


Burroughs 

**B 5000/B 6000 Series
and B 7000 Series DMS**

SYSTEM NOTES

PART 2

RELATIVE TO MARK 3.2.1 RELEASE

Copyright © 1981, Burroughs Corporation, Detroit, Michigan 48232

PRICED ITEM

“The names used in this publication are not of individuals living or otherwise. Any similarity or likeness of the names used in this publication with the names or any individual, living or otherwise, is purely coincidental and not intentional.”

Burroughs believes that the information described in this document is accurate and reliable, and much care has been taken in its preparation. However, no responsibility, financial or otherwise, is accepted for any consequences arising out of use of this information.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

Correspondence regarding this publication should be forwarded using the Remarks form at the back of the manual, or may be addressed directly to TIO West Documentation, Burroughs Corporation, P.O. Box 4040, El Monte, California 91734, U.S.A.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

TABLE OF CONTENTS

	PAGE
GENERAL	1
D3204 - B6700 System Software Support	1
D3205 - Instructions for Printing Documentation	1
D3265 - Industry Compatible Mini Disk	1
D3396 - LISTNOTES Changes	1
D4239 - B5900 Overview	3
D4243 - Modifications for "Non-BCL" Systems	8
D4248 - Intrinsic to Library Conversion	8
D4267 - Mark 32 PRI NOTESFOR Tape	11
D4323 - Corrections to Mark 32 System Notes	11
ALGOL	13
D4108 - BCL Constants Allowed	13
D4109 - "<fault stack history> ARRAY"	13
D4110 - Truthset ALPHA6 Allowed	13
D4111 - BCL Constants	13
D4112 - Dope Vectors can be Resized	13
D4162 - Call By Name Parameters	14
D4166 - Allow Underscores in Identifiers	14
D4187 - Batch Restrictions	15
D4260 - ALGOL BATCH and Arrays	15
D4312 - Double Vs. Single "By-Name" Parameter Passing	15
D4322 - "Free-Field" READ of Hex String	16
D4325 - Repeat Count, Size Field "<4030" in Formats	16
ALGOL	17
P1017 - Erroneous Attribute Error Message	17
P1124 - "IF <boolean exp> THEN ELSE" Statement	17
P1217 - Incorrect Code for "Pointer-Valued" Attributes	17
P1277 - LINEINFO	17
P1287 - "IPC-Capable", RUN, CALL Statements	17
P1300 - Incorrect Branch Using SWITCH LABEL	17
P3306 - SEG ARRAY Error for Large Data Bases	17
P3813 - "REAL (<pointer exp>, <arithmetic exp>)"	17
ATTABLEGEN	18
D4131 - Spelling Corrected	18
D4138 - New Values for SUBFILEERROR	18
BACKUP	19
D4208 - Documentation Change	19
D4250 - ND REPORT Changes	19
BACKUP	20
P1014 - ND REPORT Corrected	20
P1201 - DLBACKUP Correction	20
P1292 - Restore Printing Speed	20
P1371 - Double Printing	20
P3743 - Drop Spaces in File Title	20
P3759 - Requires LP After Punching File	20
P3796 - Report EQUAL Correction	20
BARS	21
P1015 - SPO Mode	21
P1258 - SPO Mode UNRECOGNIZED REQUEST	21
P1293 - Display of Overlay Traffic Percentage	21
BASE INTRINSICS	22
D4255 - BASEINTRINSICS Implementation	22
BASIC	23
D4155 - Flag Length Specification for EBCDIC File	23
D4168 - Negative Argument to RND	23
D4254 - Recompile "Pre-Mark 29" Codefiles	23
BASIC	24
P1162 - Compiler Capacity for String Literals	24
BASICSUPPORT	25
D3375 - New File SYSTEM/BASICSUPPORT	25
BASICSUPPORT	26
P1127 - Allow Six Characters in File Title	26
P1163 - RESTORE Statement	26
P1193 - Flush Random Binary File Buffer	26
P1299 - Matrix Inversion	26
P3307 - Array of Strings Declaration	26
BINDER	27
D4296 - Documentation Corrections	27
BINDER	28
P1289 - Bound Code Release Levels	28
P3186 - BINDER Loses Sharing Class	28
P3187 - Deimplement Installation Intrinsic Warning	28
P3213 - Syntax Errors	28
P3232 - Installation One Intrinsic	28
BURROUGHS NETWORK ARCHITECTURE	29
D3194 - BNA Implementation	29
D4134 - Implement Logging for BNA	88
CANDE	92
D4135 - Blanks in UTILITY Parameter	92
D4174 - Destination Filename Crunched	92
D4229 - HN ODT Message	92
CANDE	93

P1190	- INVALID MODE Error for BCL Files	PAGE	93
P1284	- Stack Size for PROGRAMDUMP	PAGE	93
P1285	- Changing Tankfile Problem.	PAGE	93
P1372	- Usage of RENEW	PAGE	93
P3760	- DELETE ALL Syntax.	PAGE	93
COBOL		PAGE	94
D3655	- Level 01 Data Item	PAGE	94
D4142	- AT END Allowed for Port File Read.	PAGE	94
D4218	- STOP RUN in COBOL Library.	PAGE	94
D4310	- ORGANIZATION IS SEQUENTIAL	PAGE	94
COBOL		PAGE	95
P1117	- Library Template Splits Across Area Boundary	PAGE	95
P1136	- OPEN, CLOSE Statements for Port Files.	PAGE	95
P1164	- Global ISAM Files.	PAGE	95
P1325	- Truncation Warning for Report Items.	PAGE	95
P1327	- COMPUTE Statement in WHEN Clause	PAGE	95
P1328	- BACKUPKIND Attribute	PAGE	95
P1329	- Erroneous Right Parenthesis.	PAGE	95
P1373	- Commas in OPEN Statements.	PAGE	95
P3190	- Intrinsic BINDINFO Mismatch.	PAGE	95
P3217	- MAXRECSIZE of Report Files	PAGE	95
P3265	- BLOCKSIZE Always Set	PAGE	95
P3308	- PORT Files, Perform Terminus Code.	PAGE	95
P3313	- Timedout Condition	PAGE	95
P3761	- SEG ARRAY with DATADICTINFO Set.	PAGE	95
COBOL74		PAGE	96
D4143	- AT END ALLOWED FOR PORT FILE READ.	PAGE	96
D4151	- Documentation Correction	PAGE	96
D4173	- Segment Number	PAGE	96
D4193	- Accept from Timer, 10 Digits	PAGE	96
D4202	- USAGE IS ASCII Deleted	PAGE	96
D4203	- STACK SIZE Clause Added.	PAGE	96
D4253	- I/O Status from Invalid Subfile Index.	PAGE	96
COBOL74		PAGE	97
P1018	- OPEN REVERSE	PAGE	97
P1117	- Library Template Splits Across Area Boundary	PAGE	97
P1118	- Status Key for Indexed I/O	PAGE	97
P1136	- OPEN, CLOSE Statements for Port Files.	PAGE	97
P1325	- Truncation Warning for Report Items.	PAGE	97
P1327	- COMPUTE Statement in WHEN Clause	PAGE	97
P1329	- Erroneous Right Parenthesis.	PAGE	97
P3217	- MAXRECSIZE of Report Files	PAGE	97
P3264	- OPEN EXTEND with Tape Files.	PAGE	97
P3265	- BLOCKSIZE Always Set	PAGE	97
P3761	- SEG ARRAY with DATADICTINFO Set.	PAGE	97
CONFIGURATOR.		PAGE	98
P1333	- Unknown Group ID	PAGE	98
CONTROLLER.		PAGE	99
D4136	- HOSTNSP Handling	PAGE	99
D4150	- ODT CANDIDATE Message.	PAGE	99
D4199	- Interval Logging for Files	PAGE	99
D4222	- HN ODT Message.	PAGE	99
D4245	- LT ODT Message Deimplemented.	PAGE	99
CONTROLLER.		PAGE	100
P1052	- HARDCOPY Correction.	PAGE	100
P1053	- Resource Corrections	PAGE	100
P1130	- ADM EVENT PER Correction	PAGE	100
P1165	- Correct Misspelling.	PAGE	100
P1203	- WAIT Correction.	PAGE	100
P1259	- Halt/Load Reason Check	PAGE	100
P3236	- TERM FULLPAGE Corrected.	PAGE	100
P3762	- CU Display Correction.	PAGE	100
CONTROLWARE		PAGE	101
D3191	- Controlware Vs. Firmware, Pack Controls.	PAGE	101
D4251	- Mark 32 PR1 Disk Pack Controlware Files.	PAGE	102
DATA COMMUNICATIONS		PAGE	108
D4226	- Change Name of HOSTNSP Stack	PAGE	108
D4242	- NSP Datacom Initialization Messages.	PAGE	108
DATA COMMUNICATIONS		PAGE	110
P1184	- HOST NSP Shutdown.	PAGE	110
P1185	- Datacom Error Message Correction	PAGE	110
P1186	- Error Handling in FIREOFFDCP	PAGE	110
P1187	- Unsuccessful I/O Message	PAGE	110
P1273	- B5900 Compatibility.	PAGE	110
P1334	- Retry Counts on Output Results	PAGE	110
P1335	- LINESWAP, REINITIALIZATION Corrections.	PAGE	110
P1338	- Change NSP Autorecovery.	PAGE	110
P1340	- FORGETCHECK out of STARTSYSTEM	PAGE	110
P1341	- FIREOFFDCP Problem	PAGE	110
P1342	- Datacom Table Problems	PAGE	110
P1343	- INVALID OP in DCINITIAL.	PAGE	110
P1344	- Messages for DC Initialization Failure	PAGE	110
P1345	- Loop Timeout	PAGE	111
P1358	- FIREOFFDCP Problem	PAGE	111
P1359	- Recall Timing Window	PAGE	111
P1360	- Dialout Problem.	PAGE	111
P1361	- BLOCKEXIT Mom Problem.	PAGE	111

P3205	- Change NSP Initialization Failed Message	PAGE 111
P3206	- Rejected Request	PAGE 111
P3207	- DCRECON Timing Problem	PAGE 111
P3208	- Call BLASTUNIT	PAGE 111
P3242	- Queuing between DCINITIAL/DCCONTROL	PAGE 111
P3243	- Initialization Aborted Message	PAGE 111
P3244	- Override Local Optimization in DCCONTROL	PAGE 111
P3245	- Mark Line Dead for LINEDELETED Result	PAGE 111
P3246	- Unit Left Assigned for NSP	PAGE 111
P3247	- STATION NOT READY Handling	PAGE 112
P3249	- Improve Tests	PAGE 112
P3260	- DCINITIATEMCS Stack Overflow	PAGE 112
P3261	- Miscellaneous HOSTNSP Corrections	PAGE 112
P3262	- Add Station Without Line Assignment	PAGE 112
DCALGOL		PAGE 113
D3371	- Priority Output	PAGE 113
D4234	- Documentation Changes	PAGE 113
DCALGOL		PAGE 114
P1123	- Erroneous Syntax Error on DCALGOL Message	PAGE 114
DCP PROGRAM GENERATOR		PAGE 115
P1000	- Three Address Characters	PAGE 115
P1001	- Call Designation when Station Changes	PAGE 115
P1068	- ASL Clears MAO	PAGE 115
P3218	- Priority Output Handling	PAGE 115
P3219	- Set Dial Data Bit on Each Dialout	PAGE 115
P3763	- DCP Fault 8	PAGE 115
DIAGNOSTIC MCS		PAGE 116
D4114	- Automatic DCWRITE ERROR Reporting	PAGE 116
DIAGNOSTIC MCS		PAGE 117
P1301	- SS ALL Station not Attached	PAGE 117
DMS II - GENERAL		PAGE 118
P3343	- Misleading "IOERRORS"	PAGE 118
DMS II - ACCESSROUTINES		PAGE 119
P1004	- Incorrect Key/Data Mismatch, Wrong Record	PAGE 119
P1039	- LIMITERROR Problems	PAGE 119
P1040	- Ordered Data Set Records Out of Order	PAGE 119
P1094	- Visible DBS, INQUIRY Return Software Version	PAGE 119
P1101	- Population Too High	PAGE 119
P1115	- Linearsearch Not Finding All Records	PAGE 119
P1156	- Spurious NOTFOUND After Abort	PAGE 119
P1218	- FIND PRIOR AT KEY	PAGE 119
P1219	- Usage Statistics Inconsistency	PAGE 119
P1226	- Abort Failure	PAGE 120
P1227	- Opening of Data Base Results in SEG ARRAY	PAGE 120
P1241	- Ordered Data Set Corruption	PAGE 120
P1245	- Direct Key Corrupted	PAGE 120
P1374	- DS of Linear Search	PAGE 120
P1375	- Limit Errors	PAGE 120
P1376	- NO FILE on RECOVERYINFO	PAGE 120
P3194	- Bad Open	PAGE 120
P3250	- Abort Overlooks Duplicated Audit	PAGE 120
P3327	- Excessive Memory	PAGE 120
P3328	- FORGETSPACE Timing Window	PAGE 120
DMS II - BDMSALGOL		PAGE 121
P1161	- Interlanguage Binding	PAGE 121
P1231	- Segmented Transaction Records, Record Arrays	PAGE 121
P1252	- Signed Numbers, Hex Pointers	PAGE 121
DMS II - BDMSCOBOL		PAGE 122
P1326	- Selection Expression Truncation Warning	PAGE 122
P3305	- SIB Size Increase for Large Data Bases	PAGE 122
DMS II - BDMSCOBOL74		PAGE 123
P1326	- Selection Expression Truncation Warning	PAGE 123
P3305	- SIB Size Increase for Large Data Bases	PAGE 123
DMS II - DASDL		PAGE 124
D4147	- System Allocated Areas	PAGE 124
D4213	- Chain of In Use Blocks	PAGE 124
DMS II - DASDL		PAGE 125
P1157	- NOFILE on WFL Compile Job	PAGE 125
P3258	- Deleting, Adding Global Data	PAGE 125
DMS II - DUMPDIR/LIBRARY		PAGE 126
P1308	- SEG ARRAY Error on "WRITE=", "LIST="	PAGE 126
DMS II - INQUIRY		PAGE 127
D4156	- Report Pages Split	PAGE 127
D4191	- Report Title	PAGE 127
D4195	- HEADING Format	PAGE 127
D4214	- SHOW Syntax	PAGE 127
DMS II - INQUIRY		PAGE 129
P1069	- INVALID OP	PAGE 129
P1077	- Boolean Key Data	PAGE 129
P1080	- DISPLAY Formats Output in HEADING Form	PAGE 129
P1092	- SEG ARRAY	PAGE 129
P1100	- Selection Expression Improvement	PAGE 129
P1129	- Record Function Information Not Purged	PAGE 129
P1181	- Save Prompt for Temporary Sets	PAGE 129
P1220	- Modified Bit for AA List Sorts	PAGE 129
P1221	- Subscript Value Limit	PAGE 129
P1224	- Invalid Addresses in Sort Tags	PAGE 129

P1225	- Report Title Too Large	PAGE 129
P1309	- Minus Sign on "REAL(SM,N)" Items	PAGE 130
P1310	- Use, Enforce Correct Item Field Sizes.	PAGE 130
P1321	- "SHOW ALL OF <data set name>"	PAGE 130
P3195	- Trailing Blanks on Multistatement Defines.	PAGE 130
DMS II	- INTERFACE.	PAGE 131
P1377	- INVALID INDEX.	PAGE 131
DMS II	- LOADDUMP	PAGE 132
D4190	- DATASET Change	PAGE 132
DMS II	- LOADDUMP	PAGE 133
P1222	- Incorrect Code Generated	PAGE 133
P1223	- Incorrect Source Code.	PAGE 133
P1378	- Incorrect FD Generated	PAGE 133
DMS II	- PROPERTIES	PAGE 134
P1379	- Number of Names in DMERRORS Array.	PAGE 134
DMS II	- RECOVERY	PAGE 135
P1041	- Infinite Loop in RECOVERY.	PAGE 135
P1091	- Write Error.	PAGE 135
P3257	- Rebuild Across Reorganization.	PAGE 135
P3329	- DATARECOVERY Fails to Reconstruct Row.	PAGE 135
P3821	- Rebuild/Reconstruct Abort.	PAGE 135
DMS II	- REORGANIZATION	PAGE 136
P1128	- Checksum on Reorganized Global Data.	PAGE 136
P1155	- Incorrect BCW on Embedded ordered Data Sets.	PAGE 136
P1291	- Corruption of Data	PAGE 136
P1380	- Checksum Error on Interim Tape Files	PAGE 136
P3197	- Generating Multiple Bitvectors	PAGE 136
DMS II	- UTILITY.	PAGE 137
D4215	- "<integer>" is Unique Number	PAGE 137
D4231	- Guarded Data Bases	PAGE 137
DMS II	- UTILITY.	PAGE 138
P1042	- Available Block Numbers Printed Wrong.	PAGE 138
P1070	- REBUILD Fails on File Discontinuity.	PAGE 138
P1071	- Wrong Numbers of Workers Started	PAGE 138
P1081	- Syntax Error when Restarting UTILITY	PAGE 138
P1381	- Data Base Invocation	PAGE 138
P3326	- UTILITY Recover May Corrupt Data	PAGE 138
DMS II	- WFL/COMPILEACR	PAGE 139
D4212	- INITPARTITIONS Option Removed.	PAGE 139
DMS II	- TRANSACTION PROCESSING	PAGE 140
D4161	- Application Program Restart.	PAGE 140
D4268	- Rules for Transaction Processing	PAGE 140
D4295	- Documentation Corrections.	PAGE 142
DMS II	- TFL.	PAGE 143
D4179	- INCLUDE Option	PAGE 143
DMS II	- TFL.	PAGE 144
P1116	- Duplicated Option.	PAGE 144
P1312	- Detection of Syntax Errors	PAGE 144
DMS II	- TRINTERFACE.	PAGE 145
P1313	- Bad Timestamps	PAGE 145
DMS II	- TRUTILITY.	PAGE 146
P1090	- INVALID INDEX.	PAGE 146
P1209	- Search Function Fault.	PAGE 146
DMS II	- HOSTLIB.	PAGE 147
D4144	- Increased Maximum Number of Active TPS	PAGE 147
DMS II	- HOSTLIB.	PAGE 148
P1108	- Incorrect Interval Time.	PAGE 148
P1228	- Transaction Records Exceed 2059 Bytes.	PAGE 148
P1229	- Syntax Error	PAGE 148
P1230	- Deadlock	PAGE 148
P3200	- Recovery of Journal Data File's End of File.	PAGE 148
P3201	- Cannot Save Response Transaction	PAGE 148
P3202	- Response Transaction not Saved	PAGE 148
P3339	- Error Result not Returned.	PAGE 148
DUMPALL		PAGE 149
D4209	- Hex and Decimal Record Numbers	PAGE 149
DUMP ANALYZER		PAGE 150
D4103	- Timestamp at BOT is Recorded in EOT Log.	PAGE 150
DUMP ANALYZER		PAGE 151
P1166	- Path Name with Multiple Paths.	PAGE 151
P1213	- Display MLIP Unit Status	PAGE 151
P3191	- INVALID INDEX Interrupt.	PAGE 151
P3193	- Fault on Misspelled HELP Query	PAGE 151
ESPOL		PAGE 152
D4269	- Cannot Run on "Non-BCL" System	PAGE 152
D4270	- Deimplementation Warning	PAGE 152
FILECOPY.		PAGE 153
P1253	- Default Usercode	PAGE 153
P1302	- Listing of Serial Numbers.	PAGE 153
FILEDATA.		PAGE 154
D4235	- FILEDATA Attributes.	PAGE 154
FORTAN		PAGE 155
D4204	- Files 5,6 Default to REMOTE.	PAGE 155
D4233	- MODEL1 Compiler Option	PAGE 155
D4321	- "Free-Format" Input of Hex String.	PAGE 155
D4324	- Repeat Counts, Size Field "<4030" in Formats	PAGE 155
FORTAN		PAGE 156

P1119 - READER, PRINTER Options Fail	PAGE 156
P1194 - Variable in ASSIGNMENT Statement	PAGE 156
GENERALSUPPORT	PAGE 157
D4311 - Repeat Count, Size Field "<4030" in Formats	PAGE 157
GENERALSUPPORT	PAGE 158
P1099 - Freefield Attempting to Write to Input File	PAGE 158
P1137 - Remove BCL	PAGE 158
P1303 - Formatted Read of Strings	PAGE 158
P1331 - K Modifier with Run Time Formatting	PAGE 158
P1362 - Freefield WRITE of Arrays	PAGE 158
P3214 - G FORMAT When VALUE=0	PAGE 158
P3215 - Read into String Error	PAGE 158
P3277 - F Format Output Suppresses Leading Zeroes	PAGE 158
P3278 - Repeat Counts of Zero in Formats	PAGE 158
P3309 - F Format Output Causes Incorrect Results	PAGE 158
P3312 - Misspelled Format	PAGE 158
P3314 - Formatted Output with "A*" Format	PAGE 158
P3315 - "Free-field" Input of Real, Boolean	PAGE 158
P3325 - BCL A Format Input	PAGE 159
HARDCOPY	PAGE 160
P3764 - HARDCOPY Not Reacting to HI	PAGE 160
P3765 - HARDCOPY Files Left on Disk	PAGE 160
HOSTNSPSUPPORT	PAGE 161
D4141 - HOSTNSP Implementation	PAGE 161
D4158 - Internal Efficiency Changes	PAGE 163
D4160 - "Multiple-Wait S-OP" Improvement	PAGE 163
D4223 - I/O Initiation Address Saved	PAGE 164
D4224 - Editor Process Initiation Acceleration	PAGE 164
D4225 - Save MLIP Hardware Error Flag	PAGE 164
HOSTNSPSUPPORT	PAGE 165
P1140 - HOSTNSP MLIP Error Field Retention	PAGE 165
P1141 - HOSTNSP Abort "S-OP" Correction	PAGE 165
P1142 - LISTLOOKUP	PAGE 165
P1143 - Null String Handling	PAGE 165
P1144 - Ready Queue Process Links Corrected	PAGE 165
P1145 - Set Datacount at I/O Complete	PAGE 165
P1146 - ADD STATION/ADD LINE Correction	PAGE 165
P1147 - Remove IOCB Correction	PAGE 165
P1148 - ZAP Process Correction	PAGE 165
P1149 - Remove Group to ZAP Line Processes	PAGE 165
P1150 - "Test-Op" Line Count Correction	PAGE 165
P1352 - "STATION.OUTPUTCOUNT"	PAGE 165
P1353 - "ABNORMAL-TERMINATE" RESULT Message	PAGE 165
P1354 - "IOCB.DATACOUNT"	PAGE 165
P1355 - "LINE-NOT-READY" RESULT Message	PAGE 165
P1356 - INDEXNAME "S-Op"	PAGE 165
P1357 - "LIBERATE-TO-CORRECTLY-QUEUE-WAITING-PROCESS"	PAGE 165
P4265 - Line Process Abnormal Termination	PAGE 165
INPUT-OUTPUT	PAGE 166
D4105 - MAXSUBFILES Attribute	PAGE 166
D4138 - New Values for SUBFILEERROR	PAGE 166
D4172 - AVAILABLE File Attribute	PAGE 166
D4186 - UNITNO Attribute	PAGE 166
D4228 - Optional Vs. Direct	PAGE 166
D4244 - BCL Files	PAGE 166
D4320 - "End-of-Tape" Reporting on Direct Tape Files	PAGE 166
INPUT-OUTPUT	PAGE 167
P1013 - Fix AREALENGTH File Attribute Reporting	PAGE 167
P1065 - HOSTNAME of Unassigned File	PAGE 167
P1170 - PROTECTED Vs. REWIND	PAGE 167
P1171 - Reelswitch Vs. Use	PAGE 167
P1197 - SPO Vs. UNITNO	PAGE 167
P3321 - Datacom Direct I/O	PAGE 167
P3815 - Remote Backup Files	PAGE 167
INTERACTIVEXREF	PAGE 168
D4166 - Allow Underscores in Identifiers	PAGE 168
INTERACTIVEXREF	PAGE 169
P1370 - Wrong Sequence Numbers for Aliases	PAGE 169
JOB FORMATTER	PAGE 170
D4272 - Interval Logging for Files	PAGE 170
D4273 - BNA Log Formats	PAGE 173
JOB FORMATTER	PAGE 180
P1294 - Print Unit Number of Pack	PAGE 180
LOADER	PAGE 181
P1297 - Diskcycle	PAGE 181
P1365 - Setting of Options	PAGE 181
LOG ANALYZER	PAGE 182
D4274 - BNA LOGANALYZER Changes	PAGE 182
LOG ANALYZER	PAGE 183
P1240 - Implementation of Read Extended Status	PAGE 183
LOGGER	PAGE 184
D3636 - SEG ARRAY when Totaling, Averaging Same Item	PAGE 184
LOGGER	PAGE 185
P1005 - Update YTD File with Multiple Break Items	PAGE 185
P3766 - Break on JOBQUEUEDTIME Causes SEG ARRAY	PAGE 185
P3767 - LOGGER Mixes ORGUNIT	PAGE 185
LTTABLEGEN	PAGE 186

P1010	- Complete TRRAINTABLES	PAGE 186
MAKEUSER		PAGE 187
P1102	- ACCESSCODELIST	PAGE 187
P1232	- SET LIST Problems	PAGE 187
MCP		PAGE 188
D4103	- Timestamp at BOT is Recorded in EOT Log	PAGE 188
D4104	- New Words in CLOSE Log Record	PAGE 188
D4132	- No Room for PROGRAMDUMP	PAGE 188
D4149	- Initialization Stop	PAGE 188
D4167	- Standard Tape Labels Vs. New Systems	PAGE 188
D4170	- Two New Library Parameter Types	PAGE 188
D4198	- B5900 BCL Message	PAGE 188
D4211	- SYSTEMSTATUS Type 3 Modification	PAGE 188
D4220	- "Pre-Mark 31" Codefiles on B5900	PAGE 189
D4221	- TRAINID Field in UNITTABLE	PAGE 189
D4241	- "TIME(23)" Indicates BCL Support	PAGE 189
D4264	- Catalog Block Version Implemented	PAGE 189
MCP		PAGE 190
P1008	- Stack Overflow in KANGAROO	PAGE 190
P1011	- READALABEL Codestring	PAGE 190
P1016	- Library Parameter INVALID OP	PAGE 190
P1030	- OPEN vs BACKUP	PAGE 190
P1033	- Timing Hole in DOTASKINITIATION	PAGE 190
P1034	- Restore Saved TEMPLATE MOM	PAGE 190
P1036	- DP Vs. STATUSCHANGE	PAGE 190
P1037	- DUP FAMILY NAME	PAGE 190
P1038	- Miscellaneous Corrections	PAGE 190
P1054	- SUBSPACES Attribute Erroneously Inherited	PAGE 190
P1055	- Restart ABORT Messages	PAGE 190
P1061	- Aborted Restart	PAGE 190
P1066	- READALABEL Vs. Foreign Tapes	PAGE 191
P1067	- PAST Continuation	PAGE 191
P1084	- Correct Cancelling of Nonexistent Libraries	PAGE 191
P1085	- DCHOLD Save-Core Buildup	PAGE 191
P1095	- FLATREADER Readlock Loop	PAGE 191
P1112	- Critical Block Exit and Libraries	PAGE 191
P1120	- ADM EVENT Logged	PAGE 191
P1121	- Short Disk I/O	PAGE 191
P1126	- Restart Vs. Missing Disk, Pack Family	PAGE 191
P1132	- SUSPENDER/WSSHERRIFF Suspension Improvements	PAGE 191
P1133	- Volume Library Vs. Rewind	PAGE 191
P1134	- Checkpoint/Restart	PAGE 192
P1135	- Restart Abort, PROCESSKILL, PIB Handling	PAGE 192
P1151	- SYSTEMSTATUS Type 4 Processor Buzz Loop	PAGE 192
P1172	- Remove BCL Pointers	PAGE 192
P1174	- Programdump Analyzes Interrupts Correctly	PAGE 192
P1177	- PID Releaseheader Call	PAGE 192
P1178	- Change Support Libraries	PAGE 192
P1189	- STARTUNIT Readies Units Cleanly	PAGE 192
P1205	- SHARED BY ALL Libraries Forced to Global	PAGE 192
P1207	- Backup MCP Deleted	PAGE 192
P1208	- TRAINID Vs. Subset	PAGE 192
P1246	- HDRVECTORLOCK	PAGE 192
P1247	- NEWP Vs. RLTABLEGEN	PAGE 193
P1248	- ATIGRAB Fault	PAGE 193
P1260	- LIBUSEMCPLOCK Vs. PROCESSCHANGELOCK	PAGE 193
P1262	- ALIEN I/O	PAGE 193
P1264	- GETSTATUS INVALID OP	PAGE 193
P1265	- ODT Marked as NOTREADY	PAGE 193
P1267	- COBOL BCL User Labels Vs. B5900	PAGE 193
P1268	- User Define BCL Tape Labels Vs. B5900	PAGE 193
P1286	- Bad Graph Edge	PAGE 193
P1288	- COBOL Print Files not Released	PAGE 193
P1290	- Box Number for DBS Stacks	PAGE 193
P1296	- Bad Restart after ENDF	PAGE 193
P1317	- User Vs. MCP Fault	PAGE 193
P1318	- GETSTATUS LINKF Correction	PAGE 193
P1319	- Library Parameter Type	PAGE 193
P1324	- System Hang	PAGE 194
P1363	- Direct Resize Search	PAGE 194
P1368	- SWAPPER Deadlock	PAGE 194
P1383	- STACK OVERFLOW in GETAROW	PAGE 194
P3199	- Move Intrinsic FAULTRECOVERY	PAGE 194
P3233	- SWAPPER and Data Management	PAGE 194
P3234	- Dump by BAD MOM SEARCH	PAGE 194
P3235	- Terminating Scheduled Libraries	PAGE 194
P3241	- Inheriting SUBSYSTEMID	PAGE 194
P3253	- FIBSTACK Binary Card I/O	PAGE 194
P3275	- Stack Searching Error with SWAPPER	PAGE 194
P3319	- "Core-to-Core" Reimplemented	PAGE 194
P3320	- Illegal Swapstate While in READYQ	PAGE 194
P3334	- Setting the SWAPPER Parameter MAXIOSIZE	PAGE 194
P3335	- ACTIVETIME	PAGE 195
P3340	- Unit Error Statistics Always Zero	PAGE 195
P3345	- Releaseheader	PAGE 195
P3448	- Checkpoint/Restart for Programs Using Strings	PAGE 195
P3687	- BDNAME SEG ARRAY Fault	PAGE 195

P3689	- Incorrect Compilation Date	PAGE	195
P3710	- DMSCLOSE Vs. CONTROLLER	PAGE	195
P3786	- STANDARDODISPLAY Decoding	PAGE	195
P3818	- Allow "DCSYSTEMTABLES(6)" Without Datacom.	PAGE	195
MESSAGE LEVEL INTERFACE PORT.		PAGE	196
P1060	- LOADHOST Digit Count	PAGE	196
P1111	- COBOL, ALGOL Tape SORT	PAGE	196
P1153	- Checkpoint Vs. B6900 I/O	PAGE	196
P1159	- Log DLP Errors	PAGE	196
P1160	- Exchangeable Units	PAGE	196
P1173	- RY a Saved Pack Response	PAGE	196
P1175	- Firmware Level Recorded on Dump Tape	PAGE	196
P1212	- Correct ODT UR	PAGE	196
P1214	- System Initialization	PAGE	196
P1215	- Card Punching Problem	PAGE	196
P1240	- Implementation of Read Extended Status	PAGE	196
P1263	- Readying of Packs	PAGE	197
P1269	- CANCELIO Corrections	PAGE	197
P1270	- BLASTUNIT Improvements	PAGE	197
P1272	- PATHRES Vs. Nonstatus DLP	PAGE	197
P1276	- Correct Reserve Halt/Load Path	PAGE	197
P1295	- Pack Power Off	PAGE	197
P1364	- DOIERRORIO Returns Logical R/D	PAGE	197
P1366	- TAKEUNIT Vs. "FREE PK<nn>"	PAGE	197
P1369	- WFL/CONTROLCARD Loop	PAGE	197
P3203	- Logical Unit Number Vs. "PB MT<nn>"	PAGE	197
P3209	- UR/UA Maintenance Capabilities	PAGE	197
P3237	- TEST/WAIT Bad Event Reference	PAGE	197
P3238	- BLASTUNIT, MYIOONLY	PAGE	197
P3239	- Selective Clear Vs. Hung DLP	PAGE	198
P3240	- "UA SC <unit no>" Corrected	PAGE	198
P3248	- Correct HOSTNSP Results	PAGE	198
P3254	- BINARY EOF Sensing	PAGE	198
P3341	- IOERRORTYPE Vs. "EOT/EOF"	PAGE	198
P3342	- BLASTUNIT Avoids Outboard Paths	PAGE	198
NETWORK DEFINITION LANGUAGE		PAGE	199
D4100	- NDLC Compiler User Options	PAGE	199
D4194	- Free Pound Sign	PAGE	199
NETWORK DEFINITION LANGUAGE		PAGE	200
P1003	- Error for Different Line Controls	PAGE	200
P1254	- Recursive Defines	PAGE	200
P1255	- Allow String Parameters to Defines	PAGE	200
P1307	- Setting of User Options	PAGE	200
P1332	- Alpha Labels	PAGE	200
NDLII		PAGE	201
P1019	- Detect String Size Too Large Error	PAGE	201
P1020	- Fix Fault with Use of Special String Primary	PAGE	201
P1158	- Synchronous Line Diagnostic Rate	PAGE	201
P1235	- Line Attribute Offset	PAGE	201
P1278	- Events, Interlocks in Include Lists	PAGE	201
P1279	- Fault in Group INCLUDE Declaration	PAGE	201
P1280	- "INVALID DESCRIPTOR ADDRESS (UTILACTUAL)"	PAGE	201
P1281	- Invalid Extension Skeletons	PAGE	201
P1282	- WAITFORIDLE Implemented	PAGE	201
P1347	- Algorithm Translate Table List	PAGE	201
P3222	- Duplicate Adaptor Assignment	PAGE	201
P3223	- SYSTEMWAIT Code	PAGE	201
P3256	- Control Block Arrays	PAGE	201
P3285	- IDSTORE Size Increased	PAGE	202
P3310	- COPYSTRING with "<cb variable>.TEXT"	PAGE	202
P3311	- Transmit Statement	PAGE	202
NEWP		PAGE	203
D4115	- Syntax Deleted	PAGE	203
D4166	- Allow Underscores in Identifiers	PAGE	203
NEWP		PAGE	204
P1022	- Invalid Index for CASE Expression	PAGE	204
P1023	- XREF of Procedures in MODULE Export List	PAGE	204
P1106	- String Parameters to STOP	PAGE	204
P1167	- Make Scalerrights Return an Integer	PAGE	204
NSP DUMP ANALYZER		PAGE	205
D4175	- Recognize "Mark 7 NSP" Firmware	PAGE	205
D4176	- Add Fault Tolerance	PAGE	205
NSP DUMP ANALYZER		PAGE	206
P1204	- Eliminate Firmware Code	PAGE	206
P3251	- Different Level Firmware Handling	PAGE	206
PATCH		PAGE	207
P1139	- No BCL on EBCDIC System	PAGE	207
PATCHCONTROLWARE		PAGE	208
D3128	- Initial Release of PATCHCONTROLWARE	PAGE	208
PLI		PAGE	209
P3224	- Incorrect Syntax Error	PAGE	209
PLISUPPORT		PAGE	210
P1103	- ISAM, "END-OF-FILE"	PAGE	210
P1109	- ISAM DELETE, Large FINE, COARSE Tables	PAGE	210
P1233	- GET LIST, GET DATA	PAGE	210
P1256	- Bad Parameters to DELTA	PAGE	210
P1304	- INVALID INDEX in Condition Info Table	PAGE	210

P1305 - PUT EDIT, Bit Formats	PAGE	210
P1330 - ISAM Hardware Errors	PAGE	210
P3225 - ISAM, DELETE	PAGE	210
P3226 - ISAM, COBOL START Statement	PAGE	210
P3259 - ISAM, Premature End of File.	PAGE	210
STANDALONE PRINTER DUMP (SAD)	PAGE	211
D4246 - PRINTERDUMP Implementation	PAGE	211
PTDMCP	PAGE	212
P1336 - PTD SPO Handling	PAGE	212
P1339 - PTD Path Deassignment, I/O Debug Options	PAGE	212
PERIPHERAL TEST DRIVER (PTD)	PAGE	213
D4197 - Improved PTD Path Selection	PAGE	213
D4240 - PTDTESTS Tape	PAGE	226
REMOTE JOB ENTRY	PAGE	228
D4117 - 02 and 04 Control Message Update	PAGE	228
D4118 - CLEAR <LSN> vs Virtual Stations	PAGE	228
D4163 - Remove BCL Constructs	PAGE	228
D4164 - File Transfer Floods Memory	PAGE	228
D4185 - Backup and Print Queue Rebuilding	PAGE	228
REMOTE JOB ENTRY	PAGE	229
P1024 - Session Initialization	PAGE	229
P1025 - Terminal Transfer Compatibility with SYCOM	PAGE	229
P1026 - RJE HOST to HOST Link Losing First Message	PAGE	229
P1027 - File Transfer Valid Char List Update	PAGE	229
P1110 - Usercode Word Not Zeroed Out	PAGE	229
P1168 - Contiguous Blanks	PAGE	229
P1169 - FTS Record Size, Block Size	PAGE	229
P1191 - Station ID Reconfiguration	PAGE	229
P1234 - "CP-9500 (B800)" Appears as Peer Host	PAGE	229
P1236 - OFFLINE Request Blanks Stationname Array	PAGE	229
P1237 - RUNTIMEOPTIONS Storage in Linkfile	PAGE	229
P1348 - EBCDIC to ASCII for B800 Systems	PAGE	229
P1349 - "FS1, FS2" Records Ignored	PAGE	230
P3227 - SEG ARRAY Error	PAGE	230
P3230 - Connect Reply for Inactive Station	PAGE	230
READER SORTER - RSMCP	PAGE	231
P3255 - RSPIO Parameter	PAGE	231
SCRMCP	PAGE	232
D3586 - ICMD Maintenance	PAGE	232
SCRMCP	PAGE	237
P1195 - SCR TESTII Recognizes Pack	PAGE	237
P1196 - Tags Match TEST7	PAGE	237
SORTMCP	PAGE	238
D4133 - Sort Changes Value of NEWFILE	PAGE	238
SOURCENDL	PAGE	239
D4164 - File Transfer Floods Memory	PAGE	239
SOURCENDL	PAGE	240
P1257 - RJE Host Link Reestablishment	PAGE	240
SOURCENDLII	PAGE	241
P1028 - Full Duplex Line Handling Corrected	PAGE	241
P1283 - Eliminate STRUCTURE PROTECT Faults	PAGE	241
P3228 - TTY/FULLDUPLXTTY Page, Break Handling	PAGE	241
P3229 - Upper/Lower Case Handling, Edit Buffer Size	PAGE	241
SUSPENDER	PAGE	242
P1006 - Restore SHOW Command	PAGE	242
P1007 - SUSPENDER on Monolithic Systems	PAGE	242
SYSTEMSTATUS	PAGE	243
D4291 - Documentation Change	PAGE	243
UDSTRUCTURE TABLE	PAGE	244
D4192 - IDENTITY	PAGE	244
D4232 - "CLASS=0"	PAGE	244
D4259 - SYSTEMUSER Redefined	PAGE	244
UTILITY LOADER	PAGE	245
P1261 - UTILoader VIA Syntax Vs. MCP	PAGE	245
P1266 - Save Space	PAGE	245
WORK FLOW LANGUAGE	PAGE	246
D4165 - BEND CARDS Example	PAGE	246
WORK FLOW LANGUAGE	PAGE	247
P1083 - By Reference TASK for COMPILE and GO	PAGE	247
P1087 - Eliminate BCL Constructs from WFL	PAGE	247
P1242 - USERDATA CLASS=4"80"	PAGE	247
P1243 - "\$INCLUDE * <filename>"	PAGE	247
P1244 - Multiple INCLUDE Statements	PAGE	247
P3709 - Names with Hyphens and Underscores	PAGE	247
DOCUMENTS	PAGE	248
D4152 - Mark 31 Compilation	PAGE	248
D4153 - "?RJ" ODT Message	PAGE	248
D4188 - Record Number Vs. Line Number	PAGE	248
SYSTEST - IO/IOTEST	PAGE	249
P1274 - Remove BCL Constructs	PAGE	249
SYSTEST - IO/IOINTERACT	PAGE	250
P1274 - Remove BCL Constructs	PAGE	250
SYSTEST - LANG/ALGOLSORT	PAGE	251
P3276 - Print Month Properly	PAGE	251
SYSTEST - SCR/SC01	PAGE	252
P3274 - Syntax Errors Corrected	PAGE	252
SYSTEST - SCR/ICMD	PAGE	253

DOCUMENT CHANGES NOTES (D NOTES)

GENERAL

D3204 GENERAL - "B6700" SYSTEM SOFTWARE SUPPORT

The system software Mark 3.3 release will be the last release of system software qualified for the B6700 series systems (including B6803, B6805, etc.).

With the release of system software Mark 3.4 (4Q 82), the B6700 Mark 3.3 system software will be supported via regional support centers with engineering backup as needed.

A formal announcement by the Marketing groups will be forthcoming.

D3205 GENERAL - INSTRUCTIONS FOR PRINTING DOCUMENTATION

This release includes documentation files, which may be printed by executing the OBJECT/LISTNOTES program and label-equating the file IN appropriately. Task value varies according to printer capability.

To print a file on an EBCDIC96 printer, use the following commands:

```
SITE use:
?RUN OBJECT/LISTNOTES; VALUE=0
?FILE IN=<use appropriate file title>
```

```
CANDE use:
E LISTNOTES ; FILE IN = <use appropriate file title> ; value 0
```

To print a file on an EBCDIC72 printer, use the following commands:

```
SITE use:
?RUN OBJECT/LISTNOTES; VALUE=1
?FILE IN=<use appropriate file title>
```

```
CANDE use:
E LISTNOTES ; FILE IN = <use appropriate file title> ; value 0
```

To create an EBCDIC96 printer backup tape, use the following commands:

```
SITE use:
?RUN OBJECT/LISTNOTES; VALUE=0
?FILE IN=<use appropriate file title>
?FILE LINE(KIND=PRINTER BACKUP TAPE),
SERIALNO="<use appropriate serialno>"
```

To create an EBCDIC72 printer backup tape, use the following commands:

```
SITE use:
?RUN OBJECT/LISTNOTES; VALUE=1
?FILE IN=<use appropriate file title>
?FILE LINE(KIND=PRINTER BACKUP TAPE),
SERIALNO="<use appropriate serialno>"
```

More complete documentation for LISTNOTES may be found in the GENERAL note D3396.

D3265 GENERAL - INDUSTRY COMPATIBLE MINI DISK

The B9489-16 Industry Compatible Mini Disk (ICMD) has been implemented. See ICMD/DOCUMENT on the Mark 32 PRI NOTESFOR tape for details of implementation.

D3396 GENERAL - "LISTNOTES" CHANGES

LISTNOTES is a program to process and print system notes (D- and P-notes) and other <file>/DOCUMENT files on the SYSTEMNOTES tape accompanying this release. The changes and procedures described in the note are operative for the Mark 32 PRI release (and NOTESFOR tape) as well.

LISTNOTES was released on previous system release notes tapes. The Mark 32 version has been substantially changed, particularly the interpretation of the task values and label equation.

Some of the changes include the following:

1. Default line printer file is EBCDIC96
2. Default reports are 'FINAL'; i.e., no sequence numbers
3. Option to print PCN pages only (for update releases)
4. Option to convert notes to 'NEWS' (type seqdata) files and removing headings, footings, etc.

Operating Instructions

Remote Use (via CANDE)

E LISTNOTES ; FILE IN (TITLE-X)
E LISTNOTES; VALUE CN ; FILE IN (TITLE-X)

Batch Use

RUN OBJECT/LISTNOTES ; FILE IN (TITLE-X)
RUN OBJECT/LISTNOTES ; VALUE CN ; FILE IN (TITLE-X)

The allowable values of 'N' in VALUE = CN are shown in the following table:

N	FINALTOG	UPPER % LOWERCASE	UPPERCASE ONLY	MAKETOG	PRINTER OUTPUT
0	T	T	F	F	T
1	T	F	T	F	T
2	F	T	F	F	T
3	F	F	T	F	T
9	T	F	F	T	F

C = 0 (DEFAULT) PROCESS ALL RECORDS IN THE FILE

C = 1 PROCESS ONLY PAGES WITH PCN '|' MARKS

N = 0 FINALTOG TRUE, PRINT IN UPPER AND LOWER CASE (DEFAULT VALUE)

N = 1 FINALTOG TRUE, PRINT ALL IN UPPER CASE

N = 2 FINALTOG FALSE, PRINT IN UPPER AND LOWER CASE

N = 3 FINALTOG FALSE, PRINT IN UPPER CASE

N = 9 FINALTOG TRUE AND CREATE 'DP/NEWS/X' FILES

FINALTOG TRUE (N = 0 OR N = 1)

The notes are printed in the center of the page without line numbers and sequence numbers

FINALTOG FALSE (N = 2 OR N = 3)

The notes are printed in the center of the page with line numbers on the left and two pairs of sequence numbers on the right: the first sequence number on the right is from the sequence field of the input file to the Mission Viejo text editor; the second sequence number is in steps of 100 and may be used as the sequence number to locate records in the input file to LISTNOTES, which is of type 'CDATA'.

UPPER & LOWER CASE (DEFAULT)

The records are printed on the file LINE with TRAINID=EBCDIC96. The records are not translated to uppercase.

UPPER CASE ONLY (N = 1 OR N = 3)

All records are translated to uppercase and printed on the file line with TRAINID = EBCDIC72.

PCNSTOG (C = 1, N = 0,1,2 OR 3)

Only pages which have records with the PCN (publication change notice) flag '|' are printed. This option may be used to print field update notes.

MAKETOG (N = 9)

This option processes the file 'IN = X' and creates the following files:

1. DP/NEWS/X
2. DP/NEWS/INDEX/X
3. DP/NEWS/CONTENTS/X

Each file is type 'SEQDATA'

The records in the file 'X' are converted to a file of type 'SEQ'. All headings, footings, and section headings are removed. Two or more consecutive blank lines are also removed.

The 'INDEX' file contains a record for each software item and contains the sequence numbers of the first and last record in the 'NEWS' file for the corresponding software item. It also contains one record for each D-note and P-note together with the sequence number of the first record in the 'NEWS' file for the corresponding note.

The 'NEWS' file may be used to extract portions of the system notes for publishing (in machine readable form) for system users. Extracting the CANDE D-notes could create a 'CANDE/NEWS' file, etc.

The 'CONTENTS' file contains the table of contents for the system notes.

LABEL EQUATION, FILE IN (TITLE=X)

The input file is specified with the label equation:

FILE IN (TITLE=<file title>)

Versions of LISTNOTES released prior to June 1980 required the label equation 'FILE DISKIN = X'. An attempt to use this label equation results in a warning to re-run with the correct label equation, as follows:

```
"THE LABEL EQUATION: 'FILE DISKIN (TITLE=X)'
WAS USED WITH THE OLD VERSION OF THIS PROGRAM
PLEASE USE: 'FILE IN (TITLE=X)'
PLEASE NOTE THE TASKVALUES HAVE ALSO BEEN CHANGED"
```

The options specified by the task value have also been changed from the previous releases.

D4239 GENERAL - "B5900" OVERVIEW

The B5900 data processing system is an entry-level system extending the B6000 and B7000 systems lines.

The B5900 is object-code compatible with the earlier B6000 and B7000 systems. There are two important exceptions to strict code compatibility:

The B5900 does not support the 6-bit data type (BCL).

The B5900 does not support vector-mode.

In addition, the B5900 zeroes bit 47 on all arithmetic operations. This may affect ALGOL programs which depend on preserving this bit through adds, etc.

A further restriction is that code files must have been compiled with a system release version Mark 31 (or later) compiler in order to run on the B5900.

The B5900 I/O subsystem is based upon the Message Level Interface Processor (MLIP) that was introduced with the B6900, and uses the same Data Link Processors (DLPs) that are used on the B6900. Operator and programmatic interfaces to the I/O subsystem on the B5900 are identical to those interfaces on the B6900. Differences in operator and program interfaces on the B5900/B6900 compared to the B6800 have been kept to a minimum.

A number of Mark 32 notes are referenced in this overview. These notes are contained in the file titled "OVERVIEW/DOCUMENT" on the Mark 32 PRI NOTESFOR tape. The following Mark 32 notes, also referenced in this overview, are contained on the Mark 32 PRI NOTESFOR tape: "NDLII/DOCUMENT", "ODT/DOCUMENT", and "DCAUDITOR/EXAMPLE". Two Mark 31 notes which contain useful information are also contained on the Mark 32 PRI NOTESFOR tape, as follows: "MULTIPROCESSOR/DOCUMENT" and "SHAREDRESOURCES/DOCUMENT".

The B5900 also introduces a new "soft" maintenance processor, based on the MTS2 terminal. All system initialization, confidence and maintenance routines and system state (registers, flip flops, etc.) are available on various menu-oriented screen formats. The MTS2 terminal is used as the maintenance processor when the system processor is halted and as an ODT when the system processor is running.

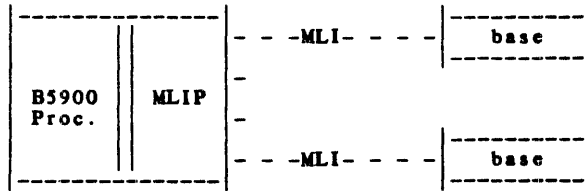
General Configuration Information

The following table lists the B5900 I/O modules and, for comparison, the approximate B6800 counterpart modules:

B5900 Hardware Modules	B6800 Counterparts
Message Level Interface (MLI)	peripheral control interface
MLI Processor (MLIP)	multiplexor
Data Link Processor (DLP)	peripheral control
base	peripheral control cabinet
Network Support Processor (NSP)	DCP
Line Support Processor (LSP)	Adapter Cluster

Other hardware components, such as peripherals, disk pack controllers (D-machines), and magnetic tape controllers (MECs), are the same on the B6800, B6900 and B5900.

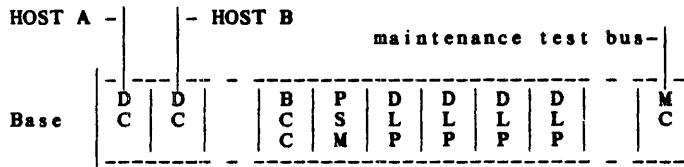
The B5900 processors communicate with the I/O subsystem through the Message Level Interface Processor (MLIP). There is one MLIP for each B5900 processor and up to 4 Message Level Interfaces (MLIs) per MLIP. Each MLI connects to a base, as shown in the following diagram:



The MLI is connected to the base through a Distribution Card. There may be up to 6 Distribution Cards in a base, allowing access from up to 6 "hosts". A host is any module that has an MLI Port, such as a B5900 processor or a Network Support Processor.

Each base includes a Base Control Card (BCC), which handles DLP access control between multiple hosts, configuration identity, and maintenance control functions.

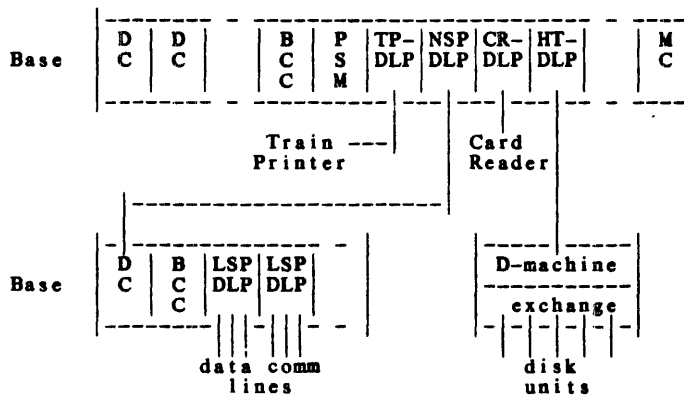
Bases that have more than one Distribution Card must have a Path Selection Module (PSM); the PSM performs priority resolution and return routing from each DLP to the appropriate host. Each base includes a Maintenance Card (MC) to connect to the maintenance test bus.



