name of the member CA JCLCheck is processing as the JOB name in report headings.
- CA JCLCheck considers members of a CA Librarian master file as unrelated, unless you also specify the PDSRELATE option.
- To use this option, you must link the required CA Librarian subroutines when you install CA JCLCheck.
- If you specify EXCLUDE with Librarian, CA JCLCheck processes all members within the range of *member1* to *membern* except those on the EXCLUDE statement. CA JCLCheck ignores any range you specify in the LIBRARIAN option if you specify INCLUDE.

See the description of the INPUT option for specification of the INPUT ddname.

#### Example: Process members starting with letter J

LIB(J J9999999) causes CA JCLCheck to process all members of the input CA Librarian master file with names starting with the letter J.

## LINECNT

The LINECNT option specifies the number of detail lines that are printed on each report page.

This option has the following format:

LINEcnt[ (*number-of-lines*) ] |NOLINecn

**Default:** LINEcnt(60)

- The LINECNT option applies to all reports CA JCLCheck prints.
- When calculating the number of lines that are desired for a physical page, allow ten lines for the CA JCLCheck standard heading information.
- If you do not specify this option, CA JCLCheck prints 50 detail lines per page (60 lines total per page).

## LIST

The LIST option requests that reports be written to the primary report file (normally SYSPRINT).

This option has the following format:

LISt(*ddname*)|NOList

**Default:** In Batch, the default is LIST(SYSPRINT). In all other modes, the default is NOList.

### ***ddname***

(Optional) Specifies an alternate ddname for the primary report file. If you omit this operand, CA JCLCheck uses SYSPRINT for its report file.

- LIST is a standard default option.
- If you do not specify this option, the primary report file is not opened and no reports are printed, regardless of other options. NOLIST (nullification of the LIST option) is normally useful only when CA JCLCheck is invoked from a time-sharing terminal and only when the TERM option is active. (If you do not specify LIST or TERM, CA JCLCheck terminates without performing its analysis.)

### **Example: Write reports to CHKRPTS file**

The following example causes CA JCLCheck to write its primary report output to the file defined by the DD statement named CHKRPTS:

```
LIS(CHKRPTS)
```

## LOCATE

The LOCATE option causes CA JCLCheck to look up data sets in the system catalog when you code UNIT and VOLSER data on the DD statement. This option is mutually exclusive and cannot be used together with the CSI option.

The NOLOCATE option suppresses catalog lookup when UNIT and VOLUME are located.

This option has the following format:

LOcate|NOLocate

**Default:** LOcate

The LOCATE option can reduce CA JCLCheck overhead but can allow data set not found conditions to go undetected.

**Note:** We do not recommend that you use this option. Instead, use the INDEX, REVINDEXT runtime options. This option is useful for checking JCL that is running on a remote CPU.

## MCOSYS

The MCOSYS option enables you to specify the data set name of a system message control option file; you can also specify a member name. The system message control option file is a text file that specifies message numbers and severities to be used instead of severities specified in the JCLMSG file. This file is dynamically allocated by CA JCLCheck.

This option has the following format:

```
MCOSys(datasetname[,membername]) | NOMCOSys
```

***datasetname***

Specifies the data set name of the message control file to be processed. This file must point to an existing sequential file that contains a logical record length of 80.

***membername***

Specifies the optional member name of the message control file to be processed. This file must point to an existing sequential file that contains a logical record length of 80.

The format of every line in the system message control option file is:

```
messagenumber,messageseverity
```

***messagenumber***

Indicates the error message number. The *messagenumber* must be present in columns 1-3. The *messagenumber* must be numeric and a valid CA JCLCheck message number in the range of 001-999.

Column 4 must contain a comma.

**messageseverity**

Indicates the message severity used when the message is issued in CA JCLCheck. The *messageseverity* must be present in columns 5-6. A *messageseverity* of **-1** indicates the message is to be suppressed.

The *messageseverity* must be numeric and within a range of -1 through 15 as shown below:

-1	Suppress the message
00 – 03	Informational severity
04 – 07	Warning severity
08 – 11	Error severity
12 – 15	Serious error severity

The *messagenumber* and *messageseverity* must be separated by a comma. If the line starts with an asterisk (\*) in column 1, it is considered to be a comment. Entries can be input in any sequence; duplicate entries are overwritten. Invalid entries result in an error. Any matching entries will be modified and unmatched entries will be added in this sequence.

While processing a message, the following order of precedence is used to establish the message severity:

1. JCLMSG (out of the box or as modified by USERMOD MZ2C023)
2. JCLCheck program that issues the message
3. MCOSYS (if option specified)
4. MCOUSR (if option specified)
5. CAZ1REXX or REXX EXEC specified in the STDREXX option (if REXXMSG option is specified)
6. CAZ1XERR (if option specified; out of the box retains all messages or as modified by USERMOD MZ1C030)

Additionally, you can elect to use the current Option Exclusion List feature (CAZ2JOEL) to disable this option, or the MCOUSR option.

**Note:** For more information on the Option Exclusion List feature (CAZ2JOEL), see Installation Options in the *Programming Guide*. For more information on the user exit CAZ1XERR, see User Exits in the *Programming Guide*.

## MCOUSR

The MCOUSR option enables you to specify the DDNAME of a user message control option file. The user message control option file specifies message numbers and severities to be used instead of severities specified in the JCLMSG file or the MCOSYS option.

This option has the following format:

```
MCOUsr([ddname]) | NOMCOUsr
```

### ***ddname***

Specifies an optional name for the DD statement pointing to the user message control file to be processed. If *ddname* is omitted, MCOUSR is used as the DDNAME. This DDNAME must be present in the JCL and must point to an existing sequential file that contains a logical record length of 80.

The format of every line in the user message control option file is:

```
messagenumber,messageseverity
```

### ***messagenumber***

Indicates the error message number. The *messagenumber* must be present in columns 1-3. The *messagenumber* must be numeric and a valid CA JCLCheck message number in the range of 001-999.

Column 4 must contain a comma.

### ***messageseverity***

Indicates the message severity used when the message is issued in CA JCLCheck. The *messageseverity* must be present in columns 5-6. A *messageseverity* of **-1** indicates the message is to be suppressed.

The *messageseverity* must be numeric and within a range of -1 through 15 as shown below:

-1	Suppress the message
00 – 03	Informational severity
04 – 07	Warning severity
08 – 11	Error severity
12 – 15	Serious error severity

The *messagenumber* and *messageseverity* must be separated by a comma. If the line starts with an asterisk (\*) in column 1, it is considered to be a comment. Entries can be input in any sequence; duplicate entries are overwritten. Invalid entries result in an error. Any matching entries will be modified and unmatched entries will be added in this sequence.

While processing a message, the following order of precedence is used to establish the message severity:

1. JCLMSG (out of the box or as modified by USERMOD MZ2C023)
2. JCLCheck program that issues the message
3. MCOSYS (if option specified)
4. MCOUSR (if option specified)
5. CAZ1REXX or REXX EXEC specified in the STDREXX option (if REXXMSG option is specified)
6. CAZ1XERR (if option specified; out of the box retains all messages or as modified by USERMOD MZ1C030)

Additionally, you can elect to use the current Option Exclusion List feature (CAZ2JOEL) to disable this option, or the MCOUSR option.

**Note:** For more information on the Option Exclusion List feature (CAZ2JOEL), see Installation Options in the *Programming Guide*. For more information on the user exit CAZ1XERR, see User Exits in the *Programming Guide*.

## MIGVOL

The MIGVOL option specifies a volume name that CA JCLCheck uses to identify the data sets that a storage management product archives or migrates (for example, Data Facility Hierarchical Storage Management (DFHSM)).

This option has the following format:

```
MIGvol(volume name)|NOMIgvol
```

**Default:** By default, CA JCLCheck recognizes the following volume names:

- MIGRAT
- ARCHIV
- ARCIVE

### ***volume name***

Defines a pseudo-name your storage management product uses to identify archived or migrated data sets. The MIGVOL option is only required when CA JCLCheck cannot recognize your archive volume name.

**Limits:** 1 to 6 characters

## MSGLEVEL

The MSGLEVEL option permits suppression of informational messages on the SYSTERM file.

This option has the following format:

MSGLevel (Info/Error) | NOMSGlvl

**Default:** MSGLevel(I)

### Info

Causes informational messages to print.

### Error

Suppresses the informational messages by setting the severity to 8 and suppressing the end of job messages on the SYSTERM file.

## MSS

The MSS option requests the VTOC of mass storage volumes when necessary to verify the existence of a data set on MSS.

This option has the following format:

Mss | NOMss

**Default:** NOMss

- CA JCLCheck performs staging after an existence check fails because an MSS volume is not mounted. CA JCLCheck flags this condition with a CAY6067W warning message if you did not specify the MSS option.
- If you have mass storage, use this option to ensure complete validation. Be aware however, that this option can increase the CA JCLCheck running time as well as contribute to contention on the staging volumes. Use this option with caution at peak periods.

## NETWORK

The NETWORK option limits the network name associated with the SYSTEM option, but NETWORK is only available if invoked from CA JCLCheck when CA APCDOC is installed.

This option is only valid with the CA APCDOC product.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

Network(*netname*) | NONEtwor

**Default:** NONEtwor

## NJBCHK

The NJBCHK option affects the validation of JCL following an NJB statement that is preceded by a route XEQ statement.

**Note:** NJB is a JCL option for JES3 users only.

This option has the following format:

NJBchk | NONJbchk

When this feature is selected, CA JCLCheck assumes that the job is to run on the current system. As usual, the JCL is validated. When this feature is not selected, CA JCLCheck assumes that the job is to run on another system. Only the syntax of the statements following the NJB statement are checked.

## NOSPIE

The NOSPIE option suppresses the SPIE function, so that CA JCLCheck abnormally terminates if there is an OCx interrupt.

This option has the following format:

NOSPIe

- Normally CA JCLCheck intercepts an OCx ABEND with a user-ABEND 1000 and prints relevant diagnostic information on the catastrophic error message and formatted dumps identical to those generated by the DEBUG option. If you specify this option, CA JCLCheck ignores the interrupt bypass and does not generate the catastrophic error message.
- You use this option typically at the request of CA Technical Support, but it is also useful in debugging user exit processing under TSO Test.

## ONEMEM

The ONEMEM option forces the ending member name to be the same as the starting member name, when only the starting member name is specified. This option applies to PDS, CA Panvalet, or CA Librarian.

This option has the following format:

```
ONEmem|NOONemem
```

Normally, CA JCLCheck interprets this syntax as a request to process from the starting name to the end.

**Note:** When this feature is enabled, there is no implicit way to process from a given member to the end. An ending member name of 9999999 must be coded explicitly.

## OPROC

The OPROC option allows you to search a user-defined procedure library ahead of all normal IBM defaults.

This option has the following format:

```
OPRoc(ddname)|NOOProc
```

**Default:** NOOProc

***ddname***

Indicates the data definition name (already allocated) to search prior to normal search sequence. This *ddname* is used to resolve both PROC, as well as INCLUDE members.

**Note:** You must pre-allocate the *ddname* to use this option. For example, the *ddname* specified must appear in your JCL (for a batch execution). Alternatively, under TSO, you must pre-allocate the *ddname* using the ALLOC command or the logon procedure.

CA Librarian and CA Panvalet PROCLIB data sets cannot be properly allocated using the TSO ALLOCATE command since ALLOCATE does not support the SUBSYS parameter. CA JCLCheck provides JCLOPROC, a TSO command processor for this purpose. At the TSO READY prompt or ISPF Option 6, enter the following:

```
JCLOPROC 'dsn' ddname LIB|PAN
```

LIB is the default. Enter the data set name in quotes unless it is to be prefixed with your TSO user ID. If the command is successful, the READY prompt is returned. Otherwise, dynamic allocation error messages or a LOCATE error will be returned. The CA Panvalet PANALLOC command may also be used for this purpose.

## OPROCLIB

The OPROCLIB runtime option enables you to specify an override procedure library.

This option has the following format:

```
OPROCLib(Library[,C])|NOOPrcLb
```

### **library**

Specifies the data set name of the override procedure library.

### **C**

Indicates to issue an information message and continue validation processing when the library is not found.

### **Usage notes:**

- You can specify the OPROCLIB option up to 256 times; the libraries are concatenated in the order they are specified.
- The libraries must be cataloged DSORG=PO files (that is, PDS or PDSe).
- The OPROCLIB runtime option is mutually exclusive with the OPROC runtime option.

### **Example: Bypass a Missing DSN**

In this example, CA JCLCheck attempts to locate DEV.PROCLIB. If the library is not found, CA JCLCheck displays an information message and validation processing continues.

```
OPROCLIB(DEV.PROCLIB,C)
```

### **Example: Specify the OPROCLIB Runtime Option under EDCHEK**

When running CA JCLCheck under EDCHEK, you have the flexibility to specify the OPROCLIB runtime option in one of two places:

- You can specify %EJCK OPROCLIB(library) on the ISPF EDIT command line.
- You can specify the OPROCLIB runtime option in the EDCHEKDD file as follows:

```
EDCHEK PROC(PROC00)           -
      LIST(EDCPRINT)          -
      OPROCLIB(EMERGENCY.PROCLIB) -
      OPROCLIB(DEV.PROCLIB,C)  -
      OPROCLIB(PROD.PROCLIB)
ALLOC FI(PROC00)  DA(SYS1.PROCLIB)
ALLOC FI(*)       DA(SYS2.PROCLIB)
ALLOC FI(*)       DA(USER.PROCLIB)
GO
```

### Example: Specify the OPROCLIB Runtime Option in Batch

This example shows how to specify the OPROCLIB runtime option in a batch job:

```
//USERID1J JOB '1234', 'DEVELOPER', MSGLEVEL=(1,1), CLASS=A,
//          MSGCLASS=X, REGION=0M, COND=(0,LT), NOTIFY=&SYSUID
//JCLCHECK EXEC PGM=JCLCHECK, PARM='OPTIONS(JCLOPTS)', COND=(0,LT)
//STEPLIB DD DSN=FID.JCK.CAILOAD, DISP=SHR
//          DD DSN=FID.JCK.CAIPLD, DISP=SHR
//SYSTEM DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPROC DD DSN=SYS1.PROCLIB, DISP=SHR
//SYSIN DD DSN=USERID1.JCK.CNTL(JCKINPUT), DISP=SHR

//JCLOPTS DD *
//          OPROCLIB(EMERGENCY.PROCLIB)
//          OPROCLIB(DEV.PROCLIB,C)
//          OPROCLIB(PROD.PROCLIB)
//
```

**Note:** For information about using the OPROCLIB runtime option in ISPF, see the [OPROC and OPROCLIB Panel \(JCK13\)](#) (see page 68).

## OPTIONS

This option overrides the default ddname for additional CA JCLCheck processing options.

This option has the following format:

Options[(*ddname*)]|NOOption

**Default:** NOOption

### ***ddname***

(Optional) Specifies the name of the DD statement defining the additional options input file. If you omit *ddname*, CA JCLCheck uses JCLOPTS as the *ddname* for this file.

You can place options in this data set (SYSIN data) anywhere between Columns 2 and 71 (inclusive), however individual options cannot be divided across lines (for example, if you specify CCLIST(50), (50) must be on the same line as CCLIST).

**Example: Override CCLIST option**

```
//STEPCK EXEC CAZ1JCHK,PARM='0(OPTS) '  
//OPTS DD *  
CCLIST(50) COND(JOBSTEP1.PROCSTP5 16)  
COND(JOBSTEP8.PROCSTP3 ABEnd)  
SXREF(NORPT)  
/*
```

## OUTFMT

The OUTFMT option is a batch-only option that requests CA JCLCheck to produce reports in alternative formats.

This option has the following format:

```
OUTfmt (Csv, Pdf) | NOOUTfmt
```

**Default:** NOOUTfmt

**OUTfmt**

Requests a report be generated in an alternative format. When specified, Csv and/or Pdf must also be specified.

**Csv**

Indicates a report will be generated in the comma separated values (CSV) format.

**Pdf**

Indicates a report will be generated in portable document format (PDF).

When OUTFMT is specified in a batch execution of CA JCLCheck, the primary report file must be a DASD-resident file. After the primary report file is written and closed, CA JCLCheck creates a CSV and/or PDF format file from the primary report file so that you can do further processing and analysis of the reports on personal workstations.

**Note:** The primary report file is usually SYSPRINT, but you can modify this using the LIST option. For more information, see Using Alternative Report Formats in the *Programming Guide*.

The use of this option requires additional JCL statements.

### Example: Create a PDF from the primary report file

This example displays the command to generate a PDF from the primary report file.

```
OUTfmt (Pdf)
```

## PANEXEC

The PANEXEC option requests support of the CA Panexec transparent loader.

This option has the following format:

```
PANExec|NOPANE
```

**Default:** NOPANE

This option lets you contain programs in a CA Panexec library. This option also requests that you use the CA Panexec transparent loader in program existence checking. When a PECNTL file exists in the step CA JCLCheck is analyzing, CA JCLCheck dynamically allocates it to its job step before issuing the BLDL macro.

- This option does not function under CA Roscoe.
- CA JCLCheck only allocates this PECNTL file if it is an existing sequential data set or member of a partitioned data set, and when the data set is on an accessible disk volume.
- CA JCLCheck does not support instream data sets and concatenated inputs.

## PANVALET

The PANVALET option specifies that the primary input is retrieved from a CA Panvalet library.

This option has the following format:

```
PANvalet([member1]...membern)]|NOPAN
```

**Default:** NOPAN

***member1***

Defines the name of the starting member CA JCLCheck processes.

***membern***

Defines the name of the ending member CA JCLCheck processes.

- Do not specify the PANVALET option in your default options.

- CA JCLCheck processes members one by one, in alphabetical member name order and treats each one as a separate job.
- The member name operands are optional. You can use one or both to restrict processing to a subset of the library. If you do not specify an operand, CA JCLCheck uses the entire CA Panvalet library as input. If you omit *membersn*, it defaults to 99999999 (implying the last member of the CA Panvalet library).
- If a member does not begin with a JOB statement, CA JCLCheck does not use the default JOB statement in the JOBCARD CSECT, but uses the actual name of the member it is processing as the JOB name in report headings.
- Members of a CA Panvalet library are unrelated, unless you also specify the PDSRELATE option.
- If you specify EXCLUDE with PANVALET, CA JCLCheck processes all members within the range of *member1* to *membersn* except those you specify in the EXCLUDE option. CA JCLCheck ignores any range you specify on the PANVALET option if you specify INCLUDE.

#### Example: Process members starting with letter J

The following example requests that CA JCLCheck process all members of the CA Panvalet library with names starting with the letter J:

```
PAN(J J9999999)
```

## PASSWORD

The PASSWORD option specifies the password to use for accessing read protected VSAM catalogs.

This option has the following format:

```
PASsword(character-string) |NOPASsw
```

#### ***character-string***

Defines the name of the password.

#### **Usage notes:**

- This option is only used if the VSAM option is also in effect.
- The password character-string operand is required. The operand is stored encrypted within CA JCLCheck and is not listed on any reports.
- If you want to provide a default password in encrypted form, contact CA Support for assistance.

## PDS

The PDS option specifies that the primary input is retrieved from a partitioned data set.

This option has the following format:

```
PDS([member1 . . . membern])|NOPDS
```

**Default:** NOPDS

***member1***

Defines the name of the starting member CA JCLCheck processes.

***membern***

Defines the name of the ending member CA JCLCheck processes.

- Do not specify the PDS option in your default options.
- Members are processed one by one, in alphabetical member name order. Each one is treated as a separate job.
- The member name operands are optional. One or both may be used to restrict processing to a subset of the library. If none is specified, CA JCLCheck uses the entire partitioned data set as input. If you omit *membern*, it defaults to 99999999 (implying the last member of the library).
- If a member does not begin with a JOB statement, CA JCLCheck uses the default JOB statement in the JOBCARD CSECT, but uses the actual name of the member being processed as the JOB name in report headings.
- The input library may be a JCL job stream library or a procedure library.
- CA JCLCheck considers members of the partitioned data set unrelated unless you also specify the PDSRELATE option.
- If you specify EXCLUDE with PDS, CA JCLCheck processes all members within the range of *member1* to *member2* except those listed on the EXCLUDE statement. CA JCLCheck ignores any range you specify on the PDS option if you specify INCLUDE.

**Example: Process members starting with letter J**

PDS(J 99999999) causes CA JCLCheck to process all members from the input partitioned data set with names that begin with J.

## PDSRELATE

The PDSRELATE option specifies that CA JCLCheck treat the primary input library as a series of consecutive, related job streams.

This option has the following format:

PDSRelat|NOPDSRel

**Default:** NOPDSRel

Use this option to extend related job data set validation and cross-reference reporting to library input.

Use this option only if the SXREF option and one of the library input options (LIBRARIAN, PANVALET or PDS) are also in effect.

## PNFMSG

The PNFMSG option forces message CAY6093E PROGRAM NOT FOUND. This message occurs when a program or an EXEC statement is not found to exist in a library mentioned in the CA JCLCheck STEPLIB/JOBLIB.

This option has the following format:

PNFmsg|NOPNFmsg

When CA JCLCheck is executed with a STEPLIB, programs in that library are normally found as an extension of the linklist search. When the library is not in the linklist, the jobs could fail without a JOBLIB or STEPLIB. This option is not needed if CA JCLCheck is linklisted or isolated in its own library.

## PRCDELTA

The PRCDELTA option limits the CA APCDOC scanning of job streams and the updating of those jobs whose cataloged procedures have changed since the last database update. You must use this option with the DELTA option. This option is only valid with PDS procedure libraries.

This option is only valid when running CA JCLCheck with CA APCDOC.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

PRCDELTA|NOPRDELta

## PREDNO

The PREDNO option scans the system without building a predecessor list.

This option has the following format:

PREDNO|NOPREDNO

This option is only valid when running CA JCLCheck with CA APCDOC Version 1.3 and above.

## PREFIX

The PREFIX option specifies the ddname prefix that is used for the dynamic allocation in non-z/OS systems and for JES3 `//*MAIN` statement CA PROCLIB support.

This option has the following format:

PREFix[(SYSDD)|(character-string)]|NOPRefix

SYSDD|*character string* is the name of the dynamic allocation prefix.

- Dynamic allocation is required to allocate any of the following:
  - Control statement data sets for the CTLSCAN option.
  - Procedure libraries for the EASYPROC option.
  - Program and source libraries for the PXREF option.
- For non-z/OS dynamic allocation to succeed, specify a DD statement with a ddname beginning with the specified PREFIX character-string in the CA JCLCheck JCL that allocates the volume containing the library. In practice, it is best to provide such a DD statement for every resident and reserved volume in the installation.
- CA JCLCheck uses SYSDD as the non-z/OS dynamic allocation prefix, when you do not specify this option or if you omit the PREFIX character-string operand.

You can also use this option with JES3 `//*MAIN ALT PROCLIB` support. PRE(DYNAL) causes CA JCLCheck to look for DD statements with names that begin with the characters DYNAL, rather than SYSDD, when performing necessary dynamic allocation.

## PREVEX

The PREVEX option suppresses the expansion of library PROCEDURE and INCLUDE members. The instream procedures are still expanded.

By default, no existence checking is performed (CAY6027 is not issued), and the NORUNTIM (No Runtime) and NOAUTOP (No Autoproc) options are also activated.

This option has the following format:

```
PREVEX(VERIFY, RUNTIME) |NOPREVEX
```

**Default:** NOPREVEX

#### **VERIFY**

(Optional) Performs existence checking for the procedure and issues message CAY6027 when it is not found.

#### **RUNTIME**

(Optional) Bypasses the automatic activation of the NORUNTIM and NOAUTOP options.

#### **Usage notes:**

- Use of the PREVEX option causes CA JCLCheck to validate less than the complete job. As a result, the analysis is incomplete and can produce inaccurate results.
- Use the PREVEX option when procedure libraries are not available or procedures inside them are not to be used. When a JCLLIB statement specifies procedure libraries and the libraries are not available, use the NOJCLLIB option to prevent error message CAY6079E (Dataset cannot be found).
- When using the PREVEX option, error message CAY6027E (Procedure not found) is not issued and procedures are not expanded, but EXEC statement keywords and symbolic parameters are checked for duplicity and syntax.
- Other error messages can be reported. For example, errors on JCL overrides of DD definitions in the unexpanded proc (since the procedure is not available). Also, procedure keywords cannot be verified so errors may exist that cannot be detected when using this option.
- Using the PREVEX option forces the use of the NOAUTOPROC and NORUNTIME options.

## **PROC**

The PROC option specifies an alternate ddname for the statement defining the procedure library.

This option has the following format:

```
PROc[(ddname, LIB|PAN)] |NOPROC
```

**Default:** PROc(SYSPROC)

*ddname* refers to a single data set DD statement or to a concatenation of procedure libraries.

LIB and PAN are optional parameters and explicitly request CA Librarian or CA Panvalet support using the specified ddname.

**Note:** When you specify the ddname and either the Lib or Pan subparameter, the ddname is limited to one data set and no concatenation.

- If you omit the option (or the *ddname* operand), CA JCLCheck uses SYSPROC as its procedure library DD statement.
- The PROC option is mutually exclusive with the AUTOPROC option.

#### Example: Specify Alternate Ddname

The following example points CA JCLCheck to the DD statement named TESTPROC for the procedure libraries:

```
PRO(TESTPROC)
```

#### Example: Specify Alternate Ddname and CA Librarian Access

The following example points CA JCLCheck to the DD statement named TESTPROC for the procedure libraries, and uses the CA Librarian access method for retrieval of records.

```
PRO(TESTPROC,LIB)
```

**Note:** Support is dynamic when you use the CA Librarian (LAM) or CA Panvalet (PAN) access method.

## PROCDLFT

The PROCDLFT option causes CA JCLCheck to search the default PROCLIB (defined by the SYSPROC DD statement) after searching any JES2 alternate PROCLIBs defined in the job stream using a /\*JOBPARM PROCLIB=ddname statement if CA JCLCheck does not find a cataloged procedure in the alternate PROCLIB.

Normally CA JCLCheck only searches the alternate PROCLIB. (The need for this option depends on how JES2 is installed at your site.)

This option has the following format:

```
PROCDLFT|NOPROCDF
```

**Default:** NOPROCDF

## PROCXREF

The PROCXREF option generates Report 11 - Symbolic Reference Report.

This option has the following format:

PROCXref|NOPROCXr

**Default:** NOPROCXr

## PULL

The PULL option requests a tape pull list of tapes used in jobs being checked.

This option is only valid when running CA JCLCheck with CA APCDOC.

This option has the following format:

PULL|NOPULL

**Default:** NOPULL

## PVTCAT

The PVTCAT option causes CA JCLCheck to look up data sets in the catalog again when the step it is analyzing has a STEPCAT DD statement.

The STEPCAT can cause CA JCLCheck to use a different catalog search order than when it first located the data set. (This feature increases CA JCLCheck overhead, but can increase accuracy if the same data set is cataloged on different volumes in different catalogs referenced in the same job.)

This option is mutually exclusive and cannot be used together with the CSI option.

This option has the following format:

PVtcat (AUTH) |NOPVtcat

**Default:** NOPVtcat

AUTH is optional and requires APF authorization. The AUTH parameter nullifies the effects of JOBCAT and STEPCAT under the CA JCLCheck execution step.

## PXREF

The PXREF option activates the program and member existence checking and reporting feature.

This option has the following format:

PXref(Rpt|NORpt)-|NOPXref

**Default:** PXref(Rpt)

### Rpt

(Optional) Generates Report 4 - Program Cross-Reference Report.

### NORpt

(Optional) Suppresses Report 4 - Program Cross-Reference Report.

- PXREF is a standard default option. If you specify NOPXREF, CA JCLCheck does not perform program or member existence checking.
- If you request the PXREF option, but omit the reporting operand, CA JCLCheck activates the feature and produces the report.
- In non-z/OS systems, program existence checking requires a SYSDDxxx DD statement for each volume containing a program library referenced by a JOBLIB or STEPLIB DD statement. In practice, if the PXREF option is used in a non-z/OS system, it is usually most convenient to add a SYSDDxxx DD statement to the CA JCLCheck cataloged procedure for every resident or reserved volume in the installation. The CTLSCAN, EASYPROC and PXREF options all use these DD statements for dynamic allocation.

**Note:** For information about reporting considerations if related job processing is invoked, see [SXREF](#) (see page 205).

### Example: Perform checking and suppress Report 4

The following example causes CA JCLCheck to perform program and member existence checking but suppresses the Report 4 - Program Cross-Reference Report.

```
PX(NOR)
```

## RDR

The RDR option specifies the ddname to use with the output job streams generated by the SUBMIT option.

This option has the following format:

```
RDr[ (ddname) ] |NORDr
```

**Default:** RDr(SYSRDR)

*ddname* is the name of the DD statement to use.

- Use this option only if the SUBMIT option is also in effect.
- If you request the SUBMIT option and omit this option or the ddname operand, CA JCLCheck defaults to SYSRDR for the DD statement defining the SUBMIT output file.

### Example: Automatically submit and write to destination

The following example causes CA JCLCheck to activate the automatic submission feature and to write those error-free jobs to the destination in the DD statement JOBG0.

```
SU RD(JOBG0)
```

## REMOTE

The REMOTE option enables remote validation of JCL on remote servers using the ChekPlex feature. Use of this option requires APF authorization.

This option has the following format:

```
REMOte|NOREMOte
```

**Default:** NOREMOte

Remote validation requires the REMVAL DD name be pre-allocated and defined in the ISPF options panel, or defined in JCL for a batch run. This DD name defines: the systems on which to do the validation, the JCL to be validated, and which procedures to use for procedure overrides.

**Note:** For information on the available options in the REMVAL DD, see The Chekplex Feature in the *Programming Guide*.

## REPORT

The REPORT option functions like the RPTRPT option, but REPORT is only available if invoked from CA APCDOC without a full version of CA JCLCheck.

This option is only valid with the CA APCDOC product.

**Note:** For more information, see the CA APCDOC documentation.

This option has the following format:

Report

## RESOLVE

The RESOLVE option requests CA JCLCheck resolve external variables that are imbedded in the JCL for a scheduling product. The variable resolution occurs before CA JCLCheck validates the JCL.

You can specify the RESOLVE option in batch runs using the CAZ1JCC7, CAZ1JCES, CAZ1JCTW, or CAZ1JCCM procedures using the RESPARM= parameter. The RESPARM= parameter is passed the CAZ1PRES pre-resolution program that runs before the JCLCHECK step in these procedures. For information on the proper usage, see the *Tutorial*.

The JCLCheck program also accepts a form of this command for EDCHEK RESOLVE processing. THIS FORM IS INTENDED FOR INTERNAL USE ONLY. The RESOLVE option requires setup that the EJCK edit macro provides and SHOULD NOT be specified to the JCLCHECK program directly.

**Note:** For information about how to use external variables with EDCHEK, see Specify the Product and Parameters using EDCHEK ISPF in the *Tutorial*.

You can use the NORESOLVE form of the command with the EJCK edit macro to prevent the external variable resolution.

This option has the following formats for use with the CAZ1PRES program:

RESolve(*product*, *dsname*[, *member*])

RESolve(*product*, DDNAME: *ddname*)

RESolve(*product*, NOPARM)

This option has the following format for use with the EJCK command:

%EJCK NORESolve

***product***

Specifies the name of the scheduling product that CA JCLCheck calls. This sub-option is required, valid values are CA7, ESP, TWS, or CTM.

***dsname***

Specifies the sequential file (LRECL 80) or PDS that contains the product-specific parameters that are used for the external variable resolution.

***member***

Specifies the PDS member (LRECL 80) that contains the product-specific parameters that are used for the external variable resolution. If the *dsname* sub-option specified is a PDS, then the *member* sub-option is required. When the *dsname* sub-option is a sequential data set, do not specify the *member*.

***ddname***

Specifies the DDNAME that is allocated to the CAZ1PRES step JCL. The DDNAME contains the product-specific parameters that are used for the external variable resolution.

**NOPARM**

Specifies that the product-specific parameters file is not used. The default values are used for all parameters. NOPARM can only be used with the CTM option. All other products require specified parameters.

**NORESolv**

Specifies not to perform External Resolution (EJCK only).

**Example: Specify a Product and Data Set Name for the CAZ1PRES RESOLVE Option**

This example shows how to specify CA WA ESP Edition as the product that CA JCLCheck calls to resolve external variables in a batch run of the CAZ1JCES procedure. The related *dsname* and *member* name are also specified.

```
//CHECK EXEC CAZ1JCES,REGION=0K,  
// RESPARM='RESOLVE(ESP,USERID1.JCK.CNTL,RESPARMS) ',
```

## REVINDEX

The REVINDEX option restricts catalog lookup and data set existence checking by ignoring the specified high-level indexes that begin with the value specified. REVINDEX treats this value as if there is a wildcard (\*) following it. This option is mutually exclusive and cannot be used together with the CSI option.

This option has the following format:

```
REVIndex (hi-index1[,hi-index2,...hi-index16])|NOREVind
```

### ***hi-index***

Specifies up to 16 names of the high-level indexes that CA JCLCheck should ignore during normal processing. For example, when looking up programs/members in system catalogs and VTOCs. At least one name is required.

You can use this option in a multi-CPU environment to prevent mounts for catalogs on volumes that are not available to the system where CA JCLCheck is being executed.

Under this option, CA JCLCheck assumes that any data set that starts with an index specified on the list is cataloged on a 3380 with a volume serial of \*XAVOL (unless the DD statement includes UNIT and VOLUME information).

### **Example: Exclude Data Sets**

The following example excludes data sets that have names beginning with the high-level index of TRU or XTR from the CA JCLCheck catalog lookup and data set existence checking:

```
REVI(TRU, XTR)
```

## REXXMSG

The REXXMSG option enables you to process messages and change message severity via the REXX EXEC specified in the STDREXX option.

This option has the following format:

```
REXXMSg | NOREXms
```

### **Default:** NOREXms

While processing a message, the following order of precedence is used to establish the message severity:

1. JCLMSG (out of the box or as modified by USERMOD MZ2C023)
2. JCLCheck program that issues the message

3. MCOSYS (if option specified)
4. MCOUSR (if option specified)
5. CAZ1REXX or REXX EXEC specified in the STDREXX option (if REXXMSG option is specified)
6. CAZ1XERR (if option specified; out of the box retains all messages or as modified by USERMOD MZ1C030)

Additionally, you can elect to use the current Option Exclusion List feature (CAZ2JOEL) to disable this option.

**Note:** For more information on the Option Exclusion List feature (CAZ2JOEL), see Installation Options in the *Programming Guide*. For more information on the user exit CAZ1XERR, see User Exits in the *Programming Guide*.

## ROSCOE

The ROSCOE option specifies that you are invoking JCLCHECK within the CA Roscoe address space. This option is specified automatically by the CA Roscoe supplied JCLCHECK monitor "JCK".

This option has the following format:

ROSCOE

**Default:** In a CA Roscoe environment, the default is ROSCOE. Outside of a CA Roscoe environment, this option is not active.

## ROSLIMIT

The ROSLIMIT option specifies the maximum number of JCL statements to process when CA JCLCheck is run as a CA Roscoe monitor routine.

Use this option only in a CA Roscoe environment.

This option has the following format:

```
ROSLimit(statement-count) |NORoLim
```

### ***statement-count***

Specifies CA JCLCheck logical statements that include all continuation statements. The actual count of statements CA JCLCheck processed is higher than the specified limit.

**Limit:** 9999

### **Usage notes:**

- All tasks compete for the virtual storage in the CA Roscoe address space. If your installation is short on storage, you can use this option to restrict the size of job streams (including PROCLIB statements) that can be processed online.
- The statement-count limit cannot be greater than the limits your AWS determines.
- Omit this option if adequate virtual storage is available. Most MVS/ESA sites would not use this option.

### **Example: Set an 80 statement limit**

The following example causes CA JCLCheck operating under CA Roscoe to terminate processing prematurely when more than 80 statements are included in the input stream.

```
ROSL(80)
```

## RPTRPT

The RPTRPT option requests production of Report 5 - Report Listing (SYSOUT) generated by the job/system. This report contains information about all the SYSOUT in a job or production system. For information about reporting considerations if related job processing is invoked, see [SXREF](#) (see page 205).

This option has the following format:

```
RPTrpt |NORptrpt
```

**Default:** NORptrpt

## RUNCPU

The RUNCPU option specifies a CPU ID to use for security validation. (RUNCPU simulates that the JCL is being run from an alternate CPU.) Specifying this option automatically turns on security validation.

This option has the following format:

`RUNCpu (CPU ID) | NORUNcpu`

**Default:** NORUNcpu

## RUNDATE

The RUNDATE option specifies a future run date and time to use for security validation. Specifying this option automatically turns on security validation.

This option is valid only for CA ACF2 r6.0 users.

This option has the following format:

`RUNDate(mm/dd/yy, [hh:mm]) | NORUNDt`

**Default:** NORUNDt

Hours and minutes (*hh:mm*) are designated in military time (24-hour clock) and are optional. You can specify the date in *mm/dd/yy* format if the DATEFMT option is turned on.

## RUNTIME

The RUNTIME option performs runtime checking of JCL. Runtime checking activates functions that verify the scanned JCL runs properly at execution time. This verification is accomplished by a wide variety of functions. For example, the execute program must be found and executable, cataloged data sets must actually exist on the disk packs or be available through the tape management system, security rules must be validated, and so forth. For more information about this option, see the SYNTAX option.

This option has the following format:

`RUNTime | NORUNTim`

**Default:** RUNTime

**Limits:** The NORUNTim and AUTOPROC options are mutually exclusive with the following four exceptions:

- JCLCheck Common Component with CA 7
- JCLCheck Common Component with Scheduler
- JCLCheck Common Component with ESP
- JCLCheck Full Product

## RV

The RV option is used internally by the ChekPlex remote validation option to indicate that the validation request is being done on a remote system. Do not specify this option in calling options or in OPTIONS dataset.

This option has the following format:

RV

## SCHEDLER

The SCHEDLER option requests basic support for CA Scheduler. Use this option only when running under CA Scheduler.

This option has the following format:

SCHedler|NOSCHEDL

## SCHENV

The SCHENV option enables validation of the scheduling-environment name parameter specified on the JOB statement.

This option has the following format:

SCHENV | NOSCHENV

**Default:** NOSCHENV

Using the SCHENV option requires JCLCheck be APF authorized, and the use of the RUNTIME option.

## SECURITY

The SECURE option requests support of the pre-validation of the security environment.

This option has the following format:

SECurity(*options*) | NOSECure

**Default:** NOSECure

Use *options* to modify security checking. Specifying the Security option automatically activates various pre-validation checks. The following are these checks:

### **PROG**ram | **NO**PROGr

Provides information about how the program on the EXEC statement is defined to the security system. Use of the DEFine sub-parameter reports whether the program is defined to the security system.

**Default:** PROGram

### **DASD**vol | **NO**DASD

Provides information from the security system about a particular volume.

**Default:** DASDvol

### **STOR**Clas | **NO**Stor

Provides information about a derived Storage Class and how it is defined to the security system. Your site ACS routines determine the derived Storage Class name. The high-level qualifier is used as the owner.

**Default:** NOStor

### **MGMT**clas | **NO**Mgmt

Provides information about a derived Management Class and how it is defined to the security system. Your site ACS routines determine the Management Class. The high-level qualifier is used as the owner.

**Default:** NOMgmt

### **SHR**upd | **NO**SHRupd

Only performs a security check for ACCESS=READ on data sets defined in the JCL as DISP=SHR.

### **PATH** | **NO**PAth

Activates security validation for UNIX PATHs. The user ID checked against comes from the USER(userid) JCLCheck option or the USER=userid JOB statement parameter.

**Default:** NOPAth

When you activate either PROGram or DASDvol, all other default security options are activated (MGMTclas, STORClas, and SHRupd). You can choose to activate or deactivate an option individually. However, you cannot deactivate basic data set level security checking which is fundamental to security checking.

Just specifying SECURITY activates the call to the CA Standard Security Facility (CAISSF). This interface makes an inquiry of the security package currently active through the CA CA90s Standard Security Facility. This interface is designed to work with CA ACF2, CA Top Secret, and any SAF-compatible product (for example, RACF).

**Note:** For more information, see the *Programming Guide*.

- Non-DFSMS users do not need to specify NOSTOR, NOMGMT.
- When using the SECURITY option, APF authorization is required.

The SECURE option is still available to users of SECURE to provide downward compatibility. The SECURE option turns off the Standard Security interface. With this option, CA JCLCheck reads protected libraries, if the user is authorized. This option is only used if the installation has the SECURE product.

#### Examples: Change the security environment

##### **NODASD**

Turns off all DASDVOL resource checking.

##### **NOLOG**

Turns off logging of security violations, preventing USERID suspensions. (DEFAULT MODE - Active Logging).

##### **NOMGMT**

Turns off SMS MGMTCLAS resource checking.

##### **NOPROG**

Turns off all PROGRAM resource checking.

##### **NOSTOR**

Turns off SMS STORCLAS resource checking.

##### **SEC**

Activates validation of resources through the Standard Security facility (CAISSF).

##### **SECurity**

(NO PROG, NODASD, NOSTOR, NOMGMT, PROGRAM(DEFINE)).

##### **SEC(PROG(DEF))**

Activates validation of resources through the CAISSF and issues messages about whether the executed program is defined to the active security package.

**SEC(SEC)**

Use only for the SECURE product.

**SEV**

The SEV option specifies the minimum severity level of errors to include on reports.

This option has the following format:

`SEv(error-level)|NOSEV`

**Default:** SEv(0)

Standard CA JCLCheck error messages are assigned the following values:

<b>0</b>	Informational
<b>4</b>	Warning
<b>8</b>	Error
<b>12</b>	Critical Error
<b>16</b>	Severe Error

Error messages that user exit processing generates can be assigned any value within the range of 0 through 16.

This option selectively suppresses messages that CA JCLCheck generates based on the numeric severity level.

Installations can also modify the severity of individual CA JCLCheck messages to any value in the range of 0 through 16.

Specifying SEV in combination with the MSGLEVEL(*error*) option, causes CA JCLCheck to use the severity that SEV specifies. MSGLEVEL(*error*) specifies severity level 8, which is the severity level that is overridden.

**Example: Suppress Messages with Severity Less Than 8**

The following example requests that CA JCLCheck suppress all informational and warning messages (any message with a severity value less than 8):

```
SE(8)
```

## SIGNON

The SIGNON and NOSIGNON runtime options control the use of the alternate user ID feature. You can use the SIGNON option only when CA JCLCheck is APF authorized.

This option has the following format:

```
SIGNon|NOSIGNon
```

### SIGNON

Uses the current user ID and makes that ID the current user for the processing duration of the job. You can establish the user ID in the following ways:

- You can specify the USER option to CA JCLCheck. When the USER option is present, this ID applies to all jobs processed in this CA JCLCheck run. The ID also overrides any user IDs that are specified in the JCL being validated.
- You can define user IDs in the JCL being validated. The most common is the specification of USER= on the JOB card. If you are using the ACF2 option, the ID can also come from one of the ACF2 specific comment cards, such as `//*LOGONID user_id`.

Each time a JOB card is processed, the user changes to the current user ID when it has changed. The job is validated using the ID that CA JCLCheck originally ran with when the USER option is not in use and the JCL contains no user ID. When all jobs have been processed, the ID is re-established to the original user.

### NOSIGNON

Prevents the SIGNON option from being used. If NOSIGNON is specified, it overrides the SIGNON specified in JCKDFLT or at execution. Placing NOSIGNON in the JCLDFLT does not prevent use of the SIGNON.

**Note:** For more information about how to use SIGNON, see the Alternate User ID topic in the *Tutorial*.

## SMS

The SMS (Storage Management Subsystem) option requests that CA JCLCheck perform JCL checking according to SMS rules. If SMS is not active, CA JCLCheck performs simulation checking, and it validates JCL based on the minimal SMS requirements. The NOSMS option requests JCL checking according to non-SMS rules, even if SMS is active.

If SMS is active and you want to use the active SMS configuration, do not specify either option.

If SMS is not active and you want to perform JCL checking according to SMS rules, specify the SMS option to simulate SMS processing.

If SMS is active and you do not want to perform JCL checking according to SMS rules, specify the NOSMS option.

If you do not specify the SMS/NOSMS option, CA JCLCheck determines your SMS environment and selects the appropriate path.

This option has the following format:

```
SMS|NOSMS
```

**Default:** If neither option is specified and SMS is active, the JCL will be validated using the active SMS configuration.

## SPACE

The SPACE option activates the direct-access space efficiency analysis and reporting feature.

This option has the following format:

```
SPace[ ([Rpt|NORpt] [min-percentage] [All|NOAll])] |NOSPace
```

**Default:** NOSPace

### RPT|NORPT

Controls whether CA JCLCheck lists direct-access space used by existing files or requested for new files. If you specify RPT, the DASD space usage is listed on Report 3 - Data Set Cross-Reference Report and Report 7 - Summary Data Set Cross-Reference Report

**Default:** NORPT

***min-percentage***

Specifies the minimum percentage of track utilization used in space efficiency checking. With this feature, if CA JCLCheck finds files using less than the specified minimum percentage of a track, a warning message is issued. The user specifies any value from 0 to 99. A value of 0 suppresses this efficiency check feature. Initially, we recommend a value of 75-80 percent.

**Default:** 0, which suppresses the feature.

**ALL|NOALL**

Controls whether the space efficiency check (specified by operand 2) is performed on both new and existing (ALL) data sets or only on new (NOALL) data sets created by the input job stream. Use this operand only if the minimum track efficiency percentage requested is greater than 0. The operands are positional. At least one of these operands must be specified if the SPACE option is requested.

**Default:** NOALL

- If this feature was specified as a system default during installation, you must code nullification of this option as NOSPA (to distinguish it from NOSP that activates the NOSPIE option).
- This option does not check the percentage of space used in a total allocation, rather, this option checks the efficiency of track utilization based on block size.

**Examples: Analyze and report on space**

The following example requests CA JCLCheck to report on space usage for all files and to issue a warning on any new files using less than 75 percent track capacity.

```
SP(RPT 75)
```

The following example requests that CA JCLCheck not report on data set DASD usage, except to issue a warning for any file (new or existing) using less than 75 percent track capacity.

```
SP(NOR 75 ALL)
```

The following example requests that CA JCLCheck report on space usage for all files but not perform any track efficiency analysis.

```
SP(RPT)
```

## SRCH

The SRCH option searches all libraries on a JOBLIB or STEPLIB concatenation, even if one of the libraries is a temporary data set.

This option has the following format:

SRCh|NOSRch

When a temporary data set is found in the search order, CA JCLCheck assumes the program being executed is found because the actual contents of the temporary library cannot be determined until runtime.

## STACK

The STACK option suppresses message CAY6158W when the file sequence count for DD statements that are stacking files on the same tape do not occur in sequential order.

There is no requirement to have DDs in order by label number, since the order in which they are opened is what counts.

This option has the following format:

STACK|NOSTACK

**Default:** NOSTACK

## STANDARD

The STANDARD option specifies the set of standards to use for job control standard compliance checking.

This option has the following format:

STandard(*standard-name1*, ... *standard-name8*)|NOSTAND

***standard-name1 ... standard-name8***

Identifies the job control standards to use for compliance checking. It also activates the Job Control Standard feature.

**Note:** When you specify more than one standard name, each standard is invoked in the order that they are coded in the standard option. Also, each standard group is *always* invoked independent of the others.

### Example: Activate Feature and Specify a Set of Standards

The following example activates the Job Control Standard Compliance feature and directs CA JCLCheck to use STD001 as the set of standards for compliance checking:

```
STANDARD(STD001)
```

## STDREXX

The STDREXX option enables you to use the REXX programming language for data center JCL standards enforcement.

The REXX EXEC specifies the REXX EXEC to use for job control standard checking.

This option has the following format:

```
STDRExx(REXX EXEC member name [,REXX EXEC DD name]) | NOSTDrex
```

### **REXX EXEC *member name***

Identifies the PDS member name that contains the REXX EXEC CA JCLCheck should use for standards enforcement. This must follow standard PDS member naming conventions. The library containing the REXX program is specified by your REXX installation defaults data definition name (ddname).

### **REXX EXEC DD name**

Specifies an optional, alternate DD name used to load the REXX EXEC member name. If omitted, the standard DD name of SYSEXEC is used.

### **NOSTDrex**

Specifies no REXX EXEC is in use.

CA JCLCheck invokes the REXX EXEC for each JCL statement.

The REXX environment created for this interface varies according to the runtime environment. This means that certain TSO/E specific functions are not valid under non-TSO/E address spaces. This is because you can run this interface under any supported environment (Batch, TSO, and so forth).

## STEPMSG

This option allows the CAY6128W message to be issued for duplicate step names within a job; otherwise, the message is not issued.

This option has the following format:

```
STEPMsg|NOSTEpms
```

## STPCAT

The STPCAT option suppresses message CAY6162W STEPCAT SHOULD BE FIRST DD STATEMENT IN STEP.

This option is mutually exclusive and cannot be used together with the CSI option.

This option has the following format:

STPCat|NOSTPcat

## SUBALLOC

The SUBAlloc option eliminates spurious CAY6066E messages that CA JCLCheck issues for VSAM data sets which are sub-allocations of a VSAM data space.

This option is mutually exclusive and cannot be used together with the CSI option.

This option has the following format:

SUBAlLoc|NOSUBALL

**Default:** NOSUBALL

## SUBCHEK

The SUBCHEK option indicates that CA JCLCheck is called from SUBCHEK, which is used when passing control to the TSO SUBMIT command.

**Note:** For more information, see the *Installation Guide*.

This option has the following format:

SUBCHEK

## SUBLIB

The SUBLIB option allows the SUBMIT option to be specified together with the PDS, PANVALET, or LIBRARIAN option so that CA JCLCheck submits jobs from a library.

This option has the following format:

```
SUBLib|NOSUBLib
```

Caution must be used with this option because improper specification of library member names in the PDS, PANVALET, or LIBRARIAN option may cause many more jobs (possibly the whole library) to be submitted.

## SUBMIT

The SUBMIT option automatically submits error-free job streams that are then written to the SYSRDR DD statement.

This option has the following format:

```
SUbit(Even|NOEven)|NOSUbit
```

**Default:** NOSUbit

### Even

Causes CA JCLCheck to submit all jobs, regardless of errors, thereby suppressing the production of a JES2 control statement.

- Installations with other spooling programs can use EVEN.
- Unless you specify the SUBLIB option, jobs are not submitted by the SUBMIT option with either the PDS, PANVALET, or LIBRARIAN options. Use this to protect against inadvertently submitting a large number of jobs in error.
- The SYSRDR DD statement (or alternate DD named by the RDR option) must describe the output destination for error-free jobs processed under this option. If this DD statement describes an internal reader, the jobs are automatically submitted to JES.
- Jobs with errors are written to the point of the error and purged with a JES2 (or HASP) /\*PURGE control statement (or /\*DEL in non-z/OS systems).

## SUPCAT

The SUPCAT option suppresses messages CAY6081E, CAY6080W, and CAY6129E when a DD has DISP=(,CATLG) for a data set that is already cataloged.

This option has the following format:

SUPCat |NOSUPCat

Some installations have a module on the system to re-catalog the data set instead of the normal course, which is to retain the previous catalog entry.

## SUPRPT

The SUPRPT option conditionally suppresses report generation based on the highest severity specified.

This option has the following format:

SUPRpt (*severity*) |NOSUPrpt

**Default:** NOSUPrpt

***severity***

Indicates the severity level at which reports are suppressed.

### Example: Suppress report generation

The following example indicates that any job with a severity less than or equal to four is suppressed.

SUPRPT(4)

## SXREF

The SXREF option activates the related job data set existence checking and reporting feature. This option specifies that the job streams CA JCLCheck analyzes are related jobs that run in succession.

This option has the following format:

SXref (NOMEM|Rpt |NORpt |Only) |NOSXref

**Default:** NOSXref

**NOMEM**

Suppresses PDS member names from the Summary Dataset Cross Reference Report.

**RPT**

Activates the related job data set validation feature and generates all selected reports (per XREF, PXREF, RPTRPT options) at both the individual job and summary (system) level.

**NORPT**

Activates the related job data set validation feature but suppresses all summary reports.

**ONLY**

Activates the related job data set validation feature and generates all selected reports (per XREF, PXREF, RPTRPT options) at only the summary (system) level (Reports 7, 8, 9). Individual job reports, Report 3 - Data Set Cross-Reference, Report 4 - Program Cross-Reference, and Report 5 - Report Listing are *not* produced.

Under this feature, CA JCLCheck remembers catalog and data set updates from job to job, simulating the execution of a system of related jobs. If a job creates and catalogs a data set and a subsequent job references the same data set using the catalog, CA JCLCheck interprets the reference as correct. When the SXREF option is not specified, the catalog data set updates from job to job are not maintained. This action can result in invalid error messages when jobs are related.

CA JCLCheck offers up to three summary reports that are associated with this feature:

- Report 7 - Summary Data Set Cross-Reference Report describes all data sets referenced in the system, the series of related jobs input to CA JCLCheck.
- Report 8 - Summary Program Cross-Reference Report describes all programs used in the system.
- Report 9 - Summary Report Listing describes all SYSOUT reports that the system produces.

The generation of these reports is controlled by a combination of the SXREF reporting operand and the individual report options (XREF, PXREF, and RPTRPT).

## SYNTAX

The SYNTAX option causes CA JCLCheck to perform syntax checking of JCL. Syntax checking validates the JCL being scanned for compliance with the rules set forth by IBM.

This option has the following format:

SYNTAX|NOSyntax

**Default:** SYNTAX

Specifying both the SYNTAX and RUNTIME options results in a normal CA JCLCheck execution with both syntactical and runtime checking. Specifying SYNTAX and NORUNT performs syntactical checking only. Specifying NOSyntax and NORUNT results in only the CA JCLCheck options being printed; the JCL is not checked. Use this combination if you need to know what options are in effect prior to running CA JCLCheck.

Your site's parameter and standard exits are called when syntax checking is active.

## SYSCHK

The SYSCHK option activates support for invoking CA JCLCheck from the IEFUJV Job Validation SMF exit point.