Each base contains up to 8 Data Link Processors (DLPs), which function as peripheral controls. The following DLP types are supported by the Mark 32 PRI software:

DLP	Peripheral
CR-DLP	B9115/6/7 Card Reader
TP-DLP	750/1100/1500 LPM Train Printer
MT-DLP	PE Tape
CP-DLP	Card Punch
HT-DLP	(Host Transfer) 225/235/206/207 Disk Pack
ODT-DLP	Operator Display Terminal
LSP-DLP	Data comm lines
NSP-DLP	Data comm network (LSPs)

A DLP can connect to a peripheral, to a peripheral exchange or controller, or, in the case of an NSP, to another base. An NSP functions as a host to its associated LSPs and connects to the base containing its LSPs through a Distribution Card.



NOTE: The diagrams above are schematic and do not represent actual configuration limits (for example, an LSP can support more than three data comm lines). Additional information about configuration is contained in Mark 32 note "D3406 CONFIGURATOR -- Soft Configuration" (of B5900 and B6900 systems).

Data Communications

The MLIP data communications subsystem consists of Network Support Processors (NSPs) and Line Support Processors (LSPs) in place of the B6800 DCPs and Adapter Clusters. NSPs and LSPs are Data Link Processors and, as such, interface to the main system and main system memory through the MLIP. An NSP acts as a host to any base that contains one or more of its associated LSPs (see diagram above).

Software changes relating to data comm, including the new NDL language and compiler (called NDLII), the DCAUDITOR program, the NSPDUMPANALYZER program, minor DCWRITE restrictions, SOURCENDLII, and the "ID" ODT message, are described or referenced in the Mark 32 note "D3202 DATACOM -- B6900 Data Communications".

Other Software Changes

The translation between unit numbers and path numbers on the B5900 and B6900 is different than on the B6800. Thus, configuration information returned when requested by a program or an operator includes path information in the new format. Changes relating to the GETSTATUS intrinsic are described in Mark 32 note "D3251 MCP GENERAL -- GETSTATUS/SETSTATUS Enhancements". Changes relating to the "OL" ODT message are described in the Mark 32 note "D3356 GENERAL -- New and Old ODT Messages".

Maintenance

The B5900 maintenance software utilizes the following hardware: the B5900 processor (and the Maintenance Processor (MP) of the B5900), the Maintenance Test Bus Facility of the MLIP bases, and the Maintenance Console. The Maintenance Console consists of the MTS2 Terminal System, the MIP (Maintenance Interface Processor) and a dual diskette drive. Different hardware paths between these units are used during different modes of operation. The console has the following modes:

1. SHELL mode - not executing any program and waiting for the user to enter commands to transfer control to another mode.
2. ODT mode - appears to be a normal ODT when the system is running with the additional capability of halting the system.
3. Maintenance mode - perform soft display or BEAM tests or Maintenance Test Bus tests when the system is halted.

Various programs, called executives, can be loaded into the MTS2. These executives are loaded from a diskette file into the MTS2 memory. When the system is running, the ODT executive causes the MTS2 to function as an ODT; otherwise, the MTS2 Terminal system controls the various maintenance functions. The executives which run when the system is halted are the following: SOFTCON (SOFT Console - performs functions of state accessing and display), BEAM (performs processor tests) and TESTBUS (performs MLIP subsystem testing). Each of these executive programs will be described in detail later. The SHELL controls the loading and execution of the various executives. Each executive resides within a diskette file. The ODT and SOFTCON executives are combined within one file called SOFTCON; the other executives' files have the same name as the executive.

The maintenance software, including all items on the SOFTCON menus, is fully described in the B5900 Test and Field Document, "B5900 Maintenance Software" (Part Number 23773310).

The new peripheral test package for the B5900 is managed by the Peripheral Test Driver (PTD). It is identical to that used on the B6900 and replaces the B6800 SCR package. PTD is described in Mark 32 PR1 note "D4197 PTDTST -- Peripheral Test Driver".

The MCP error logging procedures generate log entries in a format appropriate for the B5900. LOGANALYZER has been changed to recognize this format. See the Mark 32 note "D3355 MLIP -- Log MLIP I/O Errors".

System Operation

The B5900 operator interface to the MCP is identical to the operator interface on the B6900. Changes in the ODT commands are described in the Mark 32 note "D3356 GENERAL -- New and Old ODT Messages". The configuration file syntax is described in Mark 32 note "D3406 CONFIGURATOR -- Soft Configuration".

The major changes visible to the system operator relate to the use of the Soft Console (SOFTCON). B5900 System Initialization is fully described in "B5900 Startup Instructions" (Part Number 23773773).

System Power Up

When the MTS maintenance processor is first powered on, it loads the ODT/SOFTCON executive from the SYSTEM diskette and enters ODT mode (in case the system was running). The pair of keys CNTRL followed by H halt the system and are depressed to cause the SOFTCON executive to be entered.

When the system is first powered on, or the confidence and maintenance tests have been executed, the microcode for the processor should be loaded into the system processor via the following procedure:

1. Mount the SYSTEM diskette in one of the console diskette drives.
2. Depress the SPCFY key with the cursor on the POWER UP function on the SOFTCON operator menu.
3. Perform the Bootstrap Load function or the Cold Start function as described below.

System Initialization

There are three ways to initialize a B5900 MCP, the particular circumstances dictating which form must be used: halt/load (memory load), bootstrap load, and cool/cold start.

Halt/Load

On the B5900, a portion of memory is reserved for the "memory-resident bootstrap", a small program capable of loading the MCP from the Halt/Load unit. If the memory-resident bootstrap is valid and there is a valid MCP on the Halt/Load unit, the system can be initialized by depressing the SPCFY key with the ODT cursor on the LOAD function of the SOFTCON operator menu.

Bootstrap Load

If there is a valid MCP on any disk unit but the memory-resident bootstrap is not valid, the bootstrap should be loaded via UTILoader using the following procedure:

1. Mount the SYSTEM diskette in one of the console diskette drives.
2. Depress the SPCFY key with the ODT cursor on the BOOTLOAD function of the SOFTCON operator menu.
3. Remove the SYSTEM diskette from the console diskette drive to prevent media degradation.
4. Use the HALTLOAD function of UTILoader, as described in the Mark 32 note "D3332 UTILoader -- UTILoader on MLIP Systems", to load the memory-resident bootstrap. Specify the Halt/Load unit by entering "PK <nn>".

Cool/Cold Start

If there is no valid MCP on disk, a tape load of the MCP must be performed. UTILoader must be loaded and its TAPELOAD function used to load SYSTEM/LOADER. Then, SYSTEM/LOADER is run to copy the MCP file from tape to disk. (SYSTEM/LOADER is described in the SOG Manual, Volume II, Form No. 5001688, Article 5.) After the MCP is copied, a Bootstrap Load must be performed to initialize the memory-resident bootstrap. The complete sequence is described below:

1. Mount the SYSTEM diskette in one of the console diskette drives.
2. Depress the SPCFY key with the ODT cursor on the BOOTLOAD function of the SOFTCON operator menu.
3. Remove the SYSTEM diskette from the console diskette drive to prevent media degradation.
4. Use the TAPELOAD function of UTILoader, as described in the Mark 32 note "D3332 UTILoader -- UTILoader on MLIP Systems", to load SYSTEM/LOADER from tape.
5. Run SYSTEM/LOADER as described in the SOG Manual, Volume II (see also the Mark 32 note "D3157 LOADER -- LOADER Improvements"). After SYSTEM/LOADER has copied the MCP code file to the appropriate disk unit, the Bootstrap Load process must be performed, as described above.

Changing Halt/Load Units

To change Halt/Load units on a running system, the BOOTUNIT ODT message should be used. BOOTUNIT is described in the Mark 32 note "D3356 GENERAL -- New and Old ODT Messages". If the system is not running, the Bootstrap Load procedure can be used to establish the Halt/Load unit.

Stand-Alone Programs

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

In order to perform a Halt/Load, the memory-resident bootstrap code must be present in low-order memory. Loading a stand-alone program into low-order memory (via the TAPELOAD command of UTILoader) will overwrite the bootstrap, so that it is not possible to do a simple Halt/Load to return to the previous MCP. A Bootstrap Load sequence (described above) must be performed.

GLOSSARY OF TERMS

Base	A module that houses multiple DLPs.
BCC	Base Control Card, a base module that identifies a base and controls access to its DLPs and to the base itself.
BEAM	A resident program in the B5900 MTS2 which performs processor tests.
DC	Distribution Card, a base module that interfaces a host to a base.
DLP	Data Link Processor, a peripheral control module.
FR-DLP	(obsolete) Frame-Recognition DLP, now called LSP.
HDP	Host Dependent Port. A hardware module capable of interfacing a processor to the Message Level Interface. The HDP used to interface B6900 and B5900 processors to the MLI is specifically referred to as the Message level Interface Port. See MLIP.
Host	An I/O subsystem requestor, such as a processor (MLIP) or an NSP.
LCP	(obsolete) Line Control Processor, now called DLP.
LEM	Line Expansion Module, currently not available.
LSP	Line Support Processor DLP, a DLP that directly controls data comm lines.
MIP	Maintenance Interface Processor, the interface between the B5900 Maintenance Console and the Maintenance Processor of the B5900.
MLI	Message Level Interface, the interface between the host (MLIP) and the I/O Subsystem.
MLIP	Message Level Interface Processor, a module that controls one or more MLIs on behalf of the central processor.
MP	Maintenance Processor.
NDLII	Network Definition Language II describes datacom algorithms and network connections for the B5900.
NSP	Network Support Processor DLP, a DLP that controls one or more LSPs and provides the logical interface between the host and the data comm network.
PSM	Path Selection Module, a base module that routes messages from a DLP to one of multiple hosts.
PTD	Peripheral Test Driver, the B5900 MCP procedure that runs the peripheral test package.
SC-DLP	(obsolete) Subsystem-Controller DLP, now called NSP.
SHELL	The resident program in the B5900 MTS2, which allows loading various executives, such as SOFTCON, from diskette.
SOFTCON	The "soft console" program running in the MTS2 maintenance system.

Additional B5900 Tapes

In addition to the system tapes and maintenance tapes, the following tapes are provided for the B5900:

PTDTESTS
The PTDTESTS tapes are described in Mark 32 PR1 note "D4240 PTDTEST -- PTDTESTS Tape".

CONTROLWARE
The CONTROLWARE tapes are described in note "D4251 CONTROLWARE -- Mark 32 PR1 Disk Pack Controlware Files".

D4243 GENERAL - MODIFICATIONS FOR "NON-BCL" SYSTEMS

The B5900 hardware does not allow the use of BCL pointers. Therefore, certain changes were made throughout the software to accommodate the B5900 and future non-BCL machines. The most significant changes are detailed below:

1. If a program attempts to use a BCL pointer on a B5900, the message "B5900 IS NOT BCL CAPABLE" will be displayed followed by an invalid operator fault. (See MCP note D4198.)
2. If a program opens a file with BCL-valued attributes (e.g., INTMODE or EXTMODE) on a non-BCL system, the program will terminate with the message "BCL NOT ALLOWED ON THIS MACHINE". If a program attempts the open AVAILABLE option or the file attribute AVAILABLE on a file with BCL-valued attributes, 48 will be returned to the program and the file will not be opened. (See IN-OUTPUT note D4244).
3. TIME(23).[46:1] can be used to interrogate whether or not the system is BCL capable. The value 1 is returned for non-BCL systems and 0 for BCL-capable systems. (See MCP note D4241.)

Also refer to following system notes for changes to individual software items:

ALGOL	D4108
CANDE	P1190
ESPOL	P1138
GENERALSUPP	P1137
IOTEST	P1274
IOINTERACT	P1274
IN-OUTPUT	D4244
MCP	P1064
MCP	D4241
MCP	P1267
MCP	P1268
MCP	D4198
PATCH	P1139
RJE	D4163
WFL	P1087

D4248 GENERAL - INTRINSIC TO LIBRARY CONVERSION

Phasing Plan

The conversion of system intrinsics to system libraries began on the Mark 32 release and will continue for several releases. As of the Mark 32 PR1 release, all Burroughs-supplied intrinsics have been converted to "support" libraries written in NEWP. However, the timeframe for complete deimplementation of the old intrinsic mechanism has been lengthened considerably.

The original plan as described in Mark 32 GENERAL note D3354 called for deimplementation of the old intrinsic mechanism on the Mark 34 release. The exact release on which deimplementation will occur has not yet been decided, but the user installation intrinsic capability will be available at least through the Mark 35 release.

Before deimplementation occurs, the following capabilities will be provided.

1. Use of libraries in a COBOL program will be enhanced to provide a better substitute for installation intrinsics.
2. Programs will be permitted to access library entry points without explicit declarative syntax for each entry point.
3. The BINDER will preserve library templates.

The primary changes on the Mark 32 PR1 release are the addition of the BASIC support library and the replacement of the old intrinsic symbolics by a single skeletal intrinsic symbolic.

On the Mark 32 PR1 system software release, the intrinsics and support libraries must be initiated with the following commands:

1. SI+
2. SL GENERALSUPPORT=SYSTEM/GENERALSUPPORT
3. SL PLISUPPORT=SYSTEM/PLISUPPORT
4. SL BASICSUPPORT=SYSTEM/BASICSUPPORT

There are more details about operational aspects of intrinsics under Operational Steps.

Beginning with this release and continuing on each future release, a GENERAL note will describe the current state of the conversion process as well as the plans for the next release. No deimplementations will occur without warnings two releases in advance. The current intrinsic conversion plan is described below.

1. Mark 32 release

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

1. MCP support for the library attributes FUNCTIONNAME and LIBACCESS was added.
 2. The ODT message SL (System Library) was added to allow definition and management of system libraries, including mapping of function names to file titles. (See file "ODT/DOCUMENT" on the Mark 32 PR1 NOTESFOR tape).
 3. The ODT message PP (Privileged Program) was extended to allow the creation of privileged transparent programs. (See file "ODT/DOCUMENT" on the Mark 32 PR1 NOTESFOR tape).
 4. SYSTEM/USERSTRUCTURE now generates a patch which must be compiled into GENERALSUPPORT. (Users who use SYSTEM/USERSTRUCTURE should refer to Mark 32 USERSTRUCTURE note D3605 for details.)
 5. The MCP links user programs which reference system intrinsic procedures to either SYSTEM/INTRINSICS or the correct support library procedures.
2. Mark 32 PR1 release
1. All intrinsics have been converted to support libraries. An additional support library, SYSTEM/BASICSUPPORT, has been released. This support library contains the procedures needed to run BASIC programs. The old intrinsic symbolics (ALGOLINTRINSICS, ALGOLPLINTRINSICS, ESPOLINTRINSICS, and DCALGOLINTRINSICS) are no longer released.
 2. A skeletal intrinsic symbolic, SYMBOL/BASEINTRINSICS, has been released. It is compiled by ALGOL and then bound with the BINDER to create SYSTEM/INTRINSICS. This allows compilers access to the INTRINSICINFO array, which they require in order to recognize standard installation 1 intrinsics. The continued existence of SYSTEM/INTRINSICS also supports the binding of user installation intrinsics.
 3. The warnings about deimplementations which were promised in the Mark 32 GENERAL note D3354 are not yet provided, since the deimplementation schedule has changed.
 4. Since all the system intrinsics have been converted to support libraries on this release, the documentation for problem fixes and enhancements (P and D notes) appears under GENERALSUPPORT, BASICSUPPORT, and PLISUPPORT rather than the old intrinsic headings.
 5. The MCP links user programs which reference system intrinsic procedures to the correct support library procedures.
3. Mark 33 release
1. The ESPOL compiler will continue to be released. The planned deimplementation of ESPOL has been postponed to allow sufficient time for conversion of user installation intrinsics written in ESPOL. [Note that the ESPOL compiler will not run on the B5900 due to its dependence on BCL constructs.]
 2. SYMBOL/BASEINTRINSICS will continue to be available as the skeletal intrinsic symbolic. Binding of user installation intrinsics will continue to be supported via SYSTEM/INTRINSICS.
 3. An OFFSET function will be provided by the COBOL compilers. (This provides an alternative to the OFFSET intrinsic in the BEACON conversion package which was supplied as an installation intrinsic written in ESPOL.)
 4. If any deimplementations are necessary for the Mark 35 release, appropriate warnings will begin on the Mark 33 release.
 5. The MCP will continue to link user programs which reference system intrinsic procedures to the correct support library procedures.

Operational Steps

SYSTEM/INTRINSICS has been converted to three system support libraries: SYSTEM/GENERALSUPPORT, SYSTEM/PLISUPPORT and SYSTEM/BASICSUPPORT. Therefore, SYSTEM/INTRINSICS and these three support libraries have been released on the Mark 32 PR1 release.

SYSTEM/INTRINSICS are created by compiling SYMBOL/BASEINTRINSICS, then using the BINDER to build an intrinsic file from the resultant codefile. User installation intrinsics may also be bound to SYSTEM/INTRINSICS. This installation intrinsic capability will be available at least through the Mark 35 release.

The support libraries have reserved FUNCTIONNAMEs. These FUNCTIONNAMEs are as follows with a brief description of the library contents:

1. GENERALSUPPORT - This library contains all mathematical, formatting and miscellaneous intrinsic procedures formerly contained in SYSTEM/INTRINSICS.
2. PLISUPPORT - This library contains the support procedures for PL/I.
3. BASICSUPPORT - This library contains the support procedures for BASIC.

The SL ODT message is used to initiate support libraries, as follows:

SL <functionname> = <library code file title>

For example, GENERALSUPPORT is initiated with the following:

SL GENERALSUPPORT = SYSTEM/GENERALSUPPORT ON SYSPACE

A usercode and family name is permitted in the <library code file title>.

Only installations which use PL/I or the pre-Mark 32 ISAM need to initiate PLISUPPORT. Likewise, only installations which use BASIC need to initiate BASICSUPPORT.

In order to run the compilers, SYSTEM/INTRINSICS must have been initiated. To initiate SYSTEM/INTRINSICS, the SI (or CI) ODT message is still used.

If the appropriate support library is not running when a program needs to link to it, the program will be suspended with a "MISSING <functionname>" (e.g., "MISSING GENERALSUPPORT"). When this happens, the library may be Sled, and the program will continue.

NOTE: SLing to a support library differs from SLing to a user library in the following way: SL will immediately initiate a support library, but a user library which has been Sled is not initiated until the first time a program attempts to link to it. In either case, SL links a <functionname> to a <library code file title>. (See file "ODT/DOCUMENT" on the Mark 32 PR1 NOTESFOR tape.)

NOTE: SL differs from SI (CI) in that SL will immediately respond if the support library is not resident. (SI initiates an independent runner which waits on a "no file" condition if the intrinsic file is not resident.)

A running support library may be discontinued with SL- <functionname>. The support library will continue to run until all its users terminate. New users which require that <functionname> will be suspended until another linking SL is performed. Normally, a user should not DS a support library, since this will DS all users connected to it.

A support library may be changed at any time with SL. The former support library will continue until all of its current user programs terminate. All new user programs will link to the new support library. If the system halt/loads, all restarting user programs will link to the new support library.

Software Generation

1. The following example WFL deck shows how the support libraries may be recompiled on the Mark 32 PR1 release:

```
? BEGIN JOB COMPILE/SUPLIBS;
SUBROUTINE COMP(String LIB);
BEGIN
  TASK T;
  COMPILE SYSTEM/#LIB WITH NEWP [T] LIBRARY;
  COMPILER FILE TAPE(TITLE = SYMBOL/#LIB);
  COMPILER DATA
% CLEAR MERGE LINEINFO SEQERR NEWSEQERR
?
  IF T IS COMPILEDOK THEN
    WAIT("PP SYSTEM/" & LIB & ":TRANSPARENT",OK)
  ELSE
    ABORT "RECOMPILE " & LIB;
END; % COMP

COMP("GENERALSUPPORT");
COMP("PLISUPPORT");
COMP("BASICSUPPORT");

? END JOB
```

2. Before a system support library is Sled, it must be PPed through the ODT as follows:

PP <library title> : TRANSPARENT

This is extremely important and establishes the library as privileged transparent. (See file "ODT/DOCUMENT" on the Mark 32 PR1 NOTESFOR tape.) Essentially, this guarantees that a privileged program which invokes the support library remains privileged during the invocation, and that a non-privileged program which invokes the support library remains non-privileged.

3. The following example WFL deck shows how the SYMBOL/BASEINTRINSICS are compiled:

```

? BEGIN JOB COMPILE/INTRINSICS;

    TASK T;
    COMPILE SYSTEM/BASEINTRINSICS WITH ALGOL [T] LIBRARY;
    ALGOL FILE TAPE(TITLE = SYMBOL/BASEINTRINSICS);
ALGOL DATA
$ MERGE SEQERR NEWSEQERR LINEINFO LIBRARY INTRINSICS
?

    IF T ISNT COMPILEDOK THEN
        ABORT "ERROR IN BASEINTRINSICS COMPILE";

? END JOB

```

4. The following example WFL deck shows how the SYSTEM/INTRINSICS are now created:

```

? BEGIN JOB BIND/INTRINSICS;

    TASK T;
    BIND SYSTEM/INTRINSICS WITH BINDER [T] LIBRARY;
    COMPILER DATA
$ INTRINSICS
BIND = FROM SYSTEM/BASEINTRINSICS;
?

    IF T ISNT COMPILEDOK THEN
        ABORT "RE-BIND INTRINSICS";

? END JOB

```

5. The procedure for binding user installation intrinsics has not changed.

6. The UDSTRUCTURETABLE is no longer a code file which is bound into SYSTEM/INTRINSICS. SYSTEM/USERSTRUCTURE now generates a patch which must be compiled into the GENERALSUPPORT library. (Users who require this feature should refer to Mark 32 USERSTRUCTURE note D3605 for details of the procedure.)

D4267 GENERAL - MARK "32 PRI NOTESFOR" TAPE

This note describes the NOTESFOR tape included in the Mark 32 PR1 system software release. The tape contains the following files:

LISTNOTES and OBJECT/LISTNOTES, which is a program which can be used to print the system notes contained on the tape (see GENERAL note D3396 for a description of LISTNOTES).

NOTESFOR32360, which is a file of D- and P-notes describing the software contained in this release.

ICMD/DOCUMENT, which is a file describing the implementation of the Industry-Compatible Mini-Disk (ICMD).

MULTIPROCESSOR/DOCUMENT, which is a file describing the B6800 Multiprocessor system released on Mark 31, the documentation for which does not presently exist elsewhere.

SHAREDRESOURCES/DOCUMENT, which is a file describing the Shared Resources system released on Mark 31, the documentation for which does not presently exist elsewhere.

NDLII/DOCUMENT, which is a file containing a preliminary user's guide to the NDLII software implementation and its computer controls, the documentation for which does not presently exist elsewhere.

DCAUDITOR/EXAMPLE, which is a BD file containing output from a sample run of DCAUDITOR.

ODT/DOCUMENT, which is a file describing the Mark 32 implementation of ODT messages, the documentation for which does not presently exist elsewhere.

OVERVIEW/DOCUMENT, which is a file containing selected B6900 notes referenced in GENERAL note D4239, "B5900 Overview", the documentation for which does not presently exist elsewhere.

D4323 GENERAL - CORRECTIONS TO MARK "32" SYSTEM NOTES

The following Mark 32 system notes should be corrected:

MCP Note D3480

This note should be deleted. The information is contained in Mark 32 GENERAL note D3354, "Intrinsic to Library Conversion".

DATACOM Note D3202:

In the "DCWRITE Results" section, the ERROR RESULT tables should be corrected as follows:
the leading "0" characters should be blanks.

DOCUMENT CHANGES NOTES (D NOTES)

ALGOL

D4108 ALGOL - "BCL" CONSTANTS ALLOWED

Previously, BCL constants and TRANSLATETABLEs involving BCL were flagged with a BCL warning message or, if the NOBCL option was TRUE, they were flagged as errors. They are now allowed, whether or not the NOBCL option is TRUE.

The use of these constructs will not hinder the transportability of the code files to EBCDIC machines.

Note that if a BCL pointer is used with a TRANSLATETABLE, the pointer will still be flagged with a warning or an error depending on whether NOBCL is FALSE or TRUE.

D4109 ALGOL - "<FAULT STACK HISTORY> ARRAY"

Page 5-59 of the ALGOL Reference Manual should note the following:

The STACKHISTORY information is stored in EBCDIC form regardless of the declared <array class> of the array.

D4110 ALGOL - TRUTHSET "ALPHA6" ALLOWED

Uses of the ALPHA6 truthset will no longer be flagged as errors when the NOBCL compiler control option is TRUE or flagged with warning messages when the NOBCL option is FALSE.

D4111 ALGOL - "BCL" CONSTANTS

Previously, the use of BCL constants in a PICTURE declaration would be flagged with a warning or an error, depending on the value of the NOBCL compiler control option. BCL constants are now allowed in PICTURE declarations regardless of the setting of the NOBCL compiler control option.

D4112 ALGOL - DOPE VECTORS CAN BE RESIZED

Dope vectors may now be resized. The syntax for RESIZE statement has been changed as follows:

<resize statement>

```
-- RESIZE -- ( --<resize array designator>-- , ----->
><arithmetic expression>--<retain old>-- ) -----|
```

<resize array designator>

```
---<array designator>-----|
|<event array designator>--|
```

<event array designator>

```
---<event array identifier>-----|
|<event subarray designator>--|
```

<event subarray designator>

```
--<event array identifier>-- [ --<subscript part>--<subarray part>----->
>- ] -----|
```

Semantics

The <resize statement> causes the size of the row or dimension specified by the <resize array designator> to be changed to the size specified by the <arithmetic expression>. If RETAIN is not specified, information in the "new" array is undefined. If RETAIN is specified, the information in the "old" row or dimension is transferred into the "new" resized array until all the information is transferred or the end of the "new" array is encountered.

The row or dimension to be resized is the one in which the first * subscript occurs in the <resize array designator> part. If no subscripts are indicated or all subscripts are * subscripts, then the first dimension of the array is the one to be resized.

If the <event subarray designator> or <subarray designator> form is used, the number of subscripts indicated (including * subscripts) must be equal to the number of declared dimensions of the array.

If the array is an event array or if a dimension other than the last is being resized, then RETAIN must be specified, and the new size specified by the <arithmetic expression> must be greater than or equal to the old size of the dimension to be resized.

The size of a dimension is defined as the difference between the value of the current upper bound and the value of the lower bound plus 1 (i.e. size = current upper bound - lower bound + 1). Hence, once a dimension is resized the lower bound remains the same, but the upper bound is altered to satisfy the following relationship: new upper bound = new size + value of the lower bound - 1.

Note that if a dimension other than the first is being resized, the resulting array will have rows of different lengths.

Examples:

original declaration	RESIZE STATEMENT	valid subscript ranges
A[3:10]	RESIZE(A,11)	[3-13]
A[3:10]	RESIZE(A[*],11)	[3-13]
A[3:10]	RESIZE(A[4],11)	none (invalid resize statement)
A[0:1,0:2,0:3]	RESIZE(A,3,RETAIN)	[0-2,0-2,0-3]
A[0:1,0:2,0:3]	RESIZE(A[*,*,*],3,RETAIN)	[0-2,0-2,0-3]
A[0:1,0:2,0:3]	RESIZE(A[1,*,*],5,RETAIN)	[0,0-2,0-3] and [1,0-4,0-3]
A[2:5,3:7,2:3]	RESIZE(A[2,3,*],5)	[2, 3,2-6] and [3-5,4-7,2-3]
A[2:6,3:7,2:3]	RESIZE(A[4,*,*],7,RETAIN)	[2-3,3-7,2-3] and [4,3-9,2-3] and [5-6,3-7,2-3]

D4162 ALGOL - CALL BY NAME PARAMETERS

If an expression was being passed as a call-by-name parameter to a procedure to be processed, the results of the processed procedure which used the call-by-name parameter could be incorrect.

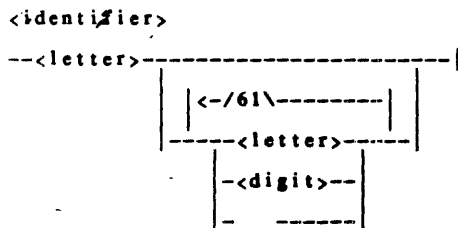
This problem has been corrected. Consequently, the definition of "critical block" given on Page 5-62 of the ALGOL Reference Manual (Form No. 5001639) should be changed to read as follows:

"The 'critical block' is the <block> of highest lexicographic level which contains the declaration of the procedure itself, call-by-name parameters, any 'thunk's associated with any of the call-by-name parameters, or the <task designator>."

D4166 ALGOL - ALLOW UNDERSCORES IN IDENTIFIERS

The definition of identifiers has been changed to include underscore ("_") characters.

The syntax diagram for an <identifier> is now the following:



Underscores are treated as significant characters in an identifier.

INTERACTIVEXREF has been changed to allow it to reference these new identifiers.

D4187 ALGOL - BATCH RESTRICTIONS

The following sentence should be added to the list of BATCH RESTRICTIONS on Page F-3 of the ALGOL Reference Manual (Form No. 5001639):

"j. When using BATCH, file-attributes may not be set to variables in the <file declaration>."

D4260 ALGOL - "ALGOL BATCH" AND ARRAYS

The ALGOL compiler no longer emits code to share arrays from one job with a succeeding job if the ALGOL BATCH facility is being used.

The following sentence on Page F-4 of the ALGOL Language Reference Manual (Form No. 5001639) should be deleted:

"The compiler makes attempts to share arrays from one individual job with succeeding jobs, eliminating many presence bit interrupts."

D4312 ALGOL - DOUBLE VS. SINGLE "BY-NAME" PARAMETER PASSING

On the Mark 34 release, the rules for type matching of call-by-name parameters will be tightened. No changes are planned for the rules for call-by-value parameters.

ALGOL has previously allowed implicit type conversions for call-by-name parameter passing, some of which could cause system-fatal errors triggered by a program fault. In order to prevent these system-fatal errors, additional restrictions will be imposed by ALGOL, as explained below.

Passing single-precision actual parameters to double-precision formal parameters and passing double-precision actual parameters to single-precision formal parameters will be de-implemented on the Mark 34 release. When the formal parameter is a simple variable and is declared call-by-value, this restriction does not hold. In this case, the compiler will generate the necessary code to correctly alter the precision of the actual parameter.

In all of the restricted cases, warning messages (on the Mark 34 release, error messages) will be given for any formal and actual parameter combinations which are to be disallowed by this change.

The following table lists the combinations of formal and actual parameters which will be affected.

FORMAL PARAMETER =====	ALLOWED ACTUAL PARAMETER =====	
	<u><type></u>	<u><type></u>
	currently -----	On Mark 34 -----
1. <simple variable>	<variable> <procedure identifier>	
A. COMPLEX	DOUBLE COMPLEX REAL INTEGER	DOUBLE COMPLEX
B. DOUBLE	DOUBLE REAL INTEGER	DOUBLE
C. REAL INTEGER	DOUBLE REAL INTEGER	REAL INTEGER

2.	<procedure identifier>	<procedure identifier>	
A.	DOUBLE	DOUBLE REAL INTEGER	DOUBLE
B.	REAL INTEGER	DOUBLE REAL INTEGER	REAL INTEGER
3.	<array identifier>	<array identifier> <direct array identifier> <value array identifier>	
A.	DOUBLE	DOUBLE REAL INTEGER	DOUBLE
B.	REAL INTEGER	DOUBLE REAL INTEGER	REAL INTEGER
4.	<direct array identifier>	<direct array identifier>	
A.	DOUBLE	DOUBLE REAL INTEGER	DOUBLE
B.	REAL INTEGER	DOUBLE REAL INTEGER	REAL INTEGER

D4322 ALGOL - "FREE-FIELD" READ OF HEX STRING

When the input to a free-field READ is a hex string, that string will be returned unaltered regardless of the type of the variable being read.

Add the following to the ALGOL Reference Manual (Form No. 5001639), Page 5-74:

"f. When the input is a hexadecimal string, that string will be returned unaltered as the value of the variable being read."

D4325 ALGOL - REPEAT COUNT, SIZE FIELD "<4030" IN FORMATS

Due to implementation limitations, the maximum number which can be used in a format as the repeat count, width, or decimal size is 4029.

Any value larger than 4029 causes a syntax error at compile time.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

ALGOL

P1017 ALGOL - ERRONEOUS ATTRIBUTE ERROR MESSAGE

An erroneous error message could occur if an attribute assignment had been preceded by a non-attribute assignment which was of the form `X := VALUE(BCL)`. The erroneous error message "BCL PROGRAMS ARE NOT PORTABLE TO EBCDIC MACHINES" would occur if the NOBCL option was TRUE. This message no longer occurs.

P1124 ALGOL - "IF <BOOLEAN EXP> THEN ELSE" STATEMENT

-If a statement assigning a value to a pointer immediately preceded a statement of the form
`IF <boolean expression> THEN ELSE;`

where the <boolean expression> referenced the pointer, a stack overflow could occur if these statements were within a loop. This problem has been corrected.

P1217 ALGOL - INCORRECT CODE FOR "POINTER-VALUED" ATTRIBUTES

In a <file declaration> or a <multiple attribute assignment statement>, the following problem occurred: an attribute would not be set if it occurred after a pointer-valued attribute had been set to a pointer. This problem has been corrected.

P1277 ALGOL - "LINEINFO"

When the LINEINFO compiler control option was true, proper line information would not be emitted for the first executable line of a new segment. This problem has been corrected.

P1287 ALGOL - "IPC-CAPABLE", RUN, CALL STATEMENTS

Effective with the Mark 31 release, ALGOL programs containing RUN statements were incorrectly marked as IPC-capable by the ALGOL compiler, and programs containing CALL statements were not marked as IPC-capable. Only ALGOL programs containing PROCESS or CALL statements are now marked as IPC-capable.

P1300 ALGOL - INCORRECT BRANCH USING "SWITCH LABEL"

When a <read statement> or <write statement> using a SWITCH LABEL with an incorrect index encountered an exception condition, an incorrect branch would be generated. This problem has been corrected. Now, if the index to the switch is an invalid value, the instruction attempting to branch is not executed and control proceeds to the next statement. This action is consistent with the documentation in the ALGOL Reference Manual (Form No. 5001639).

P3306 ALGOL - "SEG ARRAY" ERROR FOR LARGE DATA BASES

The ALGOL compiler no longer get a SEG ARRAY error when invoking very large data base descriptions.

P3813 ALGOL - "REAL (<POINTER EXP>,<ARITHMETIC EXP>)"

Previously, the following function could result in putting a double-precision operand on the stack:

```
REAL (<pointer expression>,<arithmetic expression>)
```

This action could cause problems, because the compiler always treated the result as if it were a single-precision operand.

The function has been altered so that it always returns a single-precision operand.

DOCUMENT CHANGES NOTES (D NOTES)

ATTABLEGEN

D4131 ATTABLEGEN - SPELLING CORRECTED

The HISTORYREASON task attribute value of "CARDREADXCEEDEV" should be spelled "CARDREADXCEEDEDV". The correct spelling has been implemented. Programs which used "CARDREADXCEEDEV" (the old value) should be modified to use "CARDREADXCEEDEDV".

D4138 ATTABLEGEN - NEW VALUES FOR "SUBFILEERROR"

Two new mnemonics for the file attribute SUBFILEERROR have been added:

NOFILEFOUND An attempted open on this subfile resulted in a "No File Found" result (the result from the OPEN function or AVAILABLE attribute was 2).

UNREACHABLEHOST An attempted open on this subfile resulted in an "Unreachable Host" result (the result from the OPEN function or AVAILABLE attribute was 38).

DOCUMENT CHANGES NOTES (D NOTES)

BACKUP

D4208 BACKUP - DOCUMENTATION CHANGE

The following changes should be made to the SOG Reference Manual, Volume 1 (Form No. 5011661):

Page 1-7-1 <disk file> syntax should read as follows:

<disk file>

```

-----|
| |<-/10\-----|
| | /<number> --- /<modified filename> - |

```

Page 1-7-3 Add the following after the description of "*" (special character)":

<disk file>

The <disk file> is the remaining part of the backup file name starting after the <job no> specification.

Page 1-7-7 Change the PB example in Example 2 to read as follows:

PB D 0000385/0000387/000LINE

Also, change the line:

"...followed by /<file name> exactly as specified"

to read as follows:

"...followed by <disk file> exactly as specified".

D4250 BACKUP - "ND REPORT" CHANGES

The following changes should be made to the SOG Reference Manual, Volume 1 (Form No. 5011661):

Page 1-7-4:

*REPORT Uses the columns used by outputs generated by the COBOL Report Writer feature. SYSTEM/BACKUP looks for a file with the BDNAME, "BDREPORT".

Example:

PB D 8131 0008583/000X KEY REPORT EQUAL "02"
PB "00081"1/0008583/000X" KEY REPORT EQUAL "02"

Both construct the same file title:

"BDREPORT/0008131/0008583/000X"

Page 1-7-7:

Change the following phrases in Example 2:

Change "Also, a prefix of BD or BP is assumed" to the following:

"Also, a prefix of BD, BP or BDREPORT (if a <key part> of REPORT is specified) is assumed."

Change "...constructs the file name by putting BD/ or BP/ first..." to the following:

"...constructs the file name by putting BD/, BP/ or BDREPORT/ first..."

Change the following phrase in Example 3:

Change "(BD, BP, or whatever" to the following:

"(BD, BP, BDREPORT or whatever".

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

BACKUP

P1014 BACKUP - "ND REPORT" CORRECTED

The following two problems have been corrected:

1. If the REPORT option were specified, BACKUP did not correctly append the prefix "BDREPORT".
2. If the REPORT option were specified, a section of code dealing with direct output was executed, even if ND were set.

P1201 BACKUP - "DLBACKUP" CORRECTION

Backup files created when DLBACKUP was not set to DISK or PACK could not be found. This problem has been corrected by searching on the DLBACKUP pack after not finding the file on DISK or PACK.

P1292 BACKUP - RESTORE PRINTING SPEED

BACKUP's ability to print files with one I/O per record in the backup file has been restored for all but COBOL REPORT files. This change improves BACKUP's performance.

P1371 BACKUP - DOUBLE PRINTING

A double printing problem has been corrected, as described in the following example:

Two backup files exist, TEST and TEST/TESTING, neither of which has the file name part:

<job no.>/<task no.>/<modified filename>.

If the following is entered:

PB "TEST" ND SAVE,

TEST will be printed twice and TEST/TESTING will not be printed.

This problem has been corrected; now, both TEST and TEST/TESTING are printed once.

P3743 BACKUP - DROP SPACES IN FILE TITLE

Leading spaces and spaces between file/directory name nodes are now ignored.

P3759 BACKUP - REQUIRES "LP" AFTER PUNCHING FILE

Previously, if a punch backup file was processed first, the printer backup file following it was erroneously suspended with the message "FILOUT REQUIRES EBCDIC LP". This problem has been corrected.

P3796 BACKUP - REPORT "EQUAL" CORRECTION

A problem that caused "EQUAL <string>" to not work properly has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

BARS
-----**P1015 BARS - "SPO" MODE**

Previously, only the first 80 characters of input were read from an ODT message to BARS when running in SPO mode. The problem has been corrected such that the input message may consist of up to 1514 characters.

P1258 BARS - "SPO" MODE "UNRECOGNIZED REQUEST"

Previously, when the TERM command was entered incorrectly from SPO mode, a SEG ARRAY error was generated following the "UNRECOGNIZED REQUEST" message. This problem has been corrected, allowing BARS to continue after displaying the message.

P1293 BARS - DISPLAY OF OVERLAY TRAFFIC PERCENTAGE

The scale factor applied to the overlay channel traffic (TRFC) has been modified to produce the correct percentage. Further, in tightly-coupled systems, the overlay activity display reflected only the amount contributed by one processor. This display has been corrected to display the overlay channel traffic for the entire system.

DOCUMENT CHANGES NOTES (D NOTES)

BASE INTRINSICS

D4255 BASEINTRN - "BASEINTRINSICS" IMPLEMENTATION

SYMBOL/BASEINTRINSICS is a single symbolic that replaces the following symbolics: ALGOLINTRINSICS, ALGOLPLINTRINSICS, ESPOLINTRINSICS and DCALGOLINTRINSICS. It is compiled by ALGOL and then bound by using the BINDER to create SYSTEM/INTRINSICS. This allows the compilers to access the INTRINSICINFO array as well as supporting the binding of user installation intrinsics.

SYMBOL/BASEINTRINSICS contains skeleton procedures for each of the currently-supported installation one intrinsics. These procedures are only skeletons of the actual procedures that are contained in support libraries, since these skeletons will not be invoked. At execution time, the MCP routes all uses of installation one intrinsics to the appropriate support libraries. These skeletons are needed by the BINDER to construct proper descriptions of the intrinsics for the INTRINSICINFO array.

DOCUMENT CHANGES NOTES (D NOTES)

BASIC**D4155 BASIC - FLAG LENGTH SPECIFICATION FOR "EBCDIC" FILE**

A syntax error is now given for use of the following undefined construct:

```
FILE #<filename>,<title>,<size specification>
```

This construct is undefined because EBCDIC files may be only sequential. (The size specification implies that the file is random.)

The following example program now gets a syntax error:

```
FILES *  
FILE #1, "FILE1", 200
```

D4168 BASIC - NEGATIVE ARGUMENT TO "RND"

Calling the RND (random) function with a negative argument now returns an arbitrary random number, instead of returning the same sequence of random numbers every time.

Example:

```
LET X=RND(-1)
```

D4254 BASIC - RECOMPILE "PRE-MARK 29" CODEFILES

BASIC codefiles which use strings and were compiled prior to the Mark 29 release now receive the following run-time error, followed by abnormal program termination:

```
"CODE-FILE TOO OLD. RECOMPILE."
```

The old ESPOLINTRINSICS procedures which supported the earlier version of BASIC strings are no longer provided.

If a program receives this error, it should be recompiled with the current compiler, which will link the program to the implementation of strings which has been in use since the Mark 29 release.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

BASIC

P1162 BASIC - COMPILER CAPACITY FOR STRING LITERALS

Previously, the BASIC compiler would abort with an INVALID INDEX when compiling a program with a large number of string literals. Now, the compiler's capacity for storing string literals has been substantially increased.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

DOCUMENT CHANGES NOTES (D NOTES)

BASICSUPPORT
-----**D3375 BASICSUPP - NEW FILE "SYSTEM/BASICSUPPORT"**

The file SYSTEM/BASICSUPPORT has been implemented as a system library to provide run-time support for BASIC programs.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

BASICSUPPORT

P1127 BASICSUPP - ALLOW SIX CHARACTERS IN FILE TITLE

On the Mark 30 and Mark 31 system software releases, changing a file title to a 6-character string would result in an INVALID INDEX in the intrinsics.

Example:

```
FILES *  
FILE #1, "F23456"
```

This problem has been corrected.

P1163 BASICSUPP - "RESTORE" STATEMENT

On the Mark 30 and Mark 31 releases, using the RESTORE statement would cause an INVALID INDEX in the intrinsics. This problem has been corrected.

P1193 BASICSUPP - FLUSH RANDOM BINARY FILE BUFFER

Formerly, if an existing file were opened for random binary access and then were written to, the contents of the last buffer-full (MAXRECSIZE words) would not be written to the file when it was closed. This problem has been corrected.

P1299 BASICSUPP - MATRIX INVERSION

The matrix inversion (MAT INV) procedure for BASIC formerly returned that a matrix was singular in some cases when it was not. This problem has been corrected.

P3307 BASICSUPP - ARRAY OF STRINGS DECLARATION

The declaration of an array of strings in the following form no longer causes an invalid index in the BASICSUPPORT library:

```
DIM AS (19)
```

DOCUMENT CHANGES NOTES (D NOTES)

BINDER

D4296 BINDER - DOCUMENTATION CORRECTIONS

The following changes should be made to Page 7-10 of the BINDER Reference Manual (Form No. 5001456):

1. At the top of the page, insert the following before the first line:

01 PR-RCD SIZE 42.
WORKING-STORAGE SECTION.

2. Replace the comma with a semicolon in step "e". The line should read as follows:

BIND = FROM SEP/=;

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

BINDER

P1289 BINDER - BOUND CODE RELEASE LEVELS

Previously, when codefiles were bound together, the resulting codefile had a release level equal to that of the binder that created it. The method of producing the release level has been modified, so that the release level of the bound code is now equal to the release level of the oldest codefile bound in.

P3186 BINDER - "BINDER" LOSES SHARING CLASS

When binding to a host which has a sharing class for libraries, the BINDER would lose the sharing class information and would default to SHARING=PRIVATE. This has been corrected; now, the sharing class is preserved from the HOST file.

P3187 BINDER - DEIMPLEMENT INSTALLATION INTRINSIC WARNING

Although the ability to bind installation intrinsics will eventually be deimplemented, the exact release has not yet been determined; therefore, the message stating that they will be deimplemented on release Mark 34 has been removed. The message will be re-released when necessary to provide two releases of warning prior to deimplementation.

P3213 BINDER - SYNTAX ERRORS

When binding installation intrinsics containing calls on system intrinsics that have been moved to the GENERALSUPPORT library, a syntax error was generated.

This problem has been corrected. Now, all references to system intrinsics are turned into unresolved intrinsic references, that are fixed by the MCP when they are touched.

P3232 BINDER - INSTALLATION ONE INTRINSICS

References to all but the installation one intrinsics have been removed from the BINDER'S VALUE ARRAY (see BASEINTRN note D4255) for further details).

DOCUMENT CHANGES NOTES (D NOTES)

BURROUGHS NETWORK ARCHITECTURE

D3194 BNA - "BNA" IMPLEMENTATION

INTRODUCTION

Burroughs Network Architecture (BNA) provides a means of connecting Burroughs computer systems together into networks. The purpose of these networks is distributed processing and resource sharing. BNA networks have the following attributes:

- 1 All systems in the network are peers no matter what the computing capacity.
- 2 Control is completely distributed, thus avoiding the vulnerability of a central controlling node.
- 3 The * BIAS tm routing algorithm responds automatically to changes in network topology.

The * BIAS tm routing mechanism is based on the exchange of current routing information between neighbor nodes in the network. This information consists of relative resistances between nodes. Each link between neighbor nodes is given a relative resistance factor based on the speed and efficiency of the link. Each node is assigned a resistance factor based on its speed of transmitting a message. The resistance between two nodes is the sum of resistances along the path (which might involve several links and nodes). The messages routed will always take the current path of least resistance.

* "BIAS" is a trademark of Burroughs Corporation.

- 4 The user interfaces are simple and easy to use.

BNA is divided into two major functions: Network Services and Host Services. Network Services provides the following:

- 1 Reliable data transportation from point to point in the network.
- 2 Segmentation and reassembly.
- 3 Routing.
- 4 End-to-end flow control.

Host Services is a series of high-level host-to-host protocols to provide distributed processing and resource sharing capability.

There are five phases of Networking that are visible to the user. They are as follows:

- 1) Node is NOT IN NETWORK MODE
- 2) Node is NETWORK INITIALIZING
- 3) Node is NETWORK OPERATING
- 4) NODE IN SLOW SHUTDOWN
- 5) NODE SHUTTING DOWN

"NOT IN NETWORK MODE" is anytime before the network is brought up and after it has shut down.

The node is "NETWORK INITIALIZING" during the period of time between "NET+" command and the "ENDINIT" command. During this period the network is going through its initialization sequence.

The network initialization sequence starts by going through two information entry phases. The first is the Attribute Entry Phase. The second is the Data Entry Phase. The initialization sequence changes from Attribute Entry Phase to Data Entry Phase at the first encounter with one of the following Operation Interface Messages (OIMs):

```
ADD HOST
ADD NODE
ADD CONNECTION
MODIFY STATION
MODIFY CONNECTION
DELETE HOST
DELETE NODE
DELETE CONNECTION
```

If the ENDINIT command is encountered during Attribute Entry Phase, the initialization sequence will skip the Data Entry Phase. This does not mean that the above commands will not be serviced during a later phase, only that the initialization sequence will continue without them.

Some network attributes may be set only during the Attribute Entry Phase. They include such attributes as:

```
NETWORKMAXSEGMENTSIZ (NWMSS)
NETWORKVERSION (NWVER)
VALIDATE (VAL)
```

LOCALIDENTITY (LOC)

LOCALIDENTITY is the only attribute that must be supplied by the user. All others have default values.

When the "ENDINIT" command is encountered, the network enters the "NETWORK OPERATING" phase. The network is now fully operational.

The network will enter "NODE IN SLOW SHUTDOWN" when the "NET-" command is entered. The node will remain in this phase until there are no more subports opened to remote hosts or until the "NET-NOW" command is entered, at which time, the node enters the "NODE SHUTTING DOWN" phase. The "NET-NOW" command will close all subports to remote hosts before proceeding with the shutdown sequence. Once the node completes the shutting down sequence the node is once again "NOT IN NETWORK MODE".

BNA STATIONS OVERVIEW**Definition of BNA Stations**

A node in a BNA network communicates with its neighbors by entities known as stations. A BNA station consists of the hardware and software controlling one communication path to a neighbor. There are two major categories of stations currently supported by BNA -- Global Memory stations and Burroughs Data Link Control (BDLC) stations.

Global Memory Stations

On a * GLOBAL tm Memory system, each node automatically has four Global Memory stations defined. They are named GMM01 through GMM04. Through these stations, the local node may establish connections to each of the other nodes attached to the Global Memory Module. There may be up to three other such nodes, and each node may have one Global Memory connection to each of its Global Memory neighbors. Hence, each node may have, at most, three active Global Memory stations.

* "GLOBAL Memory" is a trademark of Burroughs Corporation.

BDLC Stations

BDLC stations must be declared through NDL. They are used to communicate with other nodes via full-duplex leased lines and half-duplex switched lines. During initialization, the BNA MCS requests, from the datacom subsystem, a list of the BDLC stations which are declared in NDL to be under its control. These BDLC stations are then entered in the station table (in addition to any automatically allocated Global Memory stations). A BNA node may have as many BDLC stations as its datacom configuration can handle. These can be any combination of switched or leased lines of any speed and may be distributed among several DCPs. A BDLC station may be used to establish a connection with any other node in a BNA network, regardless of whether another station (Global Memory or BDLC) is already connected to that node.

Multiple Parallel Links

If a node has several active stations open to the same neighbor, traffic between the two nodes will be routed through all of those stations, thereby increasing the effective bandwidth to that neighbor. This ability to simultaneously send several messages over different links to the same neighbor is a feature frequently referred to as "multiple parallel links".

ESTABLISHING CONNECTIONS ON BNA STATIONS

From an operational standpoint, Global Memory and BDLC stations are virtually identical; i.e., all stations have the same set of attributes and are affected by the same set of operator commands. For example, the ESTABLISHCALL (abbreviated EC) command is used to open a permanent connection on a dedicated station or to open a dial-out connection on a switched station. The AWAITCALL (AC) command is used to open a dial-in connection on a switched station. The CLEARCALL (CC) command is used to close a connection of any type. Global Memory stations are dedicated stations; i.e., they have a permanent communication path that is always available. Thus, they are very much like BDLC stations on dedicated (e.g., leased) lines and will always have connections opened through the use of the ESTABLISHCALL command.

A connection between two stations is actually established in several steps. These reflect the fact that several levels of the hardware and software are involved and each level must establish a "dialog" with its equivalent layer in the other node. In normal operation, BNA hides these levels from users -- all required steps are performed upon receipt of an ESTABLISHCALL or AWAITCALL command. Through the use of appropriate commands in an initialization file, it is even possible to have all connections opened automatically at initialization time. However, in the event of system failure, an understanding of the underlying architecture will assist greatly in fault analysis. The following sections describe that architecture and explain the steps performed when opening a BNA link between two nodes.