For compatibility purposes, the SYSCHK option is still accepted; however, we recommend that you use the JESCHK option.

**Note:** For more information, see the JESCHK option in the *Installation Guide*.

This option has the following format:

SYSCHK

## SYSID

The SYSID option places CPU ID associated with the job on the CA APCDOC database. This option is only valid when running CA JCLCheck with CA APCDOC.

This option has the following format:

SYSID|NOSYsid

**Default:** NOSYsid

## SYSTEM

The SYSTEM option limits CA JCLCheck to only those jobs within the specified CA APCDOC systems. This option is valid only when running CA JCLCheck with CA APCDOC. You can specify up to five CA APCDOC systems.

This option has the following format:

SYSTEM(*system1*... *system5*) |NOSYSTEM

***system1...system5***

Defines the CA APCDOC system names.

## SYSUID

The SYSUID option specifies the user ID value that CA JCLCheck uses in replacement of the system symbolic parameter, SYSUID.

This option has the following format:

SYSUID(*userid*) |NOSYSUID

**(*userid*)**

Defines the replacement value for the system symbolic &SYSUID for CA JCLCheck to use.

CA JCLCheck replaces the symbolic variable SYSUID as follows:

- When the JOB statement contains USER=*userid*, CA JCLCheck uses the coded value as the replacement value for &SYSUID.
- When USER=*userid* is not coded on the JOB statement, CA JCLCheck uses the value the SYSUID runtime option specifies as the replacement value.
- When USER=*userid* is not coded in the JOB statement and SYSUID is not specified, CA JCLCheck attempts to determine the user ID dynamically.

## TERM

The TERM option generates the terminal report file, containing only the erroneous statements and their associated messages.

This option is only available in an ISPF environment.

This option has the following format:

Term[ (*ddname*) ] | NOTerm

**Default:** Term(SYSTEMM)

***ddname***

Provides for specification of an alternate ddname for the report file. If you omit the ddname, CA JCLCheck uses the SYSTEMM DD statement.

The report produced by this option is designed to route to a terminal. The output file has a logical record length of 80 and no carriage control characters. This report is useful when a time-sharing user invokes CA JCLCheck.

## TLMS

The TLMS option requests support of the CA TLMS tape library management system.

This option has the following format:

TLms(*nn*) | NOTLms

**Default:** NOTLms

***nn***

Defines the version of TLMS that you are using. Entering the version number causes CA JCLCheck to validate that version of the product on your system. Specify TLMS(55) for CA TLMS r5.5, 11.0, and higher.

This option causes CA JCLCheck to access the Volume Master File (VMF) to retrieve volume, DCB, and data set sequence information for tape data sets.

- To use this option, the required CAZ1TLMS routine must be assembled and linked during CA JCLCheck installation.
- The Volume Master File (VMF) must be allocated to the task using this option. The ddname is CAIVMFI.

## TMS

The TMS option activates CA 1 support for tape data set validation.

This option has the following format:

TMs (*nn*) | NOTMs

**Default:** NOTMs

***nn***

Defines the version of TMS that you are using. Entering the version number causes CA JCLCheck to validate that version of the product on your system.

Specify TMS(52) for CA 1 R5.2, 11.0 and 11.5.

Specify TMS(C0) for CA 1 CA 1 R120.

This option causes CA JCLCheck to access the tape management catalog (TMC) to retrieve volume, DCB, and data set sequence information for tape data sets.

To use this option, the required CAZ1TMS5 routine must be assembled and linked during CA JCLCheck installation.

## TSOE

The TSOE option activates TSOLINK. TSOLINK is a TSO Service Facility for EDCHEK, SUBCHEK, ISPF, and JCLCHECK (as a Command Processor). Use TSOLINK (only available under the IBM product TSO EXTENSIONS (TSO/E)) to obtain the authorization that is required for certain types of options, such as AUTOPROC.

This option has the following format:

TSoe | NOTSoe

**Default:** TSoe

CA JCLCheck automatically tries to identify whether you have TSO/E installed at your site. However, it cannot do this task at your site when you have a modified IKJEFTSR module. If this is the case, use this option.

CA JCLCheck ignores this option for Batch, CA Roscoe, and VM/CMS.

## USER

The USER option specifies the user ID to use with security checking.

This option has the following format:

```
USER(userid) | NOUSER
```

**Default:** NOUSER

Use this option with the SECURITY option to override the user ID of the person using CA JCLCheck.

## USERDATA

The USERDATA option allows the user to specify a data string of one to eight characters. This data is made available to all user exits and the REXX interface. The value that is specified is at the discretion of the user and CA JCLCheck does not use or verify the value.

This option has the following format:

```
USERData(string) | NOUSERDA
```

**Default:** None

## USRMSG

The USRMSG option issues the CAY6088E message only when USER is coded on a JOB statement, not the PASSWORD parameter.

This option has the following format:

```
USRMsg | NOUSRmsg
```

## UTLSEV

The UTLSEV option downgrades messages CAY6060 through CAY6085 to warnings (SEV=4) when issued for utility control statements.

This option has the following format:

```
UTLsev | NOUTLsev
```

The absence of this option causes the message severity to be issued as specified in JCLMSG.

## VIOPLUS

The VIOPLUS option indicates that the VIOPLUS product is being used.

This option has the following format:

VIOpLus|NOVIOpLs

**Default:** NOVIOpLs

If you specify this option and CA JCLCheck encounters a VIOCTL DD, CA JCLCheck validates the first qualifier of the data set name which must be either *Activate* or *Suppress*.

## VSAM

The VSAM option requests support of VSAM catalog management. This option is mutually exclusive and cannot be used together with the CSI option.

This option has the following format:

Vsam[ (Reverse|NORevers) ] |NOVsam

**Default:** NOVsam

You can specify an operand to direct the order of search of the private and system catalogs:

### Reverse

Causes CA JCLCheck to search the catalogs in reverse order (that is, the system catalog first, then the private catalog). This feature improves performance for jobs whose data sets are cataloged most often in the system catalog. However, exercise caution with this feature since its use produces incorrect results for data sets cataloged in both the system and the private catalog.

### NORevers

Requests normal search order.

**Default:** NOEVERS

This option causes CA JCLCheck to invoke the IBM utility IDCAMS when it encounters a JOBCAT or STEPCAT statement defining a VSAM catalog. If IDCAMS accepts the catalog, the utility is called instead of standard catalog management during the scope of the job or step catalog.

Without the VSAM option, CA JCLCheck handles VSAM data sets correctly only if its own JCL contains a JOBCAT or STEPCAT statement defining the correct catalog.

- In non-z/OS systems, the VSAM catalog CA JCLCheck searches must be defined in a JOBCAT or STEPCAT statement in the CA JCLCheck JCL. The VSAM option is of limited use in such systems.
- Due to the large storage requirements of IDCAMS, the minimum region size for CA JCLCheck with this option is 320 KB. Failure to specify an adequate region results in an S804 ABEND the first time a JOBCAT or STEPCAT statement is scanned.
- IDCAMS is a relatively slow utility and its use can adversely affect response time when CA JCLCheck is invoked from a time-sharing terminal.
- IDCAMS opens the catalog for an update, so appropriate clearance for CA JCLCheck is required for this option to prevent security errors.

## XEROX

The XEROX option provides syntactical validation of XPAF parameters on the output statement.

This option has the following format:

XERox|NOXERox

**Default:** NOXERox

## XREF

The XREF option requests Report 3 - Data Set Cross-Reference Report. This report provides various information for all data sets in use by the job that exists for more than one step.

This option has the following format:

Xref(NOMEM)|NOXref

**Default:** XREF

### **NOMEM**

Suppresses PDS member names from the Dataset Cross Reference Report.

## XREFAL

The XREFAL option causes CA JCLCheck to save all references to a data set in a job in the corresponding cross-reference database record.

This option has the following format:

```
XREFAL | NOXREFAL
```

Normally, only references which change the status of the data set are recorded in the Cross-Reference database (for example, DEFINE, DELETE, and CATALOG). In any event, only the first 20 references to a file are recorded for any job in the Cross-Reference database data set record. This feature must be set when running combined CA APCDOC and CA JCLCheck. This option is valid only when CA APCDOC is also installed.

## Runtime Options Syntax Summary

The following table contains a summary of the runtime options syntax:

Option Name	Default Setting	Notes
ACF2/NOACF2	CA JCLCheck attempts to dynamically determine the presence of ACF2 on the system. If ACF2 is active, ACF2 is the default; otherwise, NOACF2 is the default.	ACF2 must be active.
ACtion/NOAction	NOAction	Requires CA APCDOC.
ALLOWmnf/NOALLmnf	NOALLmnf	Related options: INCLUDE, PDS, LIBRARIAN, and PANVALET.
APPEND(AFTER, <i>jobname</i> )   (BEFORE, <i>jobname</i> )/NOAPPEND	NOAPPEND	Requires CA APCDOC.
ASM2/NOASm2	NOASm2	APF authorization is required. Specify this option if you run ASM2.
AUDit/NOAUDit	NOAUDit	
AUtoproc/NOAUtopr	NOAUtopr	APF authorization is required. Mutually exclusive with: NORUNTIM and PROC.
BLKMsg/NOBLkmsg	NOBLkmsg	
BYPRout/NOBYprou	NOBYprou	

Option Name	Default Setting	Notes
BYPRUn( <i>suboption1-suboption12</i> ) /NOByprun	NOByprun	Mutually exclusive with: NORUNTIM.
CA11( <i>stepname.procstep</i> )/NOCA11	NOCA11	Related Option: SUPCAT.
CA7([[ <i>schedule-id</i> ] [ <i>date</i> ] [ <i>time</i> ]]) /NOCa7	NOCa7	Use with the CA7CTL option.
CA7Ctl/NOCA7Ctl	NOCA7Ctl	Use with the CA7 option.
CADriver/NOCADriv	NOCADriv	Requires CA 7 or CA Scheduler.
CClist( <i>number-of-cards</i> )/NOCclist	NOCclist	
CHAnge( <i>target,dsname,member</i> ) /NOCHAnge	NOCHAnge	
CNTL/NOCNTL	NOCNTL	
CNTLOVdD/NOCNTLDD	NOCNTLDD	Requires option: CTLSCAN. Mutually exclusive with: CNTLOVER.
CNTLOVEr/NOCNTLER	NOCNTLER	Requires option: CTLSCAN. Mutually exclusive with: CNTLOVDD. Multiple CNTLOVER options allowed.
COBexit( <i>module-name</i> )/NOCOBoI	NOCOBoI	Not Available under ROSCOE. If running APF AUTH, exit must be in APF library also.
CONd[[ <i>step.procstep code</i> <i>abendcc</i> ]]/NOCONd	NOCONd	Multiple COND options allowed.
CSI/NOCSi	CSI	Mutually exclusive with: INDEX, LOCATE, REVINDEX, PVTCAT, STPCAT, SUBALLOC, and VSAM options.
CTIscan(RPT1list)/NOCTI	NOCTI	
DATEfmt(Euro)/NODAtefm	NODAtefm	Use with the RUNDATE option.
DB2( <i>subsystem,planname</i> )/NODB2	DB2 is the default if a DB2 subsystem is active; otherwise, NODB2 is the default.	
DBCS	NODBCS	
DBGjcl/NODBGjcl	NODBGjcl	Use at the request of CA Support.

Option Name	Default Setting	Notes
DBHLQ ( <i>high-level-qualifier</i> ) /NODBHLQ	NODBHLQ	Use with the DICT option. Requires CA APCDOC Version 1.1 and above.
DCBChk/NODCbchk	NODCbchk	
DEBug[[ <i>error-msg-number</i> ] [ <i>occurrence-number</i> ]]/NODEbug	NODEbug	APF authorization is required. Typically used with CA Support.
DELTA/NODELTA	NODELTA	Requires CA APCDOC.
DEQue[(List NOList)]/NODEQue	NODEQue	APF authorization is required. Use at the request of CA Support. Use in batch or under CA Roscoe – not under TSO.
DESTchk[(Jes NOJes)]/NODEStch	NODEStch	APF authorization is required.
DFLTJOBC/NODFltjc	The default used is CLASS(A).	Use with the AUTOPROC option.
Dlct/NODlct	NODlct	Requires CA APCDOC.
DICTDd( <i>ddname</i> )/NODICTdd	The default <i>ddname</i> is SYSDICT.	Requires CA APCDOC.
DIRUPD/NODIRUPD	NODIRUPD	Use with the DELTA option. Requires CA APCDOC.
DISPATCH (R M A)/NODISpat	NODISpat	
DSNErr/NODSnerr	NODSnerr	
DUmpucb/NODUmpuc	NODUmpuc	Use at the request of CA Support.
DYNexits(exit1,...exit9)/NODYNexi	NODYNexi	
EAsyproc([ONLY])/NOEasyrd	NOEasyrd	Same as the EASYRDR option.
EAsydr/NOEasyrd	NOEasyrd	Same as the EASYPROC option.
EDCHEK	There is no default.	Valid only in a TSO/ISPF session.
EDCHEL/NOEDCHEL	NOEDCHEL	Valid only under EDCHEK.
EDCHKLb/NOEDCHKL	NOEDCHKL	Valid only under EDCHEK.
EDCHKSI/NOEDCHKS	NOEDCHKS	Valid only under EDCHEK.
Error[[ <i>Start</i> ] [ <i>Inline</i> ] [ <i>End,Nor6head</i> ]] /NOERror	By default, error messages are printed in Report 6 only.	

Option Name	Default Setting	Notes
EXCLude( <i>member1...member10</i> ) /NOEXclud	There is no default.	Use only with the PDS, LIBRARIAN, or PANVALET options. Multiple EXCLUDE options allowed.
FAcom( <i>iefname</i> )	NOFAcom	Valid only on Hitachi and Fujitsu systems.
FLUSHrc( <i>nn</i> )/NOFLushr	FLUSHrc(64)	
Fulllist/NOFull	Fulllist	
GRaph[( <i>ddname</i> ,Left Right)] /NOGraph	NOGraph. However, if only GRAPH is specified, the default DD name is SYSGRAPH and the existing data sets are printed on the left side.	
HCD/NOHCD	HCD is the default for IBM z/OS operating systems. NOHCD is the default for Fujitsu and Hitachi operating systems.	APF authorization is required.
IAM/NOIAM	If IAM is active, IAM is the default; otherwise, NOIAM is the default.	To obtain the OWNER value, specify the VSAM option. To obtain the SMS storage class, do not specify the SMS or NOSMS options.
IEBmsg/NOIEbmsg	NOIEbmsg	
IMS/NOIMS	NOIMS	
INCLude( <i>member1... member10</i> ) /NOINclud	There is no default.	Use only with the PDS, LIBRARIAN, or PANVALET options. Multiple INCLUDE options allowed.
INDex( <i>hi-index1</i> [, <i>hi-index2</i> , ..., <i>hi-index16</i> ])/NOINDEX	NOINDEX	Mutually exclusive with: CSI.
INPut[( <i>ddname</i> )]/NOINPut	INPut(SYSIN)	
INTusr/NOINTusr	If omitted or if NOINTusr is specified, only the \$USER field is initialized.	
JCLlib/NOJCLlib	JCLlib	

Option Name	Default Setting	Notes
JES3/NOJES3	CA JCLCheck attempts to dynamically determine the presence of JES3 on the system. If the JES3 subsystem is active, JES3 is the default; otherwise, NOJES3 is the default.	
JESChck	There is no default.	Valid only in a JES2 environment.
JESCRunt/NOJESCru	NOJESCru	Valid only in a JESCHECK environment.
JOB/NOJob	JOB	
JOBCARD( <i>nn</i> )/NOJOBCar	JOBCARD(1)	
JOBJBRAM( <i>joblist</i> )/NOJOBJbr	NOJOBJbr	Requires CA APCDOC.
JOBSEV/NOJOBSEV	JOBSEV	
JOBType( <i>type</i> )/NOJOBTyp	JOBType(J)	
KEEP7380/NOKEep73	NOKEep73	Valid only under EDCHEK.
KILLsep/NOKIllse	NOKIllse	
LIBLIST( <i>lblast</i> )/NOLlblis	NOLlblis	Requires CA APCDOC.
LIbrarian([[ <i>member1</i> ]...[ <i>membern</i> ]])/NOLlBrar	NOLlBrar	Mutually exclusive with: PANVALET and PDS.
LINecnt[( <i>number-of-lines</i> )]/NOLINecn	LINecnt(60)	
LISt( <i>ddname</i> )/NOList	In batch, the default is LIST(SYSPRINT). In all other modes, the default is NOList.	
LOcate/NOLOcate	LOcate	Mutually exclusive with: CSI.
MCOsSys( <i>dsname</i> ,[ <i>membername</i> ])/NOMCOsSys	NOMCOsSys	
MCOUsr( <i>ddname</i> )/NOMCOUsr	NOMCOUsr	If MCOUSR is specified without a <i>ddname</i> , MCOUSR(MCOUSR) is used.
MIGvol( <i>volumename</i> )/NOMIgvol	By default, CA JCLCheck recognizes volume names MIGRAT, ARCHIV, and ARCIVE. If the option is omitted or NOMIgvol is specified, @@@@ is used.	
MSGLevel(Info/Error)/NOMSGLvl	MSGLevel(I)	

Option Name	Default Setting	Notes
Mss/NOMss	NOMss	
NETWORK( <i>netname</i> )/NONetwor	NONetwor	Requires CA APCDOC. Same as the SYSTEM option.
NJBchk/NONJbchk	NONJbchk	Valid only for JES3 users.
NOSPie	By default, CA JCLCheck issues a SPIE.	Typically used with CA Support.
ONEmem/NOONemem	NOONemem	Use only with the PDS, LIBRARIAN, or PANVALET options.
OPRoc( <i>ddname</i> )/NOOProc	NOOProc	Mutually exclusive with: OPROCLIB.
OPROCLib/NOOPrclb	NOOPrclb	Mutually exclusive with: OPROC. Multiple OPROCLIB options allowed.
Options[( <i>ddname</i> )]/NOOption	NOOption If OPTIONS is specified without the DD name, the default DD name is JCLOPTS.	
OUTfmt(Csv/Pdf)/NOOUTfmt	NOOUTfmt	Valid only in batch.
PANExec/NOPANE	NOPANE	Does not function under CA Roscoe.
PANvalet([ <i>member1</i> ].. <i>membern</i> )] /NOPAn	NOPAn	Mutually exclusive with: LIBRARIAN and PDS.
PASsword( <i>character-string</i> ) /NOPASswo	NOPASswo	Use only with the VSAM option.
PDS([ <i>member1</i> ].. <i>membern</i> )] /NOPDS	NOPDS	Mutually exclusive with: LIBRARIAN and PANVALET.
PDSRelate/NOPDSRel	NOPDSRel	Use only when the SXREF option and one of the following library input options are specified: LIBRARIAN, PANVALET, and PDS.
PNFmsg/NOPNfmsg	NOPNfmsg	
PRCDELTA/NOPRDELta	NOPRDELta	Use only with the DELTA option. Requires CA APCDOC.
PREDNO/NOPREDNO	NOPREDNO	Requires CA APCDOC Version 1.3 and above.

Option Name	Default Setting	Notes
PREfix( <i>character-string</i> )/NOPRefix	PREfix(SYSDD)	Only used in non-z/OS systems and with JES3 /*MAIN PROCLIB support.
PREVEX(VERIFY,RUNTIME)/NOPREVEX	NOPREVEX	This option forces the setting of the NOAUTOPR and NORUNTIM options.
PROc[( <i>ddname</i> , LIB PAN)]/NOPROC	PROc(SYSPROC)	Mutually exclusive with: AUTOPROC.
PROCDflt/NOPROCDf	NOPROCDf	
PROCXref/NOPROCXr	NOPROCXr	
PULI/NOPUII	NOPUII	Requires CA APCDOC.
PVtcat(AUTH)/NOPVtcat	NOPVtcat	APF authorization is required to use the optional AUTH sub-option.
PXref[(Rpt NORpt)]/NOPXref	PXref(Rpt)	
RDR[( <i>ddname</i> )]/NORDr	RDR(SYSRDR)	Use only with the SUBMIT option.
REMOte/NOREMOte	NOREMOte	APF authorization is required.
REport/NORptrpt	NORptrpt	Requires CA APCDOC. Similar to the RPTRPT option.
REVIndex( <i>hi-index1</i> [, <i>hi-index2</i> ,... <i>hi-index16</i> ])/NOREVind	NOREVind	Mutually exclusive with: CSI.
REXXMSg/NOREXXms	NOREXXms	
ROSCoe	In a CA Roscoe environment, the default is ROSCoe; otherwise, this option is ignored.	Valid only when running in a CA Roscoe environment.
ROSLimit( <i>statement-count</i> )/NOROSlim	NOROSlim	Valid only when running in a CA Roscoe environment.
RPtrpt/NORptrpt	NORptrpt	Similar to the REPORT option.
RUNCpu(CPU_ID)/NORUNcpu	NORUNcpu	Automatically turns on security validation.
RUNDate( <i>mm/dd/yy, hh:mm</i> )/NORUNDt	NORUNDt	Automatically turns on security validation. Valid only for CA ACF2 r6.0 users.

Option Name	Default Setting	Notes
RUNTime/NORUNTim	RUNTime	NORUNTim is mutually exclusive with the AUTOPROC option but there are exceptions.
RV		This option is used internally by CA JCLCheck. You should not set this option.
SCHedler/NOSCHEDI	NOSCHEDI	Valid only when running CA Scheduler.
SCHENV/NOSCHENV	NOSCHENV	APF authorization is required and the RUNTIME option must be set to use this option.
SECurity( <i>options</i> )/NOSECure	NOSECure	APF authorization is required. Designed to work with CA ACF2, CA Top Secret, RACF, or any other SAF-compatible product.
SEv( <i>error-level</i> )/NOSEV	SEv(0)	
SIGNon/NOSIGNon		APF authorization is required.
SMS/NOSMS	By default, if this option is not specified and SMS is active, the JCL will be validated using the active SMS configuration. If SMS is active and you want to use the active SMS configuration, do not specify this option. If SMS is not active and you want to perform JCL checking according to SMS rules, specify the SMS option to simulate SMS processing. If SMS is active and you do not want to perform JCL checking according to SMS rules, specify the NOSMS option. If you do not specify this option, CA JCLCheck determines your SMS environment and selects the appropriate path.	
SPace([[Rpt   NORpt] [ <i>min-percentage</i> ][All   NOAll]]) /NOSPace	NOSPace	
SRCh/NOSRch	NOSRch	
STACK/NOSTACK	NOSTACK	

Option Name	Default Setting	Notes
STandard( <i>standard-name1</i> , ... <i>standard-name8</i> )/NOSTAND	NOSTAND	
STDRExx(REXX EXEC- <i>member-name</i> , [REXX EXEC DD <i>name</i> ])/NOSTDrex	NOSTDrex is the default but if STDRExx is specified and the REXX EXEC DD name is omitted, the standard SYSEXEC DD name is used to find the user-specified REXX EXEC.	
STEPMsg/NOSTEpms	NOSTEpms	
STPCat/NOSTPcat	NOSTPcat	Mutually exclusive with: CSI.
SUBAlloc/NOSUBAll	NOSUBAll	Mutually exclusive with: CSI.
SUBCHEK	There is no default.	This options runs under TSO.
SUBLib/NOSUBLib	NOSUBLib	Use only with the SUBMIT option and with the PDS, LIBRARIAN, or PANVALET options.
Submit(Even NOEven) /NOSubmit	NOSubmit	Use with the PDS, LIBRARIAN, or PANVALET options if the SUBLIB option is specified.
SUPCat/NOSUPCat	NOSUPCat	
SUPRpt( <i>severity</i> )/NOSUPrpt	NOSUPrpt	
SXref(NOMEM Rpt NORpt Only) /NOSXref	NOSXref	Related to the XREF, PXREF, and RPTRPT options.
SYNTAX/NOSyntax	SYNTAX	This option, in conjunction with the RUNTIME option, determines the type of JCL checking performed by CA JCLCheck.
SYSChk	There is no default.	This option should not be used. Specify the identical JESCHK option instead.
SYSid/NOSYsid	NOSYsid	Requires CA APCDOC.
SYSTEM( <i>system1</i> ... <i>system5</i> ) /NOSYSTEM	There is no default.	Requires CA APCDOC. Same as the NETWORK option.
SYSUID( <i>userid</i> )/NOSYSuid		When not specified, CA JCLCheck attempts to dynamically determine the user ID.
TERM[( <i>ddname</i> )]/NOTerm	Term(SYSTEM)	Valid only when running in a TSO/ISPF environment.

Option Name	Default Setting	Notes
TLms( <i>nn</i> )/NOTLms	NOTLms	Use with the CA TLMS tape library management system.
TMs( <i>nn</i> )/NOTMs	NOTMs	Use with CA 1.
TSOE/NOTSoe	TSOE	CA JCLCheck attempts to determine whether you have TSO/E installed at your site. Use with the EDCHEK, SUBCHEK, and AUTOPROC options. Ignored when running CA JCLCheck in batch, CA Roscoe, and VM/CMS.
USER( <i>userid</i> )/NOUSER	NOUSER	Use with the SECURITY option.
USERData( <i>string</i> )/NOUSERDA	There is no default.	
USRMsg/NOUSRmsg	NOUSRmsg	
UTLsev/NOUtlsev	NOUtlsev	
VIOplus/NOVIOpls	NOVIOpls	Use with the VIOPLUS product.
Vsam[(Reverse   NORevers)]/NOVsam	NOVsam	Mutually exclusive with: CSI. CA Technologies recommends you use the CSI option instead of this option.
XERox/NOXERox	NOXERox	Use when the OUTPUT statement may contain XPAF parameters.
Xref(NOMEM)/NOXref	Xref	
XREFAL/NOXREFAI	NOXREFAI	Requires CA APCDOC.



# Chapter 4: CA JCLCheck Reports

---

This chapter provides an overview of the CA JCLCheck reports you can receive after JCL is processed. CA JCLCheck produces this first group of reports for each job in the JCL input stream:

- [Report 1 - Listing of Jobstream JCL](#) (see page 226)
- [Report 2 - Listing of Merged JCL](#) (see page 226)
- [Report 3 - Data Set Cross-Reference](#) (see page 229)
- [Report 4 - Program Cross-Reference](#) (see page 232)
- [Report 5 - Report Listing](#) (see page 233) (SYSOUT reports)
- [Report 6 - Error Messages](#) (see page 234)

Using the SXREF option, you can produce three summary reports in addition to job-oriented reports. These summary reports document the entire input production system. In this context, the production system refers to the series of jobs CA JCLCheck analyzes during a single execution. The summary reports include:

- [Report 7 - Summary Data Set Cross-Reference](#) (see page 236)
- [Report 8 - Summary Program Cross-Reference](#) (see page 238)
- [Report 9 - Summary Report Listing](#) (see page 239) (SYSOUT reports across all jobs)

A one-page listing of CA JCLCheck message output precedes the reports for the first job. This listing displays the level of CA JCLCheck in use and the options selected for the current execution. At the bottom of the last report, a message indicating NORMAL END OF CA JCLCHECK is displayed.

If you specify the GRAPH option to produce flow diagrams, the JCL is charted in standard job flow format, suitable for use in a runbook.

If you specify the AUDIT option you will produce an Audit Report with a statistical summary, input records, and averages of all related jobs, steps, and DD statements. The GRAPH and AUDIT options produce the following reports:

- [Report 10 – Flow Diagram](#) (see page 240)
- [Report 11 – Procedure Cross-Reference](#) (see page 244)
- [Audit Report](#) (see page 245)



CA		REPORT 2 - LISTING OF MERGED JCL			THURSDAY JANUARY 9, yyyy	
CA JCLCHECK		SYS: xxxxx	JOB: JCHKTEST		PGMR: CAI-JB0	
STMT NUM	STEP NAME	PROCSTEP NAME	ERR SEV	CMNT COL	STATEMENT TEXT	
1.					//JCHKTEST JOB (40100000), 'CAI-JB0'	
2.					// JCLLIB ORDER=CAI.J660.PROCLIB	
3.					/*OUTPUT 1234 F=0004,DEST=TSC32702	
4.					//TESTPROC PROC D=,U=3380,V=,PROG=IEBGENER	
					//STEP1 EXEC PGM=IEFBR14,PARM='01/03/2000'	
					//INPUTT DD DSN=&D,UNIT=&U,VOL=SER=&V,DISP=SHR	
					//DOCDB DD DSN=CAI.MTLAUREL.CAAPDOC.DOCDICT,DISP=SHR	
					//STEP12 EXEC NESTPROC	
					//NSTEP2.SYSPRINT DD SYSOUT=(J,,1234),COPIES=4	
					// PEND	
5.					// SET GDG='BAD.GDGBASE',D='CAI.J660.SYSTMST'	
6.	PROC		8		//PROC EXEC TESTPROC	
7.	PROC			L01	++TESTPROC PROC D=,U=3380,V=,PROG=IEBGENER	
8.	PROC	STEP1		L01	++STEP1 EXEC PGM=IEFBR14,PARM='01/03/91'	
9.	PROC	STEP1		L01	++INPUTT DD DSN=&D,UNIT=&U,VOL=SER=&V,DISP=SHR	
					SUBSTITUTION JCL - DSN=CAI.J660.SYSTMST,UNIT=3380,VOL=SER=,DISP=SHR	
10.	PROC	STEP1		L0	++DOCDB DD DSN=CAI.MTLAUREL.CAAPDOC.DOCDICT,DISP=SHR	
11.	STEP12			L01	++STEP12 EXEC NESTPROC	
12.	STEP12			L02	++NESTPROC PROC	
13.	STEP12	NSTEP1		L02	++NSTEP1 EXEC PGM=IEFBR14	
14.	STEP12	NSTEP1		L02	++INPUTT DD DSN=CAI.J660.SYSTMST,DISP=SHR	
15.	STEP12	NSTEP2		L02	++NSTEP2 EXEC PGM=&PROG	
					SUBSTITUTION JCL - PGM=IEBGENER	
16.	STEP12	NSTEP2		L02	++ INCLUDE MEMBER=GENERDUP	
17.	STEP12	NSTEP2		L01	XXNSTEP2.SYSPRINT DD SYSOUT=(J,,1234),COPIES=4	
18.	STEP12	NSTEP2		L02 01	I/SYSPRINT DD SYSOUT=*	
19.	STEP12	NSTEP2		L02	++SYSUT1 DD DSN=*NSTEP1.INPUTT,DISP=SHR	
20.	STEP12	NSTEP2	8	L02	++SYSUT2 DD DSN=&GDG(+1),DISP=(,CATLG,DELETE),UNIT=&U,VOL=SER=&V,	
					SUBSTITUTION JCL - DSN=BAD.GDGBASE(+1),DISP=(,CATLG,DELETE),UNIT=3380,VOL=SER	
				L02	++ SPACE=(TRK,(30,30),RLSE,,ROUND),DCB=CAI.J660.SYSTMST	

This report contains the following fields:

**STMT NUM**

Indicates the internally assigned statement number.

Use this number for reference in all CA JCLCheck error messages.

**STEP NAME**

Indicates the name of the most recent EXEC PGM= or EXEC PROC= statement.

This field is blank if the statement had no name.

#### PROCSTEP NAME

Indicates the name of the most recent EXEC PGM= statement within the current procedure.

If the current step is not from a procedure or had no name field, this column is blank.

#### ERR SEV

Indicates the severity code of the most serious error associated with this statement.

Standard values include the following:

**4**

Indicates a warning

**8**

Indicates an error

**12**

Indicates a serious error

**Note:** This may have prevented diagnosis of other errors.

The text of the error message is contained in Report 6 - Error Messages unless ER(I) is specified. This field is blank if no errors were associated with the statement.

#### CMNT COL

Indicates the number of the column where comments start.

This field is intended to bring the existence of comments to the attention of the CA JCLCheck user, so as to ensure that no blanks have been erroneously inserted. This field is blank if the JCL statement contains no comments or if it begins with `//*`.

#### STATEMENT TEXT

Indicates the text of the JCL statement.

The first two TEXT characters of the statement indicate its source:

`//` = Jobstream JCL

#### LEVEL #

Indicates the level number (*Lnn nn*) within the STATEMENT TEXT.

This is the nesting level of called JCL. *Lnn* is the nested procedure level number, and the next *nn* is the INCLUDE group nested level number. In addition to the nested level number is a two-character source code:

`XX` = JCL from a procedure library

`X/` = Overridden procedure library JCL

`++` = JCL from an instream procedure

`+/` = Overridden instream procedure JCL

`IX` = JCL from an INCLUDE member

`I/` = Overridden INCLUDE JCL

## Report 3 - Data Set Cross-Reference

The Data Set Cross-Reference report lists all data sets and members used by the job, and that exist for more than one step. Included in the report are all permanent data sets and all temporary data sets with a disposition other than NEW, DELETE.

To generate this report, specify the XREF option.

CA		REPORT 3 - DATA SET CROSS REFERENCE						PAGE 3	
CA JCLCHECK	SYS: xxxx	JOB: BK042005	PGMR: CAUSER02					WEDNESDAY SEPTEMBER 12, yyyy	
DATA SET NAME	SPACE	MEMBER	DSET	DCB INFO	DEFN	DELETE	CATLG	REFERENCED	
			VOL(S)	TYPE	FMT	LRECL, BLK	STMT	IN STATEMENTS	
KNEB001.\$J6ESTG.LISTING			>TSU028	PDSE	F	133,3325		3	
		DATACLAS: DCWRKD		STORCLAS: TSO			MGMTCLAS: TSO		
KNEB001.MYLIB.N		??????						8	
		ASMDSECT						9	
		CONVDATE						6	
		CONVTIME						11	
		IEBGENER						4	
								5	
KNEB001.REP.VOLSER2		??????						12	
KNEB001.REPORT#3			>TSU014	PDSE	F	80,3120		7	
		ASM#ONLY							
		DATACLAS: DEFAULT		STORCLAS: TSO			MGMTCLAS: TSO		
*****									

This report contains the following fields:

### DATA SET NAME

Indicates the data set name of the data set.

Generation data group members are listed with the relative generation numbers in existence before the job executes. Temporary data sets are listed with a data set name beginning with a single ampersand (&). For unnamed data sets included on this report, CA JCLCheck generates a data set name in the following format:

&nnnn

nnnn

Indicates the statement number of the statement creating the data set.

### SPACE

Indicates the space allocation for the data set.

### MEMBER

Indicates the member name of the data set.

### **VOL(S)**

Indicates the volume serial numbers of the volumes containing the data set.

Direct-access data sets created by this job with no volume explicitly assigned (including all temporary data sets) are arbitrarily assigned to the first eligible volume associated with their unit type. (When the job is actually executed, the system may choose a different volume assignment.) VOL is shown as \*VIO\* if the data set is eligible for allocation on a VIO device and as \*MSS\* for a nonspecific mass storage volume.

### **DSET TYPE**

Indicates the data set type:

#### **ALIAS**

Indicates an alias data set

#### **CART**

Indicates a cartridge

#### **DASD**

Indicates a non-VSAM, non-PDS disk data set

#### **GDGB**

Indicates a base GDG entry

#### **ISAM**

Indicates an ISAM data set

#### **PDS**

Indicates a partitioned data set

#### **PDSE**

Indicates PDS extended

#### **PDSX**

Indicates PDSE/X

#### **TABLE**

Indicates a DB2 table

#### **TAPE**

Indicates a tape data set

#### **VSAM**

Indicates a VSAM data set

**VSAM-D**

Indicates a data component

**VSAM-I**

Indicates an index component

**DCB INFO**

Indicates the DCB of this data set, including the following:

- Record format (F = fixed, V = variable, U = undefined)
- Logical record length
- Block size

If the data set has a logical record length of 0, only the block size is displayed. DCB information is taken from the DD statement for data sets created or modified in this job and from the data set label for other (previously existing) data sets.

**DEFN STMT**

Indicates the statement number of the DD statement that created this data set.

This field is blank if this job does not create the data set.

**DELETE STMT**

Indicates the statement number of the DD statement that deleted this data set.

This field contains the value EOJ if the data set was created in this job and passed from step to step without ever being kept or cataloged. This field is blank if this job does not delete this data set.

**CATLG STMT**

Indicates the statement number of the DD statement that cataloged this data set.

This field is blank if this job does not catalog this data set.

**REFERENCED IN STATEMENTS**

Indicates the statement numbers of all DD statements that reference the data set, including the creating, deleting and cataloging statements, if applicable.

**DATACLAS**

Indicates the data class associated with the data set.

**STORCLAS**

Indicates the storage class associated with the data set.

**MGMTCLAS**

Indicates the management class associated with the data set.

**Note:** DATACLAS, STORCLAS, and MGMTCLAS are fields that display for output data sets when DFSMS is installed. The values printed in these fields are the derived DFSMS data constructs used for the allocation. For more information, see the IBM documentation for Storage Management Library.

## Report 4 - Program Cross-Reference

The Program Cross-Reference report lists all programs used by the job and the libraries where they reside. This includes all programs referenced on EXEC statements and programs invoked by IMS (specified in the second PARM field parameter when the program name is DFSRRC00).

To generate this report, specify the PXREF option.

CA	REPORT 4 - PROGRAM CROSS REFERENCE		THURSDAY JANUARY 9, yyyy
CA JCLCHECK	SYS: xxxx	JOB: JCHKTEST PGMR: CAI-JB0	
-----	-----		-----
PROGRAM NAME	LIBRARY CONTAINING PROGRAM VOLUME DSNAME	REFERENCED IN STATEMENT (STEP)	
-----	-----		-----
IEBGENER	MVXAD2 SYS1.LINKLIB	15 (STEP12.NSTEP2)	
IEFBR14	SYS1.LPALIB	8 (PROC.STEP1) 13 (STEP12.NSTEP1)	

This report contains the following fields:

**PROGRAM NAME**

Indicates the program name.

**LIBRARY CONTAINING PROGRAM VOLUME**

Indicates the volume serial number of the partitioned data set in which the program is located.

**LIBRARY CONTAINING PROGRAM DSNAME**

Indicates the data set name of the partitioned data set in which the program is located.

The name is enclosed in parentheses if the data set cannot be opened for the existence check feature. The library name is shown as SYSTEM LINK LIBRARY [+nn] if the program is found in the linklist. In such a case, [+nn] indicates that the program is located in the nth library beyond SYS1.LINKLIB. The library name is shown as \*\*\* PROGRAM NOT FOUND \*\*\* if the program cannot be located in any library.

**REFERENCED IN STATEMENT (STEP)**

Indicates the statement numbers of the EXEC statements invoking this program.

The stepname (in the form *step.procstep*) is displayed in parentheses following the statement number.

**Note:** If the program is located in the STEPLIB allocated to the CA JCLCheck task, CA JCLCheck is unable to determine whether the module is really linklisted and in which library it actually resides. For information about this report, see the **PNFMSG** option.

## Report 5 - Report Listing

The Report Listing contains all SYSOUT reports produced by the job. A SYSOUT report is assumed to result from any DD statement specifying the SYSOUT parameter, including those requesting an internal reader. Information for this report is obtained from both the SYSOUT DD statement itself and from any associated JES2 /\*OUTPUT control statement.

To generate this report, specify the RPTRPT option.

STMT NUM	STEP NAME	PROCSTEP NAME	DDNAME	SYSOUT CLASS	SPECIAL FORMS	COPY COUNT	SYSOUT PROGRAM	CA DISPATCH	REPORT DESTINATION(S)
18.	STEP12	NSTEP2	SYSPRINT	J	0004	4	-----		TSC32622

This report contains the following fields:

**STMT NUM**

Indicates the statement number of the DD statement defining this report.

**STEP NAME**

Indicates the name of the most recent EXEC PGM= or EXEC PROC= statement.

This field is blank if the statement had no name.

**PROCSTEP NAME**

Indicates the name of the most recent EXEC PGM= statement within the current procedure.

If the current step is not from a procedure or had no name field, this column is blank.

**DDNAME**

Indicates the name of the DD statement defining this report.

**SYSOUT CLASS**

Indicates the SYSOUT class of this report.

This field is eight bytes long to allow for JES3 SYSOUT classes.

**SPECIAL FORMS**

Indicates the special forms number of the forms for this report.

This field is blank if you want the report to print on standard forms.

**COPY COUNT**

Indicates the number of copies requested of this report.

This field is blank if only one copy is requested.

**SYSOUT PROGRAM**

Indicates the name of the special processing program (for example, INTRDR) processing this report.

This field is blank if the report is not processed by any special program.

**CA DISPATCH**

YES indicates that the report is under the control of CA Dispatch.

**REPORT DESTINATION(S)**

Indicates the destination or destinations specified for this report.

This field is blank if default routing was requested.

## Report 6 - Error Messages

The Error Messages report lists all error messages associated with this job. It is produced only if one or more errors are detected. Direct this report to appear at the beginning or end of the report output or incorporate it with Report 2 - Listing of Merged JCL, depending on the setting of the ERROR option.

To generate this report, specify the ERROR option.

CA			REPORT 6 - ERROR MESSAGES	
CA JCLCHECK		SYS: xxxx	JOB: JCHKTEST PGMR: CAI-JB0	THURSDAY JANUARY 9, yyyy
-----				
STMT	MESSAGE			
NUM	NUMBER	MESSAGE TEXT		
-----				
6.	CAY7019E	DD STATEMENT 'SYSIN' IS REQUIRED BY PROGRAM 'IEBGENER' BUT WAS NOT PROVIDED		
20.	CAY7086E	GDG INDEX NOT BUILT		
CAY6000	2	STATEMENTS FLAGGED IN JOB 'JCHKTEST' MAXIMUM SEVERITY WAS 8		

This report contains the following fields:

**STMT NUM**

Statement number of the erroneous statement.

**MESSAGE NUMBER**

CA JCLCheck message number.

This number provides a reference if further explanation of a message is required. CA JCLCheck messages can almost always be interpreted without reference to the manual. The message number includes a suffix indicating the severity of the error.

**MESSAGE TEXT**

The text of the message. Variable text (data set names, for example) is enclosed in single quotes.

At the end of this report, a single CAY6000 message summarizes the total number of statements flagged in error and the maximum severity encountered for the job. The CAY6000 message is issued regardless of the Error Message Report being generated.

**Note:** For error messages and ABEND codes generated by CA JCLCheck, see the *Message Reference Guide*.

## Report 7 - Summary Data Set Cross-Reference

The Summary Data Set Cross-Reference report lists all permanent data sets and members used by any job within a production system (series of jobs analyzed using a single run of CA JCLCheck). It is produced only if both the SXREF and XREF options are selected. The report is similar to Report 3 - Data Set Cross-Reference, except that all data sets used by the related jobs are included. The report is sequenced by data set name.

To generate this report, specify the SX(RPT) and SX(ONLY) options.

CA		REPORT 7 - SUMMARY DATA SET CROSS REFERENCE						PAGE 1			
CA JCLCHECK		SYS: xxxx								WEDNESDAY SEPTEMBER 12, yyyy	
DATA SET NAME	SPACE	MEMBER	VOL(S)	DSET TYPE	DCB INFO	REFERENCED IN					
				FMT	LRECL, BLK	JOB (STATUS)	-	STEP	(STMT)		
KNEB001.\$J6ESTG.LISTING			>TSU028	PDSE	F 133,3325	BK042005 -					
								S1.DDD00	(3)		
KNEB001.MYLIB.N			??????			BK042005 -					
		ASMDSECT						S1.DDD3	(8)		
		CONVDATE						S1.DDD4	(9)		
		CONVTIME						S1.DDD1	(6)		
								S2.DDD2	(11)		
		IEBGENER						S1.DDD1A	(4)		
								S1.DDD1B	(5)		
KNEB001.REP.VOLSER2			??????			BK042005 -					
								S2.DDDX	(12)		
KNEB001.REPORT#3			>TSU014	PDSE	F 80,3120	BK042005 -					
		ASM#ONLY						S1.DDDA	(7)		
*****											

This report contains the following fields:

**DATA SET NAME**

Indicates the data set name of the data set.

Generation data sets are shown with the relative generation numbers as they would exist prior to execution of the first job within the production system.

**SPACE**

Indicates the space allocation for the data set.

**MEMBER**

Indicates the member name of the data set.

**VOL(S)**

Indicates the serial numbers of the volumes on which the data set resides.

Disk data sets created within this production system for which no volume is explicitly specified are arbitrarily assigned to the first eligible volume within their unit. The system may make a different assignment when the job is actually executed. VOL is shown as \*MSS\* for data sets on a nonspecific mass storage volume.

**DSET TYPE**

Indicates the data set type:

**DASD**

Indicates a non-VSAM, non-PDS disk data set

**ISAM**

Indicates an ISAM data set

**PDS**

Indicates a partitioned data set

**TAPE**

Indicates a tape data set

**VSAM**

Indicates a VSAM data set

**CART**

Indicates a cartridge

**DCB INFO**

Indicates the DCB of this data set, including the following:

- Record format: (F = fixed, V = variable, U = undefined)
- Logical record length
- Block size

If the data set has a logical record length of 0, only the block size is displayed. DCB information is taken from the DD statement for data sets created or modified in this system and from the data set label for other (previously existing) data sets.

**REFERENCE IN JOB (STATUS) STEP (STMT)**

Indicates the job and statements referencing this data set.

Each job begins a new line. If the job created, cataloged, or deleted the data set, the values DEFN, CATLG or DELETE, respectively, are listed in parentheses following the job name. The statements referencing the data set are formatted *step.procstep.ddname (stmt)*. This format allows CA JCLCheck to use the report independently of the detailed reports for each job within the system. The inclusion of the statement number permits easy reference to the detailed reports, if wanted.

## Report 8 - Summary Program Cross-Reference

The Summary Program Cross-Reference report lists all programs used by any job within a production system (series of jobs analyzed using a single run of CA JCLCheck). The report is similar to the Report 4 - Program Cross-Reference, except that all programs included in the system are reported in program name sequence. The listing includes all programs referenced on EXEC statements and programs invoked by IMS (specified in the second PARM field parameter when the program name is DFSRRC00). This report is produced only if the SXREF and PXREF options are both selected.

To generate this report, specify the SXREF and PXREF options.

CA			
CA JCLCHECK		SYS: xxxx	REPORT 8 - SUMMARY PROGRAM CROSS REFERENCE
			THURSDAY JANUARY 9, yyyy
PROGRAM NAME	LIBRARY VOLUME	CONTAINING PROGRAM DSNAME	REFERENCED IN JOB, STEP (STATEMENT)
IEBGENER	MVXAD2	SYS1.LINKLIB	JCHKTEST, STEP12.NSTEP2 (15)
IEFBR14		SYS1.LPALIB	JCHKTEST, PROC.STEP1 (8) JCHKTEST, STEP12.N

This report contains the following fields:

### PROGRAM NAME

Indicates the program name.

### LIBRARY CONTAINING PROGRAM VOLUME

Indicates the volume serial number of the partitioned data set in which the program is located.

### LIBRARY CONTAINING PROGRAM DSNAME

Indicates the data set name of the partitioned data set in which the program is located.

The name is enclosed in parentheses if the library could not be opened for the existence check. (See the Error Messages Report for the first job referencing the library for further details.) The library name is shown as SYSTEM LINK LIBRARY [+nn] if the program was found in the linklist. In such a case, "[+nn]" indicates that the program was located in the *n*th library beyond SYS1.LINKLIB. The library name is shown as \*\*\* PROGRAM NOT FOUND \*\*\* if the program could not be located in any library.

### REFERENCED IN JOB, STEP (STATEMENT)

Indicates the statements referencing the program.

References are formatted job name, step.procstep (stmt). This format allows CA JCLCheck to use the report without referencing the detailed reports for the individual jobs within the system. The inclusion of the statement number however, permits easy reference to the detailed reports.

## Report 9 - Summary Report Listing

The Summary Report Listing contains all SYSOUT reports produced by any job within a production system (series of jobs analyzed using a single run of CA JCLCheck). It is produced only if the SXREF and RPTRPT options are both selected. A SYSOUT report is assumed to result from any DD statement specifying the SYSOUT parameter, including those requesting an internal reader. Information for this report is obtained from both the SYSOUT DD statement itself and from any associated JES2 /\*OUTPUT control statement.

To generate this report, specify the RPTRPT and SXREF options.

CA		REPORT 9 - SUMMARY REPORT REPORT						THURSDAY JANUARY 9, yyyy	
CA JCLCHECK		SYS: xxxx							
STMT NUM	JOB NAME	PROCSTEP NAME	STEP NAME	DDNAME	SYSOUT CLASS	SPECIAL FORMS	COPY COUNT	SYSOUT PROGRAM	REPORT DESTINATION(S)
18.	JCHKTEST	STEP12	NSTEP2	SYSPRINT	J	0004	4	-----	TSC32622

This report contains the following fields:

### STMT NUM

Indicates the statement number of the DD statement defining this report.

### JOB NAME

Indicates the name of the job producing this report.

### PROCSTEP NAME

Indicates the name of the most recent EXEC PGM= statement within the current procedure.

If the current step is not from a procedure, or had no name field, this column is blank.

### STEP NAME

Indicates the name of the most recent EXEC PGM= or EXEC PROC= statement.

This field is blank if the statement had no name.

### DDNAME

Indicates the name of the DD statement defining this report.

### SYSOUT CLASS

Indicates the SYSOUT class of this report.

This field is eight bytes long to allow for JES3 SYSOUT classes.

#### **SPECIAL FORMS**

Indicates the special forms number of the form for this report.

This field is blank if you want to print the report on standard forms.

#### **COPY COUNT**

Indicates the number of copies of this report.

This field is blank if only one copy is requested.

#### **SYSOUT PROGRAM**

Indicates the name of the special program (for example, INTRDR) processing this report.

This field is blank if the report is not processed by any special program.

#### **REPORT DESTINATION(S)**

Indicates the destination or destinations specified for this report.

This field is blank if default routing was requested.

## **Report 10 - Flow Diagram**

The Flow Diagram report is a graphic representation of the flow of the JCL processed by CA JCLCheck. The first page lists the name of the JCL member being charted and any JOBLIB, JOBCAT, or PROCLIB data sets associated with the job. STEPLIB and STEPCAT data sets are listed at the beginning of each step. Input and output data sets, respectively, are listed on the left and right side of the process box (procstep/step/program). As much relevant information as is available from the JCL or the system catalog is also listed for each symbol:

- Data set name
- Data set disposition
- Unit and volume information (all volumes)
- Data set attributes: LRECL, BLKSIZE, DSORG, RECFM

In addition, the following information is listed for tape data sets:

- LABEL type
- File number

In addition, the following information is listed for SYSOUT data sets:

- SYSOUT CLASS
- FCB/UCS character-set identifier
- Number of copies
- SYSOUT program name

The process symbol shows the procedure and procedure step name, or instream step name, along with the program name.

Standard flowchart graphic symbols are used to represent disk, tape, card and printed files, communications lines, and online terminals. For impact printers, the symbols themselves are labeled. Non-impact printers do not have labeled symbols, as they can achieve higher graphic resolution. Each symbol includes the ddname and the corresponding statement number from the JCL.

When a chart for one step extends beyond one page, the process symbol is repeated, with page connectors on each page.

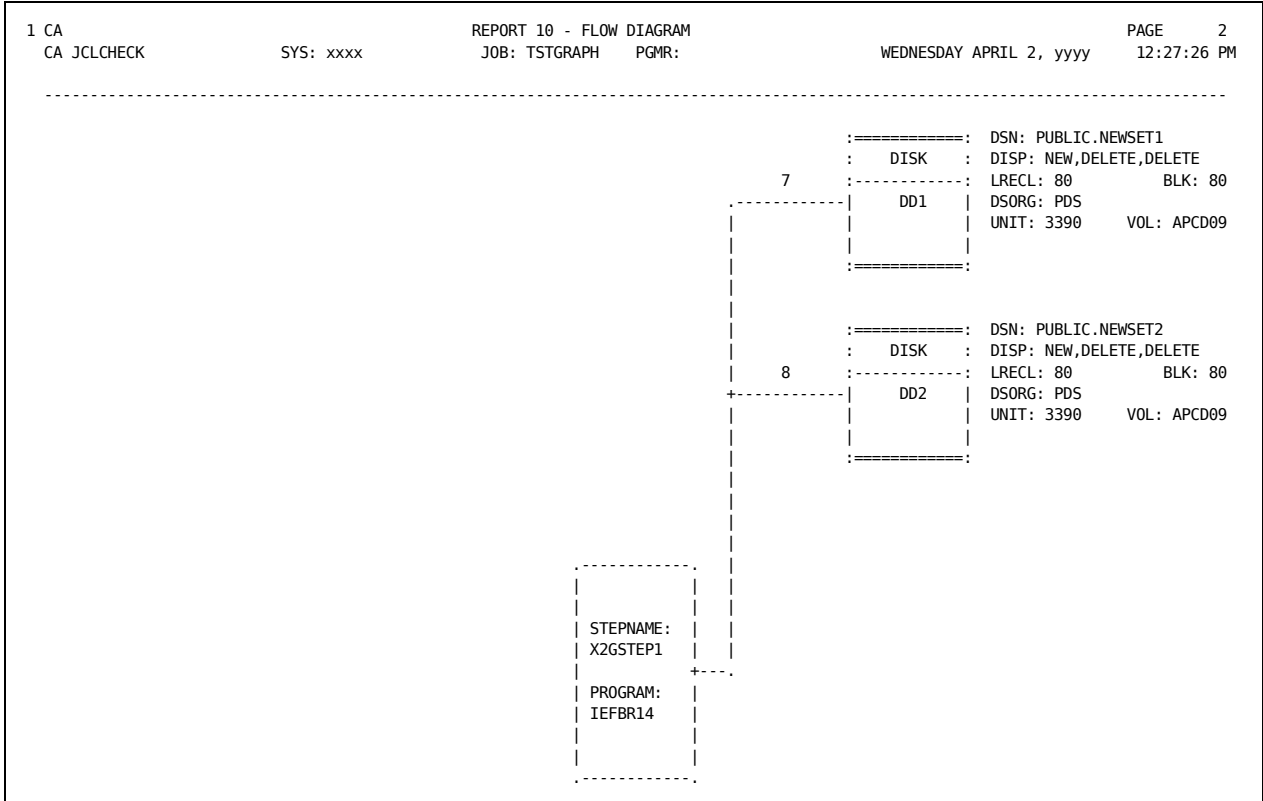
Concatenated DD statements are represented by having the file symbol duplicated. All concatenated files are listed in order of concatenation. Concatenated files that extend on to a second page do not generate repeated symbols.