Connection Port

First, the underlying physical link, known as the CONNECTION PORT, must be opened. The connection port on a dedicated link is always available -- it is "opened" solely by changing the state of the link as maintained by BNA in some internal tables. The connection port on a switched link, however, must be opened by placing a phone call. If the call is dial-out, a phone number, referred to as CALldata, must be supplied. This is described in detail in the explanations of the ESTABLISHCALL and ADD CONNECTION commands. If it is desired to open a dial-in connection, an AWAITCALL command must be issued for the appropriate station. This will place the station in an appropriate state to answer an incoming phone call. The connection port will not be opened, however, until a call is received.

Station Dialog

After the connection port is open, the stations at either end establish what is known as the STATION DIALOG. This process consists of the sending of certain control commands between the stations to ensure that they are both in the same initial state and are ready to begin transmission of information frames. BDLc stations and Global Memory stations use different commands to establish the station dialog, but the purpose of both is the same.

Validate and Attach

Once the station dialog has been opened, the stations begin to exchange information about their respective nodes. Each station sends to the other a series of messages, known as GREETINGS, which identify the sending node and contain the values of certain important BNA attributes. For example, each node sends to the other its node address, a password, the level of BNA software it is using, which prior levels of BNA it is capable of supporting, the version of the network it is running (the version is a user-chosen value used to assist in coordinating configurational changes across nodes) and the maximum size segment (message) which it is willing to handle.

Each node checks the information in the greetings it receives and verifies that the proposed connection is acceptable. This process is known as VALIDATION. Each station sends a final greeting to the other informing it either that it has passed validation, or the reason for which it has failed. If both stations pass the validation process, the connection is established and each station is ATTACHED to the ROUTER level at its respective node.

There are generally three GREETINGS exchanged between stations -- GREETING-0, GREETING-1 and GREETING-2. GREETING-0 and GREETING-1 contain the information described above, while the GREETING-2 is used only to send to the other station the results of the validation process. If, upon receipt of a GREETING-0, a node decides that the connection is not acceptable, it may omit its GREETING-1 from the exchange and immediately send the other node a GREETING-2 with a negative reply. Also, if a node is still initializing while exchanging GREETINGS with another node, that node must, upon advancing to OPERATIONAL PHASE, send an Initialization-Complete (INIT-COMPLETE) message to the other node. The INIT-COMPLETE can be considered to be a fourth greeting which is not always needed. Neither, both or only one of the nodes may send it when opening a particular connection. However, a station will neither ATTACH to its Router, nor send information frames, until it has sent and received any expected INIT-COMPLETE messages.

Validation Failure

If a validation failure occurs while opening a connection, each node displays a message showing the station and neighbor involved and the reason for failure. The message displayed will identify the failure as either a "local" or "remote" validation failure depending on whether the negative reply was generated, respectively, at the local or remote node. Thus, if one node calls another node and the calling node is not authorized to establish a connection on the station that answers the call (or, if some other error occurs) a message with the following form will be displayed at the calling node.

REMOTE VALIDATION FAILED AT <neighbor> ON <station>: <reason>

Meanwhile, a similar message, with the following form, will be displayed at the node which received the call.

LOCAL VALIDATION FAILED BY <neighbor> ON <station>: <reason>

In both cases, <neighbor> identifies the other node and <station> identifies the name, at the node displaying the message, of the station involved in the GREETINGS exchange.

<reason> may be one of the following:

1. SL VERSION NOT RESOLVED

The Station-Level Protocol Version in use by one station is not supported by the other station. The two versions of BNA in use are not compatible -- an old version of BNA is probably in use at one of the nodes.

2. NETWORK VERSION MISMATCH

The user-chosen string-valued attribute NETWORKVERSION (NWVER) is not the same at both nodes. An old initialization file is probably in use at one of the nodes. This error will not occur if the default value of NWVER (null string) is used at both nodes.

3. NMSS MISMATCH

The value of NETWORKMAXSEGMENTSIZ (NMSS) in use at the two nodes is not the same. A probable cause would be omission of the NMSS command in the initialization file of one node. This error will not occur if the default value of NMSS (128 bytes) is used at both nodes.

4. NODE ADDRESS VALIDITY FAILURE

The Other-Node-Address (ONA) in a GREETING sent to one of the nodes did not pass that node's Neighbor Validation. There are many conditions under which this error could occur. Several are listed below.

- The wrong neighbor was specified on an EC command.
- The wrong CALLDATA was specified on an EC command.
- Neighbor Validation was requested at a node, but an ADD CONNECTION command was omitted from the initialization file.
- The node who called was actually not authorized to call on the affected station.
- The LOCALIDENTITY (LOC) command in an initialization file had an incorrect value for that node's address.
- The node which received the negative reply sent an illegal node address. This would generally be a software failure and should not occur.
- Neighbor Validation is described in more detail in the sections entitled "Station Attributes", "Initialization of a BNA Node", and "Neighbor Validation".

5. SHUTDOWN IN PROGRESS

One of the nodes is shutting down BNA and refuses to open any new connections.

6. PASSWORD MISMATCH

One of the nodes sent to the other a password other than the one it was expecting. Probably, a password was misspelled in, or accidentally omitted from, the initialization file at one of the nodes. Or, perhaps one node confused the MYNODEPASSWORD (MNPW) and YOURNODEPASSWORD (YNPW) clauses in the NODEPASSWORDS (NPW) command. For a node which is processing an NPW command, MNPW is the password which it will send to the remote node, and YNPW is the password which it expects to receive from the remote node.

For each of the other nodes in a network, BNA stores a value for both MNPW and YNPW. When exchanging greetings, if either of the nodes does not receive the password it expects, the sending node will fail validation. The default value for all passwords is NULL (a null string). This default is used if no NPW command was received. A null password is also used for MNPW or YNPW if an NPW command was received which indicated, respectively, that MNPW or YNPW was to be deleted. If the password expected from a remote node (that is, YNPW) is null, then any password will be accepted, including a null password. Otherwise, the password must match exactly the YNPW entered in the NPW command.

7. INSUFFICIENT RESOURCES

The node which sent this reply does not have sufficient resources to establish an additional connection. This error will never be sent by a Large Systems BNA node.

8. INVALID GREETING

The node which sent this reply received a greeting which had an invalid length, or the greeting had an invalid value in a field but the error did not match one of the error messages listed above. This error generally indicates a failure in system software and should not occur. In the Mark 32 Large Systems release of BNA, this error is also displayed when the LINKMAXSEGSIZE (LMSS) value sent by the remote node is invalid. However, this should also not occur unless there is a failure of the system software.

Manual Commands

As noted before, the steps described above are performed upon receipt of an ESTABLISHCALL or AWAITCALL command. However, at times it may be desirable to perform the process manually. In particular, the manual process may help to identify a failed component or more easily isolate a fault. The following manual, or maintenance, commands are provided for this purpose.

1. OPEN CONNECTIONPORTDIALOG (OPEN CPD): used to open the connection port on a station. A station name must be specified. If the station has an ACU, CALLDATA may be included.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

2. **OPEN STATIONDIALOG (OPEN SD)**: used to open the station dialog on a station. A station name must be specified.
3. **VALIDATEANDATTACH (VALATT)**: used to initiate the process of exchange of greetings and validation between stations. A station name must be specified. If successful, the station is attached to the Router.

Just as the three commands described above may be used to perform the manual equivalent of an **ESTABLISHCALL**, the following commands may be used, in the order shown, to perform the manual equivalent of a **CLEARCALL**.

4. **DETACH (DET)**: used to detach a station from the Router and close the station dialog. A station name must be specified.
5. **CLOSE STATIONDIALOG (CLOSE SD)**: used to close the station dialog on a station. If the station is attached to the Router, it will also be detached. A station name must be specified.
6. **CLOSE CONNECTIONPORTDIALOG (CLOSE CPD)**: used to close the connection port on a station. A station name must be specified.

STATION ATTRIBUTES

Following is a list of the primary station attributes and a brief description of each. An understanding of these attributes and their interactions will be helpful when setting up an initialization file. Also, later descriptions of certain BNA features and commands will assume a familiarity with the terms defined below. These attributes are among those displayed in response to the **STATION** interrogation

Station Name

Global Memory stations are named **GMM01** through **GMM04**. The two digits at the end of the name represent a Global Memory channel number. Channel numbers are assigned arbitrarily to the various processors at Halt/Load time and will not necessarily remain constant. In a four-processor system, with nodes named **A** through **D**, node **A** may open its station **GMM01** at one point to establish a connection with node **B**, but after the next Halt/Load node **A** may open **GMM01** and find it connected to node **C** or **D**.

BDLC stations retain the names assigned to them in **NDL**. A BNA station name must be composed solely of alphanumeric characters and may not exceed seventeen characters in length.

Station Type

The station-type attribute has one of the following values:

- Global Memory
- BDLC Dedicated
- BDLC Switched with ACU
- BDLC Switched without ACU

The station-type attribute is set by BNA. "Global Memory" stations connect only to other nodes attached to a Global Memory module. "BDLC Dedicated" stations connect to a neighbor via a full-duplex dedicated line -- e.g., a leased phone line. "BDLC Switched with ACU" stations are attached to switched half-duplex lines and have an auto-call unit (ACU) to place outgoing calls. They also must have auto-answer capability. "BDLC Switched without ACU" stations have no ACU, but may have auto-answer capability. If a switched station has neither an ACU nor auto-answer capability, all calls must be manually dialed and answered; only outgoing calls need to be handled manually if the station has auto-answer capability.

Hardware ID

Hardware-ID is set by BNA. For Global Memory stations, it represents the channel number to global memory. For **BDLC** stations, hardware-id is the Logical Station Number (LSN) assigned in **NDL**.

Neighbor Node Address

Neighbor-Node-Address is set by BNA. When a station is closed, Neighbor-Node-Address has a value of zero (0). When open, it holds the node address of the neighbor to which the station is connected. If the station is not yet fully open, but the local node knows which neighbor is supposed to be at the other end of the link, Neighbor-Node-Address will show that neighbor's address.

AUTOINIT

AUTOINIT is a user-settable Boolean-valued attribute. If TRUE, BNA will automatically attempt, at initialization time, to open a connection via the station. If FALSE, the station will remain closed and must be opened by an operator command. Also, if a connection is closed due to failure at the remote node, or if a SAVED station is made READY, BNA will automatically try to re-open the station if AUTOINIT is TRUE. The default value for AUTOINIT is TRUE. The value may be changed at any time through Use of the MODIFY STATION command.

The action performed on the station at initialization time depends on the type of station. Global Memory stations and BDLC Dedicated stations will have the equivalent of an ESTABLISHCALL command performed. BDLC switched stations (with or without ACU) will have the equivalent of an AWAITCALL command performed. Thus, by default, every dedicated station will be opened and every switched station will be waiting for an incoming call. It is also possible to place an outgoing call automatically via a BDLC station with ACU, but this function is requested through the use of the ADD CONNECTION command, rather than the AUTOINIT attribute.

Local-Status

Local-Status is a station attribute which is maintained solely by BNA. It describes the state of the station with respect to establishing a connection to a neighbor node; i.e., Local-Status does not usually change as frames are sent and received on a station with an active connection. However, this attribute will change in response to such commands as ESTABLISHCALL and CLEARCALL. At initialization time, all stations are in Closed state. The following list shows the values of Local-Status and briefly describes each state.

- Closed station is closed
- Pend Open CPD connection port dialog is pending open
- Pend Close CPD connection port dialog is pending close
- CPD Open connection port dialog is open
- Pend Open SD station dialog is pending open
- Pend Close SD station dialog is pending close
- SD Open station dialog is open

- Pend Rcv G0 local node sent Greeting-0 -- waiting for Greeting-0 from remote node
- Pend Rcv G1 local node sent Greeting-1 -- waiting for Greeting-1 from remote node
- Pend Rcv G2 local node sent Greeting-2 -- waiting for Greeting-2 from remote node
- Pend Send Init-Complete local node still initializing -- will send Init-Complete message to remote node when Initialization is complete
- Pend Rcv Init-Complete local node waiting for Init-Complete message from remote node
- Pend Attach Router station is pending attach to the router
- Attached connection is established with neighbor -- station attached to Router at local node
- Pend Detach Router station is pending detach from the router

Saved

Saved is a user-settable Boolean-valued attribute. When TRUE, BNA will not allow any connections to be established by the saved station; i.e., all ESTABLISHCALL and AWAITCALL requests will be rejected and the function of AUTOINIT will be suspended on the affected station. Saved is set to TRUE by the SAVE STATION command, and is set to false by the READY STATION command. If the station has a connection open to a neighbor at the time a SAVE command is received, the Saved attribute will be set to TRUE as an indication that the station is "to be saved" when the connection is closed.

Link Speed and Link Efficiency

Link Speed and Link Efficiency are user-settable integer-valued attributes. Link Speed represents the speed of the link in bits-per-second. Link Efficiency represents the efficiency of the link in percent. The default value of Link Speed is 1,000,000 for Global Memory stations and 9600 for BDLC stations. The default value of Link Efficiency is 98 for Global Memory stations, 95 for BDLC permanent stations, and 45 for BDLC switched stations. These values may be changed by the MODIFY STATION command, using the speed and efficiency (EFF) clauses. These attributes may be changed only while the station is closed.

The exact values of Link Speed and Link Efficiency are not crucial to the functioning of BNA. However, the values of these two attributes are used by the Router to calculate the most efficient route to reach another node. Also, when multiple parallel links exist to a neighbor, the amount of data sent over each link will be proportional to the link's declared capacity. This capacity is computed as the product of Link Speed and Link Efficiency. Thus, an installation should include MODIFY STATION commands in its initialization file to reflect the actual speeds of its lines. The Link Efficiency attribute may also be modified to influence the Router's choice of preferred paths and to alter the load distribution on multiple parallel

links. Changing the Link Speed attribute through the MODIFY STATION command will not change the speed at which data is actually sent over a link. Such a change, when possible, is usually controlled by a setting on a modem.

Link Max Segment Size

Link Max Segment Size (LMSS) is a user-settable integer-valued attribute. It represents the size, in bytes, of the largest segment which can be transmitted over this link. Valid values are in the range 60-65,535. The default value for all stations is the value of NETWORK MAX SEGMENT SIZE (NMSS). NMSS is set by an operator command at initialization (usually in an initialization file). The value of LMSS may be changed by the MODIFY STATION command, using the MAXSEGMENTSIZ (MAXSS) clause. This attribute may be changed only while the station is closed. Its value must always be greater than or equal to the NETWORK MAX SEGMENT SIZE attribute.

Working Link Max Segment Size

Working Link Max Segment Size (WLMSS) is an integer-valued attribute which is maintained by BNA. It represents the actual maximum segment size in use by an active station. The value is determined during validation when a connection is established between two BNA nodes. The remote and local stations send to each other the values of their respective Link Max Segment Size attributes. The smaller of the two values is chosen, by both stations, for their WLMSS. This value must be greater than, or equal to, the NETWORK MAX SEGMENT SIZE (NMSS). When the default value of NMSS is used for Link Max Segment Size at both stations, WLMSS will also be equal to NMSS.

Monitor

Monitor is a user-settable Boolean-valued attribute. When set to ON, certain statistics which are accumulated for the station will occasionally be written to the system log file. The frequency of this logging function is controlled by the Station Monitor Interval attribute, which is set through the MONITOR (MON) command. The default value of Monitor is OFF. The value of Monitor can be changed by the MODIFY STATION command at any time. See the MONITOR command for more information on station-level monitoring.

Remote Busy

Remote Busy is a Boolean-valued attribute maintained by BNA. When TRUE, the station has received notification from the remote station that it is temporarily unable to accept frames. This is usually due to a short-lived buffers-full condition, but may be an indication that the remote node is taking a memory dump. If all stations connected to a neighbor node are simultaneously busy, the neighbor is also marked as busy and a timer is begun. The timer expires after the number of seconds specified in the NEIGHBOR BUSY TIMEOUT (NBTO) attribute, whose default value is 60 seconds. If all stations are still busy at the end of that period, the affected stations will all be cleared, as though a CLEARCALL command had been entered. See the description of the NBTO command for more information.

Neighbor Validation

Neighbor Validation is a user-settable Boolean-valued attribute. When TRUE, it indicates that restrictions have been placed on the station such that only certain neighbors may establish connections on that station. These restrictions are placed on the station through the ADD CONNECTION command. When Neighbor Validation is TRUE, it indicates that at least one ADD CONNECTION command has been received for the specified station. If those connection entries are deleted (see the DELETE CONNECTION command), Neighbor Validation will be reset to FALSE.

The Neighbor Validation station attribute is related to the concept of neighbor validation, which is controlled by the VALIDATE (VAL) command. However, neighbor restrictions can be placed on an individual station independently of the setting of the "Validate" options. In addition, all validation settings can be overridden on a call if the call is made using the ESTABLISHCALL command containing a neighbor phrase.

Neighbor Validation is explained more fully, with examples, in a separate section later in this document. Also, see the description, below, of the VALIDATE command in the section entitled "Initialization of the Station-Level at a BNA Node."

OPERATION

Initialization of a BNA Node

Initialization of a BNA node is performed primarily through commands in an Initialization File. These commands are used to set the values of assorted BNA attributes and to initialize internal tables. A BNA site may use these commands to select optional BNA features, to describe the configuration of the local node, to identify other nodes in the network, to specify which of those other nodes are neighbor nodes (i.e., nodes to which the local node will have a direct link), to restrict connections on certain links to only a given set of neighbors, and to request that some links be opened automatically. However, it should be stressed that use of most of these commands is optional -- appropriate default values and default actions have been

chosen for BNA so the first-time user may quickly bring up a BNA network. A simple BNA network may easily need only a handful of commands in its initialization file. These commands are organized in the following two major groups:

1. Attribute Entry Commands -- commands which set the values of BNA node attributes.
2. Configurational Commands -- commands which describe the configuration at the local node, the names and addresses of other nodes in the network and the stations by which the local node may open connections to those other nodes.

Once initialization of the node has been completed, the following types of commands may also be used:

3. Operational Commands -- commands used to establish connections with other nodes, clear connections, save and ready stations, and perform other operational requests.
4. Interrogation Commands -- commands used to interrogate the values of certain attributes, display information about the local node or remote nodes, and generate reports based on internal tables maintained by BNA.

Initialization at a BNA node actually consists of the independent initialization of several distinct levels of the software and hardware. These levels include the Station Level, the Router Level and the Port Level.

The Station Level consists of the hardware and software that control the physical links to neighbor nodes. In Large Systems BNA, this includes the DCP, the NDL (BDLC request set), the Global Memory ports, and the Station-Level-Manager (SLM) software in BNA.

The Router Level consists of the software in BNA that controls transmission of frames between origin and destination nodes. The Router maintains tables showing the configuration of the network, as seen by the local node. Using these tables, the Router can ascertain such information as which nodes are in the network, which nodes are neighbors (i.e., to which nodes the local node has a direct link), through which neighbor to route a frame to reach a non-neighbor node, by what alternate path a particular node can be reached should the preferred path become unusable, etc.

The Port Level consists of the software which controls inter-process communication (IPC). In BNA, IPC is provided to application programs through the use of Port Files (files with KIND=PORT). For example, all services provided by hosts in a BNA network (services such as File Transfer, file access and job transfer) use Port Files as a communication path.

Once initialization of these three levels is complete, a BNA node can proceed to establish connections with other nodes in the network. As these connections are established, the equivalent levels of BNA at each node establish dialogs. That is, the SLM at the local and remote nodes will exchange information, as will the two Router Levels and Port Levels. In addition to the dialogs maintained between each node and its neighbors, the Port Level establishes dialogs to each other node in the network which is providing host services.

Initialization of the Station-Level at a BNA Node

Initialization of the Station-Level at a BNA node is controlled by the Station-Level-Manager (SLM). During initialization, the SLM passes through several successive phases as initialization commands are received and processed. These commands are usually read in from an initialization file, but may be entered at an CMT or Network Services Terminal. The phases through which the SLM passes are described below.

1. Attribute Entry Phase

During Attribute Entry Phase, the SLM processes Attribute Entry Commands. These include the following.

- a. LOCALIDENTITY (LOC): used to set the Local Identity attribute, more frequently referred to as the local node address. This command is the only attribute entry command which is required -- there is no default value. It will usually be the first command in an initialization file, as most other commands will not be accepted until the local node address has been set.

Each node in a BNA network has both a host name and a node address. The host name is used by end-users to request services from other host computers in the network. The node address is used internally in BNA for routing purposes. Each node address and each host name in the network must be unique.

The LOC command is used in BNA to set only the local node address. The local host name is set via the HN ODT message, not through a network command (i.e., a command entered with an "NW" prefix). The local host name must be set prior to BNA initialization. Although the syntax of the LOC command allows inclusion of a hostname, this is used only for verification purposes. If the hostname is incorrect, the LOC command will fail. If the command is in an initialization file, processing of the file will be suspended. This feature can be used to ensure that a node is never initialized from another node's initialization file.

- b. **NETWORKVERSION (NWVER)**: used to set the NetworkVersion attribute. This command is optional -- the default value is a null (empty) string. When a station first makes contact with a neighbor node, each node sends the other the value of its NetworkVersion attribute. The connection is terminated if the two values do not agree. This attribute is supplied for the convenience of a Network Administrator who may want to coordinate, at many nodes, changes in the configuration of a network.
- c. **NETWORKMAXSEGMENTSIZ (NWMSS)**: used to set the NetworkMaxSegmentSize attribute. When a message is sent from one node in a BNA network to another, the message is split into one or more pieces known as segments. This attribute sets the value, in bytes, of the largest segment that can be sent between nodes of the network. This command is optional -- the default value is 128 bytes. When a station first makes contact with a neighbor node, each node sends the other the value of its NetworkMaxSegmentSize attribute. The connection is terminated if the two values do not agree. This ensures that the same value is used by all nodes in a network.
- d. **VALIDATE (VAL)**: used to set the HOST, ROUTER and NEIGHBOR validation options. This command is optional -- the default value for all validation options is FALSE.

If Host Validation is requested, then HOSTSERVICES dialogs will be successfully established only with known hosts; i.e., with hosts which have been explicitly declared in ADD HOST commands. If HOST validation is TRUE, HOSTSERVICES will refuse all port-level communications with nodes which were not explicitly declared to be hosts.

If Router Validation is requested, then the Router will handle traffic only to and from known nodes; i.e., nodes which have been explicitly declared in ADD NODE commands. If a router frame (i.e., an information frame with a router header) arrives whose Origin Node Address (ONA) or Destination Node Address (DNA) is not in a table of valid node addresses, the frame will be rejected. Note that in this case, the Router will not only refuse to open a dialog with an unknown node, but will also refuse to act as an intermediate transit node for that node's traffic. Router Validation may be set only by requesting that all validation options be enabled (VALIDATE=ALL) -- it can not be requested as an independent validation option.

If Neighbor Validation is requested, then links will be successfully established only between station/neighbor pairs which have been explicitly declared permissible. This is usually performed through the use of the ADD CONNECTION command. Each ADD CONNECTION command specifies one station/neighbor pair which may establish a connection. However, for dedicated or dial-out stations, if an ESTABLISHCALL (EG) command is issued and a neighbor phrase is included in the command, this connection will be successful only if the actual neighbor node is the node specified in the EC command. This is true without regard to the setting of Neighbor Validation or the previously entered ADD CONNECTION commands. Thus, the inclusion of a neighbor phrase on an EC command may be considered to override the normal Neighbor Validation function. Note, however, that when Neighbor Validation is in effect, a dial-in connection will succeed only if an ADD CONNECTION command has been issued for the affected station/neighbor pair -- there is no provision on the AWAITCALL command for specification of a neighbor.

Neighbor Validation is explained more fully in a separate section below.

2. Configuration Entry Phase

During Configuration Entry Phase, the SLM processes Configuration Entry Commands. These include the following.

- a. **MODIFY STATION (MOD STA)**: used to change the values of certain station attributes, especially to override their default values. The MODIFY STATION command may be used to change the values of AUTOINIT, SPEED, EFFICIENCY (EFF), MAXSEGMENTSIZ (MAXSS) and MONITOR (MON).
- b. **ADD CONNECTION (ADD CONN)**: used to declare which neighbors may establish connections with which stations. This is described in more detail below, under Neighbor Validation. Also, the ADD CONNECTION command may be used to store phone numbers (CALLDATA) for dial-out connections and to specify which such calls should be placed automatically at initialization time. This latter feature is provided through the use of the INITQUANTITY clause.
- c. Although normally not present in an initialization file, the MODIFY and DELETE CONNECTION commands complement the ADD CONNECTION command and may be entered at any time after the SLM has advanced to Configuration Entry Phase.
- d. **ADD NODE and ADD HOST**: these commands are not strictly Station-Level commands. However, when processing any command which references a node in the network (e.g., the ADD CONNECTION command), the SLM will check any host name against the previously declared HOSTS and, if Router Validation is set, will check any node address against the previously declared NODEs. Thus, it is usual for the ADD NODE and ADD HOST commands to precede other configuration commands.

For example, the following ADD CONNECTION would fail on an UNKNOWN HOST error.

```
ADD HOST BRANCH1 #3;
ADD CONNECTION WITH BRANCH2 ... ;
```

If Router Validation is requested, then all nodes in the network must be declared. The following ADD CONNECTION command would fail on a UNKNOWN NODE error.

```
ADD HOST BRANCH1 #3;
ADD CONNECTION WITH #2 ... ;
```

3. Start-Up Phase

During Start-Up Phase, the SLM attempts to establish all connections which have been requested to be opened automatically.

If a BDLC station has an ACU and an ADD CONNECTION command has been received which specifies an INITQUANTITY greater than zero, then an outgoing call will be placed to the neighbor specified in the ADD CONNECTION command. This will be performed as though an ESTABLISHCALL command had been entered.

For all other types of stations, an attempt will be made to open a connection only if the AUTOINIT attribute is TRUE. The default value of AUTOINIT is TRUE for all stations; it will remain TRUE unless changed by a MODIFY STATION command.

The action taken when AUTOINIT is TRUE depends on the type of station. For dedicated stations (BDLC dedicated and global memory) the equivalent of an ESTABLISHCALL command will be performed. For a BDLC switched station (with or without ACU) the equivalent of an AWAITCALL command will be performed. Note that the automatic placement of outgoing calls on switched stations (described above) is independent from the AUTOINIT attribute.

4. Operating Phase

When the SLM advances to Operating Phase, the node is operational and ready to participate in dialogs with other nodes in the network.

Neighbor Validation

Neighbor Validation refers to a BNA option that allows a node to control which other nodes in the network may be neighbors; i.e., which other nodes may establish direct links to the local node. As previously described, when a connection is first opened, each station sends to the other a series of GREETINGS. In the first greeting (GREETING-0), each station places its own local node address. At the receiving node, this field is known as the Other Node Address (ONA). If the receiving node has Neighbor Validation in effect, this ONA is checked against all neighbors which have been authorized, through ADD CONNECTION commands, to open a link via the station on which the GREETING-0 was received.

In addition to this "global" neighbor validation, which applies to all stations, each station has its own Neighbor Validation attribute which is set to TRUE upon receipt of an ADD CONNECTION command referencing that station. Thus, even if the node did not request neighbor validation (e.g., "VALIDATE=NONE", the default) in its initialization file, it could still place restrictions on individual stations.

The examples below will help to demonstrate the function of these two forms of Neighbor Validation, how they interact and how they may successfully be used to restrict connections on certain stations to only authorized neighbors.

Note that Neighbor Validation does not apply to global memory stations. That is, even if the command VALIDATE = ALL or "VALIDATE = NEIGHBOR" were entered, any global memory neighbor would be permitted to establish a connection on any global memory station. This means that the ADD CONNECTION command is never required for a global memory station. However, the ADD CONNECTION command can still be used to put neighbor restrictions on a global memory station.

In each of the following examples, assume the presence of the following configuration:

station name	station type
DIALIN	BDLC switched without ACU
DIALOUT	BDLC switched with ACU
LEASED	BDLC dedicated
host name	node address
CENTRAL	#1 (local node)
BRANCH1	#2
BRANCH2	#3

Examples:

1. When Neighbor Validation is FALSE, connections may generally be established with any neighbor by stations DIALIN, DIALOUT and LEASED. However, if an ESTABLISHCALL command is used to open a connection on DIALOUT or LEASED and that command has the phrase "WITH BRANCH2", the connection will succeed only if the node reached identifies itself

as BRANCH2 (node #3).

Example:

```
AC BY DIALIN;      % Will accept a call from any neighbor.
EC BY LEASED;     % Any neighbor may open other end of
                  % dedicated link.
EC WITH BRANCH1  % BRANCH1 must answer the phone.
  BY DIALOUT;
```

2. When Neighbor Validation is TRUE, a station may establish connections with only those neighbors which have been specially authorized. For example, if node BRANCH2 called station DIALIN, the call would be refused since no ADD CONNECTION command had been entered. However, if an ADD CONNECTION command for BRANCH2 and DIALIN were present in the initialization file (or entered manually at any time), then the call from BRANCH2 would be accepted.

Example:

```
ADD CONNECTION INCOMING WITH BRANCH2 BY STATION DIALIN;
ADD CONN IN WITH BRANCH1 BY DIALOUT;
```

Note that in the second command above, station DIALOUT, which is normally used to place outgoing calls, has been authorized to accept incoming calls from BRANCH1. This demonstrates the ability to issue both INCOMING (IN) and OUTGOING (OUT) connection commands for switched stations.

Neighbor Validation, as shown in this example, is most useful for authorizing only a select group of neighbors to call in on certain switched stations.

Neighbor Validation functions similarly for dedicated stations. For example, the following command would request BNA to allow connections between neighbor BRANCH1 and station LEASED.

```
ADD CONN PERMANENT WITH BRANCH1 BY LEASED;
```

Then, the following command would succeed if the neighbor on the other end of the dedicated connection identified itself as BRANCH1.

```
EC BY LEASED;
```

However, this installation could also omit any ADD CONNECTION commands from its initialization file and the following command would still be successful.

```
EC WITH BRANCH1 #2 BY LEASED;
```

An EC command with a neighbor phrase will succeed no matter what the value of Neighbor Validation and regardless of however many ADD CONNECTION commands have, or have not, been entered. Of course, the call must still pass neighbor validation at the neighbor node.

[It should be stressed that this discussion applies only to validation of the neighbor node address. Other attributes, such as network version, are also validated when a connection is established. These were described earlier in the section entitled "Validation Failure".]

An EC command placing an outgoing call must have a neighbor phrase. Since such a command overrides the neighbor validation feature, neighbor validation does not really apply to outgoing calls. Instead, BNA will always verify that the node who was called is the node who has answered. ADD CONNECTION commands are still useful, however, to store CALLDATA for outgoing calls. For example, the following two commands store telephone numbers by which the local node may reach BRANCH1 and BRANCH2 using station DIALOUT.

```
ADD CONN OUT WITH #2 BY DIALOUT
(CALLDATA="714-768-2313");
ADD CONN OUT WITH #3 BY DIALOUT
(CALLDATA="714-768-2314");
```

The following EC commands are sufficient to complete the connections described above.

```
EC WITH #2 BY DIALOUT;
EC WITH #3 BY DIALOUT;
```

Note that the clause "WITH <neighbor>" is required on an outgoing call both to specify which CALLDATA to retrieve and to ensure that the correct phone number was reached. In both of the examples above, host names are also acceptable and will generally be used for operational convenience. Node addresses were used only to illustrate that the neighbor specification may include host names and/or node addresses.

3. Assume that Neighbor Validation is FALSE and the following commands have been entered.

```
ADD CONN IN WITH BRANCH2 BY DIALIN;
ADD CONN PERM WITH BRANCH2 BY LEASED;
```

Although global neighbor validation has not been requested, neighbor validation has been requested on stations DIALIN and LEASED.

Example:

The following commands will put DIALIN and DIALOUT in a state where they are waiting for incoming calls.

AC BY DIALIN;
AC BY DIALOUT;

Station DIALOUT will accept an incoming call from any node. However, DIALIN will accept an incoming call only from BRANCH2. In addition, the following command will succeed only if the node reached identifies itself as BRANCH2.

EC BY LEASED;

The CLEARQUEUE command is used to bypass displaying of all messages currently queued for output to the network operator terminal.

DISCARD Command

```
-----
-- DISCARD --|
```

The DISCARD command causes a suspended load file to be discarded. A load file is suspended whenever a non-syntax error or a STOP command is encountered within the input stream of an input data file or if STOP is entered by the network operator initialization file loading process.

LOAD Command

```
-----
-- LOAD --<file title>-----|
          | - SYNTAX - |
```

The LOAD command initiates the loading of input commands from a data file. The input file must consist of a set of 80-character records. The first 72 characters of a record may contain operational commands, and the last eight characters (73-80) must contain a sequence number. Commands within the file must be separated by semicolons (;). SYNTAX specifies that the loaded file be syntax checked only and that no other action be taken on the input commands within the file.

The occurrence of a syntax error in the input file will cause the load to be aborted. Errors other than syntax errors (operational errors) will suspend the loading of the file until the load is continued or discarded by the operator.

The STOP command may be used to force suspension of the loading of an input file.

Examples:

```
LOAD BNACOMMANDS
LOAD BNA/OP/COMMANDS ON PACK
LOAD COMMANDS/FILE ON BNAPACK
```

MESSAGES Command

```
-----
---- MESSAGES ----|
      | - MSG -----|
```

The MESSAGES command displays a list of the most recent messages displayed to the Network Services operator terminal. As many messages are displayed as can be contained on one page of the output device.

NS Command

```
-----
-- NS --|
```

The NS command causes the next screen of a multi-screen display to be listed. Alternatively, CQ may be entered, ending the display, or another input command may be entered, cancelling the output display for the previous command.

RELEASE Command

```
-----
---- RELEASE ---- PRINTER ----|
      | - REL -----| | - P -----|
      | - D -----| | - D -----|
```

The RELEASE command causes an output file to be released.

Output which was diverted to a print file may accumulate in a printer backup file. When REL P is entered, the printer backup file is closed and is made available for printing.

When REL D is entered, the debugging output backup file (specified using the NO PRINT option) is closed and is made available for printing. The user must then enter "PB <job mix number>" to obtain the debugging output file.

Examples:

```
RELEASE P
REL D
```

STOP Command

```
-----
-- STOP --|
```

The STOP command causes the loading of an input data file to be suspended. This command may be entered by the operator or it may occur within the input stream of an input data file to checkpoint the file. When enacted, the following message is displayed:

```
FILE <file title> STOPPED AT LINE <line number>.
```

The network operator may then clear the suspended condition by entering either CONTINUE or DISCARD.

TEACH Command

```
----- TEACH -----|
| - HELP -- | |-<command verb>-|
|           | | - ALL -----|
```

TEACH by itself displays general information about Network Services input commands. TEACH <command verb> displays the syntax and semantics for the specified <command verb>. The following command verbs are available to the network operator:

ADD	AWAITCALL (AC)	CLEAR
CLEARCALL (CC)	CLEARQUEUE (CQ)	CLOSE
CONTINUE (CONT)	DCATTACH	DELETE (DEL)
DETACH (DET)	DISCARD	ENDINITIALIZATION (ENDINIT)
ESTABLISHCALL (EC)	HOST	HOSTUNREACHABLETIMEOUT (HUTO)
HOSTINACTIVEDISCONNECT (HID)	HOSTINACTIVETIMEOUT (HITO)	LINKRESISTANCE (LR)
LOAD	LOCALIDENTITY (LOC)	LOGGING (LOG)
MAXHOPCOUNT (MHC)	MAXRESISTANCEFACTOR (MRF)	MESSAGES (MSG)
MODIFY (MOD)	MONITOR (MON)	NEIGHBOR (NEI)
NEIGHBORBUSYTIMEOUT (NBTO)	NEIGHBORGREETINGTIMEOUT (NGTO)	NEIGHBORRESTARTTIMEOUT (NRTO)
NET	NETWORKMAXSEGMENTSIZ (NWMSS)	NETWORKOPTIONS (NO)
NETWORKVERSION (NWVER)	NODE	NODEPASSWORDS (NPW)
NODERESISTANCEFACTOR (NRF)	NODEUPTIMEOUT (NTO)	NS
OPEN	PHASE	PORTRESUMEREADY (PRR)
PORTRETRYLIMIT (PRL)	PORTSEGMENTTIMEOUT (PSTO)	PORTWINDOWSIZE (PWS)
READY	RELEASE (REL)	ROUTINGREFRESH (RREFR)
ROUTINGS (ROUT)	SAVE	SENDTEST
STARTTRACE (STARTTRA)	STATION (STA)	STOP
TEACH	TERM	TO
TUNINGS	VALIDATE (VAL)	VALIDATEANDATTACH (VALATT)
VERSION (VER)		

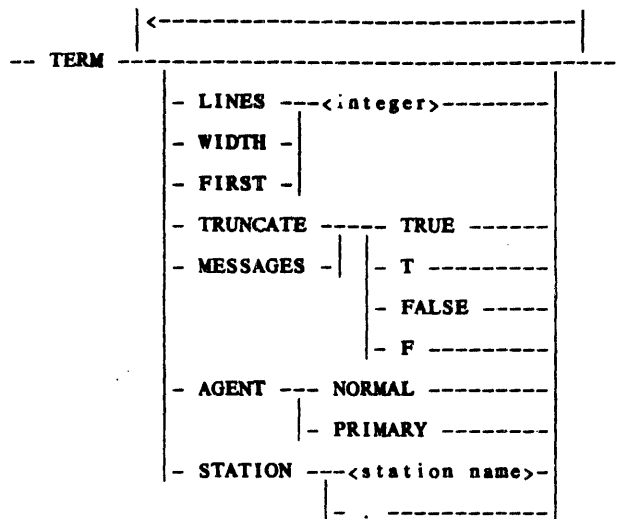
TEACH ALL displays the syntax and semantics of all <command verb>s, in alphabetical order.

HELP is synonymous with TEACH.

Examples:

```
HELP
TEACH CQ
TEACH TERM
TEACH ALL : PRINTER
```

TERM Command



The TERM command allows the operator to designate a remote station to be used as the network operator console and to control the format of the displays appearing on it. After any alteration, the new TERM settings are displayed.

TERM by itself causes the current terminal settings to be displayed.

LINES specifies the number of physical lines on the device. This option is valid for screen devices only. default = 24, maximum = 40, minimum = 5.

WIDTH specifies the number of characters on a physical line of the terminal. default = 80, maximum = 80, minimum = 20.

FIRST specifies the line number of the line on which each display is to begin. The <integer> specified may not exceed the number of lines (as established by default or by the "lines" option). This option is valid for screen devices only. default = 3, maximum = 40, minimum = 1.

TRUNCATE specifies whether or not lines of text are to be truncated (on the right) when they exceed the width of the line (as established by default or by the WIDTH option). If TRUNCATE is false, the text will be continued on the next line. The default is true.

MESSAGES specifies whether or not unsolicited messages are to be displayed automatically on the network operator console. default = true.

AGENT specifies that the remote station is to become the "primary" operator display terminal or is to maintain "normal" status as a peer of the system ODT. If "primary", unsolicited reports are displayed only to the remote station. If "normal", unsolicited reports are displayed to both the remote station and the system ODT. The default is "normal".

STATION designates the station to be used as the network operator display terminal. If no station is specified (i.e., "."), all communication to the network operator is through the system ODT. STATION <station name> causes the specified station to become the network operator console (the station may be either a screen or a hardcopy device). STATION. will close the active station, if any, and return operator communication to the system ODT. The default for this option is ".".

Examples:

```

TERM
TERM WIDTH 80 LINES 40 TRUNCATE TRUE STATION ADM102
TERM TRUNCATE FALSE
TERM STATION .

```

TO Command

```

-- TO ---<hostname>----- : --<text>--|
|
| - # <node address> -----|
| - <hostname> # <node address> -|

```

The TO command allows the operator to send text to an operator at another node. The destination node may be specified as a hostname, node address, or both.

Examples:

```
TO BLUE: NO TEST TIME TODAY
TO #10: NO TEST TIME TODAY
TO BLUE #10: NO TEST TIME TODAY
```

MAX RESISTANCE FACTOR Command

```
----- MRF -----|
| - MAXRESISTANCEFACTOR - | |----- <integer> - |
| - = - |
```

The MAXRESISTANCEFACTOR command allows the operator to either set or interrogate the max resistance factor attribute. When MRF is entered without any parameters, the attribute value is displayed. When an integer is specified, the attribute is set and the new value is displayed. The default value is 5000 and the valid range of values is 1 to 65535.

Examples:

```
MRF = 2000
MAXRESISTANCEFACTOR
MRF 500
```

MAX HOP COUNT Command

```
----- MHC -----|
| - MAXHOPCOUNT - | |----- <integer> - |
| - = - |
```

The MAXHOPCOUNT command allows the operator to either set or interrogate the max hop count attribute. When MHC is entered without any parameters, the attribute value is displayed. When an integer is specified, the attribute is set and the new value is displayed. The default value is 5 and the range of values is 1 to 255.

Examples:

```
MHC = 20
MAXHOPCOUNT
MHC 5
```

NODE RESISTANCE FACTOR Command

```
----- NRF -----|
| - NODERESISTANCEFACTOR - | |----- <integer> - |
| - = - |
```

The NODERESISTANCEFACTOR command allows the operator to either set or interrogate the node resistance factor attribute. When NRF is entered without any parameters, the attribute value is displayed. When an integer is specified, the attribute is set and the new value is displayed. The default value is 30 and the range of values is 1 to 65535.

Examples:

```
NRF = 50
NRF
NRF 75
```

NODE UP TIMEOUT Command

```
----- NTO -----|
| - NODEUPTIMEOUT - | |----- <time-interval> - |
| - = - |
```

<time interval> has the following syntax:

```

-----|
|-----| <seconds> -----|
|-----| <minutes>: -| |-----| - .<fraction> -|
|-----| - <hours>: -|

```

Where "<hours>", "<minutes>", "<seconds>" and "<fraction>" are all integers. ".<fraction>" is a fraction of a second (e.g., .75).

The NODEUPTIMEOUT command allows the operator to either set or interrogate the NODE UP TIMEOUT attribute. When NTO is entered without any parameters, the attribute value is displayed. When a <time-interval> is specified, the attribute is set and the new value is displayed. The default value is 5 seconds and the range of values is 0 to 60 seconds.

Examples:

```

NTO = 5
NODEUPTIMEOUT
NTO 7

```

NEIGHBOR RESTART TIMEOUT Command

```

-----|
|-----| NRTO -----|
|-----| - NEIGHBORRESTARTTIMEOUT -| |-----| <time-interval> -|
|-----| - = -|

```

<time interval> has the following syntax:

```

-----|
|-----| <seconds> -----|
|-----| <minutes>: -| |-----| - .<fraction> -|
|-----| - <hours>: -|

```

Where "<hours>", "<minutes>", "<seconds>", and "<fraction>" are all integers. ".<fraction>" is a fraction of a second (e.g., .75).

The NEIGHBORRESTARTTIMEOUT command allows the operator to either set or interrogate the NEIGHBOR RESTART TIMEOUT attribute. When NRTO is entered without any parameters, the attribute value is displayed. When a time-interval is specified, the attribute is set and the new value is displayed. The default value is 10 seconds and the range of values is 0 to 120 seconds.

Examples:

```

NRTO = 5
NEIGHBORRESTARTTIMEOUT
NRTO 7

```

LOCAL IDENTITY Command

```

-----|
|-----| LOC -----|
|-----| - LOCALIDENTITY -| | | |
|---|---|---|---|---|
|-----| # -- <node address> -|
|-----| - = -| |-----| -<hostname>-|

```

The LOCALIDENTITY command allows the operator to set or interrogate the identity of the local node. There is no default value for LOC and it must be set during initialization (before encountering any configuration commands). Inquiry may be done at any time. When LOC is entered without any parameters, the local hostname and node address will be displayed. When an address is specified, the local node address is set and the address is displayed. If a hostname is included, BNA will verify that it matches the local hostname. The hostname must be set, however, before BNA initialization by the HN ODT command.

Examples:

```
LOC = DETROIT #10
LOCALIDENTITY = #2
LOCALIDENTITY
```

VALIDATE Command

```
----- VAL -----|
| - VALIDATE - | |----- HOST -----| | |
| - - - - | | - NEIGHBOR -----|
| | | | - NEI -----|
| | | | - HOST, --- NEIGHBOR -|
| | | | | - NEI -----|
| | | | - NONE -----|
| | | | - ALL -----|
```

The VALIDATE command allows the operator to set or interrogate the VALIDATE options. If VAL is entered without any parameters, the current setting of the validation is displayed. If NONE is entered as a parameter, all validation is turned off. If ALL is entered as a parameter, all validation is turned on. The default value is NONE.

Examples:

```
VAL
VAL = HOST,NEI
VALIDATE ALL
VAL NEIGHBOR
```

END INITIALIZATION Command

```
----- ENDINIT -----|
| - ENDINITIALIZATION - |
```

The ENDINITIALIZATION command declares the end of the data entry phase of the node's initialization.

Examples:

```
ENDINIT
ENDINITIALIZATION
```

START TRACE Command

```
----- STARTTRA ----- <hostname> ----- TO ----->
| - STARTTRACE - | | - #<node address> -----|
| | | | - <hostname> #<node address> -|
>----- <hostname> -----|
| - #<node address> -----|
| - <hostname> #<node address> -|
```

The STARTTRACE command allows the operator to initiate a trace in the network.

Examples:

```
STARTTRA #10 TO #20
STARTTRACE BLUE TO RED #2
```

NETWORK OPTIONS Command

NO		
-	NETWORKOPTIONS	-
-	- + -	- DCPMONITOR ---
-	- - -	- DISPLAY -----
		- ERRORDP -----
		- FANCYBUG -----
		- FRAMEMONITOR -
		- NOSTATUS -----
		- PRINT -----
		- RAWBUG -----
		- ROUTERDEBUG --
		- SHOWINPUT ----
		- SLM -----
		- SLMCOMMAND ---
		- SLMDEBUG -----
		- SLMERROR -----
		- SLMMONITOR ---
		- SLMREPORT ----
		- SLMRESPONSE --
		- STAINFO -----
		- STATUSDEBUG --
		- TASKDEBUG ----

The NO command sets or resets a list of options which are used exclusively for systems debugging. During normal operation, all options should be in the reset state. The default for all options is reset.

The DCPMONITOR option writes error results from DCWRITES and logs received messages and commands in a print file.

The DISPLAY option causes debugging output to go to the ODT.

The ERRORDP option causes a programdump when an operator input message gets a syntax error.

The FANCYBUG option causes local host messages to be displayed in the same manner as the RAWBUG option, with additional interpretation.

The FRAMEMONITOR option displays summary information on the ODT for every frame sent and received by the local node. This information includes the origin node address, the destination node address and the frame length.

The NOSTATUS option inhibits the sending of status protocol request messages.

The PRINT option causes debugging output to go to the printer.

The RAWBUG option causes local host messages to be displayed on the printer or ODT, depending upon whether the PRINT option is set or reset.

The ROUTERDEBUG option will cause the display of control frames that are in error and display control frame functions as they are called.

The SHOWINPUT option causes remote host messages to be displayed on the printer or remote terminal/ODT, depending upon whether the PRINT option is set.

The SLM option is an abbreviation for the following debug options: SLMCOMMAND, SLMDEBUG, SLMERROR, SLMMONITOR, SLMREPORT, SLMRESPONSE. The abbreviation may be used to set, reset or interrogate all SLM debug options.

The SLMCOMMAND option causes information to be displayed whenever the station level manager (SLM) sends a command to a station. The display shows the station name, the LSN and the command.

The **SLMDEBUG** option causes extra debugging information to be displayed when certain SLM events occur. This option is most useful to show an operator the values of the fields received from a remote node in its station-level greetings. For example, if a remote node is failing validation due to a network-VERSION mismatch, the SLMDEBUG option may be set to show which network VERSION is in use at the remote node.

The **SLMERROR** option causes extra information to be displayed when certain errors are detected in the SLM. Usually, the SLM ignores non-fatal errors. If SLMERROR is set, it either displays more detailed information than it would otherwise, or it reports to the operator about an event it would usually ignore.

The **SLMMONITOR** option causes extra information to be displayed when certain events occur in the SLM. This option is most useful to monitor the transmission and receipt of station-level greetings. Reports are sent to the operator showing which station sent or received which greeting and the state of the station at the time the event occurred.

The **SLMREPORT** option causes information to be displayed whenever a station sends a report to the station level manager (SLM). The display shows the station name, the LSN and the report.

The **SLMRESPONSE** option causes information to be displayed whenever a station sends a response to the station level manager (SLM). All responses are sent in reply to SLM commands. The display shows the station name, the LSN, the response (positive or negative) and the reason for the response. The reason field is usually present only when a negative result is returned.

The **STAINFO** option is an abbreviation for the following three options: **SLMCOMMAND**, **SLMRESPONSE**, **SLMREPORT**. It is useful for monitoring station-level events.

The **STATUSDEBUG** option will cause the status protocol to display/print miscellaneous detailed information about the internal workings of itself if either the **DISPLAY** or **PRINT** option is set.

The **TASKDEBUG** option will cause the tasking protocol to display/print miscellaneous detailed information about the internal workings of itself if either the **DISPLAY** or **PRINT** option is set.

Examples:

```
NO
NO +
NO + RAWBUG SHOWINPUT PRINT FANCYBUG
NO - RAWBUG
```

SEND TEST Command

```
-----
-- SENDTEST --<station name>-----|
                                   |-<test-data>-|
```

<test-data> has the following syntax:

```

|<-----|
|<ebcdic letter>-----|
| |-<ebcdic number>-|
| |<-----|
| - " <ebcdic character> - " -
| |<-----|
| - HEX <hex digit>-----|
```

The **SENDTEST** command allows the operator to send a test frame across a data link. Prior to sending the frame, the connection between <station name> and another station must be opened. Optionally, a string may be specified to fill the data field in the frame. This command is used to establish confidence in the proper functioning of the data link in a maintenance environment.

Quotes within the text of <test-data> are represented by double quotes. The text must be short enough to fit in one segment; i.e., the length of the text cannot exceed the value of the **NETWORKMAXSEGMENTSZ** (NWMSS) attribute. Text entered in hexadecimal digits may be twice that long.

Examples:

```
SENDTEST GMM01 HELLO
SENDTEST BDLSTAIID HEX 1234567890ABCDEF
SENDTEST BDLCPERM01 "#$%&'((( )))"123 ABC"
SENDTEST DIALSTATION "HELLO THERE."
```

OPEN Command

```

-----
-- OPEN -- CPD -----<stationid>-----|
| - CALldata -----<calldata>- |
| - = - |
-----
-- OPEN -- SD --<station name>--|

```

The OPEN command allows the operator to manually open the connection port dialog or station dialog between a local station and its remote counterpart. The connection port dialog must be open before the station dialog can be opened and before the "SENDTEST" command can be used. If CALldata is supplied when opening the connection port on a BDLC station equipped with an automatic calling unit (ACU), an outgoing call is placed otherwise, the station will wait for an incoming call. This command is used to establish confidence in the proper functioning of the data link in a maintenance environment.

See the ADD CONNECTION command for a complete definition of <calldata>.

Examples:

```

OPEN CPD BDLCSTATION01
OPEN CPD BDLCWITHACU CALldata "768-2313"
OPEN SD BDLCSTATION01

```

CLOSE Command

```

-----
-- CLOSE --- CPD -----<station name> --|
| - CONNECTIONPORTDIALOG - |
| - SD ----- |
| - STATIONDIALOG ----- |
-----

```

The CLOSE command allows the operator to manually close the connection port dialog or station dialog between a local station and its remote counterpart. The station dialog must be closed before the connection port dialog can be closed. This command is used to establish confidence in the proper functioning of the data link in a maintenance environment.

The CLOSE SD command also detaches the station from the Router if it were previously attached.

Examples:

```

CLOSE SD GMM02
CLOSE CPD GMM02

```

VALIDATE AND ATTACH Command

```

-----
---- VALATT -----<station name>----|
| - VALIDATEANDATTACH - |
-----

```

The VALATT command allows an operator to manually initiate the validation process on a station. This consists of the exchange, with the remote station, of station-level greetings and the validation of those greetings. If validation is successful, the station is removed from manual mode, attached to the router and the link is made fully operational, as though an ESTABLISHCALL or AWAITCALL command had been entered. When the VALATT command is entered, the station must be in manual mode and the station dialog must be opened. This command is usually used following the OPEN CPD and OPEN SD commands. This command is used to establish confidence in the proper functioning of BNA in a maintenance environment.

Examples:

```

VALATT BDLCDEDICATED
VALIDATEANDATTACH BDLCSWITCHED01

```

DETACH Command

```

-----
-- DETACH -- <station name> --|

```

The DETACH command allows the operator to manually detach a station from the router. It also closes the station dialog. This command is used to establish confidence in the proper functioning of BNA in a maintenance environment.

Examples:

```
DET DED2400LINK
DETACH BDLCSWITCHED01
```

DCATTACH Command

```
--- DCATTACH --- <station name> ---|
```

The DCATTACH command allows the operator to manually attach to the BNA MCS a station which has been defined as a BNA station in the system NIF file. This command is used to establish confidence in the proper functioning of BNA in a maintenance environment.

Examples:

```
DCATTACH BDLCED
DCATTACH BDLCSWITCHED01
```

NETWORK VERSION Command

```
----- NWVER -----|
| - NETWORKVERSION - | |-----<version id>-|
| - - - - - | | - - - - - |
```

<versionid> has the following syntax:

```
| <- /16 \-----|
|-----<ebcdic letter or number>-----|
| | <- /16 \-----|
| - "-----<ebcdic character>-----" - |
```

Quotes within a string are represented by double quotes.

The NETWORKVERSION command sets the network version. All other hosts with which the local host is in communication must have the same network version. The default value is a null string. The value may be interrogated at any time. It may be set only during network initialization before any configuration commands are entered. Whenever NWVER is entered without parameters, the value is displayed. When a <versionid> is specified, the attribute is set and the new value is displayed.

Examples:

```
NWVER = NETVERSIONA
NETWORKVERSION "NETWORK VERSION A"
NWVER
```

NETWORK MAX SEGMENT SIZE Command

```
----- NWSS -----|
| - NETWORKMAXSEGMENTSIZ - | |-----<integer>-|
| - - - - - | | - - - - - |
```

The NETWORKMAXSEGMENTSIZ command sets the default max segment size for the network. The value may be interrogated at any time. It may be set only during network initialization before any configuration commands are entered. The default is 128 bytes, the maximum is 65535 bytes, and the minimum is 60 bytes. When NWSS is entered without any parameters, the value is displayed. When an integer is specified, the attribute is set and the new value is displayed.

Examples:

NWSS 256
 NETWORKMAXSEGMENTSIZ = 128
 NWSS

NEIGHBOR GREETING TIMEOUT Command

```

-----|
- NGTO -----|
- NEIGHBORGREETINGTIMEOUT - | |-----<time-interval>--|
- = - | |-----|

```

<time-interval> has the following syntax:

```

-----|
- <seconds> -----|
- <hours>: --- | | .<fraction> -|
- <minutes>: - | |-----|

```

Where "<hours>", "<minutes>", "<seconds>", and "<fraction>" are all integers. ".<fraction>" is a fraction of a second (e.g., .75).

The NEIGHBORGREETINGTIMEOUT command specifies the maximum length of time allowed for two stations to complete the greetings exchange when a connection is established. Failure to exchange greetings within the allotted time results in a greeting timeout condition and the connection is cleared. The default is 30 seconds, and the maximum is 60 seconds. The value must be greater than zero (0). When NGTO is entered without any parameters, the value is displayed. When a <time-interval> is specified, the attribute is set to the new value and the new value is displayed.

Examples:

```

NGTO = 5
NGTO
NEIGHBORGREETINGTIMEOUT 1:0.0

```

SAVE STATION Command

```

-----|
-- SAVE --- STA -----<station name>--|
- STATION - |

```

The SAVE STATION command does the following: if the station is connected, it marks the station to be saved upon disconnection; if the station is disconnected, it makes the station unavailable to be connected.

Examples:

```

SAVE STA GMM01
SAVE STATION BDLCSWITCHEDXX

```

SAVE HOST Command

```

-----|
-- SAVE -- HOST -- <hostname> -----|
- # -- <node address> - |

```

The SAVE HOST command marks the remote host as being unavailable for host communication as soon as the last subport to that host is closed.

Examples:

```

SAVE HOST BLUE
SAVE HOST RED #12

```

READY STATION Command

```

-----|
-- READY --- STA -----<station name>--|
- STATION - |

```

The **READY STATION** command does the following: if the station is connected, it removes the to-be-saved status; if the station is saved, it initializes the station as specified by the **AUTOINIT** attribute of the station.

Examples:

```
READY STA GMMO1
READY STATION BDLCSWITCHEDXX
```

READY HOST Command

```
-- READY -- HOST -- <hostname> -----|
                                   | - #<node address> - |
```

The **READY HOST** command removes the save status of the remote host. The remote host is now available for opening new subports.

Examples:

```
READY HOST BLUE
READY HOST RED #12
```

ESTABLISH CALL Command

```
----- EC ----- BY ----->
| - ESTABLISHCALL - | | - WITH --<neighbor>- | | - STA ----- |
| - STATION - |
>--<station name>-----|
| - ( -- CALLDATA -----<calldata>-- ) - |
| - = - |
```

<neighbor> has the following syntax:

```
-----<hostname>-----|
| -<hostname>-- # --<node address>- |
| - # --<node address>-----|
```

See the **ADD CONNECTION** command for a complete description of <calldata>.

the **ESTABLISHCALL** command is used to establish a new connection to a neighbor node. The station specified in the command may be a dedicated station or a switched station. The <neighbor> phrase is sometimes required, but may be included on all **ESTABLISHCALL** commands. If present, the neighbor node that answers the call will pass validation only if it is the specified neighbor.

When establishing a connection on a dedicated station or a switched station without an ACU, the neighbor clause is optional and "CALLDATA" (a phone number) is not permitted.

When establishing a connection on a BDLC switched station with an ACU, the neighbor clause is required. "CALLDATA" is required if it has not previously been defined, but is not allowed otherwise. See the **ADD CONNECTION** command for a description of how to store CALLDATA for specific neighbor/station pairs.

When establishing a connection between manual switched stations (switched stations with neither auto-dial nor auto-answer capability) at least one of the stations must be opened using the **ESTABLISHCALL** command. When an incoming connection is opened on a switched station, the station will wait for the other node to send its greeting-1 message first. If both nodes enter "AWAITCALL" commands and the call is made manually, an impasse will occur with each node waiting for the other to send a greeting.

See the **AWAITCALL** command for a description of opening an incoming (dial-in) connection on a switched station.

Examples:

ESTABLISHCALL BY STATION BDLCDDEDICATED
 EC WITH HEADQUARTERS BY BDLCDDEDICATED
 EC WITH BRANCH2 BY BDLCWITHACU (CALLDATA="9*768-2313")
 EC WITH DOWNTOWN BY DIALOUT01

CLEAR CALL Command

```

-----
CC ----- WITH --<neighbor>-----|
| - CLEARCALL - | | - BY -----<station name>-|
|               | | - STA -----|
|               | | - STATION - |
-----

```

<neighbor> has the following syntax:

```

-----<hostname>-----|
| -<hostname>-- # --<nodeaddress>-|
| - # --<nodeaddress>-----|
-----

```

The CLEARCALL command breaks one or more links to a neighbor. If a <neighbor> is specified, all stations to that neighbor are disconnected. If a station is specified, only the specified station is disconnected.

Examples:

```

CC WITH BRANCH2
CLEARCALL WITH BRANCH3
CC BY GMM01
CC BY STATION BDLCDDEDICATED

```

AWAIT CALL Command

```

----- AC ----- BY -----<station name>---|
| - AWAITCALL - | | - STA -----|
|               | | - STATION - |
-----

```

The AWAITCALL command is a variant of the ESTABLISHCALL command in which the intent is to establish connections on incoming dialed lines. The station is put into a mode in which it is awaiting a call from a potential neighbor.

Examples:

```

AC BY BDLCSWITCHEDLINE01
AWAITCALL BY STA BDLCSWITCHED02

```

ADD HOST Command

```

-- ADD -- HOST --<hostname>-- # --<node address>--|
-----

```

the ADD HOST command is used to define the name of a host computer in a BNA network and to equate it to a BNA node address. If "VALIDATE HOST" is set, the specified host is added as a valid remote host. If "VALIDATE HOST" is not set, the host is added only for the purpose of translating between <hostname> and <node address>. This allows all operator commands to use only hostnames when referring to other nodes. If ROUTER VALIDATE is set (VALIDATE option was set to ALL), the node is added to the list of nodes to which routing is allowed.

A hostname is composed of from 1 to 17 alphanumeric characters. A node address is an integer with a value from 1 to 1023, inclusive. To minimize the size of BNA tables, node addresses should be assigned sequentially, starting with 1.

Examples:

```

ADD HOST BRANCH2 #10
ADD HOST HOUSTON #255

```

ADD NODE Command

```

-- ADD -- NODE ----- # --<node address>--|
      |-----|
      |-<hostname>-|

```

The ADD NODE command is used to define the address of a node in a BNA network and to equate it to a BNA host name. If "VALIDATE ALL" is set, the specified node address is added as a valid remote node to which routing is allowed. In addition, if a <hostname> is supplied, the <hostname>/<node address> pair is stored for translation purposes. This allows all operator commands to use only hostnames when referring to other nodes.

A hostname is composed of from 1 to 17 alphanumeric characters. A node address is an integer with a value from 1 to 1023, inclusive. To minimize the size of BNA tables, node addresses should be assigned sequentially, starting with 1.

Examples:

```

ADD NODE BRANCH2 #10
ADD NODE #255

```

DELETE HOST Command

```

---- DEL ----- HOST --<hostname>-----|
      |-----|
      |- DELETE -|
      |-----|
      |- # --<nodeaddress>-|

```

The DELETE HOST command has meaning only if VALIDATE HOST is set, in which case the indicated host is removed as a valid remote host. In any case, the <hostname> to <nodeaddress> pairing is retained for translation purposes.

Examples:

```

DEL HOST BRANCH2 #10
DELETE HOST HOUSTON
DELETE HOST HOUSTON #255

```

DELETE NODE Command

```

---- DEL ----- NODE ----- # --<nodeaddress>--|
      |-----|
      |- DELETE -|
      |-----|
      |-<hostname>-|

```

The DELETE NODE command deletes the specified node. The hostname and node address are also deleted from the list of hosts.

Examples:

```

DEL NODE BRANCH2 #10
DELETE NODE #255
DELETE NODE HOUSTON #255

```

NODE Command

```

-- NODE --|

```

The NODE command displays the reachability of all known valid nodes in the network.

Example:

```

NODE

```

ROUTINGS Command

```

---- ROUT -----|
      |-----|
      |- ROUTINGS -| | - INUSE -| | - TO ----- <hostname> -----|
      |-----| |-----| |-----|
      |- VIA -| | - # -- <node address> -|
      |-----| |-----|
      |- *ALL -----|

```

The ROUTINGS command displays detailed routing information. ROUT or ROUT TO ALL returns all routings for every destination node, arranged first by destination node and then, for each destination node, all the neighbor nodes potentially usable to reach that destination. ROUT TO <hostname> or ROUT TO #<node address> returns all routings for the specified destination node, displaying each neighbor node which is potentially usable to reach that destination. ROUT INUSE or ROUT INUSE TO ALL returns in-use routing for every destination node, showing the neighbor node currently used to reach each destination. ROUT INUSE TO <hostname> or ROUT INUSE TO #<node address> returns in-use routing for the specified destination. ROUT VIA ALL returns all routings for every neighbor node, showing for each neighbor a list of all destination nodes potentially reachable through that neighbor. ROUT VIA <hostname> or ROUT VIA #<node address> returns all routings for the specified neighbor node, showing a list of all destination nodes potentially reachable through that neighbor. ROUT INUSE VIA ALL returns in-use routing information for all neighbor nodes, showing the destinations currently reachable through each neighbor. ROUT INUSE VIA <hostname> or ROUT INUSE VIA #<node address> returns in-use routing information for the specified neighbor node, showing the destinations currently reachable through that neighbor.

Examples:

```
ROUT INUSE TO HOUSTON
ROUTINGS VIA #123
ROUT INUSE
ROUT TO *ALL
```

NET Command

```
-- NET -- - - - - - - - - - - |
| - NOW - |
```

The NET command causes a shutdown on network services. NET - is the only form of the command available through the network services operations interface. NET + and NET = extensions are available through the normal systems operations interface.

NET - causes a slow shutdown of network services. During a slow shutdown, no port files may be opened to remote hosts. As soon as the last open port file is closed, network services will be terminated at the local node.

NET - NOW causes a fast shutdown of network services. All open port files are closed immediately.

Examples:

```
NET -
NET - NOW
```

PORT SEGMENT TIMEOUT Command

```
--- PSTO -----|
| - PORTSEGMENTTIMEOUT - | |----- <time interval> -|
| - - - |
```

<time interval> has the following syntax:

```
-----|
|----- <seconds> -----|
|----- <minutes>: - | | - .<fraction> - |
| - <hours>: - |
```

Where "<hours>", "<minutes>", "<seconds>" and "<fraction>" are all integers. ".<fraction>" is a fraction of a second (e.g., .7).

PORTSEGMENTTIMEOUT is used by the port level to determine how long to wait for a segment acknowledgment before resending a segment. If no time interval is specified, the current value is displayed. The default is 100 seconds and the range is 1 second to 18 hours 19 minutes and 15 seconds.

Examples:

PORTSEGMENTTIMEOUT
PSTO = 30

PORT WINDOW SIZE Command

```

----- PWS -----
| - PORTWINDOWSIZE - | |----- <integer> -|
| - = - | | - - -|
    
```

PORTWINDOWSIZE is used by the port level to determine how many unacknowledged segments may be outstanding at a time. If no integer is specified, the current value is displayed. The default value is 20 and the range is 1 to 25.

Examples:

PORTWINDOWSIZE
PWS = 5

PORT RETRY LIMIT Command

```

----- PRR -----
| - PORTRETRYLIMIT - | |----- <integer> -|
| - = - | | - - -|
    
```

PORTRETRYLIMIT is used by the port level to determine how many times a segment should be retransmitted. When the retry limit is exceeded, the effected subport will be deactivated. If no integer is specified, the current value is displayed. The default value is 2 and the range is 0 to 100.

Examples:

PORTRETRYLIMIT
PRR = 5

PORT RESUME READY Command

```

----- PRU -----
| - PORTRESUMEREADY - | |----- <integer> -|
| - = - | | - - -|
    
```

When MAXCENSUS is reached for a subport, the subport is made "NOT READY". PORTRESUMEREADY represents the percentage of MAXCENSUS at which the subport will be made "READY". If no integer is specified, the current value is displayed. The default value is 60 and the range is 1 to 99.

Examples:

PORTRESUMEREADY
PRU = 5

HOST INACTIVE DISCONNECT Command

```

----- HID -----
| - HOSTINACTIVEDISCONNECT - | |----- + -|
| - = - | | - - -|
    
```

The HOSTINACTIVEDISCONNECT command activates the HOST INACTIVE TIMEOUT. (See the HOST INACTIVE TIMEOUT for details.) If entered without any parameters, the current value is displayed. The default value is reset.

Examples:

```
HOSTINACTIVEDISCONNECT
HID = -
```

HOST Command

```
-----
-- HOST -----|
      | - <hostname> - |
```

The HOST command returns a list of hosts with summary information about each host. If <hostname> is provided, a detailed description of host <hostname> is displayed.

Examples:

```
HOST
HOST BLUE
```

ROUTING REFRESH Command

```
-----
--- RREFR -----|
      | - ROUTINGREFRESH - |
```

The ROUTINGREFRESH command causes the router to request all its neighbors to send current routing information, from which the local routing tables are updated.

Examples:

```
RREFR
ROUTINGREFRESH
```

TUNINGS Command

```
-----
-- TUNINGS --|
```

The TUNINGS command displays a collection of port and router attributes that may be used to tune the performance of the local node.

Example:

```
TUNINGS
```

MODIFY STATION Command

```
-----
--- MOD ----- STA -----<station name>-- ( ----->
      | - MODIFY - | | - STATION - |
      | <----- , ----- |
      | >-----<attrib assign>----- ) -----|
```

Where <attrib assign> is:

AUTOINIT	-----	TRUE	-----
	- = -	- T	-----
		- FALSE	-----
		- F	-----
- EFFICIENCY	-----	<integer>	-----
- EFF	-----	- = -	
- SPEED	-----	<integer>	-----
	- = -		
- MAXSEGMENTSIZ	-----	<integer>	-----
- MAXSS	-----	- = -	
- MONITOR	-----	ON	-----
- MON	-----	- = -	- OFF

The **MODIFY STATION** command modifies certain attributes of a station. This command is a configurational command and may be entered at any time.

The **AUTOINIT** attribute specifies whether a station should automatically be opened during network initialization and re-opened (under certain conditions) when the station is closed. The exact effect of **AUTOINIT** depends on the type of station.

For dedicated stations, when **AUTOINIT** is true, BNA will generally attempt to keep the station open at all times. The equivalent of an **ESTABLISHCALL** command will be performed on the station at initialization and also whenever the station is closed by the remote node. After a call is cleared by the local operator, however, the station will remain closed.

For switched stations, when **AUTOINIT** is true, BNA will attempt to keep the station in await-call mode at all times. The equivalent of an **AWAITCALL** command will be performed on the station at initialization and also whenever the station is closed. A **CLEARCALL** command from the operator will close the station, but it will immediately be set up to receive another incoming call.

The default value for **AUTOINIT** is true. **AUTOINIT** may be set or reset at any time. The **AUTOINIT** function may also be temporarily suspended by saving the station (see the **SAVE STATION** command); i.e., if a saved station is closed for any reason, it will remain closed. The **AUTOINIT** function will be resumed when the station is made ready.

The **EFFICIENCY** attribute specifies the relative efficiency of the link in per cent. The value may range from 1 to 99. The default value depends on the type of station:

GLOBAL MEMORY	98
BDLC DEDICATED	95
BDLC SWITCHED	45

Efficiency may be set only when a station is closed.

The **SPEED** attribute specifies the speed of the link in bits-per-second. The value may range from 1 to 1,500,000. The default value depends on the type of station:

GLOBAL MEMORY	100000
BDLC DEDICATED	9600
BDLC SWITCHED	9600

Speed may be set only when a station is closed.

The exact values of speed and efficiency are not crucial to the correct functioning of BNA. However, the values of these two attributes are used by the router to calculate the most efficient route to another BNA node. Also, if several links are open to a neighbor node, the volume of traffic sent on each will be proportional to the product of that station's speed and efficiency; i.e., a station with the same efficiency as another, but twice the speed, will be used to send twice the volume of data.

The **MAXSEGMENTSIZ** attribute specifies the largest message, in bytes, which can be sent via the station. The value may range from 60 to 65535. The default value is the value of the **NETWORKMAXSEGMENTSIZ** (NMSS) attribute.

MAXSEGMENTSIZ may be set only when a station is closed.

The MONITOR attribute controls the monitoring function for a station. BNA maintains certain operational statistics for each station including a count of frames sent, frames received and frames in error. When MONITOR is set to "ON", these statistics will periodically be written to the system log file and reset to zero (0). The frequency of this logging function is controlled by the value of the station monitor interval. See the MONITOR command for more details.

When MONITOR is reset to "OFF", the station monitor counts are still accumulated, but are not logged or reset. The default value for the MONITOR attribute is "OFF". When monitoring is off, the accumulated counts may be displayed through use of the COUNTS option of the station interrogation command. See the STATION command for more details.

MONITOR may be set at any time.

Examples:

```
MOD STA GMM01(EFFICIENCY=96,SPEED=500000)
MOD STA BDLCDIALOUT(AUTOINIT F,EFF 95,MAXSS 128)
MODIFY STATION BDLCSWITCHED(AUTOINIT=FALSE,MONITOR=ON)
```

CANDIDATE Command

```
----- CA -----|
| - CANDIDATE - | | - <portname> - |
```

The CANDIDATE command displays all candidates using the portname <portname>.

CANDIDATEs are offered subports that have not yet been matched.

Examples:

```
CA PORTONE
CANDIDATE MYPORT
```

PHASE Command

```
-- PHASE --|
```

The PHASE command displays the current phase of the node.

Example:

```
PHASE
```

NEIGHBOR Command

```
----- NEI -----|
| - NEIGHBOR - | | - *ACT -----|
|               | | - *ACTIVE -----|
|               | | - <neighbor> --- CONN -----|
|               | | - CONNECTION - |
|               | | - DETAIL -----|
```

NEIGHBOR or NEIGHBOR *ACTIVE returns a list of neighbors and summary information about each. (*ACTIVE specifies only active neighbors. The unqualified form specifies all neighbors, potential and active.) NEIGHBOR <neighbor> CONNECTION returns connection information with the specified neighbor. NEIGHBOR <neighbor> DETAIL returns detailed information about the specified neighbor.

Examples:

```
NEIGHBOR *ACTIVE
NEI BLUE CONN
NEIGHBOR #123 DETAIL
```

CLEAR HOST Command

```

-- CLEAR -- HOST -- <hostname> -----|
                                |- #<node address> -|
    
```

The CLEAR HOST command immediately severs all current communication to a host. New supports may be opened if a SAVE HOST command was not entered.

Examples:

```

CLEAR HOST BLUE
CLEAR HOST RED #12
    
```

ADD CONNECTION Command

```

-- ADD -- CONN ----- IN ----- WITH --<neighbor>-- BY --
          |- CONNECTION -| |- INCOMING --|
          |- OUT -----|
          |- OUTGOING --|
          |- PERM -----|
          |- PERMANENT -|

>-----<station name>-----|
|- STA -----| |- ( --<callinfo>-- ) -|
|- STATION -|
    
```

<callinfo> has the following syntax:

```

-- CALldata -----<calldata>----->
          |- = -|

>-----|
|- , -- INITQUANT -----<integer>-|
|- INITQUANTITY -| |- = -|
    
```

<calldata> has the following syntax:

```

|<-/23\-----|
-- " -----<decimal digit>----- " --|
  |- * -----|
  |- # -----|
  |- -----|
  |- S -----|
  |- E -----|
    
```

<neighbor> has the following syntax:

```

-----<hostname>-----|
|-<hostname>-- # --<node address>-|
|- # --<node address>-----|
    
```

CALLDATA represents a phone number used to call another node. The asterisk (*) and crosshatch (#) characters represent the equivalent characters on a pushbutton phone. The character "S" is used to request a one-second delay when dialing. This feature is useful when it is necessary to wait for a secondary dial-tone after, for example, dialing a preliminary access code. The character "E" is used to represent the end of the call data. It is optional. If present, it must be the last character in the string. Hyphens are solely an operational convenience to

make the CALldata more readable. CALldata may not be more than 23 characters in length, but this limit does not include hyphens.

The ADD CONNECTION command is used to define potential connections between the stations of the local node and its neighbors. The command does not, however, directly open any actual connections.

There are three directions, or types, of connections -- permanent, incoming and outgoing. Permanent connections apply only to dedicated type stations; incoming and outgoing connections apply to BDLC switched stations. Both incoming and outgoing connections may be defined for a switched station.

Each station in a node and each neighbor of the node may be referenced in multiple ADD CONNECTION commands. Each such command defines a valid station/neighbor combination. No station/neighbor pair, however, may appear in more than one connection for a particular direction.

ADD CONNECTION commands for permanent and incoming connections are similar in effect. If a station has connections defined with certain nodes, then only those nodes will be permitted to establish connections via that station. If the station has no connections defined, then no such restrictions will be present. Incoming connections apply only to incoming calls received on a switched station. A call is considered to be incoming if the await-call command was issued for the station. A call is considered outgoing if the ESTABLISHCALL command was used.

ADD CONNECTION commands for outgoing connections are somewhat different. For switched stations without an ACU, their function is similar to an incoming connection, but apply only to outgoing calls. For switched stations with an ACU they primarily provide a means of storing a telephone number used to call a particular neighbor node. They can also be used to request that, at initialization time, a call automatically be placed to that node, by the specified station. These two features are available through the use of the attributes CALldata and INITQUANTITY. Both attributes apply only to outgoing connections for stations with an ACU.

CALldata, which is required, is a phone number.

INITQUANTITY, which is optional, represents the number of calls that should be placed, at initialization time, by the specified station. Its only valid values are zero (0) and one (1); the default value is zero. When INITQUANTITY is one, the call described in the ADD CONNECTION command will be placed automatically. No more than one outgoing connection for a particular station may have INITQUANTITY equal to one. When INITQUANTITY is zero, no call will be placed at initialization time. An operator may later enter an ESTABLISHCALL command and the correct phone number will be retrieved from the information supplied in the ADD CONNECTION command.

Note that if neighbor validation has been requested at initialization time then only connections defined in ADD CONNECTION commands may be established. If no connections have been defined, all attempts to open connections will fail. However, if a connection is opened with an ESTABLISHCALL command, the optional neighbor clause, if present, takes precedence over the information in any ADD CONNECTION commands. As long as the neighbor that establishes contact matches the neighbor in the ESTABLISHCALL command, the status of neighbor validation and connection table are ignored. For more information on neighbor validation, see the VALIDATE command. For more information on opening connections, see the ESTABLISHCALL and AWAITCALL commands.

Examples:

```
ADD CONNECTION INCOMING WITH HOUSTON BY STATION BDLCdIALIN
ADD CONN OUT WITH NY BY BDLCWITHACU (CALldata="212-444-1215")
ADD CONN OUT WITH LA BY DIALOUT(CALldata "4441215",INITQUANT 1)
ADD CONN OUT WITH SF #4 BY BDLCWITHOUTACU
ADD CONN PERM WITH #5 BY STA BDLCDEDICATED
```

DELETE CONNECTION Command

```
-----
DEL ----- CONN ----- IN ----->
| - DELETE - | | - CONNECTION - | | - INCOMING -- |
|           | |           | | - OUT ----- |
|           | |           | | - OUTGOING -- |
|           | |           | | - PERM ----- |
|           | |           | | - PERMANENT - |
|           | |           |
|-----> BY <station name>-----|
| - WITH --<neighbor>- | | - STA ----- |
|           | | - STATION - |
```

```

<neighbor>
-----|
|<hostname>-----|
| | - # --<node address>- |
| | - # --<node address>- |
|-----|

```

The DELETE CONNECTION command deletes all connections by a station or a specific connection with a <neighbor>, provided none of the connections are active.

Examples:

```

DELETE CONN INCOMING WITH HOUSTON BY STATION BDLCSWITCHED
DELETE CONNECTION OUT BY STATION BDLCSWITCHED
DEL CONN PERM WITH BRANCH1 BY BDLCDEDICATED15

```

NODE PASSWORDS Command

```

----- NPW -----<neighbor>----->
| - NODEPASSWORDS - |
|-----|
|<-----|
|> /1\ MNPW ----- = *NULL -----|
| | - MYNODEPASSWORD - | | -<password>- |
| | - /1\ YNPW ----- = *NULL ----- |
| | - YOURNODEPASSWORD - | | -<password>- |

```

<neighbor> has the following syntax:

```

-----|
|<hostname>-----|
|<hostname>-- # --<node address>-|
| - # --<node address>- |
|-----|

```

<password> has the following syntax:

```

|<- /16\-----|
|-----|
| |<ebcdic letter>-----|
| | -<ebcdic number>- |
| |<- /16\-----|
| - " <ebcdic character> - " - |
| |<- /33\-----|
| - HEX ---<hex digit>- |

```

Quotes within the text of <password> are represented by double quotes.

The NODEPASSWORDS (NPW) command is used to define the passwords that nodes send to each other. The MYNODEPASSWORD (MNPW) clause is used to define the password that the local node will send to the specified neighbor node. The YOURNODEPASSWORD (YNPW) clause is used to define the password that the local node should expect to receive from the neighbor node. Setting either password to *NULL will delete any previously defined password. If MNPW is NULL, the local node will not send any password to the neighbor. If YNPW is NULL, the local node will not expect any password from the neighbor but will accept any password that is sent. If either the MNPW or YNPW clause is omitted, the corresponding password will remain unchanged from its previous value. The initial (and default) value for all passwords is NULL.

Passwords may be no more than 17 EBCDIC characters, or 34 hexadecimal digits, in length. If a password is defined to be an odd number of hexadecimal digits, it will be padded with a hex zero (0) to make it an integral number of bytes.

Examples:

```
NPW #3 MNPW = YYY
NPW YELLOW MNPW = *NULL YNPW = HEX 99AABB
NPW BLUE #2 YOURNODEPASSWORD = NULL MNPW = 'HEX'
NODEPASSWORDS #3 MYNODEPASSWORD = '<+, ' YNPW=1234
```

STATION Command

```
----- STA -----|
| - STATION - | | -<station name>-----|
|                                     |
|                                     | - STATUS - |
|                                     | - COUNTS - |
|                                     | - ALL ---- |
```

The STATION summary command is an interrogation command used to display information about the stations at the local node.

If no station name is supplied, the report will show summary information about all stations in the station list. This information includes the state of the station and, if the station is connected to a neighbor, the remote node's name and address.

If a station name, but no option, is specified, a detailed list of that station's attributes is sent to the operator.

If the STATUS option is requested, status information maintained by the station-group is reported.

If the COUNTS option is requested, certain statistics maintained by BNA for the station are reported. These statistics include the number of errors detected and the number of frames sent and received by the station. The station counts will not be displayed while the monitoring function is enabled for the station. See the "MONITOR" and "MODIFY STATION" commands for more information.

If the ALL option is requested, the information for all three reports is sent to the operator.

Examples:

```
STA
STATION BDLCSWITCHED01
STA GMM01 COUNTS
STA GMM01 ALL
```

MODIFY CONNECTION Command

```
----- MOD ----- CONN ----- OUT ----- WITH --<neighbor>-->
| - MODIFY - | | - CONNECTION - | | - OUTGOING - |
>- BY -----<station name>-----|
| - STA -----| | - ( --<callinfo>-- ) - |
| - STATION - |
```

<callinfo> has the following syntax:

```
-- CALldata -----<calldata>----->
| - - - |
>-----|
| - , --- INITQUANT -----<integer>- |
| - INITQUANTITY - | | - - - |
```

<calldata> is explained fully under the ADD CONNECTION command.

<neighbor> has the following syntax:

```

-----<hostname>-----|
|<hostname>-- # --<node address>--|
| # --<node address>-----|
    
```

The MODIFY CONNECTION command is used to change the attributes of a previously defined connection (see the ADD CONNECTION command). The only connection attributes currently supported are CALldata and INITQUANTITY. Both of these apply solely to outgoing connections defined for switched stations equipped with an ACU.

When CALldata is changed, the phone number stored in a connection table entry is changed. If an ESTABLISHCALL command is subsequently received for the specified station/neighbor pair, the new phone number will be used.

When INITQUANTITY is changed, the number of calls placed via the specified station at initialization time is changed. This value can currently be either one (1) or zero (0).

The MODIFY CONNECTION command is a configurational command. It may be entered any time after the local node address has been set.

Examples:

```

MODIFY CONNECTION OUT WITH LA BY DIALOUT(CALLDATA="1-444-1215")
MOD CONN OUT WITH BRANCH1 BY STA DIALSTATION(INITQUANT=1)
MOD CONN OUT WITH NY BY DIALOUT(CALLDATA "7682313",INITQUANT 0)
    
```

LOGGING Command

```

----- LOG -----|
| LOGGING - | |-----|
| + | |-----|
| | | MINIMUM - |
| | | MIN -----|
| | | MAXIMUM - |
| | | MAX -----|
    
```

The LOG command allows the operator to set or interrogate the LOGGING option for the local node. When "LOG" is entered without parameters, the current setting of the LOGGING option is displayed. When a parameter is provided, the option is set and the new setting is displayed. The "LOG-" form will set the LOGGING option to none. The "LOG+" form will set the LOGGING option to the standard set of events. The "LOG + MAXIMUM" and "LOG + MINIMUM" forms will set the LOGGING option to the maximum (all) and minimum sets, respectively.

Examples:

```

LOGGING
LOG +
LOG -
LOG + MINIMUM
    
```

NEIGHBOR BUSY TIMEOUT Command

```

----- NBTO -----|
| NEIGHBORBUSYTIMEOUT - | |-----<time-interval>-----|
| - = - | |-----|
    
```

<time-interval> has the following syntax:

```

----- <seconds> -----|
|-----<minutes>: - | |-----<fraction> - |
|-----<hours>: - |
    
```

Where "<hours>", "<minutes>", "<seconds>", and "<fraction>" are all integers. ".<fraction>" is a fraction of a second (e.g., .75).

The NBTO command is used to set or interrogate the value of the NEIGHBOR BUSY TIMEOUT attribute. This attribute specifies the length of time a neighbor node is allowed to remain busy before the local node clears all its connections with that neighbor.

When a BNA station is temporarily unable to accept frames, it sends to the remote station a command called RECEIVE-NOT-READY. At the remote node, that station is marked as busy, and is not used until receipt of a RECEIVE-READY command. If the remote node detects that all stations connected to a neighbor are busy, the neighbor is marked as busy and a timer is started. During the busy condition, frames which are presented to the stations for transmission are queued. When at least one of the stations again becomes ready, the timer is cancelled and the queued frames are transmitted to the neighbor node.

If the neighbor-busy timer expires before any of the stations become ready, all connections with that neighbor will be cleared as though a CLEARCALL command had been entered. Any frames which were queued at the stations will be sent back to the ROUTER for re-transmission by an alternate path. If no alternate path to the destination node exists, the frames will be discarded.

Use of the NBTO command allows a user to vary the duration of the neighbor-busy timer. The NBTO attribute is maintained in seconds. The value must be greater than 0 and no greater than 300 (5 minutes). The default value is 60 seconds. NBTO may be set or interrogated at any time. If a valid time interval is included in the command, the attribute is set to that value. If no time interval is included, the command is treated as an interrogation. The value of this attribute should be chosen with care. At all times, NBTO should be less than the product of the PORT RETRY LIMIT and PORT SEGMENT TIMEOUT attributes; otherwise, ports may be closed prematurely on segment timeout conditions. If the value chosen is too large, some traffic on the network may be interrupted for an excessive length of time. If the value chosen is too small, connections may be terminated too frequently for simple buffer-full conditions at other nodes.

Examples:

```
NBTO = 5
NBTO
NEIGHBORBUSYTIMEOUT 1:0:0
```

HOST UNREACHABLE TIMEOUT Command

```
----- HUTO -----|
| - HOSTUNREACHABLETIMEOUT - | |-----<time interval>-|
|                               | |-----|
|                               | |-----|
```

<time interval> has the following syntax:

```
----- <seconds> -----|
|-----<minutes>: -| |----- .<fraction> -|
| -<hours>: -|
```

Where "<hours>", "<minutes>", "<seconds>" and "<fraction>" are all integers. ".<fraction>" is a fraction of a second (e.g., .7).

The HUTO command sets the PORT MAX BLOCKED TIME attribute. This is the maximum length of time that a port file may be in the blocked state. If this time is exceeded, the port will be deactivated. If no time interval is specified the current value will be displayed. Default value is 30 minutes. Valid range of values is 1 minute through 1440 minutes (1 day).

Examples:

```
HOSTUNREACHABLETIMEOUT
HUTO = 120
```

HOST INACTIVE TIMEOUT Command

```
----- HITO -----|
| - HOSTINACTIVETIMEOUT - | |-----<time interval>-|
|                               | |-----|
|                               | |-----|
```



```

----- LR -----<neighbor>-----<integer>-----|
| - LINKRESISTANCE - |           | - *CALC --- |

```

The LR command sets the operations resistance factor for the specified neighbor in the router's neighbor table. If *CALC is specified the calculated resistance factor is used for the link resistance factor to the specified neighbor.

Examples:

```

LR #5 *CALC
LINKRESISTANCE BLUE #25

```

VERSION Command

```

----- VER -----|
| - VERSION - |

```

The VERSION command is used to request a report of the current network versions and protocol levels in use at the local node. The response to the command will show the current version for the station, router and port levels. It will also show which prior levels of the software the current level can support.

Examples:

```

VERSION
VER

```

NDL Considerations

The BDLC station group allows host-to-host communication on a BNA network via full-duplex leased lines or half-duplex switched lines. The Burroughs Data Link Control (BDLC) protocol is utilized to communicate between the neighbors. BDLC is a bit-oriented transparent link protocol, similar to CCITT's recommendation X.25 level 2 procedures. An Adaptor Cluster III (ACIII) is required in order to interface the physical lines to the DCP. The ACIII automatically handles zero-bit insertion, frame-check sequence calculation, and similar tasks at the front-end relieving the DCP and the operating system of these time-consuming functions.

BDLC station group functions are shared between the DCP and the operating system. All BNA BDLC stations must be defined in the NDL. The network operator input commands assume that any station names used have been previously defined in the NDL. BNA BDLC stations must have SYSTEM/BNAMCS as their MCS. A typical full-duplex station definition may be defined as follows:

STATION BDLCDEDICATED:

```

DEFAULT = STADEF.
TERMINAL = BDLCTERMINAL.
ADAPTER = 32.
MODEM = SUPERMODEM.
MCS = SYSTEM/BNAMCS.
ADDRESS = (4*0103", 4*0103").
CONTROL = QM.
RETRY = 15.

```

Typical half-duplex station definitions may be defined as follows:

STATION BDLCSWITCHEDWITHACU:

```

DEFAULT = STADEF.
TERMINAL = HBDLCTERMINAL.
ADAPTER = 32.
MODEM = M201C.
MCS = SYSTEM/BNAMCS.
ADDRESS = (4*0103", 4*0103").
CONTROL = QM.
RETRY = 15.

```

STATION BDLCSWITCHEDMANUAL:

```

DEFAULT = STADEF.
TERMINAL = HBDLCTERMINAL.
ADAPTER = 32.
MODEM = M201C.
MCS = SYSTEM/BNAMCS.
ADDRESS = (4*0103", 4*0103").
CONTROL = QM.
RETRY = 15.

```

The terminal definition for a full-duplex station must be BDLCTERMINAL while a half-duplex station must have HBDLCTERMINAL as its terminal definition. The address statement defines the receive and transmit addresses that are used for BDLC. Two different addresses are used to distinguish between commands and responses.

A full-duplex line definition may be defined as follows:

LINE L00:

```

ADDRESS = 0:10:0.
STATION = BDLCDEDICATED.
MODEM = SUPERMODEM.
ADAPTER = 10 (MODEM).
CB[0] = TYPE RECEIVE, HEADER:15, TEXT:TRUE.
CB[2] = TYPE TRANSMIT, HEADER:15, TEXT:TRUE.
CB[3] = TYPE TRANSMIT, HEADER:15, TEXT:TRUE.

```

A half-duplex line definition may be defined as follows:

LINE L02:

```

ADDRESS = 0:10:2.
STATION = BDLCSWITCHEDWITHACU.
MODEM = M201C.
ADAPTER = 10 (MODEM).
TYPE = DIALIN, DIALOUT.
ANSWER = TRUE.
ENDOFNUMBER = FALSE.
CB[0] = TYPE:RECEIVE, HEADER:15, TEXT:TRUE.
CB[1] = TYPE:TRANSMIT, HEADER:15, TEXT:TRUE.

```

```

CB[2] = TYPE:TRANSMIT,HEADER:15, TEXT:TRUE.
CB[3] = TYPE:TRANSMIT,HEADER:15, TEXT:TRUE.
CB[4] = TYPE:DIALOUT, TEXT:FALSE.

```

L04:

```

ADDRESS = 0:10:4.
STATION = BDLCSWITCHEDMANUAL.
MODEM = M201C.
ADAPTER = 10 (MODEM).
CB[0] = TYPE:RECEIVE, HEADER:15, TEXT:TRUE.
CB[1] = TYPE:TRANSMIT,HEADER:15, TEXT:TRUE.
CB[2] = TYPE:TRANSMIT,HEADER:15, TEXT:TRUE.
CB[3] = TYPE:TRANSMIT,HEADER:15, TEXT:TRUE.

```

The CB statements define the control blocks needed for BDLC message-oriented datacom. The half-duplex definition describes a line with an auto-call unit (ACU) and auto-answer capabilities. The definitions are models and must be changed for each particular line configuration. The address statement defines the DCP, adapter, and line numbers respectively. The line numbers must be incremented by two for BDLC lines.

The BDLC request sets which implement the BDLC station have been added to SOURCENDL. The BDLC station consists of the RECEIVE-FRAME-HANDLER and the TRANSMIT-FRAME-HANDLER, which transmits and receives frames up to the window size, acknowledges packets, handles REJ, FRMR, TEST, etc., messages.

The BNAMCS controls the connection port dialog and station dialog, but the BDLC request sets actually perform opening and closing the dialogs by sending SABM, UA, DM, etc., messages to the remote site.

One request set handles two-way simultaneous transmissions over BDLC dedicated lines and another controls two-way alternate operation over BDLC switched links.

DCP definitions must have the TIMESTAMP option. The terminal definition in the DCP section must have BDLCTERMINAL(MSGSPACE=3) and HBDLCTERMINAL(MSGSPACE=8) included with the other terminals defined.

When an FCS error is detected by the BDLC request sets, the text portion of the message (I-field) is returned to the BNAMCS. If FCSMONITOR is set in BNA, a report is generated displaying the I-field in hex.

OVERVIEW OF NETWORK STARTUP

In order to bring up a BNA network the following steps must be taken:

1. The appropriate hardware must be put into place (modems, adapters, * GLOBAL tm memory, etc.).
2. For all BDLc stations, the necessary NDL must be set up.
3. A network initialization file should be written.
4. The SL ODT message should be used to name the BNA code file to be used; e.g., SL BNASUPPORT=SYSTEM/BNA.

Once these steps have been taken, the system may become part of a BNA network by the operator entering NET+.

INITIALIZATION FILE EXAMPLES

1. Simple

```
LOCALIDENTITY=#1;          % the only required attribute command
ADD HOST LOSANGELES #2;    % configuration commands
ADD HOST DETROIT #3;
ADD HOST NEWYORK #4;
ENDINITIALIZATION;        % network is now operational
```

2. More Complex

```
LOC=#1:I;                  % first attribute command
NETWORKVERSION=JANUARY22:I;
NETWORKMAXSEGMENTSIZ=256:I;
VALIDATE=ALL:I;
MAXHOPCOUNT=5:I;
LOGGING+MINIMUM:I;
MONITOR INTERVAL STATION=10:0.0:I;
NEIGHBORBUSYTIMEOUT=20.0:I;
ADD HOST LOSANGELES #2:I;  % first configuration command
ADD HOST DETROIT #3:I;
ADD HOST NEWYORK #4:I;
ADD HOST PORTLAND #5:I;
MOD STA BDLCACU01 (MONITOR=ON, SPEED=4800):I;
MOD STA BDLCACU02 (SPEED=4800):I;
MOD STA BDLCDDEDICATED01 (SPEED=4800):I;
MOD STA BDLCDDEDICATED02 (SPEED=4800, MONITOR=ON):I;
MOD STA BDLCSWITCH01 (SPEED=1200, AUTOINIT=FALSE):I;
MOD STA BDLCSWITCH02 (SPEED=4800):I;
ADD CONN OUT WITH NEWYORK BY BDLCACU01
(CALLDATA="212-555-1212", INITQUANTITY=1):I;
ADD CONN OUT WITH LOSANGELES BY BDLCACU01
(CALLDATA="213-555-1212"):I;
ADD CONN OUT WITH PORTLAND BY STATION BDLCACU02
(CALLDATA="503-555-1212", INITQUANT=0):I;
ADD CONN IN WITH DETROIT BY BDLCSWITCH01:I;
ADD CONN IN WITH DETROIT BY BDLCSWITCH02:I;
ADD CONN PERM WITH LOSANGELES BY BDLCDDEDICATED01:I;
ADD CONN PERM WITH LOSANGELES BY BDLCDDEDICATED02:I;
NPW LOSANGELES MNPW=XYZZY YNPW=ZYXXZ:I;
NPW DETROIT YNPW="%&()*":I;
NPW NEWYORK MNPW=HEX ABCDEF123456 YNPW=NULL:I;
NPW PORTLAND #5 MNPW="<????>[1234]":I;
ENDINIT:I;                  % network is now operational
AWAITCALL BY STATION BDLCSWITCH01; % AUTOINIT is FALSE
EC WITH PORTLAND BY BDLCACU02; % call not placed automatically
```

COMMUNICATION BETWEEN USER PROCESSES

Communication between processes can now be performed through the standard input/output facility using files of KIND=PORT. Additional file attributes, parameters to the open and close functions, and language constructs in ALGOL, FORTRAN, PL/I, and COBOL have been defined to provide access to this facility. This note describes the general port file facility, while the specific interface defined for each language is described in a separate note for each language.

Use of Attributes

A port file has one or more associated subports (called "subfiles"), each of which may be connected to a different process. Several file attributes have been added or extended to return information about the port file or its subfiles. The following table describes the file attributes applicable to files of KIND=PORT. Attributes marked "File" apply to the port file as a whole; attributes marked "Subfile" apply to each individual subfile of the file.

<u>File Attribute</u>	<u>Notes</u>
BLOCKSTRUCTURE	File
CENSUS	File, Subfile
CHANGEDSUBFILE	File
CHANGEEVENT	File, Subfile
COMPRESSION	Subfile
CURRENTRECORD	Subfile
FILESTATE	Subfile
FRAMESIZE	File
HOSTNAME	Subfile
INPUTEVENT	File, Subfile
INTNAME	File
LASTSUBFILE	File
MAXCENSUS	Subfile
MAXRECSIZE	File, Subfile
MAXSUBFILES	File
MYHOSTNAME	File
MYNAME	File
OUTPUTEVENT	Subfile
SECURITYTYPE	File
STATE	File
SUBFILEERROR	Subfile
TITLE	File
YOURNAME	Subfile
YOURUSERCODE	Subfile

Attributes that are only file attributes may be accessed or assigned by not specifying a subfile index. If a subfile index is specified for a file attribute access or assignment, an attribute error is generated.

Those attributes which are only subfile attributes can be assigned for a particular subfile by providing a subfile index. If a subfile index of zero is specified, the attribute assignment applies to all subfiles in the file. If MAXSUBFILES is equal to one, the subfile index may be omitted; the attribute assignment will apply to the only subfile. When accessing a subfile attribute of a particular subfile, the subfile index must be specified if MAXSUBFILES is greater than one. If MAXSUBFILES is equal to one, the subfile index may be omitted; the attribute access will apply to the only subfile. If a subfile index of zero is specified for an attribute access, an attribute error is generated.

For attributes that are both file and subfile attributes, if a subfile index is not specified, the attribute access or assignment applies to the file; otherwise the attribute access or assignment applies to the subfile.

The null value for all string-valued attributes is ".".

The following attribute descriptions apply to port files:

BLOCKSTRUCTURE

The BLOCKSTRUCTURE values of FIXED and EXTERNAL apply to port files, the default being FIXED. BLOCKSTRUCTURE is meaningful only for READ operations. If BLOCKSTRUCTURE is equal to FIXED, the user's buffer is blank filled. If BLOCKSTRUCTURE is equal to EXTERNAL, only the data received is put into the user's buffer; the actual length of the data placed into the user's buffer can be determined by interrogating the CURRENTRECORD attribute.

CENSUS

The CENSUS attribute can be accessed but not assigned. If accessed as a file attribute, CENSUS returns the total number of messages queued for all subfiles. If accessed as a subfile attribute, CENSUS returns the number of messages queued for the specified subfile.

CHANGEDSUBFILE

CHANGEDSUBFILE is an access-only attribute that returns the subfile index of an arbitrary subfile whose CHANGEEVENT is "happened".

CHANGEEVENT

The subfile CHANGEEVENT is caused whenever the value of FILESTATE changes; it is reset as a side effect of interrogating the FILESTATE attribute. The CHANGEEVENT for the file has the value "happened" as long as any of the subfile CHANGEEVENTs have the value "happened". The CHANGEEVENT for the file is reset by the system after all of the subfile CHANGEEVENTs have been reset.

COMPRESSION

It is possible to compress the data sent between subfiles. Support for the compression feature is negotiated at subfile open time. Setting the value of the COMPRESSION attribute to TRUE while the subfile is open has an effect only if both hosts involved in the subfile dialog support the compression feature (if compression is not supported, the value of COMPRESSION will be FALSE even after setting it to TRUE). If compression is supported, records may be selectively compressed by changing the value of the COMPRESSION attribute.

CURRENTRECORD

This attribute returns the length, in FRAMESIZE units, of the last record read or written.

FILESTATE

The FILESTATE attribute can assume the following values:

CLOSED (0)

The initial state of a subfile is CLOSED. The subfile returns to this state when it is closed by the user.

AWAITINGHOST (1)

This state indicates that the host specified by the HOSTNAME subfile attribute is unreachable. The subfile will remain in this state until the host becomes reachable. The FILESTATE may then change to OFFERED, OPENED, or CLOSED. I/O operations are not valid when the file is in this state.

OFFERED (2)

A subfile enters this state when an open has been done and the host specified by HOSTNAME is reachable, but no matching subfile has been found. I/O operations are not valid when the file is in this state.

OPENED (3)

This state indicates that the subfile is open and may be used to send or receive data.

SHUTTINGDOWN (4)

This state indicates that the system operator has requested that communications with the host involved in the subfile dialog be terminated. This notification

gives the program the opportunity to terminate in an orderly fashion; the port remains open and all I/O operations are valid.

BLOCKED (5)

This state indicates that the remote host has become temporarily unreachable. The port remains open and all I/O operations are valid.

CLOSEPENDING (6)

This state indicates that the user has closed the subfile, but the other subfile has not yet acknowledged the closure. When close acknowledgment is received, FILESTATE changes to CLOSED.

DEACTIVATIONPENDING (7)

This state indicates that the other subfile has been closed and that this subfile has data queued for input.

DEACTIVATED (8)

This state indicates that the other subfile has been closed and that this subfile does not have data queued for input. Close is the only valid operation for a subfile in this state.

FRAMESIZE

This attribute has the same semantics that it has for other types of files. Data is always transmitted in 8-bit units, but the user program may deal with the data using other values for FRAMESIZE.

INPUTEVENT

If accessed as a file attribute, INPUTEVENT returns "happened" if the CENSUS file attribute is greater than zero. If accessed as a subfile attribute, INPUTEVENT returns "happened" if the CENSUS subfile attribute is greater than zero for the specified subfile.

INTNAME

This attribute has the same semantics that it has for other types of files.

LASTSUBFILE

This attribute contains the subfile index of the last subfile that was used for an I/O operation on the file. This value is updated only if the I/O operation was successful. LASTSUBFILE is the preferred synonym for LASTSTATION.

MAXCENSUS

MAXCENSUS specifies the number of input messages that can be queued for this subfile before the other subfile is given a "NO BUFFER AVAILABLE" indication.

MAXRECSIZE

The MAXRECSIZE attribute can be accessed or assigned as a file attribute and is access-only as a subfile attribute. As a file attribute, MAXRECSIZE is used to access or assign the maximum message text size for the port file. When interrogated as a subfile attribute, MAXRECSIZE returns the actual message text size for the subfile, which is negotiated when the subfile is being opened and may be different for each subfile.

MAXSUBFILES

This attribute specifies the maximum number of subfiles that can be opened for the file. The subfiles are assigned indices from 1 to MAXSUBFILES, inclusive.

MYNAME

MYNAME is a string-valued attribute that is used during the subfile matching process; the value of MYNAME must match the value of YOURNAME for the complementary subfile.

OUTPUTEVENT

OUTPUTEVENT is caused whenever output buffers become available and is reset by the system whenever no output buffers are available.

SECURITYTYPE

This attribute has the same semantics that it has for other types of files. The only values allowed for SECURITYTYPE are PUBLIC and PRIVATE.

STATE

The STATE attribute returns result information about the last I/O that was done on the file. The following STATE bits apply to port files:

STATE.[0:1]

This bit indicates that an error has occurred and is set in conjunction with other STATE bits.