To generate this report, specify the GRAPH runtime option.

## Operational notes

- If there are errors or severe error messages during JCL checking, CA JCLCheck does not produce a graph. This is a safety measure that safeguards against the documentation of errors and other significant problems in the graph.
- When simulating condition code checking or restarting, CA JCLCheck only graphs those steps simulated to execute. It does not graph the skipped steps and issues a warning message.
- You can allocate a SYSGRAPH DD statement that allows use of the GRAPH option with a TSO CLIST or the CA JCLCheck ISPF Interface.







This report contains the following fields:

**PROCEDURE NAME**

Indicates the name of the cataloged procedure followed by a list of symbolics.

**LIBRARY CONTAINING PROCEDURE VOLUME**

Indicates the volume on which the PROCLIB exists.

**LIBRARY CONTAINING PROCEDURE DSNAME**

Indicates the PROCLIB data set name.

**REFERENCED IN STATEMENT**

Indicates the statement number on which CA JCLCheck executes the PROC or uses the symbolic.

**Note:** An asterisk (\*) in column 1 signifies the default value.

## Audit Report

The JCL Processing Audit Report shows you a statistical summary, input records, and averages of jobs, steps, and DD statements.

To generate this report, specify the AUDIT option.

CA		JCL PROCESSING AUDIT REPORT										TUESDAY OCTOBER 24, yyyy	
CA JCLCHECK		SYS: xxxx											
-- (RECORDS) --				-- (TOTALS) --				-- (AVERAGES) --					
GRAND	SYSIN	PROCLIB	CTL-CARD	JOB	PROCS	STEPS	DDS	PROCS	STEPS	STEPS	DDS		
TOTAL	TOTAL	TOTAL	TOTAL	JOB	PROCS	STEPS	DDS	PER JOB	PER PROC	PER JOB	PER STE		
29	6	23	0	1	3	8	7	3.0	2.7	8.0			
##### END OF AUDIT REPORT FOR RUN #####													

This report contains the following fields:

**GRAND TOTAL**

Indicates the total of all input records.

**SYSIN TOTAL**

Indicates the total records read from the SYSIN DD statement.

**PROCLIB TOTAL**

Indicates the total procedure records read from any procedure libraries.

**CTL-CARD TOTAL**

Indicates the total control statements read from all control statement libraries.

**JOBS**

Indicates the total number of jobs processed.

**PROCS**

Indicates the total number of procedures processed.

**STEPS**

Indicates the total number of steps processed.

**DDS**

Indicates the total number of DD statements processed.

**PROCS PER JOB**

Indicates the average number of procedures per job.

**STEPS PER PROC**

Indicates the average number of steps per procedure.

**STEPS PER JOB**

Indicates the average number of steps per job.

**DDS PER STE**

Indicates the average number of DD statements per step

# Chapter 5: Using JCLNeat

---

When you code JCL, you can supply keywords in any order and in many ranges of columns. You can also comment this language in various ways. Many installations are not able to implement a structured format for JCL successfully; therefore, reading the JCL on listings is a tedious task for the original coder and a nightmare for support personnel.

JCLNeat standardizes the format of JCL according to the specifications of your installation.

The following terms apply to the JCLNeat component:

## **JCL statement**

A JCL statement is a logical entity consisting of the JCL operator and its positional/keyword parameters and associated data.

A JCL statement can be (and usually is) a composite of a variable number of JCL parameters.

## **logical break**

A logical break is a condition in the original JCL input stream that signals JCLNeat to process a new entity.

Logical break is typically signaled by an input member change or a new JOB statement, but it can also be signaled by an isolated PROC.

## **operator**

The operator is a type of statement in process (that is, JOB, EXEC, PROC, DD, PEND, PRINTDEV, CNTL, ENDCNTL, OUTPUT, or comments).

## **original JCL**

The original JCL is the statement lines read from the JCLNeat input stream.

## **reformatted JCL**

The reformatted JCL is the statement lines written to the JCLNeat output stream.

JCLNeat can process original JCL from any one of the following sources:

- Sequential input from statements, magnetic tape, disk, or a single partitioned data set (PDS) member.
- A PDS, which allows CA JCLCheck to process the input completely or partially, with individual members specifically included or excluded.

- Directly from CA Panvalet using the CA Panvalet Access Method (PAM), input is processed either completely or partially, with individual members specifically included or excluded.
- Directly from CA Librarian using the FAIRVS access method, input is processed either completely or partially, with individual members specifically included or excluded.

**Note:** Make sure that JCL is error-free before JCLNeat reformats it.

**Important!** The operation of the JCLNeat component and how it performs the "neating" process is controlled by various parameters, options, and keywords. You can manually supply this information in batch, or you can use ISPF panels to input this information. This chapter covers both methods. You can use either method, but you cannot use a combination of the two.

Manual control of JCLNeat is covered in the next topic.

For information on how ISPF panels control JCLNeat, see [ISPF Panels Control of JCLNeat](#) (see page 264).

## Manual Control of JCLNeat

This section covers the various ways of supplying all of the parameters, keywords, and options that the JCLNeat component needs to perform the "neating" process.

### Including/Excluding Members

JCLNeat can selectively process the members of a partitioned CA Librarian or CA Panvalet data set. To invoke selective processing, you must:

1. Specify MEMBERS=INCLUDE or MEMBERS=EXCLUDE as part of the PARM information that is supplied to JCLNeat.
2. Specify the members to INCLUDE or EXCLUDE in the MEMBERS DD data set in the following format:

```
CoLumn Number 1  
                Membername1  
                Membername2  
                Membernamen
```

For example, to specify a member named BASICPGM, enter:

```
CoLumn Number 1  
                BASICPGM
```

## Required Execution JCL for JCLNeat

The following JCL is required to invoke JCLNeat. For more information, see [JCLNeat Options](#) (see page 285).

```
//STEPNAME EXEC CAZ1NEAT,
//          OPT='option1=data1,option2=data2,optionn=datan'
//inlib   DD   DSN=old.dataset.name,DISP=SHR
//outlib  DD   DSN=new.dataset.name,DISP=OLD
//ADDCARDS DD  *
./ ADD NAME=
./ INCR SEQ1=100,INCR=100
/*
//CHANGES DD  *
(from-data1-to-data1)
(from-data2-to-data2)
(from-datan-to-datan)
/*
//KEYWORDS DD  *
EXEC keyword1
EXEC keyword2
EXEC keywordn
DD keyword1
DD keyword2
DD keywordn
DCB keyword1
DCB keyword2
DCB keywordN
PRINTDEV keyword1
PRINTDEV keyword2
PRINTDEV keywordn
OUTPUT keyword1
OUTPUT keyword2
OUTPUT keywordn
PROC symbolic1
PROC symbolic2
PROC symbolicn
/*
//MEMBERS DD  *
Membername1
Membername2
Membernamen
//GSRCTL DD  *
|verb|string|
/*
```

Use the following statement to invoke JCLNeat procedure:

**EXEC**

Required execution statement.

For a complete description of all the options available, see [JCLNeat Options](#) (see page 285).

The following are JCLNeat DD statements:

**SYSPRINT**

Receives the reports generated by JCLNeat based on your `REPORT=option` request.

This statement has a logical record length of 133 and is blocked 10 (1330 characters) by JCLNeat.

SYSPRINT is a required statement.

**SYSUT1**

Work file containing this report until a logical break is encountered in the original JCL.

The file allocation must be large enough to contain the reformatted JCL for the largest job or member in the input stream.

SYSUT1 is optional, unless the Reformatted JCL Report is requested in the `REPORT=option`.

**SYSUT2**

Work file containing this report until a logical break is encountered in the original JCL.

The file allocation must be large enough to contain the maximum number of from/to statements changed in a single job or member in the input stream.

SYSUT2 is optional, unless the Changes Report is requested in the `REPORT=option`.

**INLIB**

Points to the JCL stream JCLNeat is to reformat.

Depending on the specified `INLIB=` option, you must name this DD statement LIBIN, PANIN, PDSIN, or SEQIN.

INLIB is required.

**OUTLIB**

Specifies where JCLNeat should output its reformatted JCL.

Depending on the specified `OUTLIB=` option, you must name this DD statement PDSOUT or SEQOUT.

**Note:** When using `INLIB=PDS` and `OUTLIB=PDS`, the PDSIN and PDSOUT DD statements must not reference the same PDS or JCLNeat discontinues processing with a CAY6504S message.

**SYSUDUMP**

Use only if JCLNeat abnormally terminates with a system error message.

SYSUDUMP is optional.

**ADDCARDS**

Contains from one to five statements that CA JCLCheck places in front of each job or other logical break encountered in the input stream.

Use only if the OUTLIB=SEQ option has been specified and you want to include library control statements in the output stream.

**CHANGES**

Specifies the from-character-string and to-character-string JCLNeat uses in editing the data portion of keyword parameters.

CHANGES is optional, unless you desire global editing.

**KEYWORDS**

Contains entries for:

- Your desired keyword order for the EXEC and DD parameters and for the DCB subparameters.
- Your list of PROC symbols *not* to replace if you use the RSYM=Y option.

KEYWORDS is optional.

**MEMBERS**

Specifies the member names JCLNEAT includes or excludes when processing PDS input.

Members is optional, unless you specify the MEMBERS= option.

**GSRCNTL**

Specifies the action verbs and literals to use for manipulation of the JCL.

GSRCNTL is optional.

## Specifying Options

The JCLNeat component has a set of options that can be changed to influence how the component operates and reacts. These options are different from the CA JCLCheck runtime options and pertain only to the JCLNeat component. For a detailed description of the JCLNeat options, see [JCLNeat Options](#) (see page 285).

To specify option requests to JCLNeat, use the PARM field and the format OPTION=value. You can enter the options in any sequence. Since several of the options may contain special characters, input the entire PARM field within single quotation marks. If the number of options you are specifying makes it necessary to continue the PARM field, follow standard z/OS conventions as defined in the JCL Reference and User guides.

Alternatively, you can use the JCLNOPTS DD statement to specify the options for JCLNEAT. If the JCLNOPTS DD is used, then the EXEC PARM is ignored. This method bypasses the restriction to the length of the EXEC PARM value of 100 characters.

```
//JCLNOPTS DD *  
* Comment Card  
option1,option2  Comment  
option3          Comment  
option4          Comment
```

**Usage notes:**

- Options are specified starting in column 1 through 72 until a space is found; after a space the rest of the card is considered to be comments.
- If column 1 of the card is an asterisk (\*), then the entire card is considered a comment card.
- One or more options can be coded on a single card.
- The file must be sequential with an LRECL of 80.

## Keyword Reordering

Keyword parameters and their associated data are reordered in the reformatted JCL based on a user-specified sequence, with these exceptions:

- The setting of the JOBF option determines whether the JOB statement has its keywords reordered. Regardless of keyword order, fields, such as the OPERAND field, are still column aligned. Note that the CSECT CAZ1NCJB determines the order of the parameters within the JOB statement.
- Qualified keywords are placed on the reformatted EXEC statement in the same order as they were encountered in the original EXEC and are placed *after* the ordered keywords.

JCLNeat is distributed with tables containing a default order (alphabetical) for the EXEC and DD keyword parameters and for the DCB keyword subparameters. You can change these default tables after JCLNeat is installed, or at execution time by specifying a new table sequence in the KEYWORDS DD data set.

## KEYWORDS DD Data Set

New table sequences are provided at execution time using the KEYWORDS DD data set. The order is determined by specifying an operator of EXEC, DD or DCB in Column 1 followed by at least one space and then the keyword starting in Column 6. The keywords are sequenced based on their order in this input stream. You can intersperse operators in the KEYWORDS DD data set. However, it is difficult to easily verify the resulting table order, since the JCLNeat reporting of KEYWORDS INPUT BY USER: is printed in input sequence. Therefore, for clarity and ease of use, it is suggested that all keywords for one JCL operator be input first, followed by all the keywords for the next JCL operator, followed by all the keywords for the last JCL operator.

Keyword tables are not merged. This means that each table specification (the installation default or the execution-time KEYWORDS DD data set table) must be self-contained. You *must* specify all parameters whose sequence matters to you. The default table is *not* collated with the execution-time user table.

## EXEC Keyword Order

The JCLNeat default sequence of keyword parameters on the EXEC statement is as follows:

```
ACCT
ADDRSPC
COND
DPRTY
DYNAMNBR
PARM
PERFORM
RD
REGION
TIME
```

To override the existing order for EXEC keyword parameters, specify the desired sequence in the KEYWORDS DD data set as follows:

```
CoLumn Number 1    6
                EXEC Keyword1
                EXEC Keyword2
                EXEC Keywordn
```

**Note:** Since PGM or PROC is a positional rather than a keyword parameter, it is ignored if you specify it.

The keyword sequence specified here completely overrides the default sequence provided by JCLNeat.

Keyword parameters encountered in the original JCL and not specified in this sequence are:

- Alphabetized and placed in the reformatted JCL *before* the last keyword in the table, if the last keyword is coded in the original JCL.
- Alphabetized and placed *last* in the reformatted JCL if the last keyword in the table is not coded in the original JCL.

**Examples:**

Sample Keywords DD data set as input by user:

```
DD DSNAME
DD DISP
DD UNIT
DD VOLUME
DD SPACE
DD DCB
```

Sample DD statement prior to reformatting:

```
IN //DD1 DD DSN=SAMPLE.DSN,DISP=(NEW,KEEP),
IN // VOL=SER=DISK01,UNIT=SYSDA,
IN // DCB=BLKSIZE=8800,
IN // SPACE=(CYL,(5,1))
```

Sample DD statement after reformatting using the JCLNeat default keyword sequence:

```
OUT //DD1 DD DCB=BLKSIZE=8800,
OUT // DISP=(NEW,KEEP),
OUT // DSN=SAMPLE.DSN,
OUT // SPACE=(CYL,(5,1)),
OUT // UNIT=SYSDA,
OUT // VOL=SER=DISK01
```

Sample DD statement after reformatting using the overriding DD keyword order as specified previously:

```
OUT //DD1 DD DSN=SAMPLE.DSN,
OUT // DISP=(NEW,KEEP),
OUT // UNIT=SYSDA,
OUT // VOL=SER=DISK01,
OUT // SPACE=(CYL,(5,1)),
OUT // DCB=BLKSIZE=8800
```

## DD Keyword Order

The JCLNeat default sequence of keyword parameters on the DD statement is as follows:

AMP  
BURST  
CHARS  
CHKPT  
CNTL  
COPIES  
DCB  
DDNAME  
DEST  
DISP  
DLM  
DSID  
DSNAME  
DUMMY  
DYNAM  
FCB  
FLASH  
FREE  
HOLD  
LABEL  
MODIFY  
MVSGRP  
OUTLIM  
PROTECT  
QNAME  
SPACE  
SUBSYS  
SYSOUT  
TERM  
UCSM  
UNIT  
VOLUME

To override the existing order for EXEC keyword parameters, specify the desired sequence in the KEYWORDS DD data set as follows:

```
CoLumn Number 1 4  
                DD Keyword1  
                DD Keyword2  
                DD Keywordn
```

**Note:** You must specify the DSNAME and VOLUME keywords in their long form for JCLNeat to correctly process both their long and abbreviated forms, DSN and VOL.

The keyword sequence specified here completely overrides the default sequence defined to JCLNeat or user modified defaults.

Keyword parameters encountered in the original JCL and not specified in this sequence are:

- Alphabetized and placed in the reformatted JCL *before* the last keyword in the table, if the last keyword is coded in the original JCL.
- Alphabetized and placed *last* in the reformatted JCL if the last keyword in the table is not coded in the original JCL.

## DCB Keyword Order

The JCLNeat default sequence of keyword subparameters within the DCB parameter is as follows:

BFALN  
BFTEK  
BLKSIZE  
BUFIN  
BUFL  
BUFMAX  
BUFNO  
BUFOFF  
BUFOUT  
BUFSIZE  
CODE  
CPRI  
CYLOFL  
DEN  
DIANGS  
DSORG  
EROPT  
FRID  
FUNC  
GNCP  
INTVL  
KEYLEN  
LIMCT  
LRECL  
MODE  
NCP  
NTM  
OPTCD  
PCIF  
PRTSP  
RECFM  
RESERVE

RKP  
 STACK  
 THRESH  
 TRTCH

To provide an overriding order for the DCB subparameters at execution time, specify the desired sequence in the KEYWORDS DD data set in the following format:

```
Column Number 1 5
                DCB Keyword1
                DCB Keyword2
                DCB Keywordn
```

**Note:** The keyword sequence specified here completely overrides the default sequence provided by JCLNeat.

Keyword subparameters encountered in the original JCL and not specified in this sequence are:

- Alphabetized and placed in the reformatted JCL *before* the last keyword in the table if it is coded in the original JCL.
- Alphabetized and placed *last* in the reformatted JCL if the last keyword in the table is not coded in the original JCL.

## PRINTDEV Keyword Order

The JCLNeat default sequence of keyword parameters on the PRINTDEV statement is as follows:

BUFNO  
 CHARS  
 DATAK  
 DSHDR  
 DUMP  
 FDEFDD  
 FONTDD  
 FORMDEF  
 JOBHDR  
 JOBTRLR  
 MESSAGE  
 NPRO  
 OVLYDD  
 PAGEDEF  
 PDEFDD  
 PIMSG  
 PSEGDD  
 TRACE

To provide an overriding order for PRINTDEV keyword parameters at execution time, specify the desired sequence in the KEYWORDS DD data set in the following format:

```
Column Number 1      10
                PRINTDEV Keyword1
                PRINTDEV Keyword2
                PRINTDEV Keywordn
```

**Note:** The keyword sequence specified here completely overrides the default sequence defined to JCLNeat. Keyword parameters encountered in the original JCL and not specified in this sequence are:

- Alphabetized and placed in the reformatted JCL *before* the last keyword in the table, if the last keyword in the table is coded in the original JCL.
- Alphabetized and placed *last* in the reformatted JCL if the last keyword in the table is not coded in the original JCL.

## OUTPUT Keyword Order

The JCLNeat default sequence of keyword parameters on the OUTPUT statement is as follows:

```
BURST
CHARS
CKPTLINE
CKPTPAGE
CKPTSEC
CLASS
COMPACT
CONTROL
COPIES
DEFAULT
DEST
FLASH
FORMDEF
FORMS
GROUPID
INDEX
JESDS
LINDEX
LINECT
MODIFY
PAGEDEF
PIMSG
PRMODE
PRTY
THRESHLD
TRC
```

UCS  
WRITER

To provide an overriding order for OUTPUT keyword parameters at execution time, specify the desired sequence in the KEYWORDS DD data set in the following format:

```
Column Number 1      8
                OUTPUT Keyword1
                OUTPUT Keyword2
                OUTPUT Keywordn
```

**Note:** The keyword sequence specified here completely overrides the default sequence defined to JCLNeat or user modified defaults. Keyword parameters encountered in the original JCL and not specified in this sequence are:

- Alphabetized and placed in the reformatted JCL *before* the last keyword in the table, if the last keyword in the table is coded in the original JCL.
- Alphabetized and placed *last* in the reformatted JCL if the last keyword in the table is not coded in the original JCL.

## Global String Replacement

JCLNeat provides a global string replacement feature for any character string in the JCL.

Specify this type of string replacement in the GSRCNTL DD data set in the following format:

```
Column Number 1
                |from-string|verb|{JCL-type}
                |to-string|
```

In the preceding example, the vertical bar (|) is the delimiter for the GSRCNTL parameters. The vertical bar allows you to change any text of a JCL statement, including a slash (/).

You can use the following *verbs*:

### REPL

Replaces *from-data* with *to-data*

### DELT

Deletes the JCL STATEMENT when JCLNeat finds *from-data*

**REMV**

Deletes the line where JCLNeat finds *from-data*

**COMM**

Comments the JCL STATEMENT when JCLNeat finds *from-data*

**Important!** When the string is found in the FIRST DD of a concatenated list, the entire concatenation list is commented or deleted.

When the string is found in the SECOND or subsequent DD of a concatenated list, only the current DD statement is commented or deleted.

**INSA**

Inserts *to-data* after the *from-data* line

**INSB**

Inserts *to-data* before the *from-data* line

JCLNeat compares the *from-data* of each record and if it finds a match, executes the *verb* using *to-data* (if applicable).

**Note:** The *from-data* can be up to 63 characters in length. The *to-data* can be up to 71 characters in length.

You can use the following JCL statement identifier:

**JCL-type**

*JCL-type* is an optional JCL statement identifier that limits the search for the *from-string* to the type of JCL specified. You can use the following *JCL-types*: CNTL, COMMAND, DD, DPROC, ELSE, ENDCNTL, ENDIF, EXEC, IF, INCLUDE, JCLLIB, JOB, NJB, OUTPUT, PEND, PRINTDEV, PROC, and SET.

**Important!** JCLNeat does not ensure the accuracy of your global string changes. We recommend that you also use CA JCLCheck to verify the accuracy of the changes. Misplaced or erroneous control cards can produce unexpected results.

**Example: Change All the JCL in a PDS**

This example shows how to change all the JCL in a PDS from MSGCLASS=A to MSGCLASS=X, and how to comment out old ddnames:

```
//GSRCNTL DD *  
|MSGCLASS=A|REPL|  
|MSGCLASS=X|  
|DSN=SOME.BOGUS.DATASET|COMMENT|  
|DSN=OTHER.BOGUS.DATASET|DELT|
```

The result is as follows:

INPUT JCL:

```
//JOBNAME JOB (ACCT),programmer,CLASS=X,
//  MSGCLASS=A
//STEP01 EXEC PGM=IEFBR14
//DD1 DD DSN=SOME.BOGUS.DATASET,DISP=SHR
//DD2 DD DSN=OTHER.BOGUS.DATASET,DISP=SHR
```

REFORMATTED JCL: (JOBF=Y,FORMAT=Y)

```
//JOBNAME JOB (ACCT),
//      programmer,
//      CLASS=X
//      MSGCLASS=X
//STEP01 EXEC PGM=IEFBR14
//*DD1 DD DSN=SOME.BOGUS.DATASET,DISP=SHR
```

OR (Optionally)

NON-REFORMATTED JCL: (JOBF=N,FORMAT=N specified)

```
//JOBNAME JOB (ACCT),programmer,CLASS=X,
//  MSGCLASS=X
//STEP01 EXEC PGM=IEFBR14
//*DD1 DD DSN=SOME.BOGUS.DATASET,DISP=SHR
```

### Example: Change All OUTPUT Statements

This example shows how to change CLASS=A to CLASS=B for OUTPUT statements only:

```
//GSRCTL DD *
|CLASS=A|REPL|OUTPUT
|CLASS=B|
```

The result is as follows:

```
INPUT JCL:
//JOBNAME JOB (ACCT),programmer,CLASS=A,MSGCLASS=A
//STEP01 EXEC PGM=IEFBR14
//OUT001  OUTPUT CLASS=A
```

REFORMATTED JCL: (JOB=Y,FORMAT=Y)

```
//JOBNAME JOB (ACCT),  
//          programmer,  
//          CLASS=A,  
//          MSGCLASS=A  
//STEP01 EXEC PGM=IEFBR14  
//OUT001 OUTPUT CLASS=B
```

OR (Optionally)

NON-REFORMATTED JCL: (JOB=N,FORMAT=N specified)

```
//JOBNAME JOB (ACCT),programmer,CLASS=A,MSGCLASS=A  
//STEP01 EXEC PGM=IEFBR14  
//OUT001 OUTPUT CLASS=B
```

## Global Editing

JCLNeat provides a global editing feature for the data portion of keyword parameters and subparameters. The data portion of a keyword is the character-string *after* the = sign. For example, with a keyword parameter of UNIT=3380, *only* the data portion (that is, 3380) could be changed using the JCLNeat global editing feature.

Specify changes in the CHANGES DD data set in the following format:

```
Column Number 1  
    /from-data1/to-data1/  
    /from-data2/to-data2/  
    /from-datan/to-datan/
```

JCLNeat compares the data portion, (/from-data/) of each keyword and, if a match is found, replaces the data portion, (/to-data/). If you specify it, multiple changes to the same data can occur. For example, with:

```
//CHANGES DD *  
/3380/SYSDA/  
/SYSDA/DISK/
```

If the original JCL looks like this:

```
//DD1 DD DSN=A,UNIT=3380,DISP=SHR
```

The reformatted JCL looks like this:

```
//DD1 DD DISP=SHR,  
//      DSN=A,  
//      UNIT=DISK
```

**Note:** The *from-data* and *to-data* can be of unequal lengths and less than or equal to 34 characters each.

## Symbolic Replacement

When processing a PROCLIB member or an inline procedure (PROC/PEND) at your request (RSYM=Y), JCLNeat replaces the symbolics within the procedure with the values specified on the PROC statement unless:

- The symbolic is specifically excluded by an entry in the KEYWORDS DD data set in the following format:

```
Column Number 1    6  
                PROC Symbolic1  
                PROC Symbolic2  
                PROC Symbolicn
```

- The symbolic has a null value (SYMBOL=,) on the PROC statement.

Once replacement has been effected, all symbolics except those specifically excluded or those with a null value are then removed from the PROC statement in the reformatted JCL. You can use this feature to restructure procedure libraries by eliminating unused symbolics. Symbolics still required by the installation would be excluded in a PROC entry in the KEYWORDS DD data set.

**Note:** Global editing is performed before symbolic replacement is affected. Therefore, if you specify a CHANGES DD data set entry that modifies the data portion of a symbolic parameter, it is changed prior to being replaced in the procedure.

For example, a CHANGES DD data set entry of /3380/SYSDA/ and a PROC of:

```
//NAME1 PROC U=3380
//STEP1 EXEC PGM=A
//DD1 DD DSN=CDE,DISP=OLD,UNIT=&U,VOL=SER=XYZ
```

results in the following reformatted JCL:

```
//NAME1 PROC
//STEP1 EXEC PGM=A
//DD1 DD DISP=OLD,
//      DSN=CDE,
//      UNIT=SYSDA,
//      VOL=SER=XYZ
```

If a procedure that has had its symbolics replaced is subsequently executed and still contains qualified symbolic overrides, a JCL error results.

## ISPF Panels Control of JCLNeat

All of the methods explained previously in this chapter for batch submission of JCLNeat controls can be accomplished by using ISPF panels. The following subsections explain each of the panels that can be displayed.

Field level HELP is available on each of the panels described in this section so the fields will not be described here. This help is incorporated into the HELP/Tutorial function so that help exists at three levels:

### System level

The existing Tutorial/HELP, activated from within the other levels.

### Panel level

A pop-up box explaining the panel involved, activated by any field which does not have field level HELP support.

### Field level

A pop-up box containing an explanation of the field selected by the cursor. All of the panels described in this section have field level help available.

When activated, the pop-up features fades the current panel to a dull blue but still remains readable, and the new information overlays the current panel.