STATE.[3:1]

This bit indicates that an invalid subfile index was specified for an I/O operation.

STATE.[8:1]

This bit indicates that an I/O operation failed for one of the following reasons:

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

- a) A broadcast write failed for at least one subfile.
- b) A write with the DONTWAIT option was not done because no buffer was available.
- c) A read with the DONTWAIT option was not done because no data was available.

STATE.[9:1]
This bit indicates end-of-file.

SUBFILEERROR

The SUBFILEERROR attribute is set to one of the following values after each I/O, OPEN, or CLOSE operation that affects the subfile:

- NOERROR(0)
No error occurred during the subfile operation.
- DISCONNECTED(1)
Communication with the other subfile has been severed due to network failure.
- DATALOST(2)
During a close operation, all data was not transmitted successfully to the other subfile before the subfile was closed.
- NOBUFFER(3)
An attempted write to this subfile failed because no buffer space was available. This error can occur only if DONTWAIT was specified on the write.
- NOFILEFOUND(4)
An attempted open on this subfile resulted in a NOFILEFOUND result.
- UNREACHABLEHOST(5)
An attempted open on this subfile resulted in an UNREACHABLEHOST result.

TITLE

TITLE must be a simple name between 1 and 17 characters in length, inclusive. Its value must match the TITLE of the complementary subfile.

YOURNAME

YOURNAME is a string-valued attribute that is used during the subfile matching process; the value of YOURNAME must match the value of MYNAME for the complementary subfile.

YOURUSERCODE

The YOURUSERCODE attribute specifies the usercode under which the process opening the other (complementary) subfile must be running in order to match this subfile if SECURITYTYPE is PRIVATE. The default value for the YOURUSERCODE attribute is the usercode of the process opening this subfile. Setting the value of YOURUSERCODE to null sets the value back to the default.

Open Operations

A subfile provides a two-way, point-to-point, logical communication path between two programs. In order to establish this path, each program must describe the desired connection. The system compares connection descriptions, matches complementary descriptions, and marks the subfiles OPENED. This process is called the matching algorithm. The following subfile attributes are used by the matching algorithm:

HOSTNAME

HOSTNAME contains the name of the host on which the complementary program is running. The value of HOSTNAME for each subfile must match the value of MYHOSTNAME for the complementary subfile. A null value for HOSTNAME matches any MYHOSTNAME value.

MYNAME

The MYNAME attribute is a string of up to 100 characters that is used for matching complementary subfile descriptions. In order to match, the value of MYNAME must match the value of YOURNAME for the complementary subfile. A null value for MYNAME matches only a null value for YOURNAME.

YOURNAME

The YOURNAME attribute is a string of up to 100 characters that is used for matching complementary subfile descriptions. In order to match, the value of YOURNAME must match the value of MYNAME for the complementary subfile. A null value of YOURNAME matches any value for MYNAME.

TITLE

The TITLE attribute is in the form of a simple name (1-17 characters) and must not be null. The TITLE must match the TITLE of the complementary file. The default TITLE is the value of the INTNAME attribute.

SECURITYTYPE

Security checking is performed for each subfile as follows: If the value of its SECURITYTYPE attribute is PUBLIC, security checking is immediately successful. If

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

the value is PRIVATE, the value of its YOURUSERCODE attribute must match the usercode of the process offering the complementary subfile.

The OPEN statement requires two parameters: the subfile that is to be opened and an open option. The processing of the OPEN statement causes the matching algorithm (described above) to be invoked and causes a result to be returned indicating the success or failure of the open attempt. Acceptable values for the two parameters and the possible values for the OPEN statement result are described in the following paragraphs.

The subfile to be opened is specified by a subfile index. If the subfile index is zero, all subfiles with a FILESTATE of CLOSED are opened. When this "open all" facility is used, open results can be obtained by interrogating the SUBFILEERROR attribute. If the subfile index is greater than zero but not greater than MAXSUBFILES, only the specified subfile is opened. If no subfile index is specified and MAXSUBFILES is greater than one, an error of BADSUBFILEINDEX is returned (the open results are described below); if MAXSUBFILES is equal to one, the (only) subfile will be opened.

The following three open options apply to port files:

WAIT

WAIT is the default value. The subfile is offered for matching, and the program is suspended until a matching subfile is found. If the host specified by HOSTNAME becomes unreachable before the open is complete, the program is resumed and UNREACHABLEHOST is returned.

OFFER

OFFER causes the subfile to be offered for matching, and the program is resumed without waiting for the subfile to be matched.

AVAILABLE

AVAILABLE causes the subfile to be matched only to a complementary subfile that has been already offered. If a match is found, the subfile is opened. If no match is found, a NOFILEFOUND result is returned (the subfile is NOT left offered for subsequent matching).

If WAIT or OFFER is specified as the open option and the subfile HOSTNAME names an unknown or unreachable host, the FILESTATE will be set to AWAITINGHOST. The subfile will remain in this state until either it is closed by the user or the host becomes reachable (the open then proceeds normally). If open AVAILABLE is specified and the host is unknown or unreachable, an open result of UNREACHABLEHOST is returned.

The OPEN function may return the following values as its result:

OK(1)

The open was successful. If the open type was OFFER, this result indicates that the open process was successfully started.

NOFILEFOUND(2)

NOFILEFOUND is returned if AVAILABLE was specified for the open option and a matching subfile was not found.

UNREACHABLEHOST(38)

UNREACHABLEHOST is returned in the case where open AVAILABLE was specified and the HOSTNAME attribute names an unreachable or unknown host. UNREACHABLEHOST is also returned in the case where open WAIT was specified and the host named by the HOSTNAME attribute becomes unreachable during the process of opening the subfile.

ALREADYOPEN(40)

ALREADYOPEN is returned if the specified subfile does not have FILESTATE equal to CLOSED.

BADSUBFILEINDEX(42)

BADSUBFILEINDEX is returned if the subfile index specified was less than zero or greater than MAXSUBFILES or if a subfile index of zero was specified with the AVAILABLE open type.

As part of the open process, the value of the MAXRECSIZE attribute to be used in the conversation between the two subfiles is negotiated. The negotiated value is always the smaller of the two MAXRECSIZE values.

For languages that allow implicit file open, port files may be implicitly opened by an I/O operation on the file. If a non-zero subfile index is specified, then only that subfile will be opened. If a subfile index of zero is specified, then no implicit open action will take place. If a subfile index is not specified and MAXSUBFILES is equal to one, then the subfile will be implicitly opened.

I/O Operations

Each subfile has a unique subfile index. A program can, by specifying a subfile index, perform a read or write to a particular subfile. If the subfile index is omitted and MAXSUBFILES is equal to one, the I/O is performed on the only subfile of the file. If a subfile index of zero is specified on a read, a non-selective read is performed. The non-selective read provides the ability to read the next message from any subfile. If a subfile index of zero is specified for a write, the message is broadcast to all open subfiles. If an error occurs on any subfile

during a broadcast write, the result from the write will indicate that an error has occurred.

I/O statements may include a DONTWAIT specification, which allows the program to continue executing if the I/O operation cannot be completed immediately. If an I/O statement is prematurely terminated because of a DONTWAIT specification, a field in the STATE attribute is set to indicate this occurrence (see Use of Attributes). If a READ statement is executed and CENSUS is equal to zero, the program is suspended until data is available unless DONTWAIT is specified. If a WRITE statement is executed and no buffers are available, the program is suspended until the write can be completed unless DONTWAIT is specified.

If a subfile index less than zero or greater than MAXSUBFILES is specified or if no subfile index is specified and MAXSUBFILES is greater than one, the program attempting the I/O operation is terminated. Some languages have the ability to return a result from an I/O operation. These languages may return a "bad subfile index" indication instead of terminating the user program.

When the value of the FILESTATE attribute becomes DEACTIVATIONPENDING, all subsequent write operations return an end-of-file indication. Read operations will continue to operate normally as long as there are messages queued for input; when there are no more messages queued for input, the FILESTATE changes from DEACTIVATIONPENDING to DEACTIVATED, and all subsequent read operations return an end-of-file indication. If a program is suspended waiting for an I/O operation to complete and the other subfile closes (FILESTATE goes to DEACTIVATED), the program is resumed and end-of-file is returned.

I/O statement options, such as skip-to-channel, skip lines, and stacker selection, are ignored for port files.

When the value of the FILESTATE attribute becomes BLOCKED, I/O operations continue to function normally. This action makes it possible for programs to attempt to maintain dialogs through temporary host-unreachable conditions. However, the system may quickly run out of buffer space and write operations will cause the program to be suspended unless DONTWAIT is specified. Since there will be no incoming messages during this time, read operations will cause the program to be suspended once the queued input is exhausted, unless DONTWAIT is specified.

When the value of the FILESTATE attribute is AWAITINGHOST or OFFERED, all I/O operations will return an end-of-file condition.

The length of data transfers through a port file is dependent on the following criteria:

- a) The length indicated in the I/O statement.
- b) The MAXRECSIZE of the subfile.
- c) For READ operations, the length of the data actually received (this is a factor only if BLOCKSTRUCTURE is equal to EXTERNAL).
- d) The size of the user's buffer.

For WRITE operations, the amount of data sent is the minimum of the subfile MAXRECSIZE, the length indicated in the WRITE statement, and the size of the user's buffer. If an attempt is made to write data that is larger than MAXRECSIZE, the message is truncated to MAXRECSIZE; no indication of this truncation is given to the user. If the length specified in the WRITE is smaller than MAXRECSIZE but larger than the user's buffer, the message is truncated to the size of the user's buffer; no indication of this truncation is given to the user. The value of the BLOCKSTRUCTURE attribute has no effect on WRITE operations.

For READ operations with BLOCKSTRUCTURE equal to FIXED, the length of data delivered to the user's buffer is the minimum of the length indicated in the READ statement, the value of MAXRECSIZE, and the size of the user's buffer. If the message received is smaller than this size, the buffer is blank-filled. If the message received is larger than this size, the message is truncated without any indication given to the user. The value of CURRENTRECORD is always equal to MAXRECSIZE in this case.

For READ operations with BLOCKSTRUCTURE equal to EXTERNAL, the length of the data delivered to the user's buffer is the minimum of the length indicated in the READ statement, the value of MAXRECSIZE, the size of the user's buffer, and the size of the actual message. If the size of the actual message is larger than the length indicated in the READ statement, the value of MAXRECSIZE, or the size of the user's buffer, truncation will take place and the user will not be notified. No blank fill is done for BLOCKSTRUCTURE equal to EXTERNAL. The CURRENTRECORD attribute may be used to determine the length of actual message.

Close Operations

The execution of a CLOSE statement changes the FILESTATE of the specified subfile to CLOSEPENDING or CLOSED. Any messages that have been queued for receipt are discarded.

Closing a port file may take a significant amount of time. Because this delay is unacceptable for time-critical programs, the close option DONTWAIT is provided. If DONTWAIT is not specified, the program will be suspended while the actual close takes place. If DONTWAIT is specified, control returns immediately to the program and the process of actually closing the file takes place in parallel with the execution of the program. The program may detect when the close is complete by monitoring state changes. All other close options (PURGE, LOCK, etc.) are ignored. The CLOSE function may return the following values as its result:

OK(1)

The close was successful.

FILENOTOPEN(30)

The subfile was already closed.

BADSUBFILEINDEX(42)

The subfile index specified was less than zero or greater than MAXSUBFILES.

DATALOST(32)

All data was not successfully delivered to the destination subfile.

CLOSEALLERROR(46)

A close error occurred on any subfile while attempting to close all subfiles.

If the subfile index specified in a CLOSE statement is zero, all open subfiles are closed. When this "close all" facility is used, close results may be obtained by interrogating the SUBFILEERROR attribute. If the subfile index is greater than zero but not greater than MAXSUBFILES, only the specified subfile is closed. If no subfile index is specified and MAXSUBFILES is greater than one, an error of BADSUBFILEINDEX is returned; if MAXSUBFILES is equal to one, the (only) subfile will be closed.

USE OF USERCODES

System Users

ODT requests from other hosts are, by default, given a restricted set of capabilities. It may be desirable to give a remote operator full privileges on the local system. This may be done by marking the remote operator's usercode as a System User. ODT requests from remote System Users are then given the same privileges as ODT requests from the local ODT.

A user may be marked as a System User by turning on the SYSTEMUSER bit in the Userdatafile for that user.

Local Alias

In order for a cooperating host (henceforth known as a local host), to be able to grant access to a usercode from an initiating host (henceforth known as a remote usercode), both the remote usercode and its associated hostname must be passed to the local host. Because usercodes are unique on their own systems and hostnames are unique over the network, a remote usercode and hostname will uniquely identify a user. The local host will keep the remote usercodes and their associated hostnames in its SYSTEM/USERDATAFILE.

The remote usercode and hostname will be used for identification, but an ALIAS will be used by the local host's MCP instead of the remote usercode and hostname. The ALIAS will be a normal 17-character usercode which the MCP will treat exactly as a local usercode. The ALIAS will be kept in the local host's SYSTEM/USERDATAFILE together with the associated remote usercode and hostname.

The user interface to SYSTEM/USERDATAFILE is SYSTEM/MAKEUSER. SYSTEM/MAKEUSER is a stand-alone program that can be run only by a privileged user via WFL or CANDE.

ALIAS Syntax in SYSTEM/MAKEUSER

The ALIAS statement in SYSTEM/MAKEUSER allows the user the ability to display, add, remove, or change the ALIAS for a remote usercode and hostname.

Syntax:

<alias command>

```

-----<alias display>-----|
| -<alias delete>-----|
| -<alias add/modify>-|

```

<alias display>

```

-- ALIAS --<usercode>-----|
| - OF --<hostname>-|

```

<alias delete>

```

-- - - ALIAS --<usercode>-- OF --<hostname>--|

```

<alias add/modify>

```

----- ALIAS --<usercode>-- OF --<hostname>-- = ----->
| - + - |
>-----<usercode>-----|
| - * -----|

```

Semantics:

<alias display>

- 1) "ALIAS <usercode> OF <hostname>" will display what the "local usercode" (i.e., ALIAS) is for the "remote usercode"/"hostname" pair.
- 2) "ALIAS <usercode>" will display a list of all the "remote usercodes" of the same name, with their respective "hostnames" and "local usercodes" (i.e., ALIASes).

<alias delete>

- 1) "- ALIAS <usercode> OF <hostname>" will disallow accesses to the "local host" by "remote usercode" of "hostname".

<alias add/modify>

- 1) "+ ALIAS <usercode> OF <hostname> = <usercode>" will allow accesses to the "local host" by "remote usercode" of "hostname". The optional "+" in the "local usercode" parameter means that the "local usercode" is to be the same as the "remote usercode".
- 2) "ALIAS <usercode> OF <hostname> = <usercode>" will modify an existing "local usercode" (i.e. ALIAS).

Special Notes

Since the ALIAS statement will associate a remote usercode and hostname with a local usercode, the local usercode either must already exist or must be created. If there is no local usercode, any access to the local host by the remote usercode will then result in an "INVALID USERCODE" error being returned.

Also, if the local usercode is ever removed, all remote usercode and hostname pairs that are associated to it must also be removed. This implies that a list of all ALIASes must be kept and updated whenever any changes occur to SYSTEM/USERDATAFILE.

Examples:

```

Entering Aliases from a CANDE terminal:
RUN $SYSTEM/MAKEUSER
#RUNNING 0001
#?
*SYSTEM/MAKEUSER ON DISK 32.141 00:00 AM MONDAY, JUNE 1,1980
$SET LIST
<< OK
+ ALIAS X OF Y = *;
<< ENTERED
    ALIAS X OF Y = "*"

+ ALIAS X OF Z = A;
<< ENTERED
    ALIAS X OF Z = A

ALIAS X;
<< EXAMINED
    ALIAS X OF Y = "*"
    ALIAS X OF Z = A

ALIAS X OF Z = B;
<< UPDATED
    ALIAS X OF Z = B

ALIAS X OF Y;
<< EXAMINED
<< ALIAS = X

- ALIAS X OF Y;
<< DELETED

ALIAS U OF H = X;
***USERDATA ERR: NO SUCH USERCODE/HOSTNAME
<< **ERROR**

+ ALIAS X OF Z = A;
***USERDATA ERR: ATTEMPT TO DEFINE DUP USERCODE/HOSTNAME
<< **ERROR**

END.
#ET=0.0 PT=0.0 IO=0.0

```

If there happens to be an error while in SYSTEM/MAKEUSER, use the "IGNORE" or "END." statement. If there is an error, SYSTEM/MAKEUSER will only do syntax checking and will not enter the ALIAS into SYSTEM/USERDATAFILE.

Entering Aliases from a WFL deck:

```
?BEGIN JOB ENTER/ALIASES;  
RUN $SYSTEM/MAKEUSER;  
DATA CARD  
+ALIAS X OF Y = *;  
+ALIAS X OF Z = A;  
ALIAS X;  
ALIAS X OF Z = B;  
ALIAS X OF Y;  
-ALIAS X OF Y;  
ALIAS U OF H = X;  
+ALIAS X OF Z = A;  
END  
?END JOB
```

Example:

```
ALIAS OPERATIONS OF MIAMI = REMOTEOPERATIONS;  
ALIAS OPERATIONS OF BOISE = REMOTEOPERATIONS;  
ALIAS DUKE OF EARL = YORK;  
ALIAS UCI OF H3 = *;
```

HOST SERVICES

Host Services is a series of host-to-host protocols which provide distributed processing and resource sharing capabilities. The protocols implemented for the Mark 32 release are:

1. ODT. This allows either the system operator or users at terminals to check the status of and control tasks running at another host, or to interrogate the status of the host itself.
2. File Transfer. Using the COPY syntax, it is possible to Transfer a file from one host to another.
3. Logical I/O. Files may be created, read, written, or updated at another host.
4. Job Transfer. WFL decks may be transferred from one host to another and executed at the second host.
5. Tasking. Using normal IPC mechanisms it is possible to initiate and control tasks at another host.

ODT

In order for either a system operator or user to make inquiries or attempt to control actions taking place at another host in the network, there must be a method of detecting messages that are destined for a remote host. This is done through the system console and through CANDE by recognizing messages that start with the keyword "AT". From the console, a message may be directed to another host, e.g., one with the hostname of "SOMEWHERE", by entering the following:

AT SOMEWHERE <ANY TEXT STRING> ETX

The Controller recognizes the AT as indicating that the message is destined for a remote host. The Controller will not syntax check the message <ANY TEXT STRING>, but will simply pass the message to Host Services. A dialog will be established with the host SOMEWHERE if a) the host is known, and b) a dialog does not already exist. Host Services will transmit the message, receive the reply, and pass the reply to the Controller to be displayed on the console. CANDE users may direct commands to remote hosts by entering the following:

?AT <hostname> <command>

CANDE will pass the command to the Controller. The message will follow the same path outlined above, but the reply will be routed back to the users terminal.

A usercode is contained in each message. For requests made via the ODT, if no usercode is available (i.e., terminal usercode), the host usercode is used.

When receiving a message, the usercode in the message is tested for its privilege (i.e., whether it is a "systemuser" or not).

File Transfer

Using WFL COPY syntax, files may be copied from one host to another. The initiator of the COPY need not be the source or destination host.

Examples:

```
COPY SYSTEM/MCP TO PACK(HOSTNAME=A)
COPY X FROM YOURS(KIND=PACK,HOSTNAME=B) TO OURS(PACK)
COPY MY/FILE AS YOUR/FILE FROM PACK TO PACK(HOSTNAME=C)
ADD ABC FROM X(PACK,HOSTNAME=A) TO Y(PACK,HOSTNAME=B)
```

The following restrictions exist for the Mark 32 release:

1. The File Transfer capability is available only in new WFL.
2. A minimal subset of the COPY syntax is supported. Not supported are COMPARE, BACKUP, CATALOG, lists of multiple files, multiple sources or destinations, AREAClass, FAMILYINDEX, SINGLEPACK, INTERCHANGE volume specifications, copies involving tape as source or destination, copies involving directories.
3. The following file types cannot be transferred: FILETYPE not equal zero or three, BCL files, non-disk files, duplicated files, FILEKINDs of DIRECTORY, VERSIONDIRECTORY, SYSTEMDIRECTORY, RECONSTRUCTIONFILE, SYSTEMDIRFILE, JOBDESCFILE or XDISKFILE, installation allocated disk files, interchange pack files.

4. The restrictions listed under Logical I/O regarding file names and mnemonic-valued attributes also apply to files being transferred across systems.
5. When a command to copy files is entered on the ODT, any files which have both the <usercode> and * omitted will assume the host usercode (not *).
6. When initiating a File Transfer on a remote host via the ODT protocol, a "USER=" statement must be included.

For example, entering the following command will initiate File Transfer on host BROWN and will copy file X from host YELLOW:

```
AT BROWN USER=UC/PW; COPY X FROM DISK(HOSTNAME=YELLOW);
```

If the "USER=" statement is not included, File Transfer will give the message "ATTEMPTED FILE TRANSFER WITHOUT A USERCODE", and not copy the file.

Family substitution will not occur on any cooperating host (any host at which the transfer was not initiated). When File Transfer is performed by a job that has a usercode associated with it, that usercode will be assumed for all filenames without an explicit usercode (as usual).

File Transfer does not preserve the CREATIONDATE, USEDATE, or TIMESTAMP of the file being transferred.

It is most efficient to initiate File Transfers at either the source or destination host. When the initiator is at another host, the Job Transfer or ODT protocols may be used to initiate the job on either the source or destination host.

Logical I/O

The Host Services library permits programs to read, write, and create files at another host. The file attribute HOSTNAME can be set either in the user's program or via label-equation. This protocol requires the task to be running under a usercode.

Examples

- 1) ?BEGIN JOB FOREIGN/COMPILE;
 COMPILE PROG COBOL;
 COBOL FILE CARD(KIND=DISK,TITLE=S/PROG,HOSTNAME=D);
 ?END JOB

- 2) BEGIN
 FILE F(KIND=DISK,DEPENDENTSPECS=TRUE,TITLE="THE/FILE.",
 HOSTNAME="E.");
 ARRAY A[0:12];
 LABEL EOF;
 WHILE TRUE DO
 BEGIN
 READ(F,12,A)[EOF];
 .
 END;
 EOF:
 CLOSE(F);
 END.

- 3) BEGIN
 FILE F(KIND=PACK,MAXRECSIZE=14,TITLE="OVER/THERE.",
 NEWFILE=TRUE,AREAS=10,AREALENGTH=1400,
 HOSTNAME="E.");
 ARRAY A[0:12];
 BOOLEAN DONE;
 DO
 BEGIN
 .
 WRITE(F,12,A);
 .
 END
 UNTIL DONE;
 LOCK(F);
 END.

On the Mark 32 system software release, the following restrictions are in effect on logical I/O operations performed across systems via HOSTSERVICES:

1. Only FILETYPEs of 0, 3 and 8 are allowed.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

(FILETYPE 8 behaves the same as DEPENDENTSPECS = TRUE.)

2. The attribute KIND must be set. The only values allowed for KIND are TAPE, TAPE7, TAPE9, TAPEPE, PACK, DISK, READER, PRINTER and PUNCH.
3. No UPDATE files are allowed. UPDATEFILE must be FALSE.
4. MAXRECSIZE must be less than 65,000 characters (or equivalent).
5. USE routines are not supported.
6. In order to create a file, the attribute NEWFILE must be TRUE. If the attribute is set to FALSE, HOSTSERVICES may search for an existing file. Note that NEWFILE cannot be changed by the FA command if the program hangs on a "no file" condition.
If NEWFILE=TRUE, UNITS=CHARACTERS and INTMODE=SINGLE, the file actually created will have UNITS=WORDS.
7. No relative I/O; no keyed I/O.
8. No direct I/O.
10. There must be a USERCODE associated with the task performing logical I/O operations.
11. Use of the following attributes is not supported by HOSTSERVICES. If the attribute can be set, any value assigned before the file is opened will generate a file open error; any attempt to read or write the attribute after the file has been opened will generate an attribute error.

AREAClass
ASSIGTIME
ATTVALUE
ATTTYPE
CYLINDERMODE
ENABLEINPUT
EOF
ERRORTYPE
EXCLUSIVE
IAD
INTERCHANGE
IOCLOCKS
IOINERROR
POPULATION
PRESENT
RECEPTIONS
RECORD
RECORDINERROR
RESIDENT
ROWADDRESS
ROWSINUSE
SIZEMODE
SIZEOFFSET
SIZE2
SPEED
TANKING
TRANSMISSIONS
UNITNO

Also, any PORT only attribute, any READERSORTER only attribute, any DISKETTE only attribute, any DIRECT only attribute.

13. For each of the following mnemonic-valued attributes, the values listed below it are not supported by HOSTSERVICES. A request to open or copy a file will be refused by HOSTSERVICES if one of these attributes has been set to a value that is not supported.
 - A. SECURITY TYPE
CONTROLLED
 - B. FILEKIND
ARCHIVELOG
BACKUPPRINTER
BDDATA
CATALOG
FIRMWARE
INFOFILE
LCOBOLSL3CODE
LCOBOLSL5CODE
LCOBOLSYMBOL
MDLCODE
MDLSYMBOL
NULLFILE
RECOVERYFILE
REMOTEAUDIT

REMOTEBACKUP
 RSNETFILE
 RSPCODE
 RSSORTTABLE
 SCHEDULEFILE
 UCRFILE

C. EXTMODE or INTMODE
 BCL
 HEX

14. The attributes FILEKIND and PROTECTION are applicable only to disk files. If a file is assigned to a non-disk device, any attempt to access these attributes will generate an attribute error.
16. The file attributes FILEKIND, SECURITYTYPE, SECURITYUSE and SECURITYGUARD are ignored if a permanent disk file is assigned (NEWFILE=FALSE). If, before the file is opened, any of these attributes is set to a value different from that of the physical file on disk, the value of the file attribute will not be changed in the physical file at file assignment time. However, if the attribute is set after the file has been opened, the physical file will be changed accordingly.
17. Under HOSTSERVICES logical I/O, the default value for the file attribute TRANSLATE is FULLTRANS. The value DEFAULTTRANS is not supported; any request for DEFAULTTRANS will be changed to FULLTRANS.
18. Only file names which are valid under "new" (post-29) WFL are acceptable to HOSTSERVICES. This restriction applies to any attribute which has a file name as its value, including TITLE, FILENAME and SECURITYGUARD.
19. If the SERIALNO attribute is set before a file is opened, serial numbers for other than the first reel are ignored.
20. The TITLE of a disk file cannot be changed while the file is open, but the FILENAME can be.
21. If a file has SECURITYTYPE GUARDED, the associated guard file must grant access only by usercode; i.e., the access specification in the guardfile must be of the following form:

USERCODE X;

If the access specification is in either of the two following forms, access to the file will not be granted to the specified program:

PROGRAM Y;
 USERCODE X USING PROGRAM Y;
22. Error results for WRITE statements will not be reported until one WRITE after the WRITE statement which reuses the buffer that originally had the error, instead of exactly at the WRITE that reuses the buffer that had the error (i.e., in this case, the error is reported one buffer later than normal).
23. HOSTNAME may not be FAed once it is set.
24. MYUSE = OUT will act the same as NEWFILE = TRUE, all other values of MYUSE will act the same as NEWFILE = FALSE. If NEWFILE is set, then MYUSE is ignored. The special semantics for MYUSE = OUT and AREASIZE = 0 are not supported (these semantics are being deimplemented for local hosts on software release mark 35).
25. BASIC programs may not use logical I/O across the network.
26. The following attributes may only be set before the file is open:

AREASIZE
 FILETYPE
 MYUSE
 UNITS

These attributes cannot be set or interrogated once the file is open or closed with retention. It is suggested that the following attributes be substituted:

AREALENGTH
 BLOCKSTRUCTURE
 DEPENDENTSPECS
 FILEUSE
 FRAMESIZE
 NEWFILE
27. Any compiler will be unable to create a codefile across the network.
28. KIND = DISK will act the same as KIND = PACK across the network.

29. The default for FAMILYNAME across the network is "DISK" in all cases.

Job Transfer

WFL has been modified to interface with Host Services. This allows users to transfer entire job decks to a remote host for interpretation and execution. This feature is invoked by prefacing the job deck with a card as follows:

```
<I> AT <hostname> BEGIN JOB
```

and ending the job deck with a card as follows:

```
<I> END JOB
```

The <I> indicates the invalid character in column one of the card image. The invalid character is not allowed between the AT <hostname> and the BEGIN JOB. This insures that a deck dropped into a reader need only start with <I> BEGIN JOB in order to avoid a <I> AT <hostname> that has been left in the card reader.

The following restrictions apply to the Mark 32 release:

1. The transfer of BINARY formatted decks is not supported.
2. Passing parameters to a job that is being sent to another host to run via the JOB TRANSFER protocol is not supported by the current implementation.

The user must supply his own "BEGIN JOB" statement on "?AT <hostname>" commands entered on the ODT, CANDE "WFL AT <hostname>" commands, and "ZIP WITH <array>" statements.

Examples

```
From the ODT: ?AT MIAMI BEGIN JOB;DISPLAY "HELLO FROM LA"
From CANDE:  WFL AT MIAMI BEGIN JOB;COPY A AS B FROM DISK
```

4. It is a BNA requirement that job decks that begin with the "AT <hostname>" specification have an <I> on the "END JOB" card. For example, on a job file that is stored on disk and started by the CANDE "START" command, the following would be an appropriate terminator record:

```
?END JOB
|
+-- in column 1
```

Tasking

Tasks may be initiated at another host by setting the task attribute HOSTNAME to a hostname that is not the local host. The task attribute is available in WFL and in the user languages. It may be set at any time prior to task initiation.

This protocol requires the job to be running under a usercode.

Examples

```
1) ?BEGIN JOB FOREIGN/TASK;
   RUN T;
   OPTION=FAULT,ARRAYS;
   PRIORITY=40;
   HOSTNAME=F;
   PROCESS RUN Z;
   HOSTNAME=G;
   ?END JOB

2) BEGIN
   TASK T;
   PROCEDURE EXT; EXTERNAL;
   REPLACE T.NAME BY "FOREIGN/PROG.";
   REPLACE T.HOSTNAME BY "H.";
   PROCESS EXT[T];
   WHILE T.STATUS > 0 DO
     WAITANDRESET(MYSELF.EXCEPTIONEVENT);
   END.
```

The following limitations exist on the Mark 32 system software release:

1. Compile and go is not supported across the network.
2. The CONTINUE statement is not supported across the network.

3. The following attributes are allowed to be used across the network:

ACCESSCODE
 ACCUMPROCTIME
 BDNAM
 CHARGECODE
 CLASS
 COMPILETYPE
 DESTNAME
 DESTSTATION
 ELAPSEDLIMIT
 ELAPSEDTIME
 FAMILY
 FILECARDS
 HISTORY
 HOSTNAME
 ITINERARY
 JOBNUMBER
 MAXCARDS
 MAXIOTIME
 MAXLINES
 MAXPROCTIME
 MAXWAIT

OPTION: the following bits:

ARRAYS
 AUTORM
 BACKUP
 BDBASE
 CODE
 DSED
 FAULT
 FILES
 NOSUMMARY
 PRIORITY
 SOURCEKIND
 SOURCESTATION
 STACKHISTORY
 STACKLIMIT
 STACK
 STACKNO
 STATION
 STATUS
 STOPPOINT
 SUBSPACES
 SUBSYSTEM
 TASKVALUE
 TYPE
 CODE
 WAITLIMIT

4. Tasks initiated at a remote host may have one parameter, which is a single dimensioned array with a zero lower bound. The array will be passed by value.
5. If the task attribute FILECARDS is used, the following restrictions apply to the label equation string: WFL global files may not be used with the TASKING protocol.
6. All file names must conform to new (post-29) WFL syntax. File names must be composed of only alphanumeric characters.
7. Any specification of file KIND must conform to new (post-29) WFL syntax. No KIND lists are allowed.
8. Tasks initiated at a remote host may not access their parent job's data decks.
9. All tasks must be external code files.
10. Only semi-dependent tasks may be run.

D4134 BNA - IMPLEMENT LOGGING FOR "BNA"

NETWORK SERVICES LOGGING AND MONITORING

1. Introduction

BNA logging provides a facility for recording specific events for such purposes as billing, scheduling, reporting, or monitoring. An operator may select a group of events to log. These events, when they occur, are written to the system log file.

BNA monitoring consists of the accumulation of certain network information over a specific time interval. When monitoring is requested, a report containing the information is periodically logged in the system log file.

This document contains information about the set of events which may be logged, the user interface, and the BNA log formats.

2. Log Items

Following is a list of all BNA events and reports which may be logged. The operator has a choice of four sets of items: MINIMUM, STANDARD, ALL or NONE. (ALL causes all of the events below to be logged; NONE causes none of the events to be logged.) The reports marked with an asterisk (*) are generated by monitors, and must be individually selected by the operator (as described in Section 4) to be included in the log.

These commands are described in BNA note D3194. Reports are caused by occurrences of certain events on the system and are self-explanatory. The log formats are described in JOBFORMATTER note D4273.

NSM COMMANDS:	MINIMUM	STANDARD
SET NSM ATTRIBUTE		X
NSM REPORTS:		
PHASE CHANGE REPORT	X	X
LOG REPORT	X	X
PLM COMMANDS:	MINIMUM	STANDARD
SET PLM ATTRIBUTE		X
ADD HOST		X
SET HOST STATUS		X
DELETE HOST		X
OPEN SUBPORT (FILE LOGGING)		
CLOSE SUBPORT (FILE LOGGING)		
PLM REPORTS:	MINIMUM	STANDARD
PORT LEVEL ERROR REPORT	X	X
PORT LEVEL LOG REPORT	X	X
ROUTER COMMANDS:	MINIMUM	STANDARD
SET ROUTER ATTRIBUTE		X
NA EXISTENCE		X
LINK RF CHANGE		X
START TRACE		
ROUTING REFRESH		
RCF SEND		
ROUTER REPORTS:	MINIMUM	STANDARD
RCF RECEIVED REPORT		X
FRAME ERROR REPORT	X	X
DNA STATUS CHANGE REPORT		X
* RM SUMMARY REPORT	X	X
* RM COPY REPORT	X	X
SLM COMMANDS:	MINIMUM	STANDARD
SET SLM ATTRIBUTE		X
MODIFY STATION		X
ADD CONNECTION		X
DELETE CONNECTION		X
MODIFY CONNECTION		X
ESTABLISH CALL		X
AWAIT CALL		X
CLEAR CALL		X
SAVE		X
READY		X
SEND TEST		
LINK RESET		X
OPEN CONNECTION PORT		X
CLOSE CONNECTION PORT		X
OPEN STATION DIALOG		X
CLOSE STATION DIALOG		X
VALIDATE STATION ATTACH		X
MANUAL DETACH		X
SLM REPORTS:	MINIMUM	STANDARD
ATTACH REPORT	X	X
DETACH REPORT	X	X
NEIGHBOR RESTART REPORT	X	X
LINK RESET REPORT	X	X
* STATION LEVEL LOG REPORT	X	X
* STATION FAILURE REPORT	X	X
* STATION MONITOR REPORT	X	X
NEIGHBOR REMOTE BUSY		X
STATION VALIDATION FAILURE	X	X
BDLC TEST COMMAND RECEIVED		
BDLC TEST RESPONSE		X

3. Monitoring

BNA will include monitoring for certain station and router events. When monitoring is requested (described in the User Interface section), a report will be generated at specific time intervals and recorded in the system log file.

Station Monitoring

Station monitoring is recorded in the log under the log type "STATION MONITOR LOG REPORT".

BDLC stations will be capable of monitoring the following events:

BDLC FRAME RECEIVED
 FCS FAILURE
 BDLC FRAME SENT
 I FRAME SENT
 BDLC MEMORY ERRORS
 BDLC SHORT FRAMES
 REMOTE BUSY SET/RESET
 I RSP RECEIVED
 CONNECTION PORT DIALOG OPENED/CLOSED
 INVALID BDLC ADDRESS
 BDLC FRAME RECEIVED NOT OCTET MULTIPLE
 UNEXPECTED CONNECTION PORT DIALOG CLOSED
 NON-BNA CALLER
 LINK RESET LOCALLY
 FRMR RECEIVED
 RETRY COUNT EXCEEDED
 LINK RESET REMOTELY
 FRMR SENT
 UNEXPECTED DM/DISK RECEIVED
 STATION DIALOG REOPENED

GMM stations are capable of monitoring the following events:

GMM FRAME RECEIVED
 GMM FRAME SENT
 REMOTE BUSY SET/RESET
 CONNECTION PORT DIALOG OPENED/CLOSED
 UNEXPECTED CONNECTION PORT DIALOG CLOSED
 LINK RESET LOCALLY
 RETRY COUNT EXCEEDED
 LINK RESET REMOTELY
 STATION DIALOG REOPENED

Router Monitoring

The router is capable of accumulating traffic summary information and frame information, as follows:

ROUTER MONITOR SUMMARY REPORT - contains a summary of traffic information, such as number of frames sent and average length of frames sent.

ROUTER MONITOR COPY REPORT - stores a copy of each frame which is processed by the node.

DOCUMENT CHANGES NOTES (D NOTES)

CANDE

D4135 CANDE - BLANKS IN "UTILITY" PARAMETER

In the CANDE Reference Manual (Form No. 3011398) on Page 4-110 in the second paragraph under UTILITY semantics, the sentence containing the following:

"...if it is BOTH preceded AND followed..."

should read as follows:

"...(A blank is considered redundant if it is either preceded or followed by a character which is not a letter or digit, or if it ends the <text>.) ..."

D4174 CANDE - DESTINATION FILENAME CRUNCHED

Files created with FIND or REPLACE are now crunched files. If a crunched file is not desired, use the output option NOCRUNCH after the file name.

Examples:

FIND XFILE .X. :FILE YFILE, NOCRUNCH
REPLACE .X..BITWORD. :FILE REVISED LINES, NOCR SQUEEZE

Files created by MATCH are now crunched unconditionally.

D4229 CANDE - "HN" "ODT" MESSAGE

The HN interrogation ODT message is now allowed from a CANDE terminal.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

CANDE

P1190 CANDE - "INVALID MODE" ERROR FOR "BCL" FILES

In attempting to access BCL files on non-BCL systems, the following commands will cause an INVALID MODE error: DO, EXCLUDE, FIND, GET, INSERT, LIST, MATCH, MERGE, RANGE, RMERGE and SCHEDULE. Also, for the commands TYPE and SECURITY, if the file does exist but is not available, a "NOT AVAILABLE" message is given instead of "NO FILE".

P1284 CANDE - STACK SIZE FOR "PROGRAMDUMP"

The stack size for CANDE worker stacks has been increased to permit taking PROGRAMDUMPs.

P1285 CANDE - CHANGING TANKFILE PROBLEM

A previous change which corrected a timing problem when removing corrupted tankfiles caused the new version of a tankfile to be used only temporarily. This resulted in CANDE not remembering certain run-time parameters following a service interruption. The newly-created tankfile is now made permanent and parameters are remembered when CANDE is discontinued.

P1372 CANDE - USAGE OF "RENEW"

CANDE now correctly handles the records specified in a RENEW statement. Previously, a CANDE fault occurred when a list was done after the following command sequence:

"DEL ALL; RENEW 0-END"

P3760 CANDE - "DELETE ALL" SYNTAX

The documented syntax for DELETE ALL permits a minimum abbreviation DEL ALL; there should be no other token in the statement. This syntax is now strictly enforced; thus, DEL A is no longer accepted. A statement like the following gets an END OF STATEMENT EXPECTED error:

DEL ALL-300

DOCUMENT CHANGES NOTES (D NOTES)

COBOL

D3655 COBOL - LEVEL "01" DATA ITEM

The COBOL compiler gave a syntax error if a level 01 data item with an OCCURS clause were used as a library parameter. This problem has been corrected.

Consequently, Section 6.5 of Appendix B of the Mark 31 P- and D-Notes, "User Interface to Libraries", should be changed as follows:

Delete the following lines:

Comp-2, 01 occurs	Hex character array [0,0]
ASCII, 01 occurs	ASCII character array [0,0]
Display, 01 occurs	EBCDIC character array [0,0]

Add the following note:

"NOTE:

A level 01 data item with an OCCURS clause may not be used as a parameter."

D4142 COBOL - "AT END" ALLOWED FOR PORT FILE READ

Add the following to page 7-97 of the COBOL manual:

OPTION 1 or OPTION 2 is used for port files.

D4218 COBOL - "STOP RUN" IN "COBOL" LIBRARY

A STOP RUN executed in a COBOL library causes a normal end-of-task.

Note: This description supersedes the description of STOP RUN in Mark 31 Appendix B, "User Interface to Libraries", Section 6.1, paragraph 2.

D4310 COBOL - "ORGANIZATION IS SEQUENTIAL"

Add the following sentence to the COBOL Reference Manual (Form No. 5001464), on Page 5-17 under the heading "ORGANIZATION/FILE STATUS Clauses":

"If the "ORGANIZATION IS SEQUENTIAL" clause is specified, the access mode must not be specified as random."

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

COBOL

P1117 COBOL - LIBRARY TEMPLATE SPLITS ACROSS AREA BOUNDARY

The compiler generated library templates that crossed area boundaries, causing run-time errors when presence bit actions were taken. This problem has been corrected.

P1136 COBOL - "OPEN, CLOSE" STATEMENTS FOR PORT FILES

If FILESTATUS was not declared for a port file, the code generated for OPEN and CLOSE statement was incorrect. The correct code is now generated.

P1164 COBOL - GLOBAL "ISAM" FILES

The COBOL compiler now gives a syntax error for the use of global ISAM files.

P1325 COBOL - TRUNCATION WARNING FOR REPORT ITEMS

The compiler failed to give warning 302 "POSSIBLE TRUNCATION OF NON-ZERO DIGITS" when the receiving field was declared in the Report section. Now, the warning is given when appropriate.

P1327 COBOL - "COMPUTE" STATEMENT IN "WHEN" CLAUSE

A program with a COMPUTE statement in the WHEN clause of a SEARCH statement caused the compiler to fault with an INVALID INDEX. Now, such statements compile correctly.

P1328 COBOL - "BACKUPKIND" ATTRIBUTE

The compiler now uses the BACKUPKIND attribute in a file's description to represent a BACKUP clause in a SELECT statement.

"BACKUP TAPE" causes BACKUPKIND to be set to TAPE. "BACKUP DISK" causes BACKUPKIND to be set to DISK. Other forms of the BACKUP clause do not affect the BACKUPKIND attribute.

P1329 COBOL - ERRONEOUS RIGHT PARENTHESIS

An erroneous right parenthesis in a PICTURE string now causes syntax error 067: "PICTURE STRING ERROR". Previously, the parenthesis was treated as an insertion character.

P1373 COBOL - COMMAS IN "OPEN" STATEMENTS

On system software release Mark 32, commas in OPEN statements caused syntax error #121: "END OF STATEMENT EXPECTED". The incorrect error message is no longer emitted by the compiler.

P3190 COBOL - INTRINSIC "BINDINFO" MISMATCH

The COBOL compiler generated bindinfo which, in some cases, prevented binding with ALGOL and PL/I programs. The problem has been corrected.

P3217 COBOL - "MAXRECSIZE" OF REPORT FILES

Use of summary items in a report description sometimes caused the report file to be created with the wrong MAXRECSIZE. Now, the correct MAXRECSIZE is generated.

P3265 COBOL - "BLOCKSIZE" ALWAYS SET

The BLOCKSIZE file attribute was always being set by COBOL programs when a FILE STATUS key was specified. It is not necessary to set this attribute if the BLOCKSIZE is the same as the MAXRECSIZE. Unnecessary setting of this attribute has been eliminated.

P3308 COBOL - "PORT" FILES, PERFORM TERMINUS CODE

PORT file READ statements no longer cause erroneous returns from "performed" sections or paragraphs.

P3313 COBOL - TIMEDOUT CONDITION

For remote files with variable length records, a timedout condition would not be caught by a USE AFTER STANDARD ERROR procedure. The problem has been corrected.

P3761 COBOL - "SEG ARRAY" WITH "DATADICTINFO" SET

When compiling a program which invokes a large data base with DATADICTINFO set, the compilation no longer aborts with a SEG ARRAY error.

DOCUMENT CHANGES NOTES (D NOTES)

COBOL74

D4143 COBOL74 - "AT END ALLOWED FOR PORT FILE READ"

Add the following to page 8-75 of the COBOL74 manual:

FORMAT 1 or FORMAT 2 may be used for port files.

D4151 COBOL74 - DOCUMENTATION CORRECTION

The first sentence of the last paragraph on Page 7-34 of the COBOL ANSI-74 Reference Manual (Form No. 5011075) should state that the sign will be maintained in the right-most position (the documentation stated the left-most position).

D4173 COBOL74 - SEGMENT NUMBER

The user-specified segment number in a section heading has been changed.

Consequently, the first sentence of the second paragraph on Page 9-1 of the COBOL ANSI-74 Reference Manual (Form No. 5011075) should be changed to read as follows:

"The value of the segment number may range from 0 to 9999, allowing specification of up to 10,000 separate segments."

D4193 COBOL74 - ACCEPT FROM TIMER, "10" DIGITS

The result of the "ACCEPT <data-name> FROM TIMER" statement has been changed from 11 digits to 10 digits; the high-order digit has been eliminated.

D4202 COBOL74 - "USAGE IS ASCII" DELETED

The USAGE IS ASCII construct has been de-implemented. ASCII is no longer recognized as a COBOL74 reserved word.

D4203 COBOL74 - "STACK SIZE" CLAUSE ADDED

The STACK SIZE IS <nnn> clause has been added to the object computer entry. This clause is treated as documentation only.

D4253 COBOL74 - "I/O" STATUS FROM INVALID SUBFILE INDEX

The following changes should be made to the COBOL ANSI-74 Reference manual (Form No. 5011075), Page 6-15:

Under list item 2, Invalid Key, delete the paragraphs numbered 3 and 4.

Under list item 3, Permanent Error, add the following sentence to the end of the paragraph:

"A format 2 READ or format 2 WRITE on a port file with the contents of the ACTUAL KEY data item less than zero or greater than the number of subfiles in the file is a boundary violation."

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

COBOL74

P1018 COBOL74 - "OPEN REVERSE"

Syntax checking of the "OPEN ... REVERSE" statement has been corrected.

P1117 COBOL74 - LIBRARY TEMPLATE SPLITS ACROSS AREA BOUNDARY

The compiler generated library templates that crossed area boundaries, causing run-time errors when presence bit actions were taken. This problem has been corrected.

P1118 COBOL74 - STATUS KEY FOR INDEXED "I/O"

A value of 23 is now returned if an input-output statement executes unsuccessfully because the record identified by a key does not exist in the file.

P1136 COBOL74 - "OPEN, CLOSE" STATEMENTS FOR PORT FILES

If FILESTATUS was not declared for a port file, the code generated for OPEN and CLOSE statement was incorrect. The correct code is now generated.

P1325 COBOL74 - TRUNCATION WARNING FOR REPORT ITEMS

The compiler failed to give warning 302 "POSSIBLE TRUNCATION OF NON-ZERO DIGITS" when the receiving field was declared in the Report section. Now, the warning is given when appropriate.

P1327 COBOL74 - "COMPUTE" STATEMENT IN "WHEN" CLAUSE

A program with a COMPUTE statement in the WHEN clause of a SEARCH statement caused the compiler to fault with an INVALID INDEX. Now, such statements compile correctly.

P1329 COBOL74 - ERRONEOUS RIGHT PARENTHESIS

An erroneous right parenthesis in a PICTURE string now causes syntax error 067: "PICTURE STRING ERROR". Previously, the parenthesis was treated as an insertion character.

P3217 COBOL74 - "MAXRECSIZE" OF REPORT FILES

Use of summary items in a report description sometimes caused the report file to be created with the wrong MAXRECSIZE. Now, the correct MAXRECSIZE is generated.

P3264 COBOL74 - "OPEN EXTEND" WITH TAPE FILES

OPEN EXTEND with a tape file now works properly.

P3265 COBOL74 - "BLOCKSIZE" ALWAYS SET

The BLOCKSIZE file attribute was always being set by COBOL programs when a FILE STATUS key was specified. It is not necessary to set this attribute if the BLOCKSIZE is the same as the MAXRECSIZE. Unnecessary setting of this attribute has been eliminated.

P3761 COBOL74 - "SEG ARRAY" WITH "DATADICTINFO" SET

When compiling a program which invokes a large data base with DATADICTINFO set, the compilation no longer aborts with a SEG ARRAY error.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

CONFIGURATOR

P1333 CONFIGURATOR - UNKNOWN GROUP "ID"

An INVALID INDEX, caused when an unknown group id was encountered in an INSTALLATION statement, has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

CONTROLLER**D4136 CONTROLLER - "HOSTNSP" HANDLING**

A new keyword is available for the ID ODT message as a colon option, ":HOST".

For example, if "ID110:HOST" is entered, NSP 110 is initialized as a HOST NSP, whether or not a physical NSP110 exists and regardless of how it is declared in the NDL (as NSP or HOST).

(See HOSTNSPSUPPORT note D4141 for details of implementation.)

D4150 CONTROLLER - "ODT CANDIDATE" MESSAGE

The CANDIDATE ODT message displays information about offered supports that have not yet been matched. This command has the same function and syntax as the NW CANDIDATE command described in BNA note D3194.

```
----- CA -----|
| - CANDIDATE - | | -<port name>- |
```

D4199 CONTROLLER - INTERVAL LOGGING FOR FILES

File interval logging will log usage information of open files at time intervals. The logging will actually be done on the next file access, so that <time interval> is an approximation.

File interval logging is accomplished via the LI ODT message.

Syntax:

```
-- LI -----|
| | | | |
| - + -----|
| - - -----|
| - = --<time interval>-|
```

"LI" displays the current parameters. "LI+" and "LI-" enable and disable interval logging, respectively. "LI= <time interval>", where <time interval> must be a positive integer, sets the interval to the specified number of minutes. The default is 60; the range is 1 to 1440.

The entry format of the File Interval record is described in JOBFORMATTER note D4272.

D4222 CONTROLLER - "HN" "ODT" MESSAGE

The HN interrogation ODT message now displays the hostnames of any BNA hosts that are available for port dialogs or Host Services functions. The HN interrogation message is now allowed from a CANDE terminal.

D4245 CONTROLLER - "LT" "ODT" MESSAGE DEIMPLEMENTED

The ODT message LT (Load Trainable) has been deimplemented.

For many years, the train tables have been loaded automatically whenever needed, thus overriding the action of LT.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

CONTROLLER

P1052 CONTROLLER - "HARDCOPY" CORRECTION

This patch fixes the problem where messages were either duplicated or missed by SYSTEM/HARDCOPY.

P1053 CONTROLLER - RESOURCE CORRECTIONS

This patch fixes a problem where:

1. A job with tape resource set is stuck in a queue with MIXL=0;
2. The MIXL is then set to a non-zero number;
3. The job executes but causes the system to dump.

P1130 CONTROLLER - "ADM EVENT PER" CORRECTION

The following example now works correctly:

ADM EVENT PER <device>, PER <device> [,PER <device>,...]

P1165 CONTROLLER - CORRECT MISSPELLING

If a job is DSED out of a queue, the message "DSED BY OPERATOR" is now displayed instead of "DSED BY OPERATER".

P1203 CONTROLLER - "WAIT" CORRECTION

A problem has been corrected in which, if a WFL job were waiting on an RSVP, the job would

1. Not appear in the waiting entries list if it were the only waiting entry;

or

2. Appear in the waiting entries list if another entry exists (not a WFL job waiting on an RSVP) but with an incorrect entry count.

P1259 CONTROLLER - HALT/LOAD REASON CHECK

If the Halt/Load reason contains unviewable character(s), it is assumed to be in BCL and a conversion is made from BCL to EBCDIC. This conversion cannot be done on an EBCDIC system. Such conversions are now guarded against occurring.

P3236 CONTROLLER - "TERM FULLPAGE" CORRECTED

The FULLPAGE option on system ODTs is now preserved across Halt/Loads.

P3762 CONTROLLER - "CU" DISPLAY CORRECTION

A problem in which two blank lines were printed at the bottom of the CU display and ADM display of core usage has been corrected.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

DOCUMENT CHANGES NOTES (D NOTES)

CONTROLWARE

D3191 CONTROLWARE - CONTROLWARE VS. FIRMWARE, PACK CONTROLS

Prior to the Mark 31 system software release, FIRMWARE files were required for the B9380 (disk pack type 215) and B9385 (disk pack types 225 and 235) controllers. The firmware files were released with the following titles:

SYSTEM/FIRMWARE/215
SYSTEM/FIRMWARE/225AND235

The latter file was also copied as SYSTEM/FIRMWARE so that it could be used as the default in the LH ODT message.

Listings of the code files were also released with the following titles:

SYMBOL/FIRMWARE/215
SYMBOL/FIRMWARE/225AND235

The SYMBOL files were printer backup disk files to be used by Field Engineering.

The B9387 (disk pack types 206 and 207) controller was qualified on the Mark 31 (31.245) system software release. The FIRMWARE files described above were released on the Mark 31 PR1 release, together with two additional tapes: B9385CW and B9387CW. The contents of the B9387CW tape was described in GENERAL note D3126 on the Mark 31 and Mark 31 PR1 releases; the contents of the B9385CW tape was described in GENERAL note D3180 on the Mark 31 PR1 release. The code files on the previously-released CW tapes had titles CONTROLWARE/=; the code listings printer backup disk files had titles LISTING/=.

The above naming conventions for the code files required for the disk pack controllers will be revised by the Mark 33 system software release.

The term CONTROLWARE will be reserved for use in file titles of code files required for disk pack controllers. The term FIRMWARE will be used in file titles of other I/O subsystem code files; e.g., Data Link Processors (DLP) and Network Support Processors (NSP).

The first phase of the plan, described below, was been implemented on the Mark 32 system software release, and is operative on the Mark 32 PR1 release as well.

Note the B6900 and B5900 (referred to as "MLIP") systems require different FIRMWARE files from B6700 and B6800 systems.

The following table represents the current controller numbers and disk pack types:

Controller	Disk Pack
B9380	215
B9385 *	225,235
B9387	206,207

* Note:

Previous levels of B9385 were released as B9383 and B9384.

Mark 32 FIRMWARE and CONTROLWARE Naming Conventions

The naming conventions for the code files for B6700 and B6800 disk pack controllers remain the same as in prior releases. In addition, the file for the B9387 (disk pack types 206 and 207) controller has the following title:

SYSTEM/FIRMWARE/206AND207

Thus the code files for B6700 and B6800 are the following:

SYSTEM/FIRMWARE (copy of SYSTEM/FIRMWARE/225AND235)
SYSTEM/FIRMWARE/215
SYSTEM/FIRMWARE/225AND235
SYSTEM/FIRMWARE/206AND207

The SYMBOL/FIRMWARE files are NOT contained on the Mark 32 system tape; they are contained on the B9385CW, B9387CW and B9387CWLIST tapes.

The naming convention for MLIP systems controlware files is the following:

CONTROLWARE/BXXXX

The following files are contained on the Mark 32 software release SYSTEM tape:

CONTROLWARE/B9385 (disk pack types 225 and 235)
CONTROLWARE/B9387 (disk pack types 206 and 207)

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

The printer backup disk files on the B9385CW, B9387CW and B9387CWLIST tapes have titles CWLISTING/=.

Mark 33 CONTROLWARE Naming Conventions

The code files for MLIP systems controlware files will use the same convention as on the Mark 32 system software release.

The titles of the code files for B6700 and B6800 systems will be changed to the following:

CONTROLWARE/MPX/B9380 (disk pack type 215)
 CONTROLWARE/MPX/B9385 (disk pack type 225 and 235)
 CONTROLWARE/MPX/B9387 (disk pack type 206 and 207)

There will be no file equivalent to SYSTEM/FIRMWARE.

The printer backup disk files on the B9385CW, B9387CW and B9387CWLIST tapes will have titles CWLISTING/=.

The syntax of the LH ODT message and the checking code in SYSTEM/PATCHCONTROLWARE will be modified to accommodate these changes.

D4251 CONTROLWARE - MARK "32 PR1" DISK PACK CONTROLWARE FILES

This note replaces the following notes previously issued:

D3180 Mark 31 PR1 release
 D3126 Mark 31 release
 D3200 Mark 32 release

Controlware files for the Disk Pack Controllers B9380, B9385, and B9387 are maintained and qualified by the Downingtown plant. A release tape is forwarded to the Computer Systems Group (CSG) plants responsible for the host systems. The applicable files are contained on the system release tapes and qualified.

Controlware files used by MLIP systems Peripheral Test Driver (PTD) are described in the file PTD/DOC/MAINT/HT on the PTDTESTS tape which accompanies this release.

Note: Previous levels of disk pack controller B9385 were released as B9383 and B9384.

The titles of the files (for the appropriate combination of disk pack controllers and host transfer controls) on the master tape from the Downingtown plant are changed at each release.

The current version of the controlware files required for normal large system operations are copied (using file naming conventions described in CONTROLWARE note D3191 to the Mark 32 PR1 SYSTEM tape and titled as follows:

Title on 32 PR1 SYSTEM Tape	Title on 'CW' Tape	System, Pack Types
SYSTEM/FIRMWARE	CONTROLWARE/HSTLME #	B6800,B6700 (225,235)
SYSTEM/FIRMWARE/215	CONTROLWARE/HSTLNE #	B6800,B6700 (215)
SYSTEM/FIRMWARE/225AND235	CONTROLWARE/HSTLME #	B6800,B6700 (225,235)
SYSTEM/FIRMWARE/206AND207	CONTROLWARE/HSTLPD ##	B6800 (206,207)
CONTROLWARE/B9385 *	CONTROLWARE/HSTLKC #	B6900,B5900 (225,235)
CONTROLWARE/B9387 **	CONTROLWARE/HSTLQD ##	B6900,B5900 (206,207)

Note: B9385CW
 ## Note: B9387CW

* Note: Includes patch T3702-1 which enables binary addressing which is required for this configuration.

** Note: "Seek timeout" is reported instead of "not ready" when loss of servo phaselock occurs.

The second identifier in the file titles of the controlware files on the 'CW' tapes which are required for normal system operation are composed of the concatenation of the following items: 'HSTL', 'L', 'V'.

HSTL denotes Host Load file (B6700, B6800 system with Model III Multiplexor or MLIP system)

L denotes the combination of host controls (HTC1A or HT-DLP) and disk pack controller, indicated with a letter.

V denotes the current version, indicated with a revision letter: A,B,C, etc.

The current allowable letters for 'L' are denoted in the following table.

Note: 'Col ID' in the following table denotes the number of hex digits of relevant firmware level information.

Note: 'Section' in the following table denotes the section number in the document CWLISTING/A5615A on the B9385CW and B9387CW tapes.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

L	32 V	Col ID	Disk Pack Controllers	Disk Pack Types	Host Control	Section	System
K	C	D2C3	B9385	225,235	HTDLP	4.1	B6900,B5900
M	E	D4C5	B9385	225,235	HTC1A	4.2	B6700/B6800
N	E	D5C5	B9383	215	HTC1A	4.3	B6700/B6800
P	D	D7C4	B9387	206,207	HTC1A	4.4	B6800
Q	D	D8C4	B9387	206,207	HTDLP	4.5	B6900,B5900
R *	B	D9C2	B9385	225,235	HTC1A		B6700/B6800

* Note: 'Standalone' IVR, contained on the CW tapes, is not required for normal system operation.

Loading Controlware Files

SYSTEM/LOADER is used to load diskpack controlware files on multiplexor and MLIP systems. The format of the command is the following:

```
-- LH -- PK <nnn> --- MPX <m> PATH <p> ----->
      | - VIA <port #> <lem #> <dlp #> - |
>- <filename> FROM <tape name> -----|
```

Examples:

```
"LH PK 45 MPX 1 PATH 2 CONTROLWARE/B9387 FROM BSYSTEM ;"
"LH PK 45 VIA 0 0 5 CONTROLWARE/B9385 FROM BSYSTEM; "
```

NOTE: A space is required between PK and <pack #>.

Contents of 'CW' Tapes

All the files for normal system operation, diagnostic and maintenance operations for B6000 series systems are copied to the tapes B9385CW, B9387CW and B9387CWLIST.

The B9385CW tape contains all the controlware and CWLISTING files required for the B9385 disk pack controller for B6700, B6800, B6900 and B5900 systems.

The B9387CW tape contains all the controlware files required for the B9387 disk pack controller for B6800, B6900 and B5900 systems.

The B9387CWLIST tape contains all the CWLISTING files required for the B9387 disk pack controller for B6800, B6900 and B5900 systems.

The file CWLISTING/A5615A is a printer backup disk file containing the system notes describing the changes made to the controlware files since the previous Downingtown release. After copying from tape to disk, the file may be printed using SYSTEM/BACKUP with a command similar to the following:

```
?PB "CWLISTING/A5615A" LP14
```

For each 'CONTROLWARE/<identifier>' file, there is a corresponding 'CWLISTING/<identifier>' file.

The CWLISTING/= files are Large Systems series printer backup disk files. A single file may be printed using the following:

```
COPY CWLISTING/<identifier> FROM <tape identifier>; END
?PB "CWLISTING/<identifier>" LP14.
```

All the files may be printed using the following:

```
COPY CWLISTING/<identifier> FROM <tape identifier>; END
?PB "CWLISTING" LP14.
Applied Patches
```

The following patches have been applied (PCN bars in the right margin indicate changes since the Mark 32 release):

Patch 09221-2 BX383/4/5 Controlware Release 2.2 (Code ME)

This patch corrects the following problems:

1. If a Unit Busy R/D was generated during a conditional Subsystem Poll (indicating no units are seek ready) and the controller was lock enabled, Controlware erroneously locked the controller.
2. If a seek timeout status was detected from a drive, Controlware erroneously generated a Not Ready R/D instead of a Seek Timeout R/D.

3. If a restore had to be performed on a drive at unit select or initial seek time, the host could accidentally be put into a state which could cause an interface parity error.
4. If a Conditional Cancel was sent from the host at the same time Controlware was trying to place the host into R/D mode, the interface would hang. Controlware now checks the host status after R/D mode and reissues an R/D mode command if Conditional Cancel is present. A new R/D TAG 069 has been added to Controlware.

This patch also sets bit 02 (most significant bit=01) of the buffer memory location "PATCH_CODE" so that a Read Memory C/D may be used to determine if the patch is installed.

Host Load file (HSTLME) patch:

```
CC PATCH HSTLME 0 0 5496 4 E040
CC PATCH HSTLME 0 0 9484 4 D283
CC PATCH HSTLME 0 0 11672 4 F8B2
CC PATCH HSTLME 0 0 6956 4 6BCB
CC PATCH HSTLME 0 0 16316 4 A12C
CC PATCH HSTLME 0 0 17992 12 F871A0054AE6
CC PATCH HSTLME 0 0 16392 4 F844
CC PATCH HSTLME 0 0 16420 4 F819
CC PATCH HSTLME 0 0 16428 4 4BCE
CC PATCH HSTLME 0 0 18004 20 F84CB8FBF8C966F5C069
CC PATCH HSTLME 0 0 18024 20 6B8E66DBC049F8844A46
```

The following files have been updated:

```
SYSTEM/FIRMWARE          on SYSTEM  tape
SYSTEM/FIRMWARE/225AND235 on SYSTEM  tape
CONTROLWARE/HSTLME       on B9385CW tape
```

Patch 09921-3 B9387 Controlware Release 1.3 (Code PD)

This patch forces Controlware to generate a Seek Timeout R/D when a "loss of servo phaselock" status is received from a 207 drive. Controlware previously generated a Not Ready R/D for this condition. This patch also sets bit 09 (least significant bit=00) of buffer memory location "PATCH_CODE" so that a Ready Memory C/D may be used to determine if the patch is installed.

Host Load file (HSTLPD) patch:

```
CC PATCH HSTLPD 0 0 176956 4 0502
CC PATCH HSTLPD 0 0 184072 8 12070BA9
CC PATCH HSTLPD 0 0 189172 20 02AC1C11227C1C113203
```

Patch 09921-4 B9387 Controlware Release 1.3 (Code PD)

This patch corrects the following problems:

1. If a Unit Busy R/D was generated during a conditional Subsystem Poll (indicating no units are seek ready) and the controller was lock enabled, Controlware erroneously locked the controller.
2. If a Conditional Cancel was sent from the host at the same time Controlware was trying to place the host into R/D mode, the interface would hang. Controlware now checks the host status after R/D mode and reissues an R/D mode command if Conditional Cancel is present. A new R/D TAG 069 has been added to the Controlware.

This patch also sets bit 10 (least significant bit=00) of buffer memory location "PATCH_CODE" so that a Read Memory C/D may be used to determine if the patch is installed.

Host Load file (HSTLPD) patch:

```
CC PATCH HSTLPD 0 0 176960 4 0504
CC PATCH HSTLPD 0 0 181800 4 2512
CC PATCH HSTLPD 0 0 188780 12 02D71BAE1000
CC PATCH HSTLPD 0 0 189192 20 04721000268220001000
CC PATCH HSTLPD 0 0 189212 20 03FB1207174A25002569
```

The following files have been updated:

```
SYSTEM/FIRMWARE/206AND207 on SYSTEM  tape
CONTROLWARE/HSTLPD       on B9387CW  tape
```

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

Patch 09921-5 B9387 Controlware Release 1.3 (Code QD)

This patch corrects the following problems:

1. The Large Systems HT-DLP cannot handle a state change from State 6 (Command Descriptor) to State 4 (Control Message). Controlware no longer attempts that state change.
2. If a Conditional Cancel was set from the host at the same time Controlware was trying to place the host into R/D mode, the interface would hang. Controlware now checks the host status after R/D mode and reissues an R/D mode command if Conditional Cancel is present. A new R/D TAG 069 has been added to the Controlware.

This patch also sets bit 09 (least significant bit=00) of buffer memory location "PATCH_CODE" so that a Read Memory C/D may be used to determine if the patch is installed.

Host Load file (HSTLQD) patch

```
CC PATCH HSTLQD 0 0 176956 4 0502
CC PATCH HSTLQD 0 0 179848 4 3BA0
CC PATCH HSTLQD 0 0 189136 24 35E3360B28131223020507BE
CC PATCH HSTLQD 0 0 189028 8 12070BA6
CC PATCH HSTLQD 0 0 189160 20 34723CD3037306923248
CC PATCH HSTLQD 0 0 189180 20 240B1207179425001569
```

The following files have been updated:

```
CONTROLWARE/B9387          on SYSTEM  tape
CONTROLWARE/HSTLQD        on B9387CW  tape
```

Patch 10432-1 B9385 Controlware Release 2.2 (Code KC)

This patch corrects the following problems:

1. If there were a hardware failure, such that a HOST STATUS interrupt was generated with no exception bits set in the status, Controlware erroneously generated a Conditional Cancel R/D instead of a Host Error R/D.
2. Controlware now returns a Unit Busy R/D instead of a Disk Error R/D if a conditional C/D causes a seeking unit to be selected.

This patch also sets bit 01 (most significant bit=01) of buffer memory location "PATCH_CODE" so that a Read Memory C/D may be used to determine if the patch is installed.

Host Load File (HSTLKC) patch:

```
CC PATCH HSTLKC 0 0 5500 4 E080
CC PATCH HSTLKC 0 0 13316 4 F801
CC PATCH HSTLKC 0 0 13328 4 4BF0
CC PATCH HSTLKC 0 0 18140 8 81094741
CC PATCH HSTLKC 0 0 15436 4 4BF2
CC PATCH HSTLKC 0 0 18148 20 66ACA081A003A03546E4
```

The following files have been updated:

```
CONTROLWARE/HSTLKC        on B9385CW tape
CONTROLWARE/B9385        on BSYSTEM  tape
```

Patch 10342-4 B9387 Controlware Release 1.3 (Code QD)

This patch corrects the following problems:

1. If there were a hardware failure, such that a Host status interrupt was generated with no exception bits set in the status, Controlware erroneously generated a Conditional Cancel R/D instead of a Host Error R/D.
2. A design problem has been identified for the disk pack type 207 drive which allows messages to be corrupted if the spindle goes seek ready while a CM or DM is in progress. Also, it has been intermittently observed that the B9387-4 Disk DDP misinterprets Drive Messages. If either problem occurs when Controlware is requesting a long DM to determine the spindle type, Controlware can mistake a disk pack type 206 or disk pack type 207 spindle for a disk pack type 205 spindle. This patch forces R/DTAG 275 (Disk Error R/D) to be generated if the long DM read by Controlware from the DM register to check the spindle type does not have the continue bit (bit 8) set.
3. Controlware now returns a Unit Busy R/D instead of a Disk Error R/D if a conditional C/D causes a seeking unit to be selected.

This patch also sets bit 11 (least significant bit=00) of buffer memory location "PATCH_CODE" so that a Read Memory C/D may be used to determine if the patch is installed.

Host Load file (HSTLQD) patch:

```

CC PATCH HSTLQD 0 0 176964 4 3508
CC PATCH HSTLQD 0 0 185064 16 3FFB100003490C1C
CC PATCH HSTLQD 0 0 188568 8 12072BB3
CC PATCH HSTLQD 0 0 189212 20 25B7360B12AD081805BA
CC PATCH HSTLQD 0 0 189232 16 360B138D06B53203
CC PATCH HSTLQD 0 0 188504 4 2B8C
CC PATCH HSTLQD 0 0 189248 24 220727321581250015031535
    
```

The following files have been updated:

```

CONTROLWARE/HSTLQD      on B9387CW tape
CONTROLWARE/B9387       on BSYSTEM tape
    
```

Directories of the 'CW' Tapes

The following tables show the files contained on the 'CW' tapes. The entries in the tables are sorted by file titles and do not represent the order in which the files occur on the tapes. An asterisk (***) after "SEGS" denotes an uncrunched file.

B9385CW

LIBRARY TAPE: B9385CW HAS 9 FILES, 41110 SEGMENTS

* AFTER SEGS - FILE IS UNCRUNCHED

FILE TITLE	SEGS	TIMESTAMP	FILEKIND
*CONTROLWARE/D5X12C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/HSTLKC,	%	1000*03-16-81	FIRMWARE
*CONTROLWARE/HSTLME,	%	1000*10-29-80	FIRMWARE
*CONTROLWARE/HSTLRB,	%	1000*05-07-80	FIRMWARE
*CWLISTING/A5615A,	%	4050 05-08-80	BACKUPDISK
*CWLISTING/D5X12L,	%	9930 05-08-80	BACKUPDISK
*CWLISTING/LISTKC,	%	8060 05-08-80	BACKUPDISK
*CWLISTING/LISTME,	%	8250 05-08-80	BACKUPDISK
*CWLISTING/LISTRB,	%	6820 05-08-80	BACKUPDISK

B9387CW

LIBRARY TAPE: B9387CW HAS 26 FILES, 29050 SEGMENTS

* AFTER SEGS - FILE IS UNCRUNCHED

FILE TITLE	SEGS	TIMESTAMP	FILEKIND
*CONTROLWARE/D7MB1C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MB2C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MC1C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MD1C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MD2C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MD3C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MD4C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MD5C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MD6C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MD7C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7ME1C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MF1C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MF2C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MF3C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MF4C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MF5C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MF6C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MF7C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MH1C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MP1C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MP2C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7MP3C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/D7Y12C,	%	1000*05-07-80	FIRMWARE
*CONTROLWARE/HSTLPD,	%	1000*11-06-80	FIRMWARE
*CONTROLWARE/HSTLQD,	%	1000*03-16-81	FIRMWARE
*CWLISTING/A5615A,	%	4050 05-08-80	BACKUPDISK

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

B9387CWLIST

LIBRARY TAPE: B9387CWLIST HAS 39 FILES, 190320 SEGMENTS

* AFTER SEGS - FILE IS UNCRUNCHED

FILE TITLE	SEGS	TIMESTMP	FILEKIND
*CWLISTING/D7AA2L,	% 1260	05-08-80	BACKUPDISK
*CWLISTING/D7AA3L,	% 710	05-08-80	BACKUPDISK
*CWLISTING/D7AA4L,	% 710	05-08-80	BACKUPDISK
*CWLISTING/D7AA5L,	% 710	05-08-80	BACKUPDISK
*CWLISTING/D7AA6L,	% 730	05-08-80	BACKUPDISK
*CWLISTING/D7AB1L,	% 5770	05-08-80	BACKUPDISK
*CWLISTING/D7AB2L,	% 290	05-08-80	BACKUPDISK
*CWLISTING/D7AD1L,	% 4160	05-08-80	BACKUPDISK
*CWLISTING/D7AD2L,	% 3040	05-08-80	BACKUPDISK
*CWLISTING/D7AD3L,	% 2250	05-08-80	BACKUPDISK
*CWLISTING/D7AD4L,	% 2200	05-08-80	BACKUPDISK
*CWLISTING/D7AD5L,	% 3030	05-08-80	BACKUPDISK
*CWLISTING/D7AD6L,	% 4720	05-08-80	BACKUPDISK
*CWLISTING/D7AD7L,	% 3260	05-08-80	BACKUPDISK
*CWLISTING/D7AE1L,	% 3170	05-08-80	BACKUPDISK
*CWLISTING/D7AF1L,	% 3640	05-08-80	BACKUPDISK
*CWLISTING/D7AF2L,	% 2920	05-08-80	BACKUPDISK
*CWLISTING/D7AF3L,	% 2080	05-08-80	BACKUPDISK
*CWLISTING/D7AF4L,	% 3390	05-08-80	BACKUPDISK
*CWLISTING/D7AF5L,	% 2980	05-08-80	BACKUPDISK
*CWLISTING/D7AF6L,	% 4420	05-08-80	BACKUPDISK
*CWLISTING/D7AF7L,	% 4050	05-08-80	BACKUPDISK
*CWLISTING/D7AP0L,	% 2670	05-08-80	BACKUPDISK
*CWLISTING/D7AP1L,	% 4050	05-08-80	BACKUPDISK
*CWLISTING/D7AP2L,	% 4390	05-08-80	BACKUPDISK
*CWLISTING/D7AP3L,	% 370	05-08-80	BACKUPDISK
*CWLISTING/D7AZ1L,	% 930	05-08-80	BACKUPDISK
*CWLISTING/D7AZ2L,	% 870	05-08-80	BACKUPDISK
*CWLISTING/D7AZ3L,	% 860	05-08-80	BACKUPDISK
*CWLISTING/D7AZ4L,	% 870	05-08-80	BACKUPDISK
*CWLISTING/D7AZ5L,	% 1420	05-08-80	BACKUPDISK
*CWLISTING/D7MC1L,	% 2480	05-08-80	BACKUPDISK
*CWLISTING/D7MH1L,	% 2950	05-08-80	BACKUPDISK
*CWLISTING/D7PR1L,	% 3800	05-08-80	BACKUPDISK
*CWLISTING/D7QXRL,	% 34820	05-08-80	BACKUPDISK
*CWLISTING/D7RXRL,	% 43030	05-08-80	BACKUPDISK
*CWLISTING/D7Y12L,	% 10710	05-08-80	BACKUPDISK
*CWLISTING/LISTPD,	% 8360	05-08-80	BACKUPDISK
*CWLISTING/LISTQD,	% 8250	05-08-80	BACKUPDISK

SYSTEM/PATCHCONTROLWARE

For a description of SYSTEM/PATCHCONTROLWARE, see note D3128, Mark 32 release.

DOCUMENT CHANGES NOTES (D NOTES)

DATA COMMUNICATIONS

D4226 DATACOM - CHANGE NAME OF "HOSTNSP" STACK

To avoid confusion with the DCC stack, the external name of the HOSTNSP stack has been changed from "NSP/<nnn>" to "HOSTNSP/<nnn>".

D4242 DATACOM - "NSP" DATACOM INITIALIZATION MESSAGES

This note briefly summarizes some of the MCP messages relating to NSP datacom initialization.

* DC - UNIT <nn> IS NOT AN NSP

Either the specified unit number refers to a non-existent unit or the actual unit is not an NSP (e.g., a magnetic tape).

* DC - NSP <nn> IS ALREADY IN USE

Either the specified NSP is reserved for maintenance, the path to the specified NSP is reserved, or the path is off-line.

* DC - UNABLE TO LOAD FIRMWARE FOR NSP <nn>

Either

- 1) there are disk parity errors on the file FIRMWARE/NSP,
- 2) the firmware file has become corrupted,
- 3) the RAM firmware in the firmware file is incompatible in level with the PROM firmware in the specified NSP,
- 4) the specified NSP has insufficient memory to load the firmware file,

or

- 5) the specified NSP has a hardware problem (such as a parity error).

* DC - LSP <nn> OFF-LINE

The specified unit number refers to a non-existent unit (i.e., the unit does not appear when you do a "GC" from the ODT).

* DC - UNIT <nn> IS NOT A LSP

The actual unit is not an LSP.

* DC - LSP <nn> IS ALREADY IN USE

The specified LSP is assigned to either another NSP or to PTID.

* DC - LSP <nn> PATH ERROR

Either the specified LSP is reserved for maintenance, the path to the specified LSP is reserved, or the path is off-line.

* DC - NSP <nn> HAS NO PATH TO LSP <nn>

There is no path from the specified NSP to the specified LSP.

* DC - NSP <nn> UNABLE TO CLEAR LSP <nn>

An attempt by the MCP to perform a "selective clear" on the specified LSP via the specified NSP failed.

NSP <nn> NOT READY

An RSVP which indicates that the MCP successfully loaded the NSP firmware but then found that the NSP was not in an active state. If the stack is "OK"ed, the MCP will attempt to reload the firmware and start again.

Once an NSP base is manually cleared, the MCP must be Halt/Loaded to be able to see the NSP again, since the BCC masks were clobbered by the base clear.

Effective with MCP release 32 PR1, the MCP performs a "selective clear" on an NSP

- 1) before each attempt to load the NSP firmware,

2) when terminating the datacom stack for the NSP,

and

3) after any failed attempt to take an NSP memory dump (which will also terminate that datacom stack).

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DATA COMMUNICATIONS

P1184 DATACOM - "HOST NSP" SHUTDOWN

A problem of attempting to shutdown the HOST NSP support library when the library was not really running has been corrected.

P1185 DATACOM - DATACOM ERROR MESSAGE CORRECTION

A problem on MLIP systems where an error message was being built which was one character too long to fit in the text array has been corrected.

P1186 DATACOM - ERROR HANDLING IN "FIREOFFDCP"

A problem on MLIP systems where, under certain error conditions, a lock was not being liberated has been corrected.

P1187 DATACOM - UNSUCCESSFUL "I/O" MESSAGE

A problem where an incorrect DLP unit number was being reported on an unsuccessful I/O from an LSP has been corrected.

P1273 DATACOM - "B5900" COMPATIBILITY

Miscellaneous changes for compatibility with the B5900 have been made for datacom support. No functional changes are involved.

P1334 DATACOM - RETRY COUNTS ON OUTPUT RESULTS

A problem on MLIP systems in which transmission numbers and retry counts were not getting returned on Class 5 results has been corrected.

P1335 DATACOM - "LINESWAP" , "REINITIALIZATION" CORRECTIONS

The performance of reinitializing a faulted or suspended line on MLIP systems has been improved. Continuous reinitializations after a successful reinitialization have been avoided; also, a subsequent NDII line controller fault has been avoided.

A new feature has been implemented: the SWAPLINE reconfiguration request can be used to explicitly reinitialize any line by requesting (via a DCWRITE) that a line be swapped with itself.

P1338 DATACOM - CHANGE "NSP" AUTORECOVERY

If the NIF prefix is changed across a Halt/Load, the MCP will not attempt Autorecovery of NSPs but will instead display the following message:

"*DC-NSP AUTO RECOVERY ABORTED. NEW NIF LOADED".

Restarting the wrong NSP has been avoided.

P1340 DATACOM - "FORGETCHECK" OUT OF "STARTSYSTEM"

On B6900 systems, if datacom were active prior to a Halt/Load, a FORGETCHECK dump would occur at the end of procedure STARTSYSTEM. This problem has been corrected.

P1341 DATACOM - "FIREOFFDCP" PROBLEM

When initializing datacom by an MCS, a deadlock could result if a problem is encountered in getting to the unit or reading the NIF. This deadlock has been corrected.

P1342 DATACOM - DATACOM TABLE PROBLEMS

A problem in taking down and bringing back up an NSP when datacom tables were still allocated has been corrected.

P1343 DATACOM - "INVALID OP" IN "DCINITIAL"

On MLIP systems, when AUTODC was set and datacom was initiated by an MCS, DCINITIAL got an INVALID OP. This problem has been corrected.

P1344 DATACOM - MESSAGES FOR "DC" INITIALIZATION FAILURE

The message displayed when datacom initialization fails now includes the DCP or NSP number.

P1345 DATACOM - LOOP TIMEOUT

The major cause of a loop timeout on a scanout to the DCP (the DCP's switches being set incorrectly) has been corrected.

P1358 DATACOM - "FIREOFFDCP" PROBLEM

On MLIP systems, a problem in which a FORGETCHECK occurred when an error occurred in FIREOFFDCP has been corrected.

P1359 DATACOM - RECALL TIMING WINDOW

On MLIP systems, a problem which caused an INVALID OP in DCCONTROL when a "make station ready" operation occurred during a recall has been corrected.

P1360 DATACOM - DIALOUT PROBLEM

On MLIP systems, a problem in dialout which caused an INVALID INDEX when ENDOFNUMBER was true has been corrected.

P1361 DATACOM - "BLOCKEXIT" MOM PROBLEM

On MLIP systems, a potential problem existed where BLOCKEXIT was leaving a mom descriptor to a forgotten GETAREA. This problem has been corrected.

P3205 DATACOM - CHANGE "NSP" INITIALIZATION FAILED MESSAGE

The NSP initialization failed message has been changed to show the physical unit number instead of the relative number.

P3206 DATACOM - REJECTED REQUEST

A problem which appears when a DELETE request issued by DCRECON is rejected has been corrected. A problem has also been corrected which occurs when an unsuccessful I/O result is returned as a result to a DCRECON request.

P3207 DATACOM - "DCRECON" TIMING PROBLEM

On MLIP systems, if procedure HANDLESRESULT lost control at the wrong instant, DCRECON might be unable to queue a request to the NSP. Furthermore, DCRECON was not properly handling the error and would subsequently fault. Both of these problems have been corrected.

P3208 DATACOM - CALL "BLASTUNIT"

If DCINITIAL failed during the latter stages of NSP initialization, there were circumstances under which it would not call BLASTUNIT to clear a hung I/O to the NSP. This would result in a message "I/O HUNG OR SYSTEM ERROR" being displayed when datacom terminated and could lead to further problems when datacom was subsequently reinitialized on that NSP. All such problems have been corrected.

P3242 DATACOM - QUEUEING BETWEEN "DCINITIAL/DCCONTROL"

The queueing between DCINITIAL and DCCONTROL has been corrected. Previously, only the first item in the DCPRESULTQ was being handled.

P3243 DATACOM - INITIALIZATION ABORTED MESSAGE

The message which is displayed when an NSP aborts during initialization has been changed to include the NSP number. Also, the following problem has been corrected: An attempt was made to reference an NSP when DCINITIAL actually was initiated with DCBOOTSTROPCODE for table building only.

P3244 DATACOM - OVERRIDE LOCAL OPTIMIZATION IN "DCCONTROL"

For MLIP datacom, some local optimization in DCCONTROL that was attempted on an earlier release has been defeated, in order to make the code more readable and maintainable.

P3245 DATACOM - MARK LINE DEAD FOR "LINEDELETED" RESULT

For MLIP datacom, the following problem has been corrected: The wrong DLS was being used to mark a line as needing reinitialization following a spontaneous line deleted result from the NSP.

P3246 DATACOM - UNIT LEFT ASSIGNED FOR "NSP"

The following problem has been corrected: An LSP was being left assigned to the DCCONTROL stack when an error was encountered during the initialization of a HOSTNSP.

P3247 DATACOM - "STATION NOT READY" HANDLING

The handling of a STATION NOT READY result has been corrected for the case when multiple MAKE STATION NOT READY requests were outstanding but only one reply was forthcoming.

P3249 DATACOM - IMPROVE TESTS

It was possible that the MCP's test to determine whether datacom was active could get the wrong answer. The test has been corrected.

P3260 DATACOM - "DCINITIATEMCS" STACK OVERFLOW

If cataloging were in effect on an MLIP system, and if the AUTODC option were set so that an MCS could be automatically initiated, and if the DCINITIAL auditing option were enabled, then a fatal dump out of DCINITIATEMCS occurred because of a stack overflow fault. This problem has been corrected.

P3261 DATACOM - MISCELLANEOUS "HOSTNSP" CORRECTIONS

The following problems have been corrected:

1. The HOSTNSP SUPPORT library was left running following a DCINITIAL initialization failure.
2. Graph edges have been added between the DCCONTROL stack and the HOSTNSP SUPPORT library so that the library is marked as in use.

P3262 DATACOM - ADD STATION WITHOUT LINE ASSIGNMENT

The following problem existed on MLIP systems: DCRECON got a "BAD NIF RECORD NUMBER" error in attempting to add a station with no line assignment to a line. The problem, caused by an error in building the DCPCODE file in DCINITIAL, has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

DCALGOL

D3371 DCALGOL - PRIORITY OUTPUT

The following DCWRITE WRITE (TYPE=33) modifications have been made:

MSG[1].[47:1] = 0
MSG[1].[46:7] = Priority of output

MSG [1].[46:7] contains the priority of the message. If this field is non-zero, the DCP will insert the message into the station queue after any other messages of higher or equal priority but before messages of lesser priority. In this way, an MCS may cause messages to be transmitted in an order different from that in which they were actually given to the DCP. There are 128 different levels of priority with zero being the lowest priority and 127 being the highest the MCS can produce.

D4234 DCALGOL - DOCUMENTATION CHANGES

On Page 6-66 of the DCALGOL Reference Manual (Form No. 5011430), the parameters/fields for FRSN should read as follows:

"MSG[0].[23:24]=FRSN".

On Page 8-38, replace "3.1 file attribute TIMESTAMP" with "file TIMESTAMP".

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DCALGOL

P1123 DCALGOL - ERRONEOUS SYNTAX ERROR ON "DCALGOL" MESSAGE

The DCALGOL compiler gave an erroneous syntax error message on the following statement:

```
REPLACE P:A BY POINTER(MSG[0]) . . . ;
```

where P is a pointer, A is a character array, and MSG is a MESSAGE <identifier>.

This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DCP PROGRAM GENERATOR

P1000 DCPPROGEN - THREE ADDRESS CHARACTERS

When three address characters were used as the number of address characters in a TERMINAL DEFINITION, the correct code for using the TRANSMIT ADDRESS was not used. This problem has been corrected.

P1001 DCPPROGEN - CALL DESIGNATION WHEN STATION CHANGES

Previously, in line control assignments of the following form, the station variable was updated but the current station reference was not changed:

```
STATION[<bit no.>]=<boolean value>
```

This problem has been corrected.

P1068 DCPPROGEN - "ASL" CLEARS "MAO"

When an assignment of the following form was used in a request set, some extra bits were being set in the ASL result:

```
<byte variable> = <literal> ASL <byte variable>
```

This problem has been corrected.

P3218 DCPPROGEN - PRIORITY OUTPUT HANDLING

A problem in the DCP implementation of priority output handling, which could lead to DCP fault 4s, has been corrected.

P3219 DCPPROGEN - SET DIAL DATA BIT ON EACH DIALOUT

To avoid problems in which the data pointer valid bit of an ACIII dialout CB could be inadvertently reset, each dialout request now unconditionally sets the data pointer valid bit before using the dialout CB.

P3763 DCPPROGEN - "DCP" FAULT "8"

A DCP Fault 8 could occur or the request set could branch to some other code under the following circumstances:

1. DCP sequence mode was in use, and
2. DCP code was in main memory, not local, and
3. Initialize sequence construct was used.

This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

DIAGNOSTIC MCS

D4114 DIAGNOSTMCS - AUTOMATIC 'DCWRITE ERROR' REPORTING

DCSYSTEMTABLES will now be called to analyze DCWRITE ERRORS. This has two consequences:

1. New DCWRITE ERROR types will not cause an invalid index.
2. The text of the error given when DEBUG is set may be slightly different from what it has been in the past.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DIAGNOSTIC MCS

P1301 DIAGNOSTMCS - "SS ALL" STATION NOT ATTACHED

If a station attached to DIAGNOSTICMCS were transferred to another MCS and an "SS ALL" message were sent, DIAGNOSTICMCS would get an INVALID INDEX. This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - GENERAL

P3343 DMSII - MISLEADING "IOERRORS"

With the introduction of the B6900 (MLIP) system, end-of-tape handling was slightly modified. This change could cause misleading IOERRORs on a READ operation (IOERRORTYPE=5). This situation has been corrected; the user no longer receives an IOERROR.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - ACCESSROUTINES

P1004 ACR - INCORRECT KEY/DATA MISMATCH, WRONG RECORD

Previously, spurious key data mismatches occurred when records were retrieved via sets which allowed duplicates.

For example,

Program A found the entry for record R in the set S and saved the key value. It then initiated the I/O to read the data set record, gave up all locks, then waited for the I/O to complete. At this time, Program B changed the key for record R. Program A resumed and retrieved record R. Now, however, the saved off key value does not match the new key in the data set record.

If DATACHECK2 is set, a KEY/DATA MISMATCH occurs; otherwise, the program will retrieve the data set record with the modified key. This problem has been corrected.

P1039 ACR - "LIMITERROR" PROBLEMS

Previously, LIMITERRORs were not issued for Compact Data Sets. If LIMITERRORs were encountered on unordered Data Sets, then an error in DCB handling occurred when the structures were closed. Both of these problems have been corrected.

P1040 ACR - ORDERED DATA SET RECORDS OUT OF ORDER

When the last record was deleted in the first block of a disjoint, ordered data set, the empty block (the root block) remained in the in-use chain. The next CREATE...STORE by a program with current path undefined could cause an out-of-order condition. This problem has been corrected.

P1094 ACR - VISIBLE "DBS, INQUIRY" RETURN SOFTWARE VERSION

The software version mark has been added to the visible DBS STATUS command display. INQUIRY case number 28 returns the ACCESSROUTINES software version in the first word of its array parameter. The format of the returned word is the following:

```
[47:16] MARK
[31:16] CYCLE
[15:16] LEVEL
```

P1101 ACR - POPULATION TOO HIGH

The population of an ordered data set was being incremented on a MODIFY/STORE when the key was updated. This problem has been corrected.

P1115 ACR - LINEARSEARCH NOT FINDING ALL RECORDS

Linear searches performed on index-sequential or ordered list sets that have more than one fine table may not find all records which satisfy the key condition. This problem has been corrected.

P1156 ACR - SPURIOUS "NOTFOUND" AFTER ABORT

A spurious NOTFOUND is given by the ACCESSROUTINES in the following situation:

1. The data base contains an embedded unordered data set or an embedded index sequential, embedded ordered list or embedded unordered list set.
2. A user program does a find on the master. An abort occurs on the data base. After the abort, the user program tries to do a find on the embedded structure. A spurious NOTFOUND exception is given to the user program.

This problem has been corrected.

P1218 ACR - "FIND PRIOR AT KEY"

If a program performs a "FIND PRIOR <access> AT KEY=<value>" on a direct or ordered data set, every record prior to the current path will be returned. This problem has been corrected.

P1219 ACR - USAGE STATISTICS INCONSISTENCY

Following a STORE which is rejected with an exception, the data base usage statistics "STORE AFTER CREATE/INSERT ENTRY INTO SET" or "STORE AFTER LOCK/CHANGE DATA IN KEY" may be inconsistent. This problem has been corrected.

P1226 ACR - ABORT FAILURE

Previously, under certain circumstances, abort failed to properly update the end of file for a structure and would bring down the database with "FAILURE TO SET LASTRECORD ON DB FILE". This problem has been corrected.

P1227 ACR - OPENING OF DATA BASE RESULTS IN "SEG ARRAY"

If the name of a task that was attempting to open a data base were greater than 53 characters, the ACCESSROUTINES faulted with a SEG ARRAY error. This problem has been corrected.

P1241 ACR - ORDERED DATA SET CORRUPTION

If two or more users did CREATE/STOREs in the same data block at the same time and the data block was full, corruption could result in the data set.

The corruption caused spurious NOTFOUND exceptions. Records that had been stored could be missing in the data set. Any attempt to do any form of recovery through the portion of audit created during the time of corruption caused recovery to fault with an "INVALID TABLE SERIAL NUMBER". This problem has been corrected.

P1245 ACR - DIRECT KEY CORRUPTED

A spurious DIRECT KEY CORRUPTED error occurred when:

1. The data base contained two or more direct data sets, some of which were reblocked and others were not.
2. The first direct data set compiled into the ACCESSROUTINES was not reblocked.
3. Sequential finds were done through the access of a reblocked direct data set.

This problem has been corrected.

P1374 ACR - "DS" OF LINEAR SEARCH

If a user program were doing a linear search, it would enter user-code (DS-able code) to test each key value. If the program were DSed while executing this user code, the data base would hang. This problem has been corrected.

P1375 ACR - LIMIT ERRORS

Limit errors for ordered data sets were not previously returned to user programs. Now, however, the ACCESSROUTINES return limit errors for ordered data sets.

P1376 ACR - "NO FILE" ON "RECOVERYINFO"

Previously, if a usercoded RECOVERY were run against a non-usercoded data base, RECOVERY would come up on a NO FILE for the RECOVERYINFO file when it attempted to write the recovery region to the audit. This problem has been corrected.

P3194 ACR - BAD OPEN

Previously, if a program were DSed or got an exception while initially opening a structure, it was possible that any other program which subsequently attempted to open the structure would receive a DMOPENERROR exception. This problem has been corrected.

P3250 ACR - ABORT OVERLOOKS DUPLICATED AUDIT

Previously, if an ABORT recovery running against a data base with duplicated audit encountered an I/O error reading a block from the primary audit, it would not attempt to read the block from the duplicated audit as it should. This problem has been corrected.

P3327 ACR - EXCESSIVE MEMORY

The data base stack now uses less memory during OPEN.

P3328 ACR - "FORGETSPACE" TIMING WINDOW

A timing window in the ACCESSROUTINES buffer overlaying procedure (FORGETSPACE) could result in a number of problems, as follows:

1. DIRECT IO ATTRIBUTE ERROR #0
2. Fault in ACR code
3. Data base hanging
4. INVALID OP

This timing window has been closed.

SOFTWARE IMPROVEMENTS NOTES (P. NOTES)

DMS II - BDMSALGOL

P1161 BDMSALGOL - INTERLANGUAGE BINDING

When binding an ALGOL program which references a data base which does not have global data to a non-ALGOL program which references the same data base, an error would occur. This problem has been corrected.

P1231 BDMSALGOL - SEGMENTED TRANSACTION RECORDS, RECORD ARRAYS

If a TRANSACTION BASE had any large formats (greater than 1024 words), an assignment to a TRANSACTION RECORD or TRANSACTION RECORD ARRAY associated with the TRANSACTION BASE would cause an INVALID OP error at execution time. This problem has been corrected.

P1252 BDMSALGOL - SIGNED NUMBERS, HEX POINTERS

Using the GET verb to assign the value of a DASDL signed number to an ALGOL HEX pointer would lead to the truncation of the resulting field by 1 digit. This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - BDMSCOBOL

P1326 BDMSCOBOL - SELECTION EXPRESSION TRUNCATION WARNING

Previously, every FIND, LOCK and DELETE statement using an alphanumeric key in a selection expression was given a warning by the compiler:

"IF DATA-ITEM LARGER THAN KEY-ITEM - TRUNCATION".

The warning was given even when the DATA-ITEM was not larger than the KEY-ITEM. Now, the warning is given only when the DATA-ITEM is larger than the KEY-ITEM:

"DATA-ITEM LARGER THAN KEY-ITEM - TRUNCATION".

P3305 BDMSCOBOL - "SIB" SIZE INCREASE FOR LARGE DATA BASES

The BDMS compiler no longer faults with a SEG ARRAY error when invoking a very large data base description.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - BDMSCOBOL74

P1326 BDMSCOBOL74 - SELECTION EXPRESSION TRUNCATION WARNING

Previously, every FIND, LOCK and DELETE statement using an alphanumeric key in a selection expression was given a warning by the compiler:

"IF DATA-ITEM LARGER THAN KEY-ITEM - TRUNCATION".

The warning was given even when the DATA-ITEM was not larger than the KEY-ITEM. Now, the warning is given only when the DATA-ITEM is larger than the KEY-ITEM:

"DATA-ITEM LARGER THAN KEY-ITEM - TRUNCATION".

P3305 BDMSCOBOL74 - "SIB" SIZE INCREASE FOR LARGE DATA BASES

The BDMS compiler no longer faults with a SEG ARRAY error when invoking a very large data base description.

DOCUMENT CHANGES NOTES (D NOTES)

DMS II - DASDL

D4147 DASDL - SYSTEM ALLOCATED AREAS

DASDL always allocates extra disk areas for each data set, set and subset. These system-allocated areas are provided in addition to the user-allocated areas which were specified explicitly or computed implicitly.

For standard data sets, DASDL allocates system areas for available space tables. Enough available space is allocated to avoid run-time limit errors even if all records in the data set are deleted.

For all other structures, DASDL allocates one system area.

A syntax error results if the combination of system and user areas exceeds 1000 areas.

D4213 DASDL - CHAIN OF IN USE BLOCKS

Page C-8 of the DASDL Reference Manual (Form No. 5001480) is incomplete. ORDERED data sets should be included in the list of structures using a two-way circularly-linked list of in-use blocks. For ORDERED data sets, the head block is the root.

On Page C-31, the BAF description for NEXTBLOCKLOC should be changed to read: "Block address of next block".

On Page C-32, the BAF description for PRIORBLOCKLOC should be changed to read: "Block address of prior block".

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - DASDL

P1157 DASDL - "NOFILE" ON "WFL" COMPILE JOB

If the WFL deck zipped by DASDL to compile the data base tailored software was not available, CONTROLCARD would appear in the waiting entries for a NOFILE condition. However, the user was not informed of the no file, and the DASDL compile simply waited for CONTROLCARD to finish. The DASDL compiler now tries to open the WFL deck file before performing the ZIP. Consequently, the user will now receive a NOFILE message at the terminal if the WFL deck is not present. The FA message may be used to resolve the NOFILE condition.

P3258 DASDL - DELETING, ADDING GLOBAL DATA

The following problems have been corrected:

1. Adding GLOBAL DATA on an UPDATE after deleting it on a previous update, resulted in changing the structure numbers for all the data base structures, thereby invalidating existing programs.

Now, data base structures retain their old structure numbers and GLOBAL DATA is assigned whatever its structure number was the last time it existed.

2. Adding GLOBAL DATA for the first time on an UPDATE (i.e., it never existed previously) resulted in an error message "BLOCKSIZE TOO SMALL FOR ONE RECORD" and a subsequent DIVIDE BY ZERO fault.

Now, GLOBAL DATA can be successfully added on an UPDATE.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - DUMPDIR/LIBRARY

P1308 DUMPDIRLIB - SEG ARRAY ERROR ON "WRITE-", "LIST-"

A SEG ARRAY fault occurred when extracting structure names from a large description file. This problem has been corrected.

The first sentence of the semantics for SHOW <data set name> should be deleted. The semantics for SHOW <data set name> ITEMS and SHOW <data set name> SETS are unchanged.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - INQUIRY

P1069 INQ - "INVALID OP"

In certain cases, INQUIRY did not check the size of numbers, thus subsequent INTEGER or DINTEGER of pointers would die with an INVALID OP or INTEGER OVERFLOW. This problem has been corrected.

P1077 INQ - BOOLEAN KEY DATA

Boolean key data was handled incorrectly. This could result in either inefficient searching or invalid indices. This problem has been corrected.

P1080 INQ - "DISPLAY" FORMATS OUTPUT IN "HEADING" FORM

The DISPLAY command now formats output in the HEADING form by default whenever feasible. The only case in which INQUIRY does not default to HEADING format is when output is going to a CRT device and a record is being displayed which is so large that its heading and values will not fit on the screen. In this case, INQUIRY uses TAB format.

On previous releases, there were three additional cases in which INQUIRY used TAB by default:

1. records were being accessed with the basic DISPLAY command (a DISPLAY without a selection expression),
2. a limit of one was specified on a SELECT/DISPLAY command, and
3. output was being routed to the printer and the headings would not fit on one line.

P1092 INQ - "SEG ARRAY"

If the formal parameters to a define were very short identifiers, INQUIRY would get a SEG ARRAY fault termination when first placing the define in its tables. This fault has been removed and declaration of defines has been optimized.

P1100 INQ - SELECTION EXPRESSION IMPROVEMENT

Selection expressions may now refer to items from different data sets. The previous limitation has been eliminated.

P1129 INQ - RECORD FUNCTION INFORMATION NOT PURGED

Error response messages exited function calls without purging some structure information. This left the function unevaluated, and on any subsequent call on a function for that structure, the original error would be repeated. This problem has been corrected.

P1181 INQ - SAVE PROMPT FOR TEMPORARY SETS

INQUIRY would not ask the user if a save was desired if a temporary set was created by a generate. A quit would cause the text for the generate to be lost unless a define or virtual was done in the same session. This now works as documented, and gives the "DECLARATIONS NOT SAVED" warning.

P1220 INQ - MODIFIED BIT FOR "AA" LIST SORTS

When one part of a selection condition was satisfied by a small set of records and another part of the condition was satisfied by a large set of records, the resulting intersection was not always created properly, because the list of record addresses in the small set was not properly maintained. This problem has been corrected.

P1221 INQ - SUBSCRIPT VALUE LIMIT

Occurring items with a subscript value greater than 255 could not be displayed because of an 8-bit field size for storing subscripts. The field size has been increased to allow values up to the 1023 DASDL limit.

P1224 INQ - INVALID ADDRESSES IN SORT TAGS

When a sort option was specified on a selection condition for a data set and the selection condition tested embedded records of the data set, record addresses for the embedded records instead of the owner records were put into the sort tags. Errors would occur in attempting to retrieve records for the sorted tags.

P1225 INQ - REPORT TITLE TOO LARGE

Error checking has been added to the TITLE and GENERATE REPORT commands to ensure that report title lines do not exceed line size. Error checks also prevent page number overwrite on line one of page titles.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

P1309 INQ - MINUS SIGN ON "REAL(SM,N)" ITEMS

Data base items of type REAL(SM,N), where M and N are precision and scale values, respectively, were not written with sign on DISPLAYS and REPORTS when the value of the item was greater than -1 and less than 0. This problem has been corrected.

P1310 INQ - USE, ENFORCE CORRECT ITEM FIELD SIZES

INQUIRY dealt with fields as if their sizes were the appropriate number of bits to reach a hexade boundary; e.g., FIELD(2) would be treated as FIELD(4). This would cause invalid values to be stored in the field, a larger display size than necessary, and misleading SHOW information. INQUIRY now uses the correct size as defined in the DASDL.

P1321 INQ - "SHOW ALL OF <DATA SET NAME>"

"SHOW ALL OF <data set name>" now correctly displays only the data set requested instead of the entire data base. Also, the SHOW syntax has been changed to restrict ITEMS and SETS specifications for SHOW <identifier>, where <identifier> is not a data set. The new syntax is described in INQUIRY note D4214.

P3195 INQ - TRAILING BLANKS ON MULTISTATEMENT DEFINES

If trailing blanks were entered on the invocation of a multi-statement define, an error was given indicating that multi-statement defines must be complete statements. The trailing blanks are now ignored.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - INTERFACE

P1377 INTERFACE - "INVALID INDEX"

It was possible for INTERFACE to fail with an INVALID INDEX or pass incorrect information about the layout of a data set record back to the compiler, if a remapped data set contained a group with HIDDEN items or contained a HIDDEN group within a group. This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

DMS II - LOADDUMP

D4190 LOADDUMP - "DATASET" CHANGE

The following changes should be made to the LOADDUMP section of the DMSII Utilities and Operations Guide (Form No. 5001803):

Page 7-4:

The syntax for DATASET should read as follows:

```
-- DATASET ---<data set name>-----|
      |<data base name>--|
      |<remap name>-----|
```

Page 7-6:

The second paragraph under <database specification> should read as follows:

"The DATASET clause specifies the name of the disjoint data set, global data or remap for which LOADDUMP will generate utilities. The data set, global data or remap must be contained in the database or logical database specified in the DB clause. The RESTART data set or variable format data sets may not be specified."