There are standard format conventions that are consistent throughout all the panels:

**Commands:**

**CANcel**

End current panel session, with no action taken

**NEXT**

Process current panel, go to next panel

**PREVious**

Process current panel, go to previous panel

**PF keys:**

**Enter**

Process current panel, stay on current panel

**End**

Process current panel, go to next panel

All key functions are set by the individual user in their session profile. All panels have the current date and time at the upper right, followed by the current session's user ID.

## Batch Submission Panel Sequence

The sequence in which panels are displayed is shown in the following table. Two modes of processing present a different sequence of display: Defaults and Submit. Use Defaults when settings are to be in effect during subsequent processing, and use Submit to release JOB to perform the JCLNeat process.

For Defaults processing, JCK0210 is followed by JCK0205, followed by JCK0205A, followed by JCK0215, and so on.

For Submit processing, JCK0206 Optional Data Sets, determines which of the following panels are displayed. If you have no options specified during this panel, you skip to 10: JOB statement for submission. Other panels are displayed based on optional data sets selected. For example, you would go from sequence 4, 5, 7, 8, 9 if you did not specify Keyword Ordering on the Optional Data Sets panel.

JCK0215 has six panels that are associated with it. They are based on the six types of keyword ordering, which you select.

Defaults	Submit	Panel ID	Description
1	1	JCK0200	CA JCLCheck/SPF menu
2		JCK0210	Control options menu
3	2	JCK0205	Optional parameters
4	3	JCK0205A	Statement level options
	4	JCK0206	Optional data sets
	5	JCK0216	Member selection list for INCLUDE/EXCLUDE option
5	6	JCK0215	Keyword ordering on DD statement EXEC DD DCB OUTPUT PRINTDEV PROC
6	7	JCK0211	Add card control statements for library management
	8	JCK0212	String changes of data portion of keywords in JCL
	9	JCK0213	Global string replacement of any data in JCL
7	10	JCK0214	JOB statement for submission of JCLNeat job
	11	ISREDDE2	ISPF edit of job submission JCL
	12	JCK0214	JCK0POPR pop-up panel for verification response

## CA JCLCheck/SPF Menu (JCK0200)

The CA JCLCheck/SPF Menu panel is the starting panel for both the Defaults and Submit mode of processing.

Select Option 0 to create and maintain JCLNeat defaults. These defaults are in effect during the Submit process unless changed in either the maintenance of the defaults or during the Submit process.

Select Option 6 to format and submit a job to perform the "neating" process.

```

JCK0200 ----- CA JCLCheck/SPF MENU -----
OPTION ==>

      Date: yyyy/mm/dd      User: USERSID      Genlevel: xxxx

          0 - Option Specifications
          1 - CA JCLCheck - Foreground Invocation
          2 - CA JCLCheck - Foreground Invocation / Submit
          3 - CA JCLCheck - Batch Submit
          4 - CA 7 Interface Invocation
          5 - CA SCHEDULER Interface Invocation
          6 - JCLNeat - Batch Submit
          7 - RESOLVE - External Variable Resolution
          X - Exit

```

If you have selected Option 0, continue to the next section on the following page, which describes all of the panels associated with the Defaults mode in the sequence that they are displayed.

If you have selected Option 6, see [Submit Mode Panel Descriptions](#) (see page 273). This topic describes all of the panels associated with the Submit mode in the sequence that they are displayed.

## Defaults Mode Panel Descriptions

This section describes each of the panels used in the Defaults mode in the sequence that they are displayed.

## JCLCheck Control Options Menu (JCK0210)

When Option 0 - Option Specifications is selected on the preceding menu, a selection list of options is presented. Option 4 - CA JCLNeat Options supports the creation and maintenance of the defaults that are in effect during the Submit mode process, unless temporarily overridden while in the Submit process.

The defaults that can be set fall into the following categories:

### OPTIONAL PARAMETERS

Parameters controlling options and formats

### KEYWORD

Sort sequence of JCL command keywords

### ADDCARDS

Additional control statements to insert before each member

### JOB CARD

The JOB statement of the submitted JCL stream

These categories of settings are the ones that lend themselves to stored settings. The other categories of settings need to be set during each Submit mode process and are described in that section.

```
JCK0210 ----- CA JCLCheck/CONTROL OPTIONS MENU -----
OPTION ==>

      Date: yyyy/mm/dd      User: USERSID      Time: 12:25

      1 - Process Control Options
      2 - Output Control Options
      3 - Security Options
      4 - CA JCLNeat Options
      5 - Job Card Specification
      X - EXIT
```

## Optional Parameters for Default Mode (JCK0205)

You can maintain parameters with this panel. The parameters get formatted on the PARM command of the EXEC statement in the submitted job stream. These parameters control optional features affecting the input, format, processing, and output of the JCLNeat process. The settings that are made here are in effect on all subsequent job submits unless permanently changed here, or temporarily overridden during the Submit process.

For a full explanation of each parameter that is shown on this panel, see [JCLNeat Options](#) (see page 285). Alternatively, you can use the Field Level HELP feature available on each of the JCLNeat panels.

```
JCK0205-----CA JCLNeat xx.x - OPTIONAL Parameters (DEFAULTS)-----
COMMAND ==>
                                     yyyy/mm/dd
COMMANDs: CANCEL; NEXT or PREVIOUS panel after edit.             hh:mm
KEYs:    Enter to edit.  End to edit, go to next panel.         USER002

BCI      Block Comment Insertion           ==> Y
BCS      Blank Comment Separator           ==> N
CDEL     Comment Delete Request            ==> 0
DCBS     Split DCB Subparameters           ==> N
FCC      Fill Comment Character            ==> -
FORMAT   Format JCL? Y-Yes, N-No, B-Basic  ==> Y
JOBF     Format Job Statement               ==> Y
MKW/MQKW Maximum Keywords/Qualified        ==> 60 / 10
REXXMEM  REXX Member Name                 ==>
RSYM     Replace Symbolics                 ==> N
SECC     Start/End Comment Character       ==> *
SEQ1/INCR Sequencing Start/Increment Number ==> 10 / 10
SIM/REPORT Simulate Processing/Report Format ==> N / 7
```

## Statement Level Options for Default Mode (JCK0205A)

You can use the Statement Level Options panel to maintain additional optional parameters. These parameters are formatted onto the PARM command of the EXEC statement in the submitted job stream. These parameters control optional features affecting the formatting and output of the "neating process." In addition to the global options CEC, CSC, ICSC, OPSC, and PACK, there are also statement level options for each specific JCL statement. If a statement level option is not specified, JCLNeat uses the associated global option value. If a statement level option is specified, this value overrides the associated global option value. The settings that are made here are in effect on all subsequent job submits unless temporarily overridden during the submit process.

For a full explanation of each parameter that is shown on this panel, see [JCLNeat Options](#) (see page 285). Alternatively, you can use the Field Level HELP feature available on each of the JCLNeat panels.

```

----- CA JCLNeat xx.x - Statement Level Options (DEFAULTS) -----
COMMAND ==>
                                                    yyyy/mm/dd
COMMANDs: CANCEL; NEXT or PREVIOUS panel after edit.           hh:mm
KEYs:     ENTER to edit. END to edit, go to next panel.       USERID1

CEC      - Continuation Ending Column (4 - 71)..... => 35
JOB ==>   EXEC ==>   DD ==>   JCLLIB ==>   INCLUDE ==>
SET ==>   PROC ==>   IF ==>   OUTPUT ==>   CNTL ==>
CSC      - Continuation Start Column (4 - 16)..... => 16
JOB ==>   EXEC ==>   DD ==>   JCLLIB ==>   INCLUDE ==>
SET ==>   PROC ==>   IF ==>   OUTPUT ==>   CNTL ==>
ICSC     - Inline Comment Start Column (4 - 71)..... => 37
JOB ==>   EXEC ==>   DD ==>   JCLLIB ==>   INCLUDE ==>
SET ==>   PROC ==>   IF ==>   OUTPUT ==>   CNTL ==>
OPSC     - Operator Start Column (4 - 16)..... => 12
JOB ==>   EXEC ==>   DD ==>   JCLLIB ==>   INCLUDE ==>
SET ==>   PROC ==>   IF ==>   OUTPUT ==>   CNTL ==>
PACK     - Max Number of Keyword Parm's on Each Card (1 - 80)..... => 1
JOB ==>   EXEC ==>   DD ==>   JCLLIB ==>   INCLUDE ==>
SET ==>   PROC ==>   IF ==>   OUTPUT ==>   CNTL ==>

```

## Keyword Ordering (JCK0215)

The Keyword Ordering panels set the defaults for ordering the keywords that can be used in JCL for the JCL TYPE commands of EXEC, DD, DCB, OUTPUT, PRINTDEV, and PROC.

Each JCL TYPE has its own panel showing its related keywords. The keywords are ordered by assigning a unique (for that JCL TYPE) number in the range of 01-99. The numbers need not be sequential, but must not be duplicated or an error results in the occurrence of the duplicated number. Keywords which are not assigned an ordering number fall behind those assigned in the order they are encountered during the "neating" process.

In the example following, the keywords encountered for the EXEC statement (the defaulted starting point) are ordered PARM, COND, REGION, and TIME, followed by all the remaining keywords in the order encountered. These orders are used as defaults for subsequent Submit mode processing, unless overridden during the Submit mode process.

The JCL TYPE for selection of keywords displayed is entered on the COMMAND line. It is a number in the range of 1-6. When the display returns, the current JCL TYPE appears in the field labeled THIS SEL and the previously selected JCL TYPE appears in the field labeled LAST SEL. The appropriate keywords for that JCL TYPE appear to the right.

To be consistent, the same DO YOU WISH TO MAKE THESE SELECTIONS PERMANENT? message appears as it would in Submit mode, but here it is defaulted to Y and cannot be modified.

```
JCK0215 ----- CA JCLNeat xx.x - KEYWORD (DEFAULTS) -----
COMMAND ==>

COMMANDS: CANcel; NEXT or PREVIOUS panel after edit, save.          yyyy/mm/dd
KEYS:      Enter to edit, save. End to edit, save, next panel.      17:10
                                                USERSID

Select JCL TYPE on OPTION line, assign sequencing number by selected KEYWORD.
Keywords not selected will follow those sequenced, in alphabetic order.

----- JCL TYPE ----- KEYWORD ORDER SPECIFICATION -----
:      :      :      :      :      :      :      :      :
: THIS SEL ==> 1 :      :  ACCT  ADDRSPC  2_ COND  DPRTY  :
:      :      :  DYNAMNBR  1_ PARM  PERFORM  RD      :
:      :      :  3_ REGION  4_ TIME  :
: EXEC..... 1 :      :      :      :      :
: DD..... 2 :      :      :      :      :
: DCB..... 3 :      :      :      :      :
: OUTPUT.... 4 :      :      :      :      :
: PRINTDEV... 5 :      :      :      :      :
: PROC..... 6 :      :      :      :      :
: LAST SEL ==> :      :      :      :      :
-----

DO YOU WISH TO MAKE THESE SELECTIONS PERMANENT?  N (Y/N)
```

## ADDCARDS (JCK0211)

The ADDCARDS panel contains the defaults for up to five control statements, which are placed in front of every selected JCL statement member being processed. The name of each member is placed after the first blank of the first ADDCARD as it is processed. The subsequent ADDCARD statements are placed following the first one.

The purpose of this feature is to insert control statements for library management products such as CA Librarian, CA Panvalet, IBM IEBUPDTE, or any other product being used which requires control statements.

```
JCK0211 ----- CA JCLNeat xx.x - ADDCARDS (DEFAULTS) -----  
COMMAND ==>  
                                                    yyyy/mm/dd  
COMMANDS: CAnCel; NEXt or PREVious panel after edit, save.      17:32  
KEYS:     Enter to edit, save. End to edit, save, next panel.  USERSID  
  
Specify up to 5 JCL statements below, to be placed in front of each member  
so the OUTLIB, when specified, can be used as input where control statements  
are required (i.e. Librarian, IEBUPDTE, etc.).  
  
- JOB, MEMBER, or PROC Name copied after last character of first ADDCARD.  
  
- All subsequent ADDCARDS copied following first ADDCARD with Name.  
  
==> -ADD  
==> -HST NEW JCLCHECK FEATURE SUPPORT MODULES  
==>  
==>  
==>
```

### JOB CARD for Default Mode (JCK0214)

The JOB CARD panel contains the defaults for up to four JOB control statements. The statements are placed at the front of the job stream that is created during the Submit mode processing. The job stream comes from (and is stored in) the same place as the JCL used for other CA JCLCheck processing. Any changes that are made here are active for those processes and conversely.

For consistency, the same question is shown "Update PROFILE with these options?", although here it is defaulted to Y and cannot be modified.

The following panel also displays the contents of the PARM= parameter string based on input from the previous JCLNeat panels. These parameters are inserted into the JCLNeat job stream before submission.

```

JCK0214 ----- CA JCLNeat xx.x - JOB CARD (DEFAULTS) -----
COMMAND ==>
                                                    yyyy/mm/dd
COMMANDs: CANCEL; NEXT or PREVIOUS panel.          hh:mm
KEYs:     ENTER to save, END to save, exit to MENU.  USERID1

Job Statement Information: (Verify before saving)
=> //USERID1J JOB '113300000',USER1,NOTIFY=USERID1,MSGLEVEL=(1,1),
=> // CLASS=A,MSGCLASS=X,USER=USERID1,REGION=0M
=> /*JOBPARM ROOM=716A
=> /**
      Update PROFILE with these options      ==> Y
      Edit JCL JOB Stream before submitting  ==> Y
PARM=( BCI=Y,BCS=N,CDEL=0,CEC=35,CSC=16,DCBS=N,FCC=-,FORMAT=Y,JOBF=Y,ICSC=37, )
      ( INCR=10,INLIB=PDS,LINES=55,MKW=60,MPKW=10,OPSC=12,OUTLIB=,PACK=1,RECOV )
      ( ER=Y,REPORT=7,RSYM=N,SECC=*,SEQ1=10,SIM=N )
      ( )
      ( )
      ( )
      ( )
      ( )
      ( )
      ( )
      ( )

```

This panel completes the Defaults mode process. Pressing End now returns you to the original menu (JCK0200).

## Submit Mode Panel Descriptions

This section describes each of the panels used in the Submit mode in the sequence that they are displayed.

### Optional Parameters for Submit Mode (JCK0205)

You can review and change parameters with this panel. The parameters get formatted on the PARM command of the EXEC statement in the submitted job stream. Any changes that you make to the parameters shown are only effective for the current submission process. Otherwise, the operation and content of this panel is identical to its Defaults mode processing counterpart.

For a full explanation of each parameter that is shown on this panel, see [JCLNeat Options](#) (see page 285). Alternatively, you can use the Field Level HELP feature available on each of the JCLNeat panels.

```
JCK0205-----CA JCLNeat xx.x - OPTIONAL Parameters (SUBMIT)-----
-
COMMAND ==>
COMMANDs: CANCEL; NEXT or PREVIOUS panel after edit.          yyyy/mm/dd
KEYs:     Enter to edit. End to edit, go to next panel.      hh:mm
                                                    USER002

BCI      Block Comment Insertion           ==> Y
BCS      Blank Comment Separator           ==> N
CDEL     Comment Delete Request            ==> 0
DCBS     Split DCB Subparameters           ==> N
FCC      Fill Comment Character             ==> -
FORMAT   Format JCL? Y-Yes, N-No, B-Basic  ==> Y
JOBFB    Format Job Statement               ==> Y
MKW/MQKW Maximum Keywords/Qualified        ==> 60 / 10
REXXMEM  REXX Member Name                  ==>
RSYM     Replace Symbolics                  ==> N
SECC     Start/End Comment Character        ==> *
SEQ1/INCR Sequencing Start/Increment Number ==> 10 / 10
SIM/REPORT Simulate Processing/Report Format ==> N / 7
```

### Statement Level Options for Submit Mode (JCK0205A)

You can use the Statement Level Options panel to review and change additional optional parameters. These parameters are formatted onto the PARM command of the EXEC statement in the submitted job stream. The operation and content of this panel is identical to its Defaults mode processing counterpart except that any changes made to the parameters shown are only effective for the current submission process.

For a full explanation of each parameter that is shown on this panel, see [JCLNeat Options](#) (see page 285). Alternatively, you can use the Field Level HELP feature available on each of the JCLNeat panels.

```
----- CA JCLNeat xx.x - Statement Level Options (SUBMIT) -----
COMMAND ==>
COMMANDs: CANCEL; NEXT or PREVIOUS panel after edit.          yyyy/mm/dd
KEYs:     ENTER to edit. END to edit, go to next panel.      hh:mm
                                                    USERID1

CEC      - Continuation Ending Column (4 - 71)..... ==> 35
JOB ==>  EXEC ==>  DD ==>  JCLLIB ==>  INCLUDE ==>
SET ==>  PROC ==>  IF ==>  OUTPUT ==>  CNTL ==>
CSC      - Continuation Start Column (4 - 16)..... ==> 16
JOB ==>  EXEC ==>  DD ==>  JCLLIB ==>  INCLUDE ==>
SET ==>  PROC ==>  IF ==>  OUTPUT ==>  CNTL ==>
ICSC     - Inline Comment Start Column (4 - 71)..... ==> 37
JOB ==>  EXEC ==>  DD ==>  JCLLIB ==>  INCLUDE ==>
SET ==>  PROC ==>  IF ==>  OUTPUT ==>  CNTL ==>
OPSC     - Operator Start Column (4 - 16)..... ==> 12
JOB ==>  EXEC ==>  DD ==>  JCLLIB ==>  INCLUDE ==>
SET ==>  PROC ==>  IF ==>  OUTPUT ==>  CNTL ==>
PACK     - Max Number of Keyword Parms on Each Card (1 - 80)..... ==> 1
JOB ==>  EXEC ==>  DD ==>  JCLLIB ==>  INCLUDE ==>
SET ==>  PROC ==>  IF ==>  OUTPUT ==>  CNTL ==>
```

## Optional Data Sets (JCK0206)

The Optional Data Sets panel allows specification of optional data sets and sources of input data to support additional special features of the system. The panel consists of two sections:

### Member Libraries

Identifies the input and output JCL member data sets and format.

### Additional Control

Identifies the additional optional control data required.

This example shows the format that is used to specify error conditions. In errors that are syntax or process related, a short message appears in the upper right corner and the cursor positions itself on the error. If you need more information, press the HELP PF Key (usually PF1) for a longer message on line 3. Pressing the HELP PF Key again activates the Field Level HELP.

The bottom lines indicate which of the five subsequent panels display. This example shows all five panels display. If the panel ADDCARD to Output equals N, that panel would not display.

```
JCK0206 ----- CA JCLNeat xx.x - OPTIONAL Datasets (SUBMIT)   ERROR ON OUTLIB
COMMAND ==>
                                                    yyyy/mm/dd
COMMANDs: CANCEL; NEXT or PREVIOUS panel after edit.          hh:mm
KEYs:     ENTER to edit. END to edit, go to next panel.      USERID1

INLIB Dataset Name... ==> 'USERID1.JCK.FILE'
Library Type... ==> SEQ (SEQ,PDS,PAN,LIB)
Volume Serial.. ==> _____ (If Not CATALOGED)
OUTLIB Dataset Name.. ==> 'USERID1.NOTEXIST.PRINT'
Library Type... ==> SEQ (SEQ,PDS)
Volume Serial.. ==> _____ (If not CATALOGED)

MEMBERS Selected..... ==> _____ (*,Include,Exclude,blank for SEQ)
KEYWORDS Ordering.... ==> Y (Y/N)
CHANGES to Strings... ==> Y (Y/N)
ADDCARDS to Output... ==> Y (Y/N)
GSRCNTL Control..... ==> Y (Y/N)
```

## Member Selection List (JCK0216)

The Member Selection List panel allows the selection of JCL members from the specified INLIB library for "neat" processing. In the Optional Data Sets panel, the field labeled MEMBERS Selected determines whether this panel is displayed, and if so, the use of this panel:

\*

Process all members of the library.

### INCLUDE

Display MEMBERS panel for selection to include.

### EXCLUDE

Display MEMBERS panel for selection to exclude.

This panel lists all the members on a PDS, PANVALET, or Librarian JCL Library for selection. In the example shown here, it is a PDS and two members have been selected (selection is by any non-blank character). Members specified here have their names copied to and input as data to the MEMBERS DD in the JCLNeat Job stream submitted for execution as follows:

```
//MEMBERS DD *
member1
member1+n
/*
```

JCK0216 ----- CA JCLNeat xx.x - MEMBERS (SUBMIT)							Row 1 to 22 of 132	
COMMAND ==>							SCROLL ==> CSR_	
COMMANDS: CANcel; NEXT or PREVIOUS panel after edit, save.							yyyy/mm/dd	
KEYS: Enter to edit, save. End to edit, save, go to next panel.							17:07	
							USERSID	
Selection to INCLUDE - PDS INLIB: 'APC.DEVJCK.ASM'								
s member	added	last change		userid	size	changes		
_ \$JCLXAWA	98/08/14	00/07/17	17:01	GOURI01	92	0		
_ #README	98/12/11	99/06/16	08:04	MALDE01	14	14		
s AAMACRO	00/07/12	00/07/18	15:22	USERSID	42	34		
s CA\$DSECT	99/11/16	99/11/16	12:53	USERSID	15	4		
_ CAZ0NEXE	00/05/30	00/05/30	11:24	USERSID	12	6		
_ CAZ1JREX	98/08/14	99/04/26	15:32	MALDE01	1130	0		
_ CAZ1NBLD	98/08/14	99/04/26	15:33	MALDE01	767	0		
_ CAZ1NEDN	98/08/14	00/07/18	10:02	MOOJ001	838	0		
_ CAZ1NGBL	98/08/14	99/06/16	07:02	MALDE01	438	149		
_ CAZ1NGPP	98/08/14	99/04/26	15:33	MALDE01	303	0		
_ CAZ1NGSR	98/08/14	99/04/26	15:34	MALDE01	701	0		
_ CAZ1NIDO	98/08/14	99/06/16	07:03	MALDE01	224	3		
_ CAZ1NIRO	98/08/14	99/06/16	06:49	MALDE01	1253	126		
_ CAZ1NRXE	98/08/14	00/02/29	17:18	MOOJ001	153	0		
_ CAZ1NRXI	98/08/14	99/04/26	15:34	MALDE01	643	0		
_ CAZ1NSUB	98/08/14	99/04/26	15:35	MALDE01	664	0		

## Keyword Ordering (JCK0215)

The KEYWORD panels allow for the review and change of ordering of keywords used in the JCL TYPE commands of EXEC, DD, DCB, OUTPUT, PRINTDEV, and PROC.

In the Optional Data Sets panel, the field labeled KEYWORDS Ordering determines the use of this panel. If it contains a Y, the system expects all six JCL TYPEs to be reviewed and ordering set to the Defaults or changed to whatever is appropriate. This panel is not displayed if N was selected.

The resulting ordered keywords are copied to and input as data to the KEYWORDS DD in the JCLNeat job stream submitted for execution. Only the JCL TYPE selected and reviewed participate in the processing, a positive response is required for inclusion and no assumptions are made.

In the example following, the default starting point is JCL TYPE 1 (EXEC). Once it is reviewed, pressing End includes it in the process.

Each of the subsequent KEYWORD (SUBMIT) screens are handled in the same manner.

```
JCK0215 ----- CA JCLNeat xx.x - KEYWORD (SUBMIT) -----
COMMAND ==>
                                                    yyyy/mm/dd
COMMANDS: CANCEL; NEXT or PREVIOUS panel after edit, save.          17:10
KEYs:      Enter to edit, save. End to edit, save, next panel.     USERSID

Select JCL TYPE on OPTION line, assign sequencing number by selected KEYWORD.
Keywords not selected will follow those sequenced, in alphabetic order.

.---- JCL TYPE ----- KEYWORD ORDER SPECIFICATION -----
:                               : 10 ACCT   _9 ADDRSPC  _8 COND   _7 DPRTY  :
: THIS SEL ==> 1 :         _6 DYNAMNBR  _5 PARM   _4 PERFORM  _3 RD    :
:                               :         _2 REGION    _1 TIME                               :
: EXEC..... 1 :                               :                               :
: DD..... 2 :                               :                               :
: DCB..... 3 :                               :                               :
: OUTPUT.... 4 :                               :                               :
: PRINTDEV... 5 :                               :                               :
: PROC..... 6 :                               :                               :
: LAST SEL ==> :                               :                               :
-----

DO YOU WISH TO MAKE THESE SELECTIONS PERMANENT? N (Y/N)
```

## ADDCARDS (JCK0211)

The ADDCARDS panel allows insertion of up to five control statements to be inserted before each processed module being placed on the OUTLIB sequential data set. This panel allows the OUTLIB to be used as input to source library products, for example, CA Librarian, CA Panvalet, or IBM IEBUPDTE. The data strings entered here result as input to JCLNeat as shown:

```
//ADDCARDS DD *
-ADD
-HST JCLNEAT CONTROL MODULES
/*
```

When the submitted job stream processes the members, the name of the member is copied after the first blank of the first control statement. The remaining control statements are copied following the first statement.

```
JCK0211 ----- CA JCLNeat xx.x - ADDCARDS (SUBMIT) -----
COMMAND ==>
                                         yyyy/mm/dd
COMMANDS: CANCEL; NEXT or PREVIOUS panel after edit, save.          17:32
KEYS:      Enter to edit, save. End to edit, save, next panel.     USERSID

Specify up to 5 JCL statements below, to be placed in front of each member
so the OUTLIB, when specified, can be used as input where control statements
are required (i.e. Librarian, IEBUPDTE, etc.).

- JOB, MEMBER, or PROC Name copied after last character of first ADDCARD.
- All subsequent ADDCARDS copied following first ADDCARD with Name.

==> _____
==> _____
==> _____
==> _____
==> _____
```

## CHANGES (JCK0212)

Global string editing provides the ability to search for a from-string and when found change it to the value specified in the to-string. This panel allows the creation of these data strings that will be used as search criteria to scan all the data portions of the JCL keyword parameters, and when found change them as indicated. The data strings entered here will result as input to JCLNeat as shown:

```
//CHANGES DD *
/SYSDA/SYSALLDA/
/*
```

```

JCK0212 ----- CA JCLNeat xx.x - CHANGES (SUBMIT) - Row 1 to 1 of 1
COMMAND ==>                                     SCROLL ==> CSR_

COMMANDS: CANcel; NEXT OR PREVIOUS panel after edit, save.          yyyy/mm/dd
KEYS:     Enter to edit, save. End to edit, save, next panel.      17:37
                                                USERSID

Specify the STRINGS of data to be found and changed in the data part
of the Keyword parameters. The /from-string/ is searched and when found
changed to the to-string/ . Only the /from-string/ begins with a / but
both must end with one. Actions (act) Insert, Delete (I,D) of string sets,
to Update over-key the data to be changed.

act                                     /from-string/to-string/

***** Bottom of data *****

```

### GSRCNTL (JCK0213)

Global string editing lets you search for a from-string. When found, the from-string is changed to the value specified in the to-string. This panel lets you create these data strings. The data strings are used as search criteria to scan all the data portions of the JCL keyword parameters, and when found change them as indicated. The data strings entered here result as input to JCLNeat as shown:

```

//GSRCNTL DD *
|UNIT=SYSALLDA|INSA|
|DISP=OLD|
|DISP=SHR|DELT|
/*

```

```

JCK0213 ----- CA JCLNeat xx.x - GSRCNTL (SUBMIT) ----- Row 1 of 2
COMMAND ==>                                     SCROLL ==> PAGE

COMMANDS: CANcel; NEXT or PREVIOUS panel after edit.              yyyy/mm/dd
KEYS:     ENTER to edit. END to edit and go to next panel.        hh:mm
                                                USERID

Specify STRING sets for each change you want to make. Each set consists of a
from string, verb, statement filter and a to string. Sets can be Inserted,
or Deleted by specifying I,D in the 'Act(I,D)' field.

Verbs: REPL(replace with TO:), DELT(delete JCL stmt), COMM(comment JCL stmt),
      INSA(insert TO: after found line), INSB(insert TO: before found line)

ACT (I,D) _ VERB: INSA      JCL statement filter: _____
FROM: UNIT=SYSALLDA_____
TO : DISP=OLD_____
ACT (I,D) _ VERB: DELT     JCL statement filter: _____
FROM: DISP=SHR_____
TO : _____

```

## JOB CARD for Submit Mode (JCK0214)

The JOB CARD panel lets you review the JOB JCL statement and its associated JCL. You can verify and change the way that the submitted job executes. Optionally, the following two functions are available from this panel:

- Update PROFILE with these options (Permanently save the JCL as the default).
- Edit the JCL JOB Stream before submitting (Review the generated JCL).

```
JCK0214 ----- CA JCLNeat xx.x - JOB CARD (SUBMIT) -----
COMMAND ==>

COMMANDs: CANCEL; NEXT or PREVIOUS panel.                yyyy/mm/dd
KEYs:     ENTER to save, END to save, exit to MENU.      hh:mm
                                                    USERID1

Job Statement Information: (Verify before saving)
=> //USERID1J JOB '113300000',USER1,NOTIFY=USERID1,MSGLEVEL=(1,1),
=> // CLASS=A,MSGCLASS=X,USER=USERID1,REGION=0M
=> /*JOBPARM ROOM=716A
=> /**
    Update PROFILE with these options      ==> Y
    Edit JCL JOB Stream before submitting  ==> Y
PARM=( BCI=Y,BCS=N,CDEL=0,CEC=35,CSC=16,DCBS=N,FCC=-,FORMAT=Y,JOBF=Y,ICSC=37, )
      ( INCR=10,INLIB=PDS,LINES=55,MKW=60,MQKW=10,OPSC=12,OUTLIB=,PACK=1,RECOV )
      ( ER=Y,REPORT=7,RSYM=N,SECC=*,SEQ1=10,SIM=N )
      ( )
      ( )
      ( )
      ( )
      ( )
      ( )
      ( )
      ( )
      ( )
```

This panel also displays the contents of the PARM= parameter string. The contents are based on input from the previous JCLNeat panels. These parameters are inserted into the JCLNeat job stream before submission.