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - LOADDUMP

P1222 LOADDUMP - INCORRECT CODE GENERATED

LOADDUMP was incorrectly generating source statements for a data item of a compact data set which occurs depending on another data item. The number of instances of the occurring data item (or record item) moved was the DASDL specified maximum. This would result in an INVALID INDEX during execution. LOADDUMP now generates source statements which will move the minimum of either the DASDL specified maximum or the value of the DEPENDING ON data item.

P1223 LOADDUMP - INCORRECT SOURCE CODE

LOADDUMP was generating incorrect source code when GLOBALDATA was specified as the data set. GLOBALDATA and its remaps are now handled correctly.

P1378 LOADDUMP - INCORRECT "FD" GENERATED

If the last item in a group were "EXCLUDED", LOADDUMP would incorrectly omit all items following the excluded item from the FD record description, resulting in an incorrect DUMP/LOAD symbolic. This was true only if the FD were being generated from the data set description. This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - PROPERTIES

P1379 PROPERTIES - NUMBER OF NAMES IN "DMERRORS" ARRAY

The number of names in the DMERROR array was incorrect. This array is INCLUDED when a BDMS compiler is being compiled and contains the identifiers used to represent DMSII exception categories (e.g., NOTFOUND). This number was not incremented when the new categories FATALERROR and INTEGRITYERROR were added; consequently, the BDMS compilers would syntax the use of either of the new category names. The problem has been corrected; however, the BDMS compilers must be recompiled using the corrected DATABASE/PROPERTIES symbolic to resolve the problem.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - RECOVERY

P1041 RECOVERY - INFINITE LOOP IN "RECOVERY"

When RECOVERY detected a problem with an audit block it had read, it printed the audit block. If the audit block were corrupted and the auditsize field in the control word of the record contained 0 or 1, an infinite loop occurred. The problem has been corrected.

P1091 RECOVERY - WRITE ERROR

RECOVERY was writing the wrong size record to the ROWLOCKOUTAUDIT file when the record was at the end of a disk row. This caused RECOVERY to terminate with a WRITE ERROR ON ROWLOCKOUTAUDIT FILE. This problem has been corrected.

P3257 RECOVERY - REBUILD ACROSS REORGANIZATION

A structure discontinuity is caused when (1) the structure is initialized by UTILITY and (2) when a structure is reorganized. A STRDC audit record is written for each structure that has been initialized or reorganized.

A REBUILD across a structure discontinuity (STRDC) that was caused by a REORG was erroneously reinitializing the structure.

This would cause RECOVERY to:

1. Get addresscheck or checksum errors if these options were set. This would happen each time an audit record was encountered after the STRDC that caused RECOVERY to read a block beyond the EOF of the reinitialized structure.
2. Abort with an "INVALID TABLE SERIAL NUMBER" if sets were involved.

If RECOVERY did not have one of the above problems, data corruption could occur unnoticed. This problem has been corrected.

P3329 RECOVERY - "DATARECOVERY" FAILS TO RECONSTRUCT ROW

Previously, under certain conditions, DATARECOVERY could corrupt the last segment of the last block of a row in a data file. If the file were checksummed, DATARECOVERY would fail to reconstruct the row. This problem has been corrected.

P3821 RECOVERY - REBUILD/RECONSTRUCT ABORT

Previously, Rebuild or Reconstruct could abort with an "INVALID TSN" in before image application under the following conditions:

- a. Rebuild or Reconstruct was started from two or more dumps taken at different times, and the area in the audit corresponding to the time period from the start of the first dump to the end of the last dump contained a Recovery Region.

or

- b. The initial Rebuild or Reconstruct run was abnormally terminated at a certain point on the audit (e.g., Point A) and it was then restarted at another point on the audit (preceding Point A) which was separated from Point A by a Recovery Region.

This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - REORGANIZATION

P1128 REORG - CHECKSUM ON REORGANIZED GLOBAL DATA

Reorganization of global data has been corrected to include checksum in the reorganized global data when appropriate.

P1155 REORG - INCORRECT "BCW" ON EMBEDDED ORDERED DATA SETS

When reorganizing an embedded ordered data set with subblocks greater than 1 subblock per block, the ODLASTAVAIL field of the last BCW in the block was being assigned an incorrect value. This value was causing data base failures on subsequent updating of the corresponding subblock. This problem has been corrected.

P1291 REORG - CORRUPTION OF DATA

Performing garbage collection on a data set which was the data set to last undergo file format or record format conversion could result in data corruption of that data set. This problem has been corrected.

P1380 REORG - CHECKSUM ERROR ON INTERIM TAPE FILES

REORG would give a checksum error on interim tape files when the data base did not have ADDRESSCHECK set. This problem has been corrected.

P3197 REORG - GENERATING MULTIPLE BITVECTORS

The REORG compile failed (syntax error) if the specifications indicated that MULTIPLE BITVECTORS of a data set were to be generated. This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

DMS II - UTILITY

D4215 UTIL - "<INTEGER>" IS UNIQUE NUMBER

The following sentence on Page 4-37 of the DMSII Utilities and Operations Guide (Form No. 5001803):

"<integer> is the structure number assigned by the DASDL compiler."

should be changed to read as follows:

"<integer> is a unique number."

D4231 UTIL - GUARDED DATA BASES

Add the following to the DMSII Utilities & Operations Guide (Form No. 5001803) as a new topic after Page 4-43:

When UTILITY does an "on-line" dump, the task "UTILITY/<dbname>" may open the data base and all "DUMPWORKER<nn>/<tapename>" tasks will open the data base. It is not practical to specify all of these tasks in the guardfile guarding the data base. It is suggested that UTILITY be made a privileged program when the data base is guarded, via the ODT message "PP".

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - UTILITY

P1042 UTIL - AVAILABLE BLOCK NUMBERS PRINTED WRONG

UTILITY was printing all of the NEXTBLOCKLOC word in decimal instead of just the block address field. This no longer occurs.

P1070 UTIL - "REBUILD" FAILS ON FILE DISCONTINUITY

REBUILD failed when a file discontinuity was encountered when two or more UTILITY dumps were input to a rebuild. The failure occurred when a single structure appeared on two UTILITY dump tapes and the first dump was taken prior to structure initialization and the second dump was taken immediately after structure initialization. On the Mark 31 release, when the first dump was specified before the second dump in the UTILITY RECOVER command, RECOVERY failed with the following message:

"CANNOT RECOVER ACROSS FILE DISCONTINUITY WITH OLD FILE"

On the Mark 32 release, RECOVERY worked correctly but applied more audit images than necessary. This problem has been corrected.

P1071 UTIL - WRONG NUMBERS OF WORKERS STARTED

UTILITY restarted incorrectly when multiple workers were used. During the first restart, an extra dumpworker/tapeworker was processed. If UTILITY were again DSed and restarted, UTILITY terminated abnormally. This failure occurred under the following circumstances:

UTILITY syntax must have WORKER=N where N is less than the number of physical tapes needed. More than one dump needs to be specified so that not all dumpworkers/tapeworkers are started in order for the workers (WRITEVOLUMES/READVOLUMES). For example:

"OPTIONS(WORKERS=2) OFFLINE DUMP= TO DUMP1; = TO DUMP2; = TO DUMP3"

where DUMP1 and DUMP2 each start up 1 WRITEVOLUME.

This problem has been corrected.

P1081 UTIL - SYNTAX ERROR WHEN RESTARTING "UTILITY"

If the options NOZIP or WORKERS were specified and UTILITY was restarted following a Halt/Load or DS, an erroneous syntax error was produced. This problem has been corrected.

P1381 UTIL - DATA BASE INVOCATION

Previously, the entire data base was invoked whether the data base was audited or not. Now, the entire data base is invoked for non-audited data bases. Only the restart data set is invoked for audited data bases.

P3326 UTIL - "UTILITY" RECOVER MAY CORRUPT DATA

A UTILITY recover was corrupting data under the following circumstances:

1. The recover was done from the dump tape, so that more than one READVOLUME was going at the same time, and
2. The data base file was being read in by more than one of the READVOLUMES, and
3. Recovery ran successfully.

Under this set of circumstances, the data base file would contain rows whose first block had zeroes in the first 30 words. This no longer occurs.

DOCUMENT CHANGES NOTES (D NOTES)

DMS II - WFL/COMPILEACR

D4212 COMPILEACR - "INITPARTITIONS" OPTION REMOVED

Because the SYSTEM/DMCONTROL option INITIALIZE PARTITIONS has been deimplemented (described in Mark 32 DMCONTROL note D3119), the DATABASE/WFL/COMPILEACR deck no longer recognizes the INITPARTITIONS option. Setting this option caused the WFL deck to run SYSTEM/DMCONTROL to initialize the partition directory in the control. The DASDL compiler no longer generates this option when constructing the parameter string to be passed to the WFL deck when the ZIP option is set.

DOCUMENT CHANGES NOTES (D NOTES)

DMS II - TRANSACTION PROCESSING

D4161 TPS - APPLICATION PROGRAM RESTART

Section VII, paragraph 6, of the Transaction Processing System, released with the Mark 32 system software release as TPS/DOCUMENT on the SYSTEMNOTES tape, should be replaced by the following:

*6. Application Program Restart

Application program restart is very simple when the Transaction Processing system is used. An application program need only provide for restart following a DS or Halt/Load interruption. When restart is necessary, each program must log on by calling LOGONTRUSER. This is necessary because all users are automatically logged off following a failure. The system saves each user's last successful input transaction and its response. Programs may retrieve these transaction records by calling the Transaction Library procedures RETURNRESTARTINFO and RETURNLASTRESPONSE with the program's user id number. These transaction records must contain all information needed to restart the program."

Note: Mark 32 TPS note D3334 incorrectly described the TPS document as "Appendix C"; the TPS document was "TPS/DOCUMENT" on the Mark 32 SYSTEMNOTES tape.

D4268 TPS - RULES FOR TRANSACTION PROCESSING

Some confusion has resulted from the use of Database applications which span Libraries, Process families, and even other Databases. In order to eliminate further confusion, the following discussion reviews the rules for transaction processing. The rules described below apply across Databases, Libraries, and Process families. These rules are explained by the use of a few examples, some of which reference the following example Shared Library.

EXAMPLE LIBRARY

```

:
:
DATABASE PAYROLL: EMPLOYEE, RDS;
:
:
PROCEDURE BTR;
BEGIN
  BEGINTRANSACTION RDS;
END;
:
PROCEDURE ETR;
BEGIN
  ENDTRANSACTION RDS;
END;
:
PROCEDURE STOREEMP;
BEGIN
  % STORE AN EMPLOYEE RECORD
END;
:
EXPORT BTR, ETR, STOREEMP;
:
:

```

1. A program may perform only one transaction at a time; i.e., a program which performs BEGINTRANSACTION must perform ENDTRANSACTION before performing another BEGINTRANSACTION.

Example:

```

:
:
DATABASE PAYROLL : RDS;
:
:
BEGINTRANSACTION RDS;
BEGINTRANSACTION RDS; % AUDITERROR WOULD RESULT
:
:

```

Example:

```

:
:
DATABASE PAYROLL : RDS;
DATABASE ACCOUNTING : RDS;
:
:
BEGINTRANSACTION RDS OF PAYROLL;
BEGINTRANSACTION RDS OF ACCOUNTING; % AUDITERROR WOULD
:                                     % RESULT
:

```

2. Only one program may be in transaction state against a Database declaration at a time.

Example:

If Program #1 calls procedure BTR, and Program #2 calls BTR before Program #1 calls ETR, then Program #2 will receive an AUDITERROR.

Program #1	Program #2
-----	-----
BTR;	
ETR;	BTR; % AUDITERROR WILL RESULT

3. In order to perform an update operation against an audited Database, the Database declaration must be in transaction state.

Example:

```

:
:
DATABASE PAYROLL: RDS;
:
BEGINTRANSACTION RDS;
STORE EMPLOYEE; % LEGAL
ENDTRANSACTION RDS;
:
:

```

Example:

```

:
:
BTR;
STOREEMP; % LEGAL
ETR;
:
:

```

Example:

Program #1	Program #2
-----	-----
BTR;	
ETR;	STOREEMP; % LEGAL

Example:

```

:
:
DATABASE PAYROLL : EMPLOYEE, RDS;
:
STORE EMPLOYEE; % INVALID OP - NOT IN TRANSACTION STATE
:
:

```

Example:

```

:
:
DATABASE PAYROLL : EMPLOYEE, RDS;
:
BTR;
STORE EMPLOYEE; % INVALID OP - IN TRANSACTION STATE AGAINST
ETR; % THE WRONG DECLARATION
:
:

```

Example:

```

:
DATABASE PAYROLL : EMPLOYEE, RDS;
:
BEGINTRANSACTION RDS;
STOREEMP;      % INVALID OP - IN TRANSACTION STATE AGAINST
                %
                % THE WRONG DECLARATION
ENDTRANSACTION RDS;
:
:

```

4. The program which enters transaction state against a Database declaration must be the one to end transaction state.

Example:

```

PROGRAM #1          PROGRAM #2
-----
BTR;
                   STOREEMP;
                   ETR;      % AUDITERROR WOULD RESULT

```

D4295 TPS -- DOCUMENTATION CORRECTIONS

The following changes should be made to the documentation for the Transaction Processing System (described in Appendix C of the Mark 31 P- and D-Notes):

1. Exception #306

The description of exception #306 states "Journal must be open input". This statement does not adequately describe the cause of the error. The statement should read "Must perform SEEKTRANSACTION before attempting READTRANSACTION".

2. Add Exception #506 Semantics

Exception #506 -
The Update library called procedure SAVEINPUT without first performing BEGINTRANSACTION.

3. Add Exception #507 Semantics

Exception #507 -
The Update library called procedure SAVERESPONSE without first performing MIDTRANSACTION. This error will be returned only if the Update library enters transaction state.

DOCUMENT CHANGES NOTES (D NOTES)

DMS II - TFL

D4179 TFL - "INCLUDE" OPTION

The TFL INCLUDE option was described incorrectly in the original Transaction Processing System documentation. The correct description is as follows:

"INCLUDE

The INCLUDE option permits source language input from files other than CARD and TAPE files. Each INCLUDE option specifies the title of the input file in quotes and indicates which portion of the file is to be included. The included card images are compiled in place of the INCLUDE option. INCLUDE options may appear anywhere in the input. INCLUDE options must not be nested; i.e., included input must not contain INCLUDE options.

<file name> specifies the title of the file to be used as input. <base> and <limit> indicate which portion of the file is to be included. <base> specifies the sequence number of the first card image to be included. <limit> specifies the sequence number of the last card image to be included. If no <base> is stipulated, inclusion begins with the first record in the file. If no <limit> is indicated, the last record of the file is the last record included. When neither is specified, the entire file is included.

Included input must have the same record size and format as the CARD file. <base> and <limit> should only be used on properly sequenced files."

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - TFL

P1116 TFL - DUPLICATED OPTION

If the data file attribute "DUPLICATED ON <packname>" is specified in the TFL DEFAULTS statement, a syntax error is issued when the host library is compiled. This problem has been corrected.

P1312 TFL - DETECTION OF SYNTAX ERRORS

Previously, if syntax errors were detected during the compilation of a transaction base, TFL did not DS itself; consequently, examination of the task attribute COMPLETEDOK would not indicate whether syntax errors had been detected. This problem has been corrected. If syntax errors are detected, the following message is displayed and the TFL processor is DSed:

**** SYNTAX ERRORS ****.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - TRINTERFACE

P1313 TRINTERFACE - BAD TIMESTAMPS

If more than one transaction base were invoked in a single program, when a subsequent CREATE statement was executed, the wrong timestamps were placed in the transaction record. The timestamps of the most recently-invoked transaction base were always used. This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - TRUTILITY

P1090 TRUTILITY - "INVALID INDEX"

If TRBASE/UTILITY was compiled with a transaction base containing more formats, subformats or items than UTILITY allowed, UTILITY would be terminated with an INVALID INDEX. This problem has been corrected.

P1209 TRUTILITY - SEARCH FUNCTION FAULT

If the TRUTILITY search function is executed using a journal data file that contains multiple transaction records per block, TRUTILITY could abnormally terminate with an INVALID INDEX. This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

DMS II - HOSTLIB

D4144 HOSTLIB - INCREASED MAXIMUM NUMBER OF ACTIVE "TPS"

For a single transaction base, the maximum number of journals that can be concurrently accessed by user programs is 128. Similarly, the maximum number of concurrent users of a single journal is 128. These limits have been increased from their previous value of 16.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DMS II - HOSTLIB

P1108 HOSTLIB - INCORRECT INTERVAL TIME

If the Host library's statistics gathering began before midnight and finished after midnight, the summary statistics had an incorrect elapsed time interval. This problem has been corrected.

P1228 HOSTLIB - TRANSACTION RECORDS EXCEED "2059" BYTES

Previously, the HOSTLIB would not accept any transaction record which was greater than the largest system transaction; i.e., 2059 bytes long. This problem has been corrected.

P1229 HOSTLIB - SYNTAX ERROR

If the HOSTLIB were compiled against a very large transaction base, it was possible for the compilation to terminate with a syntax error. The error indicated that the compiler attempted to generate a code segment which exceeded the maximum allowable size. This problem has been corrected.

P1230 HOSTLIB - DEADLOCK

If, as a result of writing a transaction record to a journal data file, the transaction was split across two files, a deadlock would occur. All processing against the journal control file and all data files would cease. This problem has been corrected.

P3200 HOSTLIB - RECOVERY OF JOURNAL DATA FILE'S END OF FILE

If the host library were attempting to write a transaction record whose length was greater than the block size of the data file, and a Halt/Load occurred, it was possible the host library would not be able to recover the end of the data file. This problem has been corrected.

P3201 HOSTLIB - CANNOT SAVE RESPONSE TRANSACTION

If an INQUIRY or UPDATE user directly called procedure SAVERESPONSE from an update library, the user's response transaction would not be saved in the journal control file. This problem has been corrected.

P3202 HOSTLIB - RESPONSE TRANSACTION NOT SAVED

If one TPS user were "purged" and a new TPS user were subsequently "created", the response record for the newly-created user could not be retrieved. Consequently, calls on RETURNRESTARTINFO and RETURNLASTRESPONSE produced unpredictable results. This problem has been corrected.

P3339 HOSTLIB - ERROR RESULT NOT RETURNED

If a transaction record were created in a format which was subsequently deleted as a result of a TFL update, and if this transaction record were submitted to the HOSTLIB for tanking or processing, it was possible that the transaction would be accepted without the HOSTLIB returning an exception. This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

DUMPALL

D4209 DUMPALL - HEX AND DECIMAL RECORD NUMBERS

DUMPALL now prints 1-relative numbers for both hex record numbers and decimal record numbers. The LASTRECORD printed in the heading is still 0-relative.

DOCUMENT CHANGES NOTES (D NOTES)

DUMP ANALYZER

D4103 DUMPANALY - TIMESTAMP AT "BOT" IS RECORDED IN "EOT" LOG

The timestamp for the time at which a task was initiated has been added to the EOT log record. This timestamp identifies the month, day, year, hour, minute, and second of task/job initiation. The new word is referenced by EOTBOTTIMESTAMP and is the 34th word of EOT type log records on the Mark 32 release.

In addition, DUMPANALYZER will print the BOT timestamp when analyzing the PIB.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

DUMP ANALYZER

P1166 DUMPANALY - PATH NAME WITH MULTIPLE PATHS

Previously, when multiple paths were involved, the path name was reported incorrectly on MLIP system. This problem has now been corrected; the path name appearing with a CQ message will be consistent with the information printed for all IOCBs in that Command Queue.

P1213 DUMPANALY - DISPLAY "MLIP" UNIT STATUS

The DUMPANALYZER now correctly deciphers UNITSTATUS table entries on MLIP systems.

P3191 DUMPANALY - "INVALID INDEX" INTERRUPT

B6800 processors can generate an INVALID INDEX interrupt literal that looks like 4"8800020" instead of 4"9000020". The DUMPANALYZER now recognizes it as an INVALID INDEX.

P3193 DUMPANALY - FAULT ON MISPELLED "HELP" QUERY

When "HELP <option>" was entered with the <option> misspelled, a SEG ARRAY error could occur, resulting in a DUMPANALYZER fault. The error has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

ESPOL

D4269 ESPOL - CANNOT RUN ON "NON-BCL" SYSTEM

Compiling with ESPOL or running SYSTEM/TABLEGEN (the table generator for the ESPOL compiler) on a non-BCL system will cause: (1) a message to be displayed stating that the program cannot run on a non-BCL system, and (2) the program to discontinue execution.

D4270 ESPOL - DEIMPLEMENTATION WARNING

ESPOL will be deimplemented on a future release. A message is now produced to say that the compiler will be deimplemented on some future release. Originally, the compiler was to be deimplemented on the Mark 33 release.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

FILECOPY

P1253 FILECOPY - DEFAULT USERCODE

Filenames given for INCLUDE, FILES and EXCLUDE statements will assume the usercode of the job under which FILECOPY is running (if one exists), unless:

1. a usercode was specified, or
2. it is a system file, or
3. the filename given was "=" (all files)

P1302 FILECOPY - LISTING OF SERIAL NUMBERS

Previously, when a large number of tape serial numbers were used, FILECOPY raised a SEG ARRAY error. FILECOPY has been corrected to allow the listing of the serial numbers over several lines, if needed.

DOCUMENT CHANGES NOTES (D NOTES)

FILEDATA

D4235 FILEDATA - "FILEDATA" ATTRIBUTES

Add the following to Page 5-3-3, SOG Reference Manual, Volume 1 (Form No. 5011661), below the "Semantics":

"An ATTRIBUTES request with TAPE specification is treated as the request:
"ATTRIBUTES:NAMEONLY TAPE=<tape specification>"."

DOCUMENT CHANGES NOTES (D NOTES)

FORTRAN

D4204 FORTRAN - FILES "5,6" DEFAULT TO "REMOTE"

Add the following item to the discussion of "COMPILATIONS THROUGH CANDE" on Page 18-9 of the FORTRAN Reference Manual (Form No. 5001506):

"g. FILES and FILE6 default to KIND=REMOTE"

D4233 FORTRAN - "MODELI" COMPILER OPTION

The default value of the MODELI compiler option, described in the FORTRAN Reference Manual (Form No. 5001506) on Page 19-19 has been changed from RESET to SET.

D4321 FORTRAN - "FREE-FORMAT" INPUT OF HEX STRING

Add the following to the notes on the FREE-FORMAT INPUT AND OUTPUT section of the FORTRAN Reference Manual (Form No. 5001506), Page 12-22:

"When the input is a hexadecimal string, that string will be returned unaltered as the value of the variable being read."

D4324 FORTRAN - REPEAT COUNTS, SIZE FIELD "<4030" IN FORMATS

Due to implementation limitations, the maximum number which can be used in a format as the repeat count, width, or decimal size is 4029.

Any value larger than 4029 causes a fatal run-time error at run-time formatting.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

FORTRAN

P1119 FORTRAN - "READER, PRINTER" OPTIONS FAIL

If the compiler was compiled with the option MULTI reset (disallowing batch compilations), the options READER and PRINTER received syntax errors. This problem has been corrected.

P1194 FORTRAN - VARIABLE IN "ASSIGNMENT" STATEMENT

XREFANALYZER did not print an asterisk in front of the sequence number when a variable was used in an ASSIGNMENT statement. This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

GENERSUPPORT

D4311 GENERSUPP - REPEAT COUNT, SIZE FIELD "<4030" IN FORMATS

Due to implementation limitations, the maximum number which can be used in a format as the repeat count, width, or decimal size is 4029.

Any value larger than 4029 causes a syntax error at compile time (for ALGOL) or a fatal run-time error at run-time formatting (for FORTRAN).

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

GENERALSUPPORT

P1099 GENERALSUPP - FREEFIELD ATTEMPTING TO WRITE TO INPUT FILE

When input data required multiple records and the file positioning was done using a record number, the intrinsics would fail attempting to write to the input file. This problem has been corrected so that the records following the <record number> will be read if more data is needed to satisfy the <list>.

P1137 GENERALSUPP - REMOVE "BCL"

All uses of BCL for conversion to and from octal have been removed. For breakpoint commands "/BCL" and "&BCL", an error message is returned if the system does not support BCL.

P1303 GENERALSUPP - FORMATTED READ OF STRINGS

If an attempt were made to read a string using an <A*> format and the length were zero, the length was improperly updated. Depending on how the string was subsequently used, a multitude of errors could occur. This problem has been corrected.

P1331 GENERALSUPP - "K" MODIFIER WITH RUN TIME FORMATTING

When the K format modifier (FORTRAN) was used with run-time formatting of real numbers (e.g., KF10.2), commas were not produced. The commas are now produced.

Note: The K format modifier in FORTRAN is equivalent to the P format modifier in ALGOL.

P1362 GENERALSUPP - FREEFIELD "WRITE" OF ARRAYS

When using "*" or "*" freefield format to print an array row with non-zero lower bounds, the subscripts are now printed correctly.

P3214 GENERALSUPP - "G FORMAT" WHEN "VALUE=0"

When writing a value of 0.0 in G format (FORTRAN), where the width of the field is less than or equal to the number of digits to the right of the decimal place + 5 (W<=d+5), unprintable characters would be printed. This problem has been corrected.

P3215 GENERALSUPP - READ INTO STRING ERROR

When a string was being read, its present size was being used as the maximum size and truncation was performed if the size were smaller than the editing phrase. This problem has been corrected, so that the full size of the editing specification is used when reading into a string variable.

P3277 GENERALSUPP - "F" FORMAT OUTPUT SUPPRESSES LEADING ZEROES

Certain F formats with large field widths (e.g., F34.0) would produce leading zeroes. This problem has been corrected.

P3278 GENERALSUPP - REPEAT COUNTS OF ZERO IN FORMATS

A repeat count of zero when used for repeating parenthesized formats now causes a syntax error.

P3309 GENERALSUPP - "F" FORMAT OUTPUT CAUSES INCORRECT RESULTS

Certain F formats with small field widths (e.g., F4.2) caused incorrect results to be printed. The correct results are now generated.

P3312 GENERALSUPP - MISSPELLED FORMAT

Previously, if the last item in a format were an integer, a syntax error was not generated. Now, a syntax error is generated when a repeat count without an editing specification occurs at the end of a format.

P3314 GENERALSUPP - FORMATTED OUTPUT WITH "A*" FORMAT

Certain cases of A* formats could cause either an INVALID OP in the formatters or an INCORRECT FORMAT error. These problems have been corrected.

P3315 GENERALSUPP - "FREE-FIELD" INPUT OF REAL, BOOLEAN

When reading Booleans and reals using free-field formatting, unexpected results were returned. The formatter now returns values as follows:

Reals: If the input is a simple integer, the result is an integer.
 If the input is a Real (e.g., 12.0), the result is in normalized form.
 If the input is a hex constant (e.g., 4"A3"), the result is identical to the input.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

Booleans: If the input is an integer, that integer is returned.
If the input is a Real, a normalized value is returned.
If the input is a hex constant, the result is that hex constant.

P3325 GENERALSUPP - "BCL A" FORMAT INPUT

When reading BCL data with an A format into an array, an INVALID OP could occur. This problem has been corrected; now, arrays can be used as the target of A formats with BCL data.

SOFTWARE IMPROVEMENTS NOTES (P. NOTES)

HARDCOPY

P3764 HARDCOPY - "HARDCOPY" NOT REACTING TO "HI"

HARDCOPY was not always initiating PRINTCOPY when "HI" was entered at the ODT. This problem has been corrected.

P3765 HARDCOPY - "HARDCOPY" FILES LEFT ON DISK

When HARDCOPY fills up its files, it initiates PRINTCOPY, which should print the file and then delete it; however, PRINTCOPY was not deleting the file. This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

HOSTNSPSUPPORT

D4141 HOSTNSP - "HOSTNSP" IMPLEMENTATION

The HOSTNSP is a Host System software implementation of the Network Services Processor (NSP) used for Data Communications on MLIP systems. It is in the form of an MCP Support Library internally named HOSTNSPSUPPORT and called only by the MCP.

The intent of the HOSTNSP is to provide fully-functional Data Communication and backup capabilities for entry-level systems. This is accomplished by the elimination of the NSP hardware, and thus requires only the host and LSP DLPs. This type of operation is not "free" in that host system processor time and memory are used. Therefore, it is advisable that systems with a large Data Communication network and traffic should utilize NSP hardware. Regardless of the presence of physical NSPs on a system, some of the "diagnostic" options available with the HOSTNSP make it a useful tool for NDLLI Algorithm/Editor debugging. All NDLLI functions are supported by the HOSTNSP, and there are no differences visible to a user program. A NIF created for a physical NSP will run on either physical NSPs or HOSTNSPs with the selection made by the operator with the invoking ODT input message:

ID <nn> - invokes the physical NSP <nn>
ID <nn>:HOST - invokes a HOSTNSP in place of the physical NSP<nn>

When it is desired to always invoke a HOSTNSP, "HOST <nn>" may be specified in NDLLI in place of "NSP <nn>". A simple "ID <nn>" ODT input command is sufficient in this case.

Multiple copies of HOSTNSP are allowed as well as any combination of physical NSPs and HOSTNSPs.

The HOSTNSPSUPPORT library is "shared by all" such that one copy of the library is invoked, but there is one visible stack for each HOSTNSP initiated. Each HOSTNSP initiated runs as an independent runner with a visible name of "HOSTNSP<nn>". The implementation language used is NEWP.

Preconditions for invoking a HOSTNSP:

1. A path must exist from the host to the I/O base containing the LSP DLPs.
2. If the title of the library file is other than "SYSTEM/HOSTNSPSUPPORT", an ODT command of the form "SL HOSTNSPSUPPORT = SYSTEM/HOSTNSPSUPPORT" must be entered to establish the name of the HOSTNSP library file.

NOTE: A physical NSP need not be present on the system.

There is a difference between the physical NSP and the HOSTNSP in the area of NSP Dumps. A dump of a physical NSP is written to a disk file and may be subsequently analyzed by use of an NSP Dump Analyzer program. A dump of a HOSTNSP is an analyzed dump along with a program dump directed to a printer backup file; no additional analysis program is required.

In addition to the NSP dump, there are several run-time diagnostic options that may be utilized to diagnose problems. They are set and reset via the TASKVALUE attribute of the visible HOSTNSP stack (ODT "HI" command).

TASKVALUE bit no. (value)	Function
0 (1)	Force an NSP dump. This option is utilized to directly request the HOSTNSP to take an NSP dump. One dump will be taken and the request bit reset. This option may be set by: "<mix number>HI <value including bit 0>".
1 (2)	Trace I/O. Dump the NSP IOCB at the completion of each I/O operation. This option will cause the HOSTNSP to analyze the NSP IOCB for each LSP I/O operation completion. This option may be set by: "<mix number>HI <value including bit 1>". Once set, this option will remain set until reset via: "<mix number>HI <value not including bit 1>".

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

- 2 (4) Trace Requests and Results to/from DCCONTROL.
- This option will cause the HOSTNSP to trace the REQUESTS and RESULTS as they are handled.
- This option may be set by:
 "<mix number>HI <value including bit 2>".
 Once set, this option will remain set until reset via:
 "<mix number>HI<value not including bit 2>".
- 3 (8) Trace S-ops.
- Trace the S-ops of Line Control and Editor Processes.
- This option will cause the HOSTNSP to trace the S-OPs of Line Controllers and Editors. Whenever a Line Controller or Editor Process is selected for execution, the Process number, Line index, process PC Segment and Offset are dumped to the trace file. For each S-Op that is executed, the S-Op and PC Offset are dumped. Procedure Calls, Branches taken, and Expression Stack changes are noted and dumped.
- When execution of the process is interrupted a count of the number of S-ops executed is dumped.
- This option may be set via:
 "<mix number>HI <value including bit 3>"
 and will remain set until reset via:
 "<mix number>HI <value not including bit 3>".
- 5 (32) Trace Line Control/Editor Processes.
- Trace the start/stop of Line Control and Editor Processes.
- This option causes the HOSTNSP to dump information relevant to the execution of Line Control Processes and Editor Processes.
- The process index, PC segment and offset are dumped when HOSTNSP starts execution of a process.
- Process "calls" and "returns" are dumped along with the PC segment:offset information. This allows a high level trace of the operation of the process.
- When the execution of the process is interrupted, the ending PC segment:offset is dumped along with the number of S-ops executed and the process time expended.
- This option may be set via:
 "<mix number>HI<value including bit 5>"
 and will remain set until reset via:
 "<mix number>HI<value not including bit 5>".
- 6 (64) Trace "Status" operations.
- This option causes the HOSTNSP to dump information pertaining to the initiation of all Status Handler operations.
- This option may be set via:
 "<mix number>HI <value including bit 6>"
 and will remain set until reset via:
 "<mix number>HI<value not including bit 6>".

- 7 (128) Trace Switched Handler Processes.
- Trace start/stop of Switched Line Connect/Disconnect operations.
- This option causes the HOSTNSP to dump information pertaining to the initiation and any invocation of a Switched Line Handler process.
- This option may be set via:
 "<mix number>HI<value including bit 7>"
 and will remain set until reset via:
 "<mix number>HI<value not including bit 7>".

- 8 (256) Trace Switched Handler S-ops.
- Trace the S-ops of Switched Line Connect/Disconnect operations.
- This option causes the HOSTNSP to trace the execution of all Switched Line Handler Processes. Each S-op generated or executed along with its address will be dumped.
- This option may be set via:
 "<mix number>HI<value including bit 8>"
 and will remain set until reset via:
 "<mix number>HI<value not including bit 8>".

- 9 (512) S-Op Timings.
- This option, once set, remains set for the duration of the execution of the HOSTNSP. The process time for each S-op type is accumulated along with a count of the number of times the individual S-op is executed.
- As the HOSTNSP performs its shut-down process the information accumulated is analyzed and dumped. For each S-op, the information provided is:
- S-op type
 - Number of times executed
 - Total process time (in milliseconds)
 - Average process time per execution of the S-op (in microseconds)
- The primary purpose of this option is for optimizing the interpreter portion of the HOSTNSP implementation.
- This option may be set via:
 "<mix number>HI<value including bit 9>".

TASKVALUE is picked up by HOSTNSP whenever it changes, and all of the trace bits are set/reset according to the value in TASKVALUE. More than one option may be set by specifying a value which is the sum of the values for the desired options. A "<mix number>HI 34" for instance will set Trace I/O and Trace Line Control/Editor Processes.

The <mix number> in the discussion of the options is the mix number of the HOSTNSP.

The information for all of the above options is directed to the same printer backup file. The file is closed at the completion of an NSP dump and at termination of the HOSTNSP.

NOTE: The use of these trace options will have an adverse effect on system performance and may utilize a significant amount of printer backup disk.

D4158 HOSTNSP - INTERNAL EFFICIENCY CHANGES

Several internal efficiency changes have been made to improve HOSTNSP operations.

D4160 HOSTNSP - 'MULTIPLE-WAIT S-OP' IMPROVEMENT

The operation of the multiple-wait S-op has been accelerated.

D4223 HOSTNSP - "I/O" INITIATION ADDRESS SAVED

A diagnostic aid has been created by saving the code segment:offset of the INITIATEIO S-op initiating an I/O. This saved offset is displayed as part of tracing I/O operations.

D4224 HOSTNSP - EDITOR PROCESS INITIATION ACCELERATION

The initiation and termination of Editor processes has been accelerated.

D4225 HOSTNSP - SAVE "MLIP" HARDWARE ERROR FLAG

The MLIP HARDWARE ERROR flag is now saved for diagnostic usage.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

HOSTNSPSUPPORT

P1140 HOSTNSP - "HOSTNSP MLIP" ERROR FIELD RETENTION

A mechanism has been provided to retain MLIP error conditions in HOSTNSP I/O to the LSP.

P1141 HOSTNSP - HOSTNSP ABORT "S-OP" CORRECTION

The abort S-op now causes an NSP dump to occur as an aid to NDII debugging.

-P1142 HOSTNSP - "LISTLOOKUP"

LISTLOOKUP is now handled by software rather than by the LISTLOOKUP operator.

P1143 HOSTNSP - NULL STRING HANDLING

The handling of null strings has been corrected.

P1144 HOSTNSP - READY QUEUE PROCESS LINKS CORRECTED

Several corrections have been made for NSP ready queue links for NDII processes.

P1145 HOSTNSP - SET DATACOUNT AT "I/O" COMPLETE

Operation of the control block TEXTSIZE attribute at completion of LSP input operations has been corrected.

P1146 HOSTNSP - "ADD STATION/ADD LINE" CORRECTION

ADD STATION and ADD LINE requests now work correctly during reconfiguration.

P1147 HOSTNSP - REMOVE "IOCB" CORRECTION

Control blocks (IOCBs) in progress at the time of a DELETE LINE will now be handled properly.

P1148 HOSTNSP - "ZAP" PROCESS CORRECTION

A DELETE LINE request will now correctly cause line process termination and delete the line.

P1149 HOSTNSP - REMOVE GROUP TO "ZAP" LINE PROCESSES

REMOVE GROUP now causes termination of line processes for all lines within the group.

P1150 HOSTNSP - "TEST-OP" LINE COUNT CORRECTION

The test S-op now returns the correct line count.

P1352 HOSTNSP - "STATION.OUTPUTCOUNT"

"STATION.OUTPUTCOUNT" is now handled correctly in the Purge Request S-op.

P1353 HOSTNSP - "ABNORMAL-TERMINATE" RESULT MESSAGE

The line number in the ABNORMAL-TERMINATE RESULT message is now correct.

P1354 HOSTNSP - "IOCB.DATACOUNT"

The "IOCB.DATACOUNT" has been corrected.

P1355 HOSTNSP - "LINE-NOT-READY" RESULT MESSAGE

The LINE-NOT-READY RESULT message is now correctly generated.

P1356 HOSTNSP - INDEXNAME "S-OP"

The INDEXNAME S-op no longer gives an INVALID OP fault when executed.

P1357 HOSTNSP - "LIBERATE-TO-CORRECTLY-QUEUE-WAITING-PROCESS"

A process waiting on a lock is now correctly queued in the NSP Ready Queue when the lock is liberated.

P4265 HOSTNSP - LINE PROCESS ABNORMAL TERMINATION

Abnormally-terminated processes are now correctly delinked and removed.

DOCUMENT CHANGES NOTES (D NOTES)

INPUT-OUTPUT

D4105 IN-OUTPUT - "MAXSUBFILES" ATTRIBUTE

The effect of setting MAXSUBFILES in a file declaration on a label equation (with some subfile attributes also set, such as HOSTNAME or YOURNAME) is as if MAXSUBFILES were set first regardless of the actual order. That is, the subfile attributes will be applied to all the subfiles.

D4138 IN-OUTPUT - NEW VALUES FOR "SUBFILEERROR"

Two new mnemonics for the file attribute SUBFILEERROR have been added:

NOFILEFOUND An attempted open on this subfile resulted in a "No File Found" result (the result from the OPEN function or AVAILABLE attribute was 2).

UNREACHABLEHOST An attempted open on this subfile resulted in an "Unreachable Host" result (the result from the OPEN function or AVAILABLE attribute was 38).

D4172 IN-OUTPUT - "AVAILABLE" FILE ATTRIBUTE

A new value has been added for the file attribute AVAILABLE, as follows:

38 The specified host was not reachable.

D4186 IN-OUTPUT - "UNITNO" ATTRIBUTE

The description of the UNITNO attribute on Page 3-57 of the I/O Subsystem Reference Manual (Form No. 5001779), should be changed as follows:

Insert the following paragraph after the characteristics:

"For a disk file that is OPEN, the UNITNO attribute returns the core index of the disk file header. For any other file, the UNITNO attribute returns the hardware unit number to which the file is assigned, or is to be assigned."

Replace the first sentence with the following:

"The UNITNO attribute can be set to the unit number to which the file is to be assigned."

The remainder of the description of UNITNO is unchanged.

D4228 IN-OUTPUT - OPTIONAL VS. DIRECT

Direct files that are OFed now return an EOF result descriptor to READs.

As a result of this change, Page 3-38 of the I/O Subsystem Manual (Form No. 5001779), under "OPTIONAL", the second paragraph, should now read as follows:

"Non-present optional files are input-only files. All read statements result in end-of-file action. Writing on the file is illegal and will cause program termination."

D4244 IN-OUTPUT - "BCL" FILES

An open error is now given for trying to open a file with INTMODE or EXTMODE BCL on a B5900 system (B5900 systems do not support BCL).

D4320 IN-OUTPUT - "END-OF-TAPE" REPORTING ON DIRECT TAPE FILES

On Multiplexor systems, indications of end-of-tape in the attributes IORESULT and IOERRORTYPE are suppressed when reading direct tape files.

This action does not take place on MLIP systems; thus, all READs occurring after the end-of-tape marker return a value of IORESULT indicating end-of-tape (and possibly end-of-file as well). The attribute IOERRORTYPE returns a value indicating end-of-tape, unless end-of-file is encountered, in which case end-of-file is reported instead.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

INPUT-OUTPUT

P1013 IN-OUTPUT - FIX "AREALENGTH" FILE ATTRIBUTE REPORTING

The AREALENGTH file attribute now reports the proper value when interrogated for an open file. The value returned is now correct when BLOCKSIZE is not a multiple of sectors.

P1065 IN-OUTPUT - "HOSTNAME" OF UNASSIGNED FILE

The HOSTNAME attribute now returns the local host name if no value has been set by the user.

P1170 IN-OUTPUT - "PROTECTED" VS. "REWIND"

Sometimes when a disk file with PROTECTION=PROTECTED was closed with a REWIND statement, the EOF pointer would not be updated correctly. This problem has been corrected.

P1171 IN-OUTPUT - REELSWITCH VS. USE

If a COBOL program were DSed while executing a TAPE "USE AFTER ENDING PROCEDURE", the system would take a FORGETCHECK dump. This problem has been corrected.

P1197 IN-OUTPUT - "SPO" VS. "UNITNO"

Sometimes, when OPENing a SPO file with UNITNO set, the MCP would issue the RSVP "REQUIRES SC<nn>" because the ODT was not ready. SPO file assignment has been changed so that it no longer depends on the ready status of the unit.

P3321 IN-OUTPUT - DATACOM DIRECT "I/O"

Datacom direct I/O no longer causes system faults when used by a swapjob on a multiprocessor system.

P3815 IN-OUTPUT - REMOTE BACKUP FILES

A problem with some files with KIND=REMOTE and BACKUPKIND=DISK was causing SEG ARRAY faults. This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

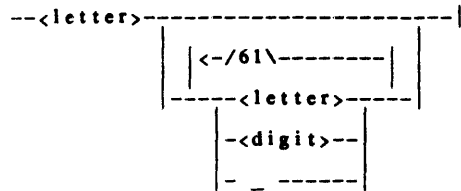
INTERACTIVEXREF

D4166 IXREF - ALLOW UNDERSCORES IN IDENTIFIERS

The definition of identifiers has been changed to include underscore ("_") characters.

The syntax diagram for an <identifier> is now the following:

<identifier>



Underscores are treated as significant characters in an identifier.

INTERACTIVEXREF has been changed to allow it to reference these new identifiers.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

INTERACTIVEXREF

P1370 IXREF - WRONG SEQUENCE NUMBERS FOR ALIASES

A problem with aliases to procedures in alternative modules having the wrong sequence numbers has been corrected.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

DOCUMENT CHANGES NOTES (D NOTES)

JOB FORMATTER

D4272 JOBFORMAT - INTERVAL LOGGING FOR FILES

File interval logging logs usage information of open files at time intervals.

The File Interval record has a major type of 1 and a minor type of 10. The entry format is listed below.

File Interval Record

Word	Format
0	Record Group Description (standard format)
1	Current date, in binary Julian format (standard format).
2	Current time, in units of 2.4 microseconds (standard format).
3	Log entry type -- major and minor classifications, and length of log entry (standard format).
4	File size in units of words; valid for disk only.
5	Link to internal name. Field [19:20] gives the word of the log entry where the name starts. Names are in standard form. The word number refers to the log entry after it is rebuilt into a single entry; that is, the record group description words of the second and succeeding entries are skipped in this count. Field [39:20] gives the length of the name in units of words.
6	Link to external name. Format is same as word 5.
7	Description word. Field [39:8] gives the unit number. Field [31:8] gives the unit type. Field [23:8] gives the value of the FILEKIND file attribute. Field [7:7] gives the value of the MYUSE file attribute.
8	Name qualification information. Field [42:15] gives the USASI generation number. Field [27:11] gives the file SAVEFACTOR. Field [16:17] gives the creation date (in binary Julian format).
9	File blocking information. Field [47:16] gives the file BLOCKSIZE. Field [27:11] gives the file MINRECSIZE. Field [15:16] gives the file MAXRECSIZE.
10	Reserved for transactioncount.
11	Elapsed I/O time.
12	Miscellaneous information. For a tape file: Bit 15 is on for a multi-reel file, and field [14:15] gives the reel number. For a disk file: Bit 47 is on for an IAD file. Bit 46 is on for a permanent file. Bit 44 is on for an updated file. Bit 43 is on for a crunched file. Bit 42 is on for a protected file. Bit 41 is on for a multipack disk pack file. Bit 40 is on for an Interchange disk pack file. Bit 32 is on for a duplicated file, and field [31:4] contains the number of copies. Field [27:4] gives the speed. Field [23:24] gives the value of the AREASIZE attribute. For units other than tape or disk, this word will contain zeroes.
13	Serial number for a tape file (six EBCDIC characters); zero for other units.
14	Unit information. Field [25:14] contains the unit cycle number. Field [11:8] contains the unit version number.
15	File parameters: Bit 46 will be set if the file is a valid subfile. Field [39:16] will contain the subfile index if valid.
16	Physical count. Field [47:24] contains the count for reads. Field [23:24] contains the count for writes.
17	Value of FORMMSSG.
18	Link to YOURHOSTNAME value (for port files).
19	Link to YOURNAME value (for port files).
20	Value for MAXSEGSIZE attribute (for port files).
21	Link to YOURUSERCODE value (for port files).
22-28	LOGGING-INFO values (for port files).

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

Word 22 contains the number of messages sent.
Word 23 contains the number of messages received.
Word 24 contains the number of segments sent.
Word 25 contains the number of segments received.
Word 26 contains the number of retransmissions.
Word 27 contains the number of control frames sent.
Word 28 contains the subfile index.
29+ Variable-length data

New records have been added to the File Close record, for port information. The new format is listed below.

File Close Record

Word -----	Format -----
0	Record Group Description (standard format)
1	Current date, in binary Julian format (standard format).
2	Current time, in units of 2.4 microseconds (standard format).
3	Log entry type -- major and minor classifications, and length of log entry (standard format).
4	File size in units of words; valid for disk only.
5	Link to internal name. Field [19:20] gives the word of the log entry where the name starts. Names are in standard form. The word number refers to the log entry after it is rebuilt into a single entry; that is, the record group description words of the second and succeeding entries are skipped in this count. Field [39:20] gives the length of the name in units of words.
6	Link to external name. Format is same as word 5.
7	Close description word. Field [39:8] gives the unit number. Field [31:8] gives the unit type. Field [23:8] gives the value of the FILEKIND file attribute. Field [15:8] gives the close error number. Field [7:7] gives the value of the MYUSE file attribute. Bit 0 is an error flag, indicating that field [15:8] is valid.
8	Name qualification information. Field [42:15] gives the USASI generation number. Field [27:11] gives the file SAVEFACTOR. Field [16:17] gives the creation date (in binary Julian format).
9	File blocking information. Field [47:16] gives the file BLOCKSIZE. Field [27:11] gives the file MINRECSIZE. Field [15:16] gives the file MAXRECSIZE.
10	Reserved for transactioncount.
11	Elapsed I/O time.
12	Miscellaneous information. For a tape file: Bit 15 is on for a multi-reel file, and field [14:15] gives the reel number. For a disk file: Bit 47 is on for an IAD file. Bit 46 is on for a permanent file. Bit 44 is on for an updated file. Bit 43 is on for a crunched file. Bit 42 is on for a protected file. Bit 41 is on for a multipack disk pack file. Bit 40 is on for an Interchange disk pack file. Bit 32 is on for a duplicated file, and field [31:4] contains the number of copies. Field [27:4] gives the speed. Field [23:24] gives the value of the AREASIZE attribute. For units other than tape or disk, this word will contain zeroes.
13	Serial number for a tape file (six EBCDIC characters); zero for other units.
14	Unit information. Field [25:14] contains the unit cycle number. Field [11:8] contains the unit version number.
15	Close parameters: Bit 46 indicates whether a valid subfile was closed. Field [39:16] contains the subfile index. Field [23:8] contains the disposition. Field [15:8] contains the association. Field [7:8] contains the close type.
16	Physical count. Field [47:24] contains the count for reads. Field [23:24] contains the count for writes.
17	Value of FORMSSG.
18	Link to YOURHOSTNAME value (for port files).
19	Link to YOURNAME value (for port files).
20	Value for MAXSEGSIZE attribute (for port files).
21	Link to YOURUSERCODE value (for port files).
22-28	LOGGING-INFO values (for port files). Word 22 contains the number of messages sent. Word 23 contains the number of messages received. Word 24 contains the number of segments sent.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

- Word 25 contains the number of segments received.
- Word 26 contains the number of retransmissions.
- Word 27 contains the number of control frames sent.
- Word 28 contains the subfile index.
- 29+ Variable-length data

D4273 JOBFORMAT - "BNA" LOG FORMATS

BNA log records will comprise a new major classification of log entries. All BNA records will fall under the major type BNA (11). Each of the entry types listed on the following pages constitute a specific minor type within the BNA major type.

The standard system log format uses the first four words of every log entry as follows:

Word 0	Record Group Description
	[47:8] Cardinality
	[39:8] Number of records in this entry
	[31:16] Job number
	[15:16] Task number
Word 1	Date, binary (Julian)
Word 2	Time of day, 2.4 microseconds
Word 3	Type of entry
	[47:16] Length of entry in words, as follows:
	[47:6] Length of fixed part of record
	[41:10] Length of entire log record
	[31:16] Major type (Job, MCS, BNA, etc.)
	The value will be eleven (11) for major type BNA.
	[15:16] Minor type (BOJ, EOJ, Open, Close, etc.)
	These values are listed with the formats on the following pages.

Records are set up so that fixed-length values are placed at the beginning of an entry. Any variable-length values (attribute names, frame text, etc.) are placed after the fixed-length values, and have a link pointing to them from the fixed-length area. The standard link format used by Log analyzer contains the following: field [19:20] gives the word of the log entry where the information starts, and field [39:20] gives the length of the name, in units of words.