## ISPF Edit of the Job Submission JCL

If you responded to 'Edit JCL JOB Stream before submitting' with a Y, the following panel is shown. This panel allows the submission JCL to be reviewed and changed, and optionally manually submitted with the SUBMIT command. Alternatively, you can allow the system to submit it.

```

File Edit Confirm Menu Utilities Compilers Test Help
-----
ISREDDE2  SYS00222.T163707.RA000.USERSID.R0131100      Columns 00001 00072
Command ==>                                         Scroll ==> CSR
***** ***** Top of Data *****
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 //JOBNAME JOB (ACCT),NAME
000002 //*
000003 //*
000004 //*
000005 //*****//
000006 //*                                     *//
000007 //*  JCLNEAT BATCH SUBMISSION PROCEDURE FROM ONLINE JCLCHECK MENU  *//
000008 //*                                     *//
000009 //*****//
000010 //JCLNEAT EXEC PGM=CAZ1NEAT,
000011 //          REGION=0K
000012 //JCLNOPTS DD *
000013 INLIB=PDS
000014 BCS=Y
000015 FCC=-
000016 JOBF=N
000017 MEMBERS=INCLUDE
000018 REPORT=0

```



This feature permits you to construct an input stream for subsequent IEBUPDTE, CA Librarian, or CA Panvalet processing.

**Note:** If you request sequential output but do not specify any ADDCARDS, the JCLNeat resultant output is a continuous JCL stream.

The option combination of INLIB=SEQ and OUTLIB=PDS is not supported by JCLNeat.

## Using JCLNeat under CA Roscoe

The JCKNRPF permits you to reformat JCL. It consists of only two panels: the Option panel, which can be hidden, and the File Specification panel.

Normally, the OPTIONS panel is hidden from view. This *hiding* is accomplished if you enter Y in response to the question "Bypass this panel?" on the OPTIONS panel. This is the last question on the panel. This panel reappears if you use a single optional parameter when executing the RPF.

The following are valid values for the argument and the results:

### EXEC\_JCKNRPF PANEL

PANEL argument (passed parameter) causes the Hidden Panel to display so that you can optionally make changes.

The RPF also permits the use of OPTION or OPTIONS as alternate arguments to achieve the same results.

### EXEC\_JCKNRPF RESET

Forces all parameters to reset to the default values listed in the *Reference Guide* for Version 6.0 or higher.

The RPF also permits use of DEFAULT, DEFAULTS, INITIALIZE, and INITIAL as alternate arguments to achieve the same result.

**Note:** The PANEXEC option is not available under CA Roscoe.

If the OPTIONS panel is not hidden and you can enter values in it, you are presented with a File Specification panel. The File Specification panel permits you to choose from a wide variety of I/O options. Select any of the following logical files for input/output:

- CA Roscoe library member
- AWS
- Sequential data set
- PDS member

Any combination is permitted. For example, you can reformat AWS-to-AWS, PDS(MEM)-to-RosLib, and so on. This RPF permits you to reformat the same PDS member. When this occurs, the RPF informs you that the member is replaced. However, we recommend that you do not replace members until you gain a mastery of how the reformatting option works.

**Important!** YOU CAN REFORMAT BUT YOU CANNOT UNFORMAT! If you do not replace the member, you can always return to it.

You must decide on a logical file to hold the messages for JCLNeat, either by using the TERMINAL or by selecting a flat file (sequential output) to hold the reports from JCLNeat. If you decide on a flat file, the RPF follows the General File Allocation technique described previously.

# Chapter 6: JCLNeat Options

---

The JCLNeat component contains default options that are entirely different from the CA JCLCheck runtime options. The JCLNeat default options control how the JCLNeat component operates and interprets data. You can change these defaults as discussed in the chapter "Using JCLNeat." With each execution of JCLNeat, you supply required changes to the defaults.

JCLNeat evaluates the requested options at the beginning of its processing. An error in an option specification results in JCLNeat terminating its execution. JCLNeat analyzes all of the options specified prior to terminating to diagnose multiple errors.

Specification of certain options determines the ddname that CA JCLCheck OPENS for processing by JCLNeat.

## Summary of JCLNeat Options by Function

This section lists the JCLNeat options by function, it also gives a brief description of each option.

### Input Options

Input options specify source JCL information.

#### **INLIB**

Specifies the data type of the input for this invocation of JCLNeat, determining the ddname to OPEN.

#### **MEMBERS**

Selects members of the input partitioned data set or source management library to process.

#### **MKW**

Specifies the maximum number of keywords expected on an input JCL statement.

#### **MQKW**

Specifies the maximum number of qualified keywords expected on an input EXEC JCL statement.

## Comment Options

Comment options specify how JCLNeat formats comments in the JCL.

### **BCI**

Boxes stand-alone comments in the reformatted JCL.

### **BCS**

Inserts a blank comment statement after each JCL statement.

### **CDEL**

Specifies the disposition of comments CA JCLCheck encounters in the input stream.

### **FCC**

Specifies the fill character JCLNeat uses in the stand-alone comments box.

### **ICSC**

Specifies the column number on which to start inline comments on the reformatted JCL statement.

Additionally, the following statement level options for each specific JCL statement are available:

- ICSCJ—for JOB statements
- ICSCEX—for EXEC statements
- ICSCDD—for DD statements
- ICSCIF—for IF/ELSE/ENDIF statements
- ICSCIN—for INCLUDE statements
- ICSCJL—for JCLLIB statements
- ICSCOP—for OUTPUT statements
- ICSCPR—for PROC/PEND statements
- ICSCS—for SET statements
- ICSCCN—for CNTL/PRINTDEV/ENDCNTL statements

### **SECC**

Specifies the character for the start and end of a stand-alone comments box.

## Debugging Options

Debugging options specify diagnostic information that is useful to CA Technical Support.

### **RECOVER**

Specifies whether to use the JCLNeat recovery environment.

## JCL Statement Reformatting Options

JCL statement reformatting options specify how to reformat the JCL for JCLNeat.

### CEC

Specifies the ending column number to use on a JCL statement for placement of keywords and associated data before generating a continuation statement.

Additionally, the following statement level options for each specific JCL statement are available:

- CECJ—for JOB statements
- CECEX—for EXEC statements
- CECDD—for DD statements
- CECIF—for IF/ELSE/ENDIF statements
- CECIN—for INCLUDE statements
- CECJL—for JCLLIB statements
- CECOP—for OUTPUT statements
- CECPR—for PROC/PEND statements
- CECS—for SET statements
- CECCN—for CNTL/PRINTDEV/ENDCNTL statements

### CSC

Specifies the starting column number for the first keyword on each JCL statement.

Additionally, the following statement level options for each specific JCL statement are available:

- CSCJ—for JOB statements
- CSCEX—for EXEC statements
- CSCDD—for DD statements
- CSCIF—for IF/ELSE/ENDIF statements
- CSCIN—for INCLUDE statements
- CSCJL—for JCLLIB statements
- CSCOP—for OUTPUT statements
- CSCPR—for PROC/PEND statements
- CSCS—for SET statements
- CSCCN—for CNTL/PRINTDEV/ENDCNTL statements

### DCBS

Controls splitting of the DCB subparameters across several statements.

### **JOBF**

Specifies formatting of the JOB statement.

### **OPSC**

Specifies the starting column number for the JCL operator on the reformatted JCL statement.

Additionally, the following statement level options for each specific JCL statement are available:

- OPSCJ—for JOB statements
- OPSCEX—for EXEC statements
- OPSCDD—for DD statements
- OPSCIF—for IF/ELSE/ENDIF statements
- OPSCIN—for INCLUDE statements
- OPSCJL—for JCLLIB statements
- OPSCOP—for OUTPUT statements
- OPSCPR—for PROC/PEND statements
- OPSCS—for SET statements
- OPSCCN—for CNTL/PRINTDEV/ENDCNTL statements

### **PACK**

Specifies the maximum number of keyword parameters that can be placed between the starting and ending columns.

Additionally, the following statement level options for each specific JCL statement are available:

- PACKJ—for JOB statements
- PACKEX—for EXEC statements
- PACKDD—for DD statements
- PACKIF—for IF/ELSE/ENDIF statements
- PACKIN—for INCLUDE statements
- PACKJL—for JCLLIB statements
- PACKOP—for OUTPUT statements
- PACKPR—for PROC/PEND statements
- PACKS—for SET statements
- PACKCN—for CNTL/PRINTDEV/ENDCNTL statements

**REXXMEM**

Specify the REXX member name for standards and manipulation of JCL.

When specifying REXXMEM in ISPF panels JCKN01 or JCK0205, you can use standard IBM member pattern characters to display a member list for the data sets that are allocated to your SYSEXEC DD. Use the asterisk (\*) for multiple characters and the per cent sign (%) for a single character. Select the member that you want to use from the list and press Enter. The selected member name fills in as the REXXMEM value.

## Output Options

Output Options specify additional JCL output processing information for JCLNeat.

**FORMAT**

Specifies how to format the JCL.

**INCR**

Specifies the increment between sequence numbers for reformatted JCL statements.

**LINES**

Specifies the number of lines to print on each report page.

**OPTDESC**

Provides a brief description of all JCLNeat options in effect.

**OUTLIB**

Specifies the data type of the JCLNeat output, determining the ddname to OPEN.

**REPORT**

Specifies the types of reports to generate.

**RSYM**

Indicates whether JCLNeat should replace the symbolic parameters in PROC statements in the original JCL in the reformatted JCL.

**SEQ1**

Specifies the beginning sequence number for each new member or job in the output stream.

**SIM**

Specifies simulation of JCLNeat processing.

## Description of JCLNeat Options

This section describes the JCLNeat options.

### BCI

The Block Comment Insert (BCI) option specifies whether JCLNeat boxes stand-alone comments in the reformatted JCL. A box is composed of an opening delineation line (defined by the SECC and FCC options), a blank comment line, the stand-alone comment, a blank comment line, and a closing delineation line.

JCLNeat places Column 4 of the original comment in Column 6 of the reformatted comment, followed by the first 65 columns of the original comment.

This option has the following format:

BCI=Y|N

#### Y

Boxes stand-alone comments in the reformatted JCL. This is the default.

#### N

Does not box stand-alone comments in the reformatted JCL. Specifying N causes JCLNeat not to produce a box. Specify N to eliminate production of multiple boxes during repeat runs of the JCLNeat utility.

#### Example: Box a stand-alone comment

This example PARM sets BCI to yes, which produces the boxed JCL that follows:

```
PARM='BCI=Y,SECC=*,FCC=-'
```

```
Column Number
12345678
/**-----*
/**                               *
/**STAND-ALONE COMMENT           *
/**                               *
/**-----*
```

## BCS

The Blank Comment Separator (BCS) option specifies whether JCLNeat inserts a blank comment statement after each JCL statement.

This option has the following format:

BCS=N|Y

**N**

Does not insert a comment statement. This is the default.

**Y**

Inserts a `/**` comment statement after each JCL statement in the reformatted JCL.

## CDEL

The Comment Delete (CDEL) option determines the disposition of comments CA JCLCheck encounters in the input stream.

This option has the following format:

CDEL=0|1|2|3|4|9

**0**

Indicates delete no comments. This is the default.

**1**

Indicates delete all comments.

**2**

Indicates delete stand-alone comments.

**3**

Indicates delete blank stand-alone comments.

**Note:** If you specify CDEL=3, the blank stand-alone comment statements that JCLNeat inserts in response to BCS=Y are not deleted.

**4**

Indicates delete inline comments.

9

Requests that comments, imbedded and inline, are to be locked or associated with the keyword that precedes it.

An inline comment is one that is on the same card as the keyword.

An imbedded comment is one that is on a line by itself within the current logical statement.

**Example: Inline comment**

```
//DDNAME DD DSN=MY.DSN,      This is an inline comment
//          DISP=SHR
```

**Example: Imbedded comment**

```
//DDNAME DD DSN=MY.DSN,
//* This is an imbedded comment
//          DISP=SHR
```

In the examples above, both the inline and imbedded comments will be locked (Associated) with the DSN keyword.

## CEC

The Continuation End Column (CEC) option specifies the ending column number to use on a JCL statement. If additional keywords exist on the input JCL statement, CA JCLCheck generates continuation lines.

This option has the following format:

*CEC=column-number*

**column-number**

Defines a column from 4 to 71. The default is Column 35.

Additionally, the following statement level options for each specific JCL statement are available:

- CEJ—for JOB statements
- CECEX—for EXEC statements
- CECDD—for DD statements
- CECIF—for IF/ELSE/ENDIF statements
- CECIN—for INCLUDE statements
- CECJL—for JCLLIB statements

- CECOP—for OUTPUT statements
- CECPR—for PROC/PEND statements
- CECS—for SET statements
- CECCN—for CNTL/PRINTDEV/ENDCNTL statements

If a statement level option is not specified, JCLNeat uses the associated global option value. If a statement level option is specified, this value overrides the associated global option value.

**Note:** JCLNeat attempts to divide the parameter at a logical point such as, a comma, when PACK=1 and the data portion of the keyword exceeds CEC. PACK=1 means only one keyword parameter per reformatted JCL statement line. If you cannot divide the keyword parameter, the data exceeds CEC. When the data exceeds column 71, JCLNeat backspaces the keyword start (CSC) until it can contain the entire keyword on the JCL statement line.

## CSC

The Continuation Start Column (CSC) option specifies the starting column number for the first keyword on each JCL statement line.

This option has the following format:

*CSC=column-number*

### **column-number**

Defines a value from 4 through 16. The default is Column 16.

For JCL containing the operator, the CSC is the ending column of the operator + 1 if the specified column is not available.

Additionally, the following statement level options for each specific JCL statement are available:

- CSCJ—for JOB statements
- CSCEX—for EXEC statements
- CSCDD—for DD statements
- CSCIF—for IF/ELSE/ENDIF statements
- CSCIN—for INCLUDE statements
- CSCJL—for JCLLIB statements
- CSCOP—for OUTPUT statements
- CSCPR—for PROC/PEND statements

- CSCS—for SET statements
- CSCCN—for CNTL/PRINTDEV/ENDCNTL statements

If a statement level option is not specified, JCLNeat uses the associated global option value. If a statement level option is specified, this value overrides the associated global option value.

**Note:** JCLNeat overrides CSC when a reformatted JCL statement cannot contain a single keyword parameter and its data. See the [CEC](#) (see page 292) option for a description of this processing logic.

## DCBS

The DCB split (DCBS) option controls the splitting of DCB subparameters across several statements.

This option has the following format:

DCBS=N|Y

**N**

Specifies not to split DCB subparameters. This is the default.

**Y**

Specifies that JCLNeat place one DCB subparameter per statement. If you specify Y, JCLNeat ignores the value in CEC for the DCB subparameters.

## FCC

The Fill Comment Character (FCC) option specifies the character JCLNeat uses in Columns 5-70 of the boxes JCLNeat produces for stand-alone comments.

This option has the following format:

FCC=*character*

The default character is a hyphen (-).

## FORMAT

The FORMAT option specifies if the JCL is formatted before generating the output file.

This option has the following format:

FORMAT=Y|N|B

### Y

Turns on formatting. This option is the default.

### N

Turns off formatting (REXXMEM cannot be used with N).

### B

Turns on Basic formatting. Basic formatting allows REXXMEM to be used while minimizing the changes to the original format JCL.

### Usage notes:

- If you are using FORMAT=N without global string replacement, JCLNeat effectively performs a COPY operation.
- JCLNeat is designed to perform formatting. Therefore, you cannot specify FORMAT=N as a system installation default. Additionally, this option is not available under the ISPF environment.
- When FORMAT=B is specified, JCLNeat overrides several options to adjust them as needed based on the input JCL. Any values that are specified for these options are not used. These options include:  
CSC, ICSC, BCI, BNS, CDEL, CEC, JOBF, OPSC, PACK, AND RSYM.
- When using FORMAT=B, changes that REXX makes to the \$CA.OPTRESET and \$CA.OPTS variables are ignored. Changing JCLNeat Options from REXX is not supported when using Basic formatting.
- FORMAT=B minimizes changes to the original formatting, with the following exceptions:
  1. The output JCL includes changes that the REXXMEM user REXX routine makes. These changes could modify the lengths of parameters, add additional parameters, or add and delete entire statements. These changes affect the resulting JCL.

- Parameters that contain multiple subparameters, such as DCB=, COND=, VOL=, LABEL=, and PARM=, could possibly not format as entered. This format change occurs when they are split across multiple lines in the original JCL. Generally, if the parameter can fit on a single line then the multiple lines are consolidated on a single line. If the parameter requires multiple lines, the split of the parameter is done without regards for the original format. Comments that are contained within a parameter that was split across lines are moved to separate comment lines.

## ICSC

The Inline Comment Start Column (ICSC) option specifies the column number on which to start inline comments on the reformatted JCL statement.

This option has the following format:

*ICSC=column-number*

### ***column-number***

Specifies a number from 4 through 71. A specification of column 71 causes JCLNeat to right-justify the inline comment on the JCL statement. The default is Column 37.

Additionally, the following statement level options for each specific JCL statement are available:

- ICSCJ—for JOB statements
- ICSCJL—for JCLLIB statements
- ICSCDD—for DD statements
- ICSCIF—for IF/ELSE/ENDIF statements
- ICSCIN—for INCLUDE statements
- ICSCOP—for OUTPUT statements
- ICSCPR—for PROC/PEND statements
- ICSCS—for SET statements
- ICSCCN—for CNTL/PRINTDEV/ENDCNTL statements

If a statement level option is not specified, JCLNeat uses the associated global option value. If a statement level option is specified, this value overrides the associated global option value.

## INCR

The Increment (INCR) option specifies the increment between sequence numbers for reformatted JCL statements.

This option has the following format:

`INCR=number`

***number***

Specifies a value to increment between sequence numbers. This is the default is 10.

**Limits:** 1-99999999

## INLIB

The Input Library (INLIB) option specifies the data type of the input for this invocation of JCLNeat, determining the ddname to OPEN.

This option has the following format:

`INLIB=LIB|PAN|PDS|SEQ`

**LIB**

Specifies CA Librarian input and opens the LIBIN DD statement.

**PAN**

Specifies CA Panvalet input and opens the PANIN DD statement.

**PDS**

Specifies partitioned data set input and opens the PDSIN DD statement.

**SEQ**

Specifies sequential input and opens the SEQIN DD statement. The default is SEQ.

**Note:** JCLNeat does not support the options combination INLIB=SEQ and OUTLIB=PDS.

When using INLIB=PDS and OUTLIB=PDS, the PDSIN and PDSOUT DD statements must not reference the same PDS or JCLNeat discontinues processing with a CAY6504S message.

## JOBF

The JOB statement Format (JOBF) option specifies formatting of the JOB statement.

This option has the following format:

JOBF=Y|N

**Y**

Formats the JOB statement. This is the default.

**N**

Does not format the JOB statement.

## LINES

The LINES option specifies the number of lines to print on each report page. This includes headers and blank lines JCLNeat inserts.

This option has the following format:

LINES=*number*

***number***

Specifies the number of lines per page. The default is 55.

**Limits:** 10 through 72

## MEMBERS

The MEMBERS option selects members of the input partitioned data set or source management library to process.

This option has the following format:

MEMBERS=\*|INCLUDE|EXCLUDE

**\***

Specifies processing of all the members in the input data set. This is the default.

**INCLUDE**

Specifies processing of only the members listed in the MEMBERS DD data set.

**EXCLUDE**

Specifies processing of all the members of the input data set except those listed in the MEMBERS DD data set.

## MKW

The Maximum Keyword (MKW) slots option specifies the maximum number of keywords expected on an original JCL statement. It ensures adequate working storage for JCLNeat to process the statement. We recommend that you set MKW at 60 or higher. CA JCLCheck issues a CAY6508E message if the value is too low to process the JCL.

This option has the following format:

*MKW=number*

**number**

Specifies the maximum number of keywords. The default is 60.

**Limit:** 10-999

## MQKW

The Maximum Qualified Keyword (MQKW) slots option specifies the maximum number of qualified keywords JCLNeat expects on an original EXEC JCL statement. MQKW ensures that there is adequate working storage to process an EXEC procedure JCL statement.

This option has the following format:

*MQKW=number*

**number**

Specifies the maximum number of keywords. The default is 10.

## OPSC

The Operator Start Column (OPSC) option specifies the starting column number for the JCL operator on the reformatted JCL statement.

This option has the following format:

*OPSC=column-number*

**column-number**

Specifies the starting column number. The default is Column 12.

**Limits:** 4 through 16

Additionally, the following statement level options for each specific JCL statement are available:

- OPSCJ—for JOB statements
- OPSCEX—for EXEC statements
- OPSCDD—for DD statements
- OPSCIF—for IF/ELSE/ENDIF statements
- OPSCIN—for INCLUDE statements
- OPSCJL—for JCLLIB statements
- OPSCOP—for OUTPUT statements
- OPSCPR—for PROC/PEND statements
- OPSCS—for SET statements
- OPSCCN—for CNTL/PRINTDEV/ENDCNTL statements

If a statement level option is not specified, JCLNeat uses the associated global option value. If a statement level option is specified, this value overrides the associated global option value.

## OPTDESC

The OPTDESC option provides a brief description of all JCLNeat options in effect. This information is displayed on the Options Report, see [JCLNeat Reports](#) (see page 285).

This option has the following format:

OPTDESC=Y|N

**Y**

Specifies to include the brief description. This is the default.

**N**

Specifies not to include the description.

## OUTLIB

The OUTLIB option specifies the data type of the JCLNeat output, determining the ddname to OPEN.

This option has the following format:

OUTLIB=PDS|SEQ

### PDS

Specifies output to a partitioned data set. The PDSOUT DD statement is opened.

### SEQ

Specifies output to a sequential data set. The SEQOUT DD statement is opened. This is the default.

**Note:** CA JCLCheck does not support the options combination INLIB=SEQ and OUTLIB=PDS.

When using INLIB=PDS and OUTLIB=PDS, the PDSIN and PDSOUT DD statements must not reference the same PDS or JCLNeat discontinues processing with a CAY6504S message.

## PACK

The PACK option specifies the maximum number of keyword parameters that can be placed between the starting and ending columns on each JCL statement.

This option has the following format:

PACK=*max-number*

### *max-number*

Specifies the maximum number of keyword parameters. The default is 1.

**Limits:** 1 through 80

Additionally, the following statement level options for each specific JCL statement are available:

- PACKJ—for JOB statements
- PACKEX—for EXEC statements
- PACKDD—for DD statements
- PACKIF—for IF/ELSE/ENDIF statements
- PACKIN—for INCLUDE statements
- PACKJL—for JCLLIB statements

- PACKOP—for OUTPUT statements
- PACKPR—for PROC/PEND statements
- PACKS—for SET statements
- PACKCN—for CNTL/PRINTDEV/ENDCNTL statements

If a statement level option is not specified, JCLNeat uses the associated global option value. If a statement level option is specified, this value overrides the associated global option value.

Usage notes:

- For compatibility with earlier releases, PACK=N and PACK=Y are still accepted. PACK=N is treated as PACK=1 (each reformatted JCL statement contains only one keyword parameter and its data). PACK=Y is treated as PACK=80 (each reformatted JCL statement contains as many keyword parameters as possible within the limits of the starting column and Column 71).
- JCLNeat converts any inline comments that it encounters into stand-alone comments and output before the reformatted JCL statement.
- The following rules apply when using DCBS and PACK together:
  - When PACK=1 (or PACK=N) and DCBS=Y are specified, the DCB subparameters are each split onto separate JCL statements.
  - With all other combinations of PACK and DCBS, the string of DCB subparameters are all considered one single keyword parameter.

## RECOVER

The RECOVER option specifies whether JCLNeat intercepts ABEND conditions for diagnosis.

This option has the following format:

RECOVER=Y|N

**Y**

Establishes the recovery environment. This is the default.

**N**

Does not establish a recovery environment.

## REXXMEM

The REXXMEM option invokes a REXX EXEC to process the input JCL and update it according to the rules specified in the REXX EXEC.

This option has the following format:

REXXMEM=*member-name*

### ***member-name***

Indicates the REXX program name for update processing. This member name is loaded from the SYSEXEC ddname.

### **Usage notes:**

- This option is mutually exclusive with the FORMAT=N option. JCL is always reformatted when the REXX interface is active. FORMAT=B can be used to minimize reformatting.
- Include a SYSEXEC ddname either in the JCL (batch run) or in your installations TSO logon PROC/CLIST. If you use EXECUTIL SEARCHDD(YES/NO) under TSO, it determines whether you need SYSEXEC, SYSPROC, or both. See your systems programmer or the IBM manual for a complete explanation of the EXECUTIL command.
- The member is loaded from the installations default library. The installation default library is site-specific (the default is the SYSEXEC DD).
- When specifying REXXMEM in ISPF panels JCKN01 or JCK0205, you can use standard IBM member pattern characters to display a member list for the data sets that are allocated to your SYSEXEC DD. Use the asterisk (\*) for multiple characters and the per cent sign (%) for a single character. Select the member that you want to use from the list and press Enter. The selected member name fills in as the REXXMEM value.

## REPORT

The REPORT option specifies the combination of reports to generate.

This option has the following format:

REPORT=0|1|2|3|4|5|6|7

You can generate reports by selecting one of the following:

**0**

Specifies reporting of only the options used and error messages

CA JCLCheck always includes the Option Report with any of the other reporting options 1-7.

**1**

Requests the Original JCL Report (Report 1)

**2**

Requests the Reformatted JCL Report

**3**

Requests the Changes Report

**4**

Requests the Original and Reformatted Reports

**5**

Requests the Original Report and the Changes Report

**6**

Requests Reports Reformatted Report and the Changes Report

**7**

Requests the Original Report, the Reformatted Report, and the Changes Report. This is the default.

## RSYM

The RSYM option indicates whether JCLNeat should replace the symbolic parameters in PROC statements in the original JCL in the reformatted JCL.

This option has the following format:

RSYM=Y|N

### Y

Replaces the symbolic parameters in the reformatted JCL. JCLNeat does not replace symbolics with a null value (SYMBOL=,). You can specify other symbolics to exclude from replacement as an entry in the KEYWORDS DD data set.

### N

Does not replace the symbolic parameters. This is the default.

## SECC

The SECC option specifies the character to use for the start and end of a stand-alone comments box. JCLNeat places character in Columns 5 and 70 of the stand-alone comment statements JCLNeat uses to create a box when you specify BCS=Y.

This option has the following format:

SECC=*character*

### *character*

Specifies a one-character field for start and end stand-alone comments box. The default is \* (asterisk).

**Limits:** The character cannot be & (ampersand).

## SEQ1

The SEQ1 option specifies the beginning sequence number for each new member or job in the output stream.

This option has the following format:

SEQ1=*number*

### *number*

Specifies that sequencing restarts with the specified *number* at a logical break in the original JCL. The default is 10.

**Usage notes:**

- To prevent the generation of sequence numbers, code SEQ1=0. Any other value for SEQ1 results in the generation of sequence numbers in Columns 73-80.
- Specifying SEQ1=0 also causes the clearing (blanks) of existing sequence numbers in columns 73-80. It is similar to the ISPF UNNUM option.
- If the ISPF NUM ON option is in effect for the members JCLNeat is processing, ISPF overrides JCLNeat and generates sequence numbers regardless of the setting of the SEQ1 option.

## SIM

The SIM option simulates all JCLNeat processing, without writing the results to the output file. All reports are produced as if the JCL were updated. This allows you to preview the JCL that is affected, before actually making the change.

This option has the following format:

SIM=Y|N

**Y**

Turns on simulation.

**N**

Turns off simulation. JCLNeat produces an output file, with the formatted JCL, changed JCL, or both. This is the default.

**Notes:**

- Under ISPF, this option does not format the JCL in the EDIT work area. Since there are not any reports under ISPF, there are not any reports to simulate.
- SIM is a batch runtime only option. You cannot specify SIM=Y as a system installation default.

## JCLNeat Options Syntax Summary

The following table summarizes the JCLNeat Options:

---

BCI=Y|N

---

BCS=Y|N

---

CDEL=0|1|2|3|4|9

---

CEC=*column-number*

---

---

CECCN=*column-number*

---

CECDD=*column-number*

---

CECEX=*column-number*

---

CECIF=*column-number*

---

CECIN=*column-number*

---

CECJ=*column-number*

---

CECJL=*column-number*

---

CECOP=*column-number*

---

CECPR=*column-number*

---

CECS=*column-number*

---

CSC=*column-number*

---

CSCCN=*column-number*

---

CSCDD=*column-number*

---

CSCEX=*column-number*

---

CSCIF=*column-number*

---

CSCIN=*column-number*

---

CSCJ=*column-number*

---

CSCJL=*column-number*

---

CSCOP=*column-number*

---

CSCPR=*column-number*

---

CSCS=*column-number*

---

DCBS=Y|N

---

FCC=*character*

---

FORMAT=Y|N|B

---

ICSC=*column-number*

---

ICSCCN=*column-number*

---

ICSCDD=*column-number*

---

ICSCEX=*column-number*

---

ICSCIF=*column-number*

---

ICSCIN=*column-number*

---

ICSCJ=*column-number*

---

---

ICSCJL=*column-number*

---

ICSCOP=*column-number*

---

ICSCPR=*column-number*

---

ICSCS=*column-number*

---

INCR=*number*

---

INLIB=LIB|PAN|PDS|SEQ

---

JOBF=Y|N

---

LINES=*number*

---

MEMBERS=\*|INCLUDE|EXCLUDE

---

MKW=*number*

---

MQKW=*number*

---

OPSC=*column-number*

---

OPSCCN=*column-number*

---

OPSCDD=*column-number*

---

OPSCEX=*column-number*

---

OPSCIF=*column-number*

---

OPSCIN=*column-number*

---

OPSCJ=*column-number*

---

OPSCJL=*column-number*

---

OPSCOP=*column-number*

---

OPSCPR=*column-number*

---

OPSCS=*column-number*

---

OPTDESC=Y|N

---

OUTLIB=PDS|SEQ

---

PACK=*max-number*

---

PACKCN=*max-number*

---

PACKDD=*max-number*

---

PACKEX=*max-number*

---

PACKIF=*max-number*

---

PACKIN=*max-number*

---

PACKJ=*max-number*

---

---

PACKJL=*max-number*

---

PACKOP=*max-number*

---

PACKPR=*max-number*

---

PACKS=*max-number*

---

RECOVER=Y|N

---

REXXMEM=*member-name*

---

REPORT=0|1|2|3|4|5|6|7

---

RSYM=Y|N

---

SECC=*character*

---

SEQ1=*number*

---

SIM=Y|N

---



# Chapter 7: JCLNeat Reports

---

JCLNeat produces up to four reports for each execution. The following are the reports:

- JCLNeat Options Report  
A Listing of options and features used in this execution
- JCLNeat Original Report  
The Original JCL
- JCLNeat Reformatted Report  
The Reformatted JCL
- JCLNeat Changes Report  
Changes From and To JCL statements generated as a result of your global editing requests

If you are using the GSRCTL DD in your execution JCL to perform global string replacement, you will receive a different version of the reports listed here. These reports are present in this chapter.

If you are using the JCLNeat REXMEM option, you will receive a different version of the reports. These reports are also present in this chapter.

# Options Report

This report describes the options and features that were used for JCL reformatting during this job run.

```
CA                               JCLNeat OPTIONS
CA JCLCHECK                      SYS: xxxx

PARM FIELD OPTIONS SPECIFIED BY USER:

INLIB=PDS
OUTLIB=SEQ
MEMBERS=INCLUDE
PARMS IN EFFECT THIS RUN:

BCI=Y
BCS=N
CDEL=0
CEC=35
CECJ=35
CSC=16
DCBS=N
FCC=-
FORMAT=Y
JOBF=Y
ICSC=37
INCR=10
INLIB=PDS
LINES=55
MEMBERS=INCLUDE
MKW=60
MQKW=10
OPSC=12
OPTDESC=Y
OUTLIB=SEQ
PACK=N
RECOVER=Y
REPORT=7
REXXMEM=
RSYM=N
SECC=*
SEQ1=10
SIM=N
```

This report contains the following fields:

**PARM FIELD OPTIONS SPECIFIED BY USER:**

The options specified by the user in the JCLNeat PARM field are listed in the order in which they were coded.

**PARMS IN EFFECT THIS RUN:**

Alphabetical list of the values used for all of the JCLNeat options during this run.

## Specifying GSRCNTL DD Control Cards

If you are using global string replacement (GSRCNTL DD), the Options report output looks like the following:

```

CA                               JCLNEAT OPTIONS                               PAGE 001
  CA JCLCHECK                     SYS: xxxx

  PARM FIELD OPTIONS SPECIFIED BY USER:

    INLIB=PDS
    OUTLIB=SEQ
    MEMBERS=INCLUDE
    PACK=Y
    PARS IN EFFECT THIS RUN:

    BCI=Y
    BCS=N
    CDEL=0
    CEC=35
    CSC=16
    DCBS=N
    FCC=-
    FORMAT=Y,
    JOBF=Y
    ICSC=37
    INCR=10
    INLIB=PDS
    LINES=55
    MEMBERS=INCLUDE
    MKW=60
    MQKW=10
    OPSC=12
    OPTDESC=Y
    OUTLIB=SEQ
    PACK=Y
    REPORT=7
    RSYM=N
    SECC=*
    SEQ1=10

  GSRCNTL CARDS INPUT BY USER:

|UNIT=3350|REPL|                 * REPLACE STRING WITH STRING ON NEXT LINE
|UNIT=(SYSDA,2)|                 * REPLACEMENT STRING
|CLASS=A|REPL|                   * REPLACE STRING WITH STRING ON NEXT LINE
|CLASS=X|                         * REPLACEMENT STRING
|CLASS=A|INSA|                   * FIND CLASS=A AND INSERT A LINE AFTER IT
|// ROLL=(NO,NO)|                * NEW JCL CARD FOR INSERTION
|CLASS=A|INSB|                   * FIND CLASS=B AND INSERT A LINE BEFORE IT
|// PRTY=15|                     * NEW JCL CARD FOR INSERTION
|DD2|DELT|                       * FIND DD1 AND DELETE THE LINE
|DD3|COMM|                       * FIND DD3 AND COMMENT OUT THE LINE

```

This report contains the following fields:

**PARM FIELD OPTIONS SPECIFIED BY USER:**

The options specified by the user in the JCLNeat PARM field are listed in the order in which they were coded.

**PARMS IN EFFECT THIS RUN:**

Alphabetical list of the values used for all of the JCLNeat options during this run.

**GSRCNTL CARDS INPUT BY USER**

Produces a list of all control statements used by global string replacement for this execution.

## Original JCL Report

This report is a sequential listing of the statements read from the input stream specified to JCLNeat. Each statement line is prefixed with an identifier of IN. A new page begins when a logical break is encountered in the input stream.

This report also contains the JCLNeat error messages documented in the *Message Reference Guide*. Many of the error messages are then followed by the statement line that was being processed when the error was encountered. This statement is prefixed with an identifier of IMAGE. A new page begins after an error message and its associated IMAGE record is printed.

```
CA                               JCLNeat ORIGINAL JCL
CA JCLCHECK                      SYS: xxxx

IN //BROWN01F JOB '40700000',BROWN,MSGCLASS=X,CLASS=K,NOTIFY=BROWN01    00010034
IN //IEF PROC                                                              00020034
IN //STEP1 EXEC PGM=IEFBR14                                                00021034
IN //DD1 DD DSN=BROWN01.TEST.TAPE,DISP=(NEW,DELETE),UNIT=3380,          00022034
IN //  SPACE=(TRK,(1,1)),                                                  00022134
IN //  DCB=(RECFM=VBS,LRECL=80,BLKSIZE=3200)                             00023034
IN //  PEND                                                                00024034
IN //STEP1 EXEC IEF                                                        00024134
IN //                                                                      00024634
IN //IEF PROC                                                              00024734
IN //STEP1 EXEC PGM=IEFBR14                                                00024834
IN //DD1 DD DSN=BROWN01.TEST.TAPE,DISP=(NEW,DELETE),UNIT=3380,          00024935
IN //  SPACE=(TRK,(1,1)),                                                  00025035
IN //  DCB=(RECFM=VBS,LRECL=80,BLKSIZE=3200)                             00025135
IN //  PEND                                                                00025434
IN //STEP1 EXEC IEF                                                        00025534
```

## Specifying GSRCNTL DD Control Cards

If you are using global string replacement (GSRCNTL DD), the Original JCL report output looks like the following:

CA	JCLNeat ORIGINAL JCL
CA JCLCHECK           SYS: xxxx	
IN //BELST023 JOB (40100000),'CAUSER1',	0001000
INSB // PRTY=15,	
IN //    CLASS=A,	0002000
REP //    CLASS=X,	0002000
IN //    MSGCLASS=X	0003000
INSA //    ROLL=(NO,NO)	
IN //STEP01 EXEC PGM=IEFBR14	0004000
IN //DD0 DD UNIT=3350,VOL=SER=SER001,SPACE=(TRK,0),DCB=(BUFNO=7)	0005000
REP //DD0 DD UNIT=(SYSDA,2),VOL=SER=SER001,SPACE=(TRK,0),	0005000
REP // DCB=(BUFNO=7)	
IN //DD1 DD UNIT=3350,VOL=SER=SER001,SPACE=(TRK,0)	0006000
REP //DD1 DD UNIT=(SYSDA,2),VOL=SER=SER001,SPACE=(TRK,0)	0006000
IN //DD2 DD UNIT=3380,VOL=SER=SER002,SPACE=(TRK,0)	0007000
DEL //DD2 DD UNIT=3380,VOL=SER=SER002,SPACE=(TRK,0)	0007000
IN //DD3 DD UNIT=3390,VOL=SER=SER003,SPACE=(TRK,0)	0008000
COM // *DD3 DD UNIT=3390,VOL=SER=SER003,SPACE=(TRK,0)	0008000

### Output

When you use the GSRCNTL parameter, the report identifies each of the changes by prefixing each line of output with one of following:

#### COM

Commented JCL statement

#### DEL

Deleted JCL statement

#### IN

Input JCL line (FROM string)

#### INSA

Inserted JCL statement after

**INSB**

Inserted JCL statement before

**REP**

Line string replacement

## Reformatted JCL Report

This report is a sequential listing of the JCL statements generated by JCLNeat in reformatting the JCL. Each statement line is prefixed with an identifier of OUT. A new page begins when a logical break is encountered in the input stream.

CA	REFORMATTED JCL
CA JCLCHECK	SYS: xxxx
OUT //BROWN01F	JOB '40700000', 00000010
OUT //	BROWN, 00000020
OUT //	MSGCLASS=X, 00000030
OUT //	CLASS=K, 00000040
OUT //	NOTIFY=BROWN01 00000050
OUT //*	00000060
OUT //IEF	PROC 00000070
OUT //STEP1	EXEC PGM=IEFBR14 00000080
OUT //*	00000090
OUT //DD1 DD	DCB=(BLKSIZE=3200,LRECL=80,RECFM=VBS), 00000100
OUT //	DISP=(NEW,DELETE), 00000110
OUT //	DSN=BROWN01.TEST.DISK, 00000120
OUT //	SPACE=(TRK,(1,1)), 00000130
OUT //	UNIT=3380 00000140
OUT //*	00000150
OUT //	PEND 00000160
OUT //STEP1	EXEC IEF 00000170
OUT //*	00000180
OUT //	
OUT //IEF	PROC 00000190
OUT //STEP1	EXEC PGM=IEFBR14 00000200
OUT //*	00000210
OUT //DD1 DD	DCB=(BLKSIZE=3200,LRECL=80,RECFM=VBS), 00000220
OUT //	DISP=(NEW,DELETE), 00000230
OUT //	DSN=BROWN01.TEST.DISK, 00000240
OUT //	SPACE=(TRK,(1,1)), 00000250
OUT //	UNIT=3380 00000260
OUT //*	00000270
OUT //	PEND 00000280
OUT //STEP1	EXEC IEF 00000290
OUT //*	00000300

## Changes Report

This report is a before and after listing of the JCL resulting from the global editing requests specified in the CHANGES DD data set. Each *before* original JCL statement is prefixed with FRM, and each *after* reformatted JCL statement is prefixed with an identifier of TO.

All the FRM JCL statement lines are printed first, followed by all the TO JCL statement lines. A new page begins when a logical break is encountered in the input stream.

CA	JCLCHECK	SYS: xxxx	JCLNeat CHANGES	
FRM //DD1	DD	DSN=BROWN01.TEST.TAPE,DISP=(NEW,DELETE),UNIT=3380,		00022034
FRM //		SPACE=(TRK,(1,1)),		00022134
FRM //		DCB=(RECFM=VBS,LRECL=80,BLKSIZE=3200)		00023034
TO //DD1	DD	DCB=(BLKSIZE=3200,LRECL=80,RECFM=VBS),		00000100
TO //		DISP=(NEW,DELETE),		00000110
TO //		DSN=BROWN01.TEST.DISK,		00000120
TO //		SPACE=(TRK,(1,1)),		00000130
TO //		UNIT=3380		00000140
FRM //DD1	DD	DSN=BROWN01.TEST.TAPE,DISP=(NEW,DELETE),UNIT=3380,		00024935
FRM //		SPACE=(TRK,(1,1)),		00025035
FRM //		DCB=(RECFM=VBS,LRECL=80,BLKSIZE=3200)		00025135
TO //DD1	DD	DCB=(BLKSIZE=3200,LRECL=80,RECFM=VBS),		00000220
TO //		DISP=(NEW,DELETE),		00000230
TO //		DSN=BROWN01.TEST.DISK,		00000240
TO //		SPACE=(TRK,(1,1)),		00000250
TO //		UNIT=3380		00000260

## JCLNeat Options Report Using REXX

This report describes the options and features that were used for JCL reformatting during this job run.

```
CA                                     JCLNEAT OPTIONS
CA JCLCHECK                          SYS: xxxx

PARM FIELD OPTIONS SPECIFIED BY USER:

INLIB=PDS
OUTLIB=SEQ
MEMBERS=INCLUDE
REXXMEM=CAZ1NREX
PARMS IN EFFECT THIS RUN:

BCI=Y
BCS=N
CDEL=0
CEC=35
CECJ=35
CSC=16
DCBS=N
FCC=-
FORMAT=Y
JOBF=Y
ICSC=37
INCR=10
INLIB=PDS
LINES=55
MEMBERS=INCLUDE
MKW=60
MQKW=10
OPSC=12
OUTLIB=SEQ
PACK=N
RECOVER=Y
REPORT=7
REXXMEM=CAZ1NREX
RSYM=N
SECC=*
SEQ1=10
SIM=N
MEMBERS TO BE INCLUDED BY USER:

NEAT@W96
```

## JCLNeat Original JCL Report Using REXX

This report is a sequential listing of the statements read from the input stream specified to JCLNeat using the REXXMEM option. See the JCLNeat Original JCL report in this chapter for more information about the Original JCL Report.

CA	JCLNEAT ORIGINAL JCL	
CA JCLCHECK	SYS: xxxx	
IN //BELST029 JOB (40100000), 'S. BELL', CLASS=K, MSGCLASS=X,		00010000
IN // NOTIFY=BELST02		00020000
REPL //BELST029 JOB (40100000), 'S. BELL', CLASS=K, MSGCLASS=A,		00010000
IN //STEP0 EXEC PGM=OLDPGM		00030000
DELT //STEP0 EXEC PGM=OLDPGM		00030000
IN //STEP1 EXEC PGM=IEFBR14		00031000
INSA //STEPLIB DD DISP=SHR, DSN=CAI.J6B0.CAILOAD		
IN //DD1 DD DISP=(,CATLG), DSN=BELST02.SSC95.DATA,		00040000
IN // UNIT=BOGUS, SPACE=(CYL,(1,1),RLSE),		00050000
IN // DCB=(RECFM=FBA, LRECL=80, BLKSIZE=80)		00060000
REMV // UNIT=BOGUS, SPACE=(CYL,(1,1)),		00050000
REMV // SPACE=(CYL,(1,1)),		00050000
REPL // DCB=(RECFM=FBA, LRECL=80, BLKSIZE=0)		00060000
IN //DD2 DD DISP=SHR, DSN=OBSOLETE.DATASET		00070000
COMM // *DD2 DD DISP=SHR, DSN=OBSOLETE.DATASET		

This is a sample of the JCLNeat Reformatted JCL report.

CA	REFORMATED JCL	
CA JCLCHECK	SYS: xxxx	
OUT //BELST029 JOB (40100000),		00000010
OUT // 'S. BELL',		00000020
OUT // CLASS=K,		00000030
OUT // MSGCLASS=A,		00000040
OUT // NOTIFY=BELST02		00000050
OUT //STEP1 EXEC PGM=IEFBR14		00000060
OUT //STEPLIB DD DISP=SHR,		00000070
OUT // DSN=CAI.J6B0.CAILOAD		00000080
OUT //DD1 DD DCB=(BLKSIZE=0, LRECL=80, RECFM=FBA),		00000090
OUT // DISP=(,CATLG),		00000100
OUT // DSN=BELST02.SSC95.DATA,		00000110
OUT // SPACE=(CYL,(1,1))		00000120
OUT //**-----*		00000130
OUT //**		* 00000140
OUT // *DD2 DD DISP=SHR, DSN=OBSOLETE.DATASET		00000150



# Glossary

---

## **Audit report**

The *Audit report* is a CA JCLCheck report, which produces a statistical summary of jobs, steps, and DD statements, input records of jobs, steps, and DD statements, and averages of all of the jobs, steps, and DD statements included in a particular run.

## **ChekPlex**

The *ChekPlex option* is an optional feature that enables remote validation of JCL. You can issue a JCL validation request from a local system, and the validation takes place on another system.

## **CLIST**

A *CLIST* is a command list that allocates the required data sets and calls the TSO communication program.

## **CSV**

CA JCLCheck can produce reports in Comma Separated Values format. This allows for the converted report file to be read and displayed on a personal workstation by Excel or a similar spreadsheet product.

## **CTLSCAN**

The *CTLSCAN option* reads and interprets control statements of certain IBM utility programs and the CA XCOM Data Transport program.

## **Data Set Cross-Reference report**

The *Data Set Cross-Reference report* is CA JCLCheck Report 3, which lists all data sets using a job that exists for more than one step. Included in the report are all permanent and temporary data sets with a disposition other than NEW,DELETE.

## **dictionary support**

CA APCDOC is a CA product that maintains a dictionary of information concerning jobs, job networks, data sets, procedures, and programs.

## **EDCHEK option**

The *EDCHEK option* is an optional feature that allows CA JCLCheck to run as an edit macro under ISPF.

## **EDCHEL option**

The *EDCHEL option* enables the expansion of procedure and include statements when the EJCK macro is invoked from EDCHEK.

---

**Error Messages report**

The *Error Messages report* is CA JCLCheck Report 6, which lists all error messages associated with the job. CA JCLCheck produces this report if it detects one or more errors. You can direct this report to the beginning or end of the report output or incorporate it with the Listing of Merged JCL Report (Report 2), based on the setting of the ERROR option.

**Flow Diagram report**

The *Flow Diagram report* is CA JCLCheck Report 10. This report is a graphic representation of the flow of the JCL that CA JCLCheck processed. The first page lists the name of the JCL member being charted and any JOBLIB, JOBCAT, or PROCLIB data sets associated with the job.

**ISPF**

*ISPF*, Interactive Systems Productivity Facility, is an IBM utility that acts as an extension of the operating system under TSO.

**IXR**

*IXR*, Intelligent Transparent Restore, allows you to access information regarding archived data sets.

**JCL card**

A *JCL card* is an 80-character card image that is part of a complete JCL statement.

**JCL statement**

A *JCL statement* is a logical entity consisting of the JCL operator and its positional/keyword parameters and associated data. A JCL statement can be (and usually is) a composite of a variable number of JCL cards.

**Job Control Standards**

*Job Control Standards* are standards that comply with a specific site and are enforced by CA JCLCheck. They consist of a Standards Rules database and a series of panels that allow the administrator to set up criteria to produce specific actions based on JCL variables (that is, JOB.JOBNAME, EXEC.STEPNAME).

**Listing of Jobstream JCL report**

The *Listing of Jobstream JCL report* is CA JCLCheck Report 1, which is an 80-80 listing of the JCL as submitted. It also includes control statement statements if the CCLIST option was requested. Generation of this report can be controlled with the JOB option.

**Listing of Merged JCL report**

The *Listing of Merged JCL report* is CA JCLCheck Report 2, which is a listing of the entire JCL for the job, including procedure library statements.

**logical break**

A *logical break* is a condition in the original JCL input stream that signals JCLNeat to process a new entity. A logical break is typically signaled by an input member change or a new JOB statement, but is also signaled by an isolated PROC.

---

**online**

The *online* method immediately processes information at the time of data entry. Usually, the results are displayed on the screen.

**operator**

The *operator* is the type of statement in process (that is, JOB, EXEC, PROC, DD, PEND, PRINTDEV, CNTL, ENDCNTL, OUTPUT, or comments).

**original JCL**

The *original JCL* is the JCL unmodified statements prior to JCLNeat reformatting.

**PDF**

CA JCLCheck can produce reports in Portable Document Format. This allows for the converted report file to be read and displayed on a personal workstation.

**predecessor**

A *predecessor*, in CA JCLCheck, consists of jobs related to or preceding the job being analyzed. CA JCLCheck obtains information pertaining to the predecessor jobs from the CA APCDOC Cross-Reference database.

**PROCLIB**

The *PROCLIB* is the system or user Procedure Library, which CA JCLCheck searches for cataloged procedures.

**Program Cross-Reference report**

The *Program Cross-Reference report* is CA JCLCheck Report 4, which lists all programs that the job uses and the libraries in which the jobs reside.

**reformatted JCL**

The *reformatted JCL* is the JCL statements written to the JCLNeat output stream.

**Report Listing**

The *Report Listing* is CA JCLCheck Report 5, which lists all SYSOUT reports the job produces. A SYSOUT report is assumed to result from any DD statement specifying the SYSOUT parameter, including those requesting an internal reader.

**REXX**

*REXX*, R(estructured) EX(tended) (e)X(ecutor), is an IBM programming language that includes environment commands and mathematical functions.

**status code**

A *status code* is a one-character code that identifies the status of a job or run. (Example: A = abended)

**SUBCHEK option**

The *SUBCHEK option* is an optional feature of CA JCLCheck that allows JCLCheck to run as a front-end processor for the TSO/SUBMIT command.

---

**Summary Data Set Cross-Reference report**

The *Summary Data Set Cross-Reference report* is CA JCLCheck Report 7, which lists all permanent data sets used by any job within a production system. CA JCLCheck only produces this report if you select both the SXREF and XREF options. The report is similar to the Data Set Cross-Reference Report (Report #3), except that all data sets the related jobs use are included. The report is sequenced by data set names.

**Summary Program Cross-Reference report**

The *Summary Program Cross-Reference report* is CA JCLCheck Report 8, which lists all programs used by a job within a production system. This report is similar to the Program Cross-Reference Report (Report #4), except that all programs included in the system are reported in program name sequence.

**Summary Report listing**

The *Summary Report listing* is CA JCLCheck Report 9, which lists all SYSOUT reports produced by any job within a production system. It is produced only if both the SXREF and RPTRPT options are selected.