Type	Description	Word	Contents
1	Set NSM Attribute	4	link to attribute name
2	Set PLM Attribute	5	link to attribute value
3	Set Router Attribute	6	result
4	Set SLM Attribute		0 - OK 1 - UNKNOWN ATTRIBUTE 2 - INVALID VALUE 3 - READ-ONLY ATTRIBUTE
		7+	variable-length data from words 4 and 5
5	NSM Phase Change	4	new phase 2 - STARTNET 3 - DATA-ENTRY 4 - WAIT-STATION 5 - WAIT-ROUTER 6 - WAIT-PORT 7 - NORMAL-OPERATION 11 - SLOW-SHUTDOWN 12 - FAST-SHUTDOWN 13 - ROUTER-SHUTDOWN 14 - STATION-SHUTDOWN
6	NSM Log Report PLM Add Host	4	Implemented as "LC" ODT command. link to hostname
7	NSM Add Node	5	host information: node address [15:16] MAXSEGSIZE [31:16] translate-only flag [32:1] reachability flag [33:1]
		6	result 0 - OK 51 - DUPLICATE HOSTNAME 66 - HOSTNAME/NA MISMATCH
		7+	variable-length data from word 4
8	PLM Set Host Status	4	link to hostname
		5	host information: ready/saved flag [0:1] 0 = ready 1 = saved clear flag [1:1]
		6	result 0 - OK 64 - UNKNOWN HOSTNAME
		7+	variable-length data from word 4
9	PLM Delete Host	4	link to hostname
10	NSM Delete Node	5	host information: node address [15:16] delete/allow-trans [16:1] 0 = delete host 1 = allow translation
		6	result 0 - OK 52 - HOST COMMUNICATING 53 - HOST VALIDATE FALSE 54 - ROUTER VALIDATE FALSE 64 - UNKNOWN HOSTNAME
		7+	variable-length data from word 4
11	PLM Error Report	4	error information: error source [1:2] Port Level Manager = 1 Port Level = 2 Port = 3 error code [10:8]
		5	link to additional error information
		6+	variable-length data from word 5
12	PLM Log Report		For the Subport Close and Subport Periodic data records, additions will be made to the current File Close record, to incorporate port information, as follows: 16 Link to YOURHOSTNAME

(port files only)
 17 Link to YOURNAME value
 (port files only)
 18 MAXSEGSIZE (port files)
 19 Link to LOGGING-INFO
 (port files only)
 20 Link to YOURUSERCODE
 (port files only)
 21+ Variable-length data

In addition, dialog established, dialog terminated, validation information report, and AX for missing remote have the following format:

4 report information
 remote node address [15:16]
 report type [31:16]
 dialog est [16:1]
 dialog term [17:1]
 validation info [18:1]
 unchanged mss [19:1]
 (currently unused)
 AX missing remote [20:1]

The remainder of the record depends on the report type:

Dialog established and dialog terminated:

5 remote hostname
 6 local subport index
 7 termination reason
 (dialog term report only)
 8 variable-length data

Validation info report:

5 remote host name
 6 remote incarnation id
 7 variable-length data

AX for missing remote report:

5 link to text of AX match
 6 variable-length data

13 Router Link RF Change 4 neighbor NA
 5 new factor
 6 result
 0 - OK
 65 - UNKNOWN NA

14 Start Trace 4 trace source NA
 5 trace destination NA
 6 trace reference
 7 result
 0 - OK
 38 - SOURCE NA UNREACHABLE
 37 - DESTINATION NA UNREACHABLE

15 Routing Refresh

16 Router Control Frame Send 4 link to remote hostname
 5 link to frame text
 6+ variable-length data

17 Router Control Frame Received Report 4 link to remote hostname
 5 link to frame text
 6+ variable-length data

18 Router Frame Error Report 4 destination node address
 5 neighbor node address
 6 error information:
 error source [7:8]
 local = 0
 remote = 1
 error code [15:8]
 0 - OK
 1 - UNDELIV MSG
 2 - TCNT OVERFLOW
 3 - MSS ERROR
 4 - INVALID ATTR ID
 5 - INVALID NODERF VALUE
 6 - UNKNOWN CTL DES
 7 - NO SPACE IN RNT

```

          9 - TRACE RESULT
            DESTINATION UN
          10 - INVALID ROUTER PHASE
          12 - NETCH, SNA NOT EXIST
          13 - INCONSISTENT DEFAULTS
          15 - FRAME TOO SHORT
          16 - FRAME LEVEL ERROR
          17 - UNKNOWN RFT
          18 - DNA INVALID
          19 - UNKNOWN ONA
          21 - ILLEGAL NEWRF VALUE
          22 - LINKCH, XNNA NOT EXIST
          23 - CANNOT DELETE NEIGHBOR
          24 - ATTACH, NEIGHBOR NOT
              EXIST
          25 - RESTART, NEIGHBOR NOT
              EXIST
          26 - DNASTREQ,DNA NOT EXIST
          27 - LINKCH, VERSION
              MISMATCH
          7  link to frame text
          8+ variable-length data

19  DNA Status Change      4  Destination NA
    Report                  5  Neighbor NA
                              6  status information (1):
                                MAXSEGSIZE [15:16]
                                PFMAXSEGSIZE [31:16]
                                DNA resistance factor [47:16]
                              7  status information (2):
                                hop count [15:16]
                                reachability [16:1]
                                type of change:
                                  neighbor node addr [47:1]
                                  max seg size [46:1]
                                  PF max seg size [45:1]
                                  reachability [44:1]
                                  resistance factor [43:1]
                                  hop count [42:1]

20  RM Copy Report        4  Neighbor NA (recv from)
                              5  Neighbor NA (sent to)
                              6  link to frame
                              7+ variable-length data

21  RM Summary Report     4  array passed from RM, with
                              one word of frame information
                              for each node pair in use:
                                control frames sent w/o error
                                average length (in bytes)
                                info frames sent w/o error
                                average length (in bytes)
                                frames sent w/error
                                average length (in bytes)

24  Router Node Existence 4  node information
    Report                  node address [15:16]
                              operation [16:1]
                                0 - ADD
                                1 - DELETE
                              5  result
                              6+ variable-length data

25  SLM Modify Station     4  link to station name
                              5  reserved for link to
                              profile name (Add only)
                              6  link to attribute list
                              7  result
                                0 - OK
                                82 - INVALID STATION NAME
                                84 - STATION ATTRIBUTE NOT
                                    SETTABLE
                                120 - INVALID MODIFY OPTION
                                148 - STATION NOT IN CORRECT
                                    STATE
                              8+ variable-length data

30  SLM Add Connection     4  neighbor NA
                              5  direction
                                permanent = 1
                                incoming = 2
                                outgoing = 3
                              6  link to name
    
```

		7	link to call data
		8	init-quantity
		9	result
			0 - OK
			147 - STATION NOT IN CORRECT STATE
			110 - STATION IN ENSEMBLE
			82 - INVALID STATION NAME
		10+	variable-length data
31	SLM Modify Connection	4	neighbor NA
		5	direction
			permanent = 1
			incoming = 2
			outgoing = 3
		6	link to name
		7	link to call data
		8	init-quantity
		9	result
			0 - OK
			147 - STATION NOT IN CORRECT STATE
			84 - STATION ATTRIBUTE NOT SETTABLE
		10+	variable-length data
32	SLM Delete Connection	4	neighbor NA
		5	direction
			permanent = 1
			incoming = 2
			outgoing = 3
		6	link to name
		7	reserved for link to call data (Add and Modify)
		8	reserved for init-quantity (Add and Modify)
		9	result
			0 - OK
			147 - STATION NOT IN CORRECT STATE
			113 - CONNECTION BUSY
			82 - INVALID STATION NAME
		10+	variable-length data
33	SLM Clear Call	4	neighbor NA
		5	link to station/ensemble name; bit 40 indicates station or ensemble
		6	link to call-data (Establish only)
		7	result
			0 - OK
			145 - ALREADY CLOSED
			82 - INVALID STATION NAME
		8+	variable-length data
34	SLM Establish Call	4	neighbor NA
		5	link to station/ensemble name; bit 40 indicates station or ensemble
		6	link to call-data (Establish only)
		7	result
			0 - OK
			147 - STATION NOT IN CORRECT STATE
			113 - CONNECTIONS TO NEIGHBOR
			85 - INVALID ENSEMBLE
			82 - INVALID STATION NAME
		8+	variable-length data
35	SLM Await Call	4	link to station/ensemble name; bit 40 indicates station or ensemble
		5	result
			0 - OK
			87 - ENSEMBLE NOT INCOMING-CAPABLE
			82 - INVALID STATION NAME
		6+	variable-length data
36	SLM Send Test	4	link to station name
		5	link to text
		6	result
			0 - OK

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

		7+	variable-length data	
37	BDLC Test Command	4	link to station name	
	Received Report	5	link to text	
38	BDLC Test Response	6+	variable-length data	
	Received Report			
39	SLM Link Reset	4	link to station name	
		5	reset origin:	
			0 - LOCAL	
			1 - REMOTE	
		6	BDLC Reason Code	
40	SLM Open Connection Port	4	link to station name	
		5	link to call data	
		6	result	
			0 - OK	
			147 - STATION NOT IN CORRECT STATE	
			24 - RETRY COMPLETED	
		7+	variable-length data	
41	SLM Close CPD	4	link to station name	
		5	result	
			0 - OK	
			147 - STATION NOT IN CORRECT STATE	
		6+	variable-length data	
42	Validate Station Attach	4	link to station name	
		5	result	
			0 - OK	
			145 - ALREADY CLOSED	
			82 - INVALID STATION NAME	
		6+	variable-length data	
43	Manual Detach	4	link to station name	
		5	result	
			0 - OK	
			147 - STATION NOT IN CORRECT STATE	
			113 - CONNECTIONS TO NEIGHBOR	
			82 - INVALID STATION NAME	
		6+	variable-length data	
44	Save	4	link to station name	
		5	result	
			0 - OK	
			143 - ALREADY SAVED	
			101 - INVALID SLM PHASE	
			82 - INVALID STATION NAME	
		6+	variable-length data	
45	Ready	4	link to station name	
		5	result	
			0 - OK	
			147 - STATION NOT IN CORRECT STATE	
			82 - INVALID STATION NAME	
		6+	variable-length data	
46	Open Station Dialog	4	link to station name	
		5	result	
			0 - OK	
			147 - STATION NOT IN CORRECT STATE	
		6	variable-length data	
47	Close Station Dialog	4	link to station name	
		5	result	
			0 - OK	
			147 - STATION NOT IN CORRECT STATE	
			145 - ALREADY CLOSED	
		6+	variable-length data	
48	SLM Attach Report	4	link to station name	
		5	neighbor NA	
		6	VAN id	
		7	link information:	
			WLMSS	[15:16]
			link speed	[37:22]
			link efficiency	[45:8]
		8+	variable-length data	

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

49	SLM Detach Report	4	link to station name
		5	neighbor NA
		6+	variable-length data
50	Neighbor Restart Report	4	neighbor NA
51	Neighbor Remote Busy Report	4	neighbor NA
52	Station Log Report	4	link to station name
		5	neighbor NA
		6	node information:
			Disconnect indicator [0:1]
		7	link to report and report type:
			report type [47:8]
			frame received = 4
			IRSP received = 7
			CPD open closed = 8
			invalid BDLC address = 9
			BDLC frame not octet
			multiple = 10
			unexpected CPD close = 12
			non-BNA caller = 13
			link reset local = 14
			FRMR received = 15
			retry exceeded OSD = 16
			link reset remotely = 17
			FRMR sent = 18
			unexpected DM disk
			received = 19
			SD reopened = 20
			set remotebusy = 24
			clear remotebusy = 25
		8+	variable-length data
53	Station Validation Failure	4	link to station name
		5	neighbor NA
		6	node information:
			local/remote [16:1]
			local = 0
			remote = 1
		7	link to greeting 1 or
			greeting 2 message
		8+	variable-length data
54	Monitor Report	4	link to station name
		5	neighbor NA
		6	station type
			0 = GMM
			1 = BDLC
		7	frames sent
		8	frames received
		9	I-frames sent (BDLC)
		10	FCS failures (BDLC)
		11	memory errors (BDLC)
		12	short frames (BDLC)
		13+	variable-length data
56	OIM Report	4	origin
		5	link to text of OIM
		6+	variable-length data

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

JOB FORMATTER

P1294 JOBFORMAT - PRINT UNIT NUMBER OF PACK

A problem that caused LOGANALYZER (which includes code from JOBFORMATTER) to not print the unit number of a pack for OPEN/CLOSE operations has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

LOADER

P1297 LOADER - DISKCYCLE

Catalog and directory files loaded by the Mark 32 LOADER on cold start had their CYCLE set to 0. This error has been corrected; the CYCLE is now set to 1.

P1365 LOADER - SETTING OF OPTIONS

LOADER no longer faults when an option is set or reset.

DOCUMENT CHANGES NOTES (D NOTES)

LOG ANALYZER

D4274 LOGANALY - "BNA LOGANALYZER" CHANGES

A new operator option, BNA, will be added to the list of operator options. The format will be the same as for other options.

BNA log reports will have the following general format:

```

12/17/79 06:07:50 7312 PLM  CMD  DIALOG TIMEOUT ATT CHANGED
                                     TO 30 MINUTES
12/17/79 06:07:52 7450 SLM  RPT  NEIGHBOR NODE 7 RESTARTED
12/17/79 06:10:55 7792 NSM  CMD  HOST GREEN, NODE 15 DELETED
12/17/79 06:11:23 7800 NSM  CMD  PHASE CHANGED TO WAIT-PORT
12/17/79 06:12:38 7852 RTR  CMD  ROUTER NODE EXISTENCE REPORT:
                                     NODE # 2 ADDED

```

The level (PLM, SLM, NSM, or RTR) tells which BNA level the entry came from. The type (CMD or RPT) indicates whether the entry logs the occurrence of a command or a report.

The following changes should be made to the LOGANALYZER section of the SOG Reference Manual, Volume 2 (Form No. 5001688):

<u>Page</u>	<u>Modification</u>
6-1-1	Add BNA to the list of operator options, between MCS and JOB.
6-1-2	Add the following at the end of the page: LOG 1400 7/8/80 TO 1600 7/8/80 BNA Retrieves all BNA entries between 1400 and 1600 hours on July 8, 1980.
6-1-4	Add the following at the end of the page: BNA Prints a history of BNA activity. See Figure 20.
6-1-21A	Add Figure 20, showing a typical BNA log report as described above.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

LOG ANALYZER

P1240 LOGANALY - IMPLEMENTATION OF READ EXTENDED STATUS

A "read extended status" operation is done on MLIP systems for those DLPs which require a "read extended status" to unlock a control (currently, just the HT).

LOGANALYZER will print out the extended status if required. In the case where the extended status itself was not obtainable, the result descriptor from the "read extended status" is printed.

In order to retrieve the correct extended status information, the latest level of DLP microcode is required.

Installations should ensure that their HT-DLP hardware is up to date.

Performance of older DLPs is not impaired; however, spurious extended status results may occur.

DOCUMENT CHANGES NOTES (D NOTES)

LOGGER
-----D3636 **LOGGER - "SEG ARRAY" WHEN TOTALING, AVERAGING SAME ITEM**

Previously, when **LOGGER** ran a report with totaling and averaging for the same item, a **SEG ARRAY** error would occur because **LOGGER** was not designed to do both totaling and averaging for an item. Totals and averages could be done on different items but not on the same item. Now, totals and averages can be done on the same item. A bit has been added in each word of the totals and averages array to indicate totaling for that item. Previously, one bit was used to indicate averaging if set and totaling if reset.

Also, the printing of totals and averages has been split into two lines.

As a result, the examples in the **SOG Reference Manual, Volume 2 (Form No. 5001688), Pages 7-1-30, 7-1-32 and 7-1-36**, should be changed to show a line for totals and a line for averages.

Effective with this change, the current **LOGGER** should not be run against year-to-date files which were created prior to this change; the results could be erroneous. Those files should be processed using the existing **LOGGER** which first created them.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

LOGGER

P1005 LOGGER - UPDATE 'YTD' FILE WITH MULTIPLE BREAK ITEMS

SYSTEM/LOGGER generated the error message "UPDATE REPORT DOES NOT MATCH YEAR-TO-DATE FILE" when updating a year-to-date file with multiple break items.

It now runs normally.

P3766 LOGGER - BREAK ON "JOBQUEUEDTIME" CAUSES "SEG ARRAY"

When LOGGER referenced a break item beyond word 60 of the JOBSUMMARY file (i.e., JOBENTRYTIME or JOBQUEUEDTIME), a SEG ARRAY error would occur because items were added to JOBSUMMARY without increasing the size of the OLDVALUES array. The size of OLDVALUES has been increased from 60 words to 64 words, thus avoiding the SEG ARRAY error.

P3767 LOGGER - "LOGGER" MIXES "ORGUNIT"

LOGGER did not distinguish between unit and LSN for ORGUNIT. If a peripheral unit and station had the same number, LOGGER reported both as the same ORGUNIT, making it impossible to know from where a task or job entered the mix.

Now, LOGGER distinguishes unit and LSN for ORGUNIT by printing "UNIT:" or "LSN:" in front of the corresponding number.

Effective with this change, LOGGER reports for year-to-date files created prior to this change cannot be run with this change. Those files should be processed using the existing LOGGER which created them.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

LTTABLEGEN

P1010 LTTABLEGEN - COMPLETE "TRAIPTABLES"

1. LTTABLEGEN will now build a SYSTEM/TRAIPTABLE file that contains traintables for TRAINIDs, 1 through 37.
2. Train printers that return a trainid of zero were causing problems. The MCP uses the default traintable for such printers. However, if a program requested a specific trainid, even if it was the default id, the MCP wouldn't load it. Now, if the requested id is the default (or subset of it) the MCP will load the table.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

MAKEUSER

P1102 MAKEUSER - "ACCESSCODELIST"

An error in the Mark 32 release of MAKEUSER has been corrected. Symptoms of the error include the following:

1. When entering ACCESSCODELIST items that use USERCODE/PASSWORD pairs, the error message "ARRAY TOO LARGE" appears, and MAKEUSER terminates.
2. When a non-privileged user tries to use MAKEUSER, an error message "USERDATA FAULT #8" appears instead of the appropriate "SECURITY VIOLATION", and MAKEUSER terminates.

P1232 MAKEUSER - "SET LIST" PROBLEMS

A problem has been corrected in which, if LIST were SET, ACCESSCODELIST and FILENAMELIST would not always be displayed.

DOCUMENT CHANGES NOTES (D NOTES)

MCP

D4103 MCP - TIMESTAMP AT "BOT" IS RECORDED IN "EOT" LOG

The timestamp for the time at which a task was initiated has been added to the EOT log record. This timestamp identifies the month, day, year, hour, minute, and second of task/job initiation. The new word is referenced by EOTBOTTIMESTAMP and is the 34th word of EOT type log records on the Mark 32 release.

In addition, DUMPANALYZER will print the BOT timestamp when analyzing the PIB.

D4104 MCP - NEW WORDS IN "CLOSE" LOG RECORD

Two new words have been added to CLOSE log record. The first word contains the physical I/O counts of the file, and the second word contains the link to the FORMMESSAGE or FORMID of the file.

The physical I/O count word is referenced by CLOSEPHYCNTX at word #15 of the CLOSE log record and is divided into two fields: the read count (PHYRD) is at [47:24], and the write count (PHYWR) is at [23:24].

The FORMMESSAGE of the file is recorded in the CLOSE log record if it is not null or ".". The link to this information is referenced by CLOSEFORMMSGX at word #16 of the CLOSE log record.

D4132 MCP - NO ROOM FOR "PROGRAMDUMP"

If PROGRAMDUMP did not have enough room on the stack to execute, it would just exit without any message.

PROGRAMDUMP will now print out a message if there is not enough space to execute.

D4149 MCP - INITIALIZATION STOP

A conditional HALT has been inserted in the MCP initialization code. If Conditional Halt is enabled at Halt/Load time, the system stops after the MCP kernel has been established but before such activities as label reading or disk/pack complementing have begun. On a B6800 multiprocessor system, only the halt/load processor senses the conditional halt.

The register contents at the stop are:

```
A:4'11111111111111'
B:D[0] setting
X:4'00008F800000'      X and Y form a letter "I"
Y:4'00001F100000'      in the B6700 binary display
```

(All four registers have tag=2.)

The initialization stop is intended to facilitate establishing special hardware or software test situations. It is a replacement for the "M" stop that existed in Mark 30 and earlier versions of the MCP.

D4167 MCP - STANDARD TAPE LABELS VS. NEW SYSTEMS

New systems, starting with the B5900, will not support "Burroughs Standard Tape Labels" (i.e., the tape labels only written on the B5500 and initially on the B2500/B3500). If such a tape is mounted on a new system, the tape will come up "UNLABELLED". The new systems will continue to recognize the B6700/B6800/B6900 "USASI-style" tape labels.

D4170 MCP - TWO NEW LIBRARY PARAMETER TYPES

The library parameter checking procedure now recognizes two new parameter types, direct files and direct arrays. Originally, these were flagged as unknown parameter types.

D4198 MCP - "B5900 BCL" MESSAGE

BCL pointers are not supported on the B5900. Any use of BCL pointers causes the offending program to be terminated and the following message emitted: "B5900 IS NOT BCL CAPABLE". The fault reason is INVALID OP, which can be trapped (in ALGOL) using the ON INVALIDOP fault statement.

D4211 MCP - "SYSTEMSTATUS" TYPE "3" MODIFICATION

The SYSTEMSTATUS mix information call (Group 3) has been modified to return a result of TRUE with field [i2:1]=1 and [i1:8]=33 if the mix number is that of an active session. Please note that the field [i2:1] is only valid if [i1:8] is 33.

D4220 MCP - "PRE-MARK 31" CODEFILES ON B5900

Codefiles compiled on pre-Mark 31 level software cannot be run on the B5900. The MCP will terminate the execution of any such codefile.

D4221 MCP - "TRAINID" FIELD IN "UNITTABLE"

The TRAINID field in the UNITTABLE has been widened to accommodate train IDs from 0 to 255, thus requiring changing the layout of the UNITTABLE word.

The description of SYSTEMSTATUS cases 4 and 13 in the SYSTEMSTATUS Reference Manual should be modified as follows:

Page 6-8 (Type 4 - General Unit Information)

UTRAINID	is now at field 34:8
ULABLEREAD	is now at field 39:1
URESERVEDFORMAINT	is now at field 37:1
ULPFMED	is now at field 36:1
USTKASSIGNED	is now at field 35:1

Page 13-4 (Type 13, V2=1 - General Unit Information)

The train ID of the printer is in field 21:8.

D4241 MCP - "TIME(23)" INDICATES BCL SUPPORT

Bit 46 of the result from TIME(23) will be returned as 1 for systems which do not support BCL, and 0 for systems which do support BCL.

The complete layout for TIME(23) is the following:

{47:1}	Machine has vector mode
{46:1}	1 for EBCDIC systems 0 for EBCDIC or BCL systems
{45:22}	** unused **
{23:16}	Serial number of system
{7:8}	System type; e.g., 68,69,78

D4264 MCP - CATALOG BLOCK VERSION IMPLEMENTED

A level for catalog blocks has been created for Mark 32 MPCs. The level for all catalog blocks created, entered or changed by a Mark 32 PR1 MCP is 1.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

MCP

P1008 MCP - STACK OVERFLOW IN "KANGAROO"

An unexpected stack overflow in KANGAROO has been corrected.

P1011 MCP - "READALABEL" CODESTRING

READALABEL would get an INV OP when processing an installation supplied "codestring" that involved 6-bit pointers.

This problem has been corrected.

P1016 MCP - LIBRARY PARAMETER "INVALID OP"

This patch corrects the problem where when checking library parameters, an INV OP would occur. This only happened when there was more than one procedural entry point and the last one checked came up with parameter mismatch.

P1030 MCP - "OPEN" VS "BACKUP"

When a backup file was DSed for "NO BUFFER SPACE", the system would often hang. This problem has been corrected.

P1033 MCP - TIMING HOLE IN "DOTASKINITIATION"

When initiating a library from the linker, there was a timing hole where the library could unfreeze before the USER had a chance to add himself as a USER of that library. When the USER eventually tried to add himself as a USER of that library stack, the stack itself could be gone. This problem has been corrected.

P1034 MCP - RESTORE SAVED "TEMPLATE MOM"

For CANCEL to work it is necessary to save off the codefile address and length of the original template. When CANCEL was implemented, this information was saved in the template right after the library parameter block. If, after saving off this information, the LPB is expanded, the CANCEL information is destroyed. The CANCEL information is now saved and restored when the LPB is expanded.

P1036 MCP - "DP" VS. "STATUSCHANGE"

Certain combinations of ODT unit numbers and diskpack unit numbers were preventing MEMDUMP from working correctly. This problem has been corrected.

P1037 MCP - "DUP FAMILY NAME"

If an error occurred while rebuilding a disk that had a DUP FAMILYNAME, then the "DUP name condition" would persist until the next haltload. The MCP has been corrected so that the "DUP name" condition will go away as soon as any other "rebuild error" occurs.

P1038 MCP - MISCELLANEOUS CORRECTIONS

1. TAPEPARITYRETRY will assume tape moved if parity error (C81) occurs.
2. TAPEPARITYRETRY will not use NEWPERETRY algorithm to recover from errors encountered while writing VOL1 label records.
3. Tape block count and record count error messages now include number expected and number read.

P1054 MCP - "SUBSPACES" ATTRIBUTE ERRONEOUSLY INHERITED

A task initiated with SUBSPACES=0 will no longer inherit the subspaces attribute of its parent.

P1055 MCP - RESTART "ABORT" MESSAGES

Messages for an aborted restart are now always reported under the restarting job mix number. Previously those messages were reported under either the job or the task mix number according to the different steps of a restart.

P1061 MCP - ABORTED RESTART

Due to an error during modification of CALLKANGAROO, an aborted restart tried to kill the job using a wrong stack number. This situation could result in abnormal termination of other jobs or stacks. This problem has been corrected.

P1066 MCP - "READALABEL" VS. FOREIGN TAPES

The MCP has the capability of recognizing a limited set of tape labels from systems other than B6000/B7000 systems; e.g., B1700/B1800 and B3500/B4700 systems. A problem existed on the Mark 32 release such that some or all of this set of tape labels was no longer recognized; i.e., appeared "unlabelled". This problem has been corrected.

P1067 MCP - "PAST" CONTINUATION

1. The PAST file became corrupted if there were a large number of disk packs in the volume library. This problem has been corrected.
2. When POing a pack, PO now turns off the pack before calling PACKREMOVE; thus, the operator wait has been reduced.

P1084 MCP - CORRECT CANCELLING OF NONEXISTENT LIBRARIES

Previously, the MOM pointed back into the codefile without fixing up the library descriptor. If a library was not running and CANCEL was called for that library, the library descriptor pointed to where the library template previously was. This problem has been corrected.

P1085 MCP - "DCHOLD" SAVE-CORE BUILDUP

Each call on the DCALGOL HOLD intrinsic could cause an allocation of save memory (via GETAREA) which was not returned. The error, which has been corrected, occurred in any non-swap task running in a monolithic system.

P1095 MCP - "FLATREADER" READLOCK LOOP

A timing hole that could cause a readlock loop when the MCP procedure FLATREADER was used has been corrected.

P1112 MCP - CRITICAL BLOCK EXIT AND LIBRARIES

A task that processes a procedure of a library from the same block that contains the library template, and then takes a critical block exit, is no longer system fatal.

P1120 MCP - "ADM EVENT" LOGGED

A problem where ADM EVENT <option> would appear in the log as options being set/reset has been corrected.

P1121 MCP - SHORT DISK "I/O"

Both B6700 and B6800 systems test the length of READs and WRITEs. If the result descriptor indicates that not all the data was transferred, IOFINISH simulates an I/O error by setting the result descriptor error field to 4'D'.

P1126 MCP - RESTART VS. MISSING DISK, PACK FAMILY

Restart does not try to change checkpoint files on a missing DISK or PACK family.

Previously, this required an OF for the missing family. Checkpoint and restart still require a family named DISK or a family named PACK.

If both families were used during the checkpoint, the operator must ensure that they are both present before the rerun; otherwise, the restart will fail.

P1132 MCP - "SUSPENDER/WSSHERRIFF" SUSPENSION IMPROVEMENTS

On the Mark 31 system software release, the WSSHERRIFF and SUSPENDER could choose frozen libraries to be suspended. On Mark 32, they could also choose new style DMSII data bases for suspension.

When a stack chosen to be suspended was not in a state where it could be STED BY SYSTEM, it was possible for WSSHERRIFF or SUSPENDER to continue issuing repeated suspension requests. This prevented them from choosing any other stack for suspension and effectively prevented them from suspending anybody. It could also cause a very gradual buildup of save core.

These problems have been corrected.

P1133 MCP - VOLUME LIBRARY VS. REWIND

A problem with the volume library has been corrected. Formerly, if a cataloged tape file were created, the tape closed with a REWIND statement, the file title changed and a new file written, the volume library was not updated to reflect the title change. This discrepancy has been corrected.

P1134 MCP - CHECKPOINT/RESTART

Checkpoint now controls the visibility of the job stack. A new abort result (29) is given when the job stack is not visible. A warning is also displayed any time the checkpoint is possible, but cannot be assured for another execution of the job, due to the subsystem specifications. Checkpoint is also able to handle jobs or tasks or both under SWAPPER, as soon as the visibility is assured. The use of VISIBILITY=GLOBAL for the job, or the specification of a subsystem with only one box, will always permit checkpoints.

All jobs are restarted from a checkpoint in global and all tasks with their original subsystem name. None of them are restarted under SWAPPER.

A semidependent task may take a checkpoint and be restarted, when it follows the visibility rules.

P1135 MCP - RESTART ABORT, "PROCESSKILL, PIB" HANDLING

The handling of PIBs in restart has been changed. This will allow the restart to be aborted at some critical points without system dumps. Also, a problem, caused by an uninitialized PIB left in memory by JOBSTARTER, has been corrected.

P1151 MCP - "SYSTEMSTATUS" TYPE "4" PROCESSOR BUZZ LOOP

Under infrequent circumstances, a SYSTEMSTATUS type 4 specific unit request could cause the processor to go into a hard loop buzzing UNITABLOCK. The problem has been corrected.

P1172 MCP - REMOVE "BCL" POINTERS

Tapesearch no longer uses BCL pointers.

P1174 MCP - PROGRAMDUMP ANALYZES INTERRUPTS CORRECTLY

Programdump now correctly analyzes hardware interrupt literals on the following systems: B5900, B6700, B6800, B6900, B7700, B7800.

P1177 MCP - "PTD" RELEASEHEADER CALL

PTD no longer enters a fault loop when a valid, but non-baddisk, file title is entered in the baddisk test cases.

P1178 MCP - CHANGE SUPPORT LIBRARIES

When changing support libraries, the old support library would remain in use if it had any users linked to it. Furthermore, any new tasks linking to that support library would also use the old one. Now, the old support library remains frozen only while existing users are linked to it; any new users will link to the new library.

P1189 MCP - "STARTUNIT" READIES UNITS CLEANLY

Various problems existed on B6900 systems with readying units through the ODT RY message. PHYSICALIO, when called to determine the physical ready status of the unit, was basing its determination on a table that was not always current at the instant of the invocation.

These problems have been corrected by having PHYSICALIO always issue test ops to those units for which a test op is meaningful and reporting the result of the operations. Those units that do not support test ops (e.g., NSP DLP) are always considered to be physically ready.

P1205 MCP - "SHARED BY ALL" LIBRARIES FORCED TO GLOBAL

Libraries that had "SHARING=SHARED BY ALL" could be located in local memory on Multiprocessor systems if the task that caused the library to be initiated had its subsystem attribute set. This problem has been corrected.

P1207 MCP - BACKUP "MCP" DELETED

The backup MCP was deleted at the first I/O error instead of at approximately the 804th error. Also, a retry was not attempted; instead, an error message was displayed on the ODT. This problem may have caused strange memory dumps for installations using backup MCPs. It has been corrected.

P1208 MCP - "TRAINID" VS. SUBSET

Print files that requested a particular TRAINID were being incorrectly printed if the correct train was not on-line. This problem has been corrected.

P1246 MCP - "HDRVECTORLOCK"

A NOT READY I/O error while updating a disk pack directory could "hang" the entire system. This problem has been corrected.

P1247 MCP - "NEWP" VS. "RLTABLEGEN"

SYSTEM/RLTABLEGEN now works properly with the NEWP MCP.

P1248 MCP - "ATTGRAB" FAULT

Faults in ATTGRAB (which retrieves file attributes for user programs) no longer cause FAULT IN DO CODE memory dumps.

P1260 MCP - "LIBUSEMCPLOCK" VS. "PROCESSCHANGELOCK"

LIBUSEMAPLOCK must be procured before getting PROCESSCHANGELOCK. FIXALLDESC of INTRINSICHANDLER got the lock in the wrong order. This problem has been corrected.

P1262 MCP - "ALIEN I/O"

On a B6800 multiprocessor system, dumps by BAD ALIEN I/O were occasionally produced. This problem has been corrected.

P1264 MCP - "GETSTATUS INVALID OP"

An INVALID OP no longer occurs with GETSTATUS mix request Case #11 (GSPROGUSERLNK) when YOURNAME=0.

P1265 MCP - "ODT" MARKED AS "NOTREADY"

At times, the CONTROLLER would not get its responses to input printed at the ODT. This problem has been corrected.

P1267 MCP - "COBOL BCL" USER LABELS VS. "B5900"

On new systems, beginning with the B5900, if the first use of BCL by a COBOL program is in a user-supplied routine being invoked by either PREADUSERSLABELS or PWRITEUSERSLABELS, the resulting INVALID OP (which previously caused a dump) is trapped by a fault declaration in either PREADUSERSLABELS or PWRITEUSERSLABELS.

P1268 MCP - USER DEFINE "BCL" TAPE LABELS VS. "B5900"

On new systems, beginning with the B5900, READALABEL skips any user-defined tape label formats which use BCL data.

P1286 MCP - BAD GRAPH EDGE

Programs that use data bases were sometimes getting dumps by "NON EMPTY GRAPH2" when they went to EOJ abnormally. This problem has been corrected.

P1288 MCP - "COBOL" PRINT FILES NOT RELEASED

On Mark 32, when a COBOL program closed a backup file, that file was not being released correctly; i.e., the file was left as "in use" even after the program went to EOJ. This problem has been corrected.

P1290 MCP - BOX NUMBER FOR "DBS" STACKS

If GETSTATUS were called with TYPE=0, SUBCLASS=1, and MASK=1[22:1], no box number was returned for data base stacks. This problem has been corrected.

P1296 MCP - BAD RESTART AFTER "ENDF"

In table edit mode, the operator following ENDF was not being executed if the ENDF occurred on a paged array boundary and the float flip-flop was set. This problem has been corrected.

P1317 MCP - USER VS. "MCP" FAULT

On previous releases, special code had been added to the diagnostics MCP to prevent user fault handling code from being entered as a result of an MCP fault. It is no longer feasible to add this special code. Now, a non-fatal MCP fault may sometimes be trapped by a user fault statement.

P1318 MCP - "GETSTATUS LINKF" CORRECTION

When doing a directory call (TYPE=3) in GETSTATUS, if RETURNFULLNAME is specified, LINKF should point to the last file name for which the requested information was completely returned; however, on the last file returned from the directory, the LINKF field was incorrect. This problem has been corrected.

P1319 MCP - LIBRARY PARAMETER TYPE

If a parameter in the template matches a parameter in the directory, except that the template's parameter type is less than the directory's, a parameter mismatch message would sometimes be displayed. This problem has been corrected.

P1324 MCP - SYSTEM HANG

The MCP could hang if the operator did an OT on a swapped-out task. This problem has been corrected.

P1363 MCP - DIRECT RESIZE SEARCH

A problem in resizing direct arrays has been corrected. Possible symptoms included a control-state loop and dump by RESIZE FAULT.

P1368 MCP - "SWAPPER" DEADLOCK

A deadlock condition involving SWAPPER and PACKETBUILDERLOCK has been corrected.

P1383 MCP - "STACK OVERFLOW" IN "GETAROW"

A stack overflow from GETAROW has been corrected.

P3199 MCP - MOVE INTRINSIC "FAULTRECOVERY"

The procedure FAULTRECOVERY (used by FORTRAN) has been moved from the intrinsics into the MCP, because this procedure is very closely tied to the ways in which the system runs and is better suited to be in the MCP.

P3233 MCP - "SWAPPER" AND DATA MANAGEMENT

Data management tasks running in swapspace that attempt to initiate the abort procedure, or swapjobs that attempt to initiate offspring with parameters, no longer cause a system hang if the initiation fails. In addition, the data management abort procedure will always be run in main memory to ensure that it runs.

P3234 MCP - DUMP BY "BAD MOM SEARCH"

An error in the Mark 32 MCP, that caused fatal dumps by BAD MOM SEARCH when a segment of RESIDENT ("sticky") memory was moved, has been corrected.

P3235 MCP - TERMINATING SCHEDULED LIBRARIES

When trying to change a permanent library to a temporary one (via the SL- ODT message), if the library were scheduled, an INVALID OP occurred because the library directory was not yet set up. This problem has been corrected.

P3241 MCP - INHERITING "SUBSYSTEMID"

Libraries which did not have compiled-in subsystem specifications were inheriting the SUBSYSTEMID of their initiators, thus causing SHAREDYALL libraries to be put into local. This problem has been corrected.

P3253 MCP - "FIBSTACK" BINARY CARD "I/O"

FIBSTACK was building a BCL buffer descriptor for binary card reads; this action caused an INVALID IOCB WORD MLIP error on MLIP systems. Now, an EBCDIC buffer descriptor is built.

P3275 MCP - STACK SEARCHING ERROR WITH "SWAPPER"

An error in the stack-searching mechanism that caused dumps by "BAD PACKET" and/or "BAD BLOCK COUNT" on B6800 Multiprocessor systems with SWAPPER running has been corrected.

P3319 MCP - "CORE-TO-CORE" REIMPLEMENTED

The Mark 32 stack-searching code had inadvertently deimplemented core-to-core overlaying on single-processor systems and in local boxes of multiprocessor systems. Core-to-core overlaying has been reimplemented.

P3320 MCP - ILLEGAL SWAPSTATE WHILE IN "READYQ"

A problem in the process switching code for swapjobs could cause the diagnostic dump "BAD MVSTK GEORGE".

The non-diagnostic symptom was that the swapjob was left in the READYQ while in a strange swapstate (e.g., on disk).

This problem has been corrected.

P3334 MCP - SETTING THE "SWAPPER" PARAMETER "MAXIOSIZE"

The MCP no longer causes a dump by "CONTROLLER" when setting the MAXIOSIZE parameter when SWAPPER is not running.

P3335 MCP - "ACTIVETIME"

The MCP compile-time option ACTIVETIME now works correctly.

P3340 MCP - UNIT ERROR STATISTICS ALWAYS ZERO

The peripheral unit error statistics reported by SYSTEMSTATUS and in maintenance log entries were always zero. In addition to the incorrect reporting, the reliability factor (RF ODT message) was always 100%. This problem has been corrected; the unit error statistics are now updated properly when an error occurs.

P3345 MCP - RELEASEHEADER

RELEASEHEADER was not returning 0 when the call was successful, resulting in soft releaseheader errors when closing IC files. This problem has been corrected.

P3448 MCP - CHECKPOINT/RESTART FOR PROGRAMS USING STRINGS

The string pool of a program is now saved in the CP file and correctly restored.

P3687 MCP - "BDNAME" "SEG ARRAY" FAULT

A problem has been corrected where setting BDNAME to an identifier of 2 characters or less without a usercode caused a dump.

P3689 MCP - INCORRECT COMPILATION DATE

The MCP was incorrectly writing the timestamp instead of the compilation date/time to the log. This problem has been corrected.

As a result of this change, some pre-Mark 32 codefiles (all FORTRAN and some COBOL) will not have compile times listed in the job summary.

P3710 MCP - "DMSCLOSE" VS. "CONTROLLER"

When a data base stack went away, the CONTROLLER was not notified of the current in-use count of the data base.

This would cause the CONTROLLER to fail to release the data base stack number after the data base had gone to EOJ. The next task using that stack number would appear in the active entry list but would have the name of the old data base.

This problem has been corrected.

P3786 MCP - "STANDARDTODISPLAY" DECODING

A problem, where STANDARDTODISPLAY omitted the trailing right parenthesis when converting "(<usercode>) ON <pack>" stored in standard form, has been corrected.

P3818 MCP - ALLOW "DCSYSTEMTABLES(6)" WITHOUT DATACOM

When DCSYSTEMSTABLES was called from CANDE with Case 6 (return text of DCWRITE error), it would return an error if datacom were inactive. This has been changed so that it will work whether or not datacom is active. Thus, CANDE will no longer get a Fault 14 attempting to analyze a DCWRITE error #87 (uninitialized DCP) caused by datacom having terminated.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

MESSAGE LEVEL INTERFACE PORT

P1060 MLIP - "LOADHOST" DIGIT COUNT

1. LOADHOST reports the correct digit count when the LOADHOST operation fails with a DLP/disk pack controller interface timeout (206/207 disk packs).
2. After a successful HOSTLOAD on 206/207/235 disk pack controllers, if a READUNITSTATUS operation results in a releasecode of 0000 or an unrecognizable unitidcode, an error is reported.

In both cases, the path is marked OFFLINE.

P1111 MLIP - "COBOL, ALGOL" TAPE "SORT"

The following problems have been corrected:

1. SORT was calling INITIATEDIRECTWORDIO to do backward tape I/Os; however, the MLIP does not allow backward word-mode I/O. SORT now uses INITIATEDIRECTCHARIO for its backward I/Os.
2. Physical I/O was returning "short block" in cases where there was not a "real" short block condition. Physical I/O now only returns "short block" when the DLP reports "short block" and the amount of data transferred is not equal to the I/O length requested.

P1153 MLIP - CHECKPOINT VS. "B6900 I/O"

Previously, checkpoint initiated an I/O save the overlay information dope vector; this I/O was invalid on a B6900 system. This problem has been corrected by using an intermediate buffer.

P1159 MLIP - LOG "DLP" ERRORS

LOGDLPERRORS of peripheral configurator no longer takes a dump if called when DIAGNOSTICS is reset; instead, the procedure becomes a no-op.

P1160 MLIP - EXCHANGEABLE UNITS

GETUNITINFO now has a new case, UNITXCHANGEABLEV, in addition to the case UNITXFERABLEV. On multiplexor systems, these cases are the same. On B6900 systems, UNITXFERABLEV is always true (all units are capable of being freed/acquired and all units have OL information); UNITXCHANGEABLEV returns true for all exchangeable unit (i.e., disk and tape).

P1173 MLIP - "RY" A SAVED PACK RESPONSE

Whenever a saved and labelled pack was readied through the ODT ("RY PK.."), the system would reset the saved bit but would then report "UNIT IN USE".

Now the system notices that the pack is labelled and reports that the unit is ready.

P1175 MLIP - FIRMWARE LEVEL RECORDED ON DUMP TAPE

On B5900 and B6900 systems, the system type and MLIP firmware level reported by the "READ MLIP STATUS" operation is stored on memory dump tapes. The MLIP firmware level is printed out as part of the dump heading when the dump tape is analyzed.

P1212 MLIP - CORRECT "ODT" "UR"

When reserving an ODT on a MLIP system (e.g., "UR SCI"), a dump was being taken by KEYIN. This problem has been corrected.

P1214 MLIP - SYSTEM INITIALIZATION

A problem existed on MLIP systems during system initialization such that a fault in D0 code could occur on some illegal configurations. This problem has been corrected.

P1215 MLIP - CARD PUNCHING PROBLEM

A problem which existed on MLIP systems involving punching cards has been corrected.

P1240 MLIP - IMPLEMENTATION OF READ EXTENDED STATUS

A "read extended status" operation is done on MLIP systems for those DLPs which require a "read extended status" to unlock a control (currently, just the HT).

LOGANALYZER will print out the extended status if required. In the case where the extended status itself was not obtainable, the result descriptor from the "read extended status" is printed.

In order to retrieve the correct extended status information, the latest level of DLP microcode is required.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

Installations should ensure that their HT-DLP hardware is up to date.

Performance of older DLPs is not impaired; however, spurious extended status results may occur.

P1263 MLIP - READYING OF PACKS

When Halt/Loading on MLIP systems with only one path to packs, only the Halt/Load unit would be READYed. This problem has been corrected; now, all packs are READYed after a Halt/Load.

P1269 MLIP - "CANCELIO" CORRECTIONS

The mechanism for cancelling I/Os on MLIP systems could cause occasional faults and dumps, usually when clearing units. These problems have been corrected.

P1270 MLIP - "BLASTUNIT" IMPROVEMENTS

The procedure BLASTUNIT (used for clearing units) has been extensively modified to improve its ability to handle hung I/Os on MLIP systems.

P1272 MLIP - "PATHRES" VS. NONSTATUS "DLP"

When entering UR- for a path (DLP) that does not understand TEST/WAIT operations (e.g., datacom NSP and LSP DLPs), a dump would be taken. This problem has been corrected.

P1276 MLIP - CORRECT RESERVE HALT/LOAD PATH

On MLIP systems with multiple Halt/Load unit paths, reserving the path used to get to the Halt/Load unit would result in a dump on the ensuing Halt/Load.

This problem has been corrected, as follows:

1. When the Halt/Load path is reserved, another path will be used on the next Halt/Load.
2. When UTILoader is used to Halt/Load the system, if a reserved path is either selected by default or specified by the user, that path will automatically be un-reserved at Halt/Load time.

P1295 MLIP - PACK POWER OFF

A problem which prevented a "PO PK" ODT command on an MLIP system from actually physically powering off the pack has been corrected.

P1364 MLIP - "DOIOERRORIO" RETURNS LOGICAL "R/D"

DOIOERRORIO was returning the logical result descriptor (as the value of the procedure) only if the I/O was actually initiated; recovery routines were not being notified if, for example, BLASTUNIT had run on the unit as the result of its having been cleared. This problem has been corrected.

P1366 MLIP - TAKEUNIT VS. "FREE PK<NN>"

A problem in SETSTATUS in which "SV PK <NN>, FREE PK <nn>" would cause a dump has been corrected.

P1369 MLIP - "WFL/CONTROLCARD" LOOP

Making the card reader not ready just after it had been made ready and staged the first card caused WFL to go into a loop. This problem has been corrected.

P3203 MLIP - LOGICAL UNIT NUMBER VS. "PB MT<NN>"

If "PB MT<nn>" were entered on a B6900, an incorrect <nn> was displayed on the ODT. This problem has been corrected.

P3209 MLIP - "UR/UA" MAINTENANCE CAPABILITIES

Previously, it was not possible to place a 206/207 pack into maintenance mode. Now, this can be done.

P3237 MLIP - "TEST/WAIT" BAD EVENT REFERENCE

On B6900 systems, after failing to cancel a TEST/WAIT operation, an invalid event reference was left in the TEST/WAIT IOCB. If the IOCB ever completed, a dump occurred as a result of attempting to use the invalid event reference. This problem has been corrected.

P3238 MLIP - "BLASTUNIT, MYIOSONLY"

On B6900 systems, BLASTUNIT now clears "MYIOSONLY" when called in the "BLEOFV" case. This avoids a possible dump by the procedure trying to cancel a TEST/WAIT operation because BLASTUNIT cancelled the CANCEL or DISCONTINUE operation.

P3239 MLIP - SELECTIVE CLEAR VS. HUNG "DLP"

The following problem has been corrected: A hung DLP was encountered during initialization, causing an error IOCB to complete.

P3240 MLIP - "UA SC <UNIT NO>" CORRECTED

A fatal dump could occur if an ODT were unreserved with its path off-line. This problem has been corrected.

P3248 MLIP - CORRECT "HOSTNSP" RESULTS

A problem existed which could cause the HOSTNSP to report "unsuccessful I/O" errors after the system had taken a memory dump when there had been terminal activity while the dump was in process. This problem has been corrected.

P3254 MLIP - "BINARY EOF" SENSING

The BINARY end-of-file card is now sensed properly.

P3341 MLIP - IOERRORTYPE VS. "EOT/EOF"

On MLIP systems, the direct file attribute IOERRORTYPE no longer returns EOT rather than EOF when reading tape files if the tapemark causing the end-of-file condition is beyond end-of-tape.

P3342 MLIP - "BLASTUNIT" AVOIDS OUTBOARD PATHS

On MLIP systems, BLASTUNIT would fault when called for a unit (LSP) which had "outboard" paths; i.e., paths from an NSP rather than directly from the host system. This would cause a dump if the LSP were cleared from the ODT, or the HOSTNSP library terminated abnormally. This problem has been corrected.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

DOCUMENT CHANGES NOTES (D NOTES)

NETWORK DEFINITION LANGUAGE

D4100 NDL - "NDL" COMPILER USER OPTIONS

Page D-8 of the NDL Reference Manual (Form No. 5001522) states that "a word on an option control card, which is not a standard option, is recognized as a user-defined option". The manual does not state, but should, that only the first five characters and the length of user-defined options are significant. Thus, the two options, \$SET ABCDEX and \$RESET ABCDEY, are treated as two references to the same option, while \$POP ABCDEXY would be considered a separate user option.

D4194 NDL - FREE POUND SIGN

On Page 5-63 of the NDL Reference Manual (Form No. 5001522), the third paragraph should be amended to read as follows:

"A free pound sign is any pound sign not in a <string> or <remark>. Unlike ALGOL, NDL <define definitions> may not be nested in any manner whatsoever."

The fifth paragraph should be amended to read as follows:

"<define definitions> may reference other <define definitions> in any mutually non-recursive manner."

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

NETWORK DEFINITION LANGUAGE

P1003 NDL - ERROR FOR DIFFERENT LINE CONTROLS

When two stations on the same line used different terminals, and these terminals used different line controls, the compiler would use only one of the line controls and not generate a syntax error. This problem has been corrected.

P1254 NDL - RECURSIVE DEFINES

Recursive defines in SOURCENDL would cause the NDL compiler to terminate with a fault. This problem has been corrected.

P1255 NDL - ALLOW STRING PARAMETERS TO DEFINES

The NDL compiler would get a compiler error if a string were used as an actual parameter to a define. This problem has been corrected.

P1307 NDL - SETTING OF USER OPTIONS

The NDL compiler, when handling a dollar card of the form "\$ SET <name> = <dollar exp>", would immediately perform \$SET <name>. Then, it would SET <name> to the value of <dollar exp>. This sequence worked correctly except in the case where <name> appeared in <dollar exp>. In that case, erroneous results could occur. Such situations are now handled properly.

P1332 NDL - ALPHA LABELS

After the Mark 30 PR1 release, alpha labels could no longer be used as actual parameters to defines if the label was to be defined inside the body of the define. This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

NDLII

P1019 NDLII - DETECT STRING SIZE TOO LARGE ERROR

When a string was declared larger than 255 characters, the declaration was not being detected as a syntax error; instead, the requested size modulo 256 was taken as the length.

A syntax error will now be generated.

P1020 NDLII - FIX FAULT WITH USE OF SPECIAL STRING PRIMARY

Use of STATION.RESULT.TEXT, STATION.REQUEST.TEXT, or <cb-id>.TEXT as a <special string primary> would result in an INVALID ADDRESS fault.

This problem has been corrected.

Also, string expressions were not being allowed in many places where they should be allowed. String expressions are now allowed as parameters to TAKE, DROP, STRINGADD, STRINGSUBTRACT, INTEGER, HEAD, TAIL, TRIM, SUBSTRING, and TIMEINTERVAL; and in the statements TRANSFER, COPYSTRING, STORE, TRANSMIT; and in conditional expressions.

P1158 NDLII - SYNCHRONOUS LINE DIAGNOSTIC RATE

The NDLII compiler was not emitting information to the NIF file to handle the diagnostic rate specification for a line class declaration of type SYNC. The problem has been corrected.

P1235 NDLII - LINE ATTRIBUTE OFFSET

Line attributes are now entered in a LINEDEC record of the NIF starting at the proper offset. Previously, the last byte of the fixed portion of a LINEDEC record was overwritten by user attributes.

P1278 NDLII - EVENTS, INTERLOCKS IN INCLUDE LISTS

In LINE and STATION include lists, the NDLII compiler was allowing events and interlocks to be declared in external declarations but not in non-external parts of INCLUDE declarations. Event and interlock declarations are now allowed in non-external parts but not in external portions.

P1279 NDLII - FAULT IN GROUP "INCLUDE" DECLARATION

Declaring a group INCLUDE list at the global level could cause the compiler to abort with the message "NO ALTERNATIVE--GETINDEX OF INFO". The problem has been corrected.

P1280 NDLII - "INVALID DESCRIPTOR ADDRESS (UTILACTUAL)"

If a syntax error were detected in the protocol module, the NDLII compiler could abort with the message "INVALID DESCRIPTOR ADDRESS (UTILACTUAL)" while compiling the configuration module. The problem has been corrected.

P1281 NDLII - INVALID EXTENSION SKELETONS

The NDLII compiler could produce invalid station and group extension skeletons when more than one algorithm was declared in the protocol module. The problem has been corrected.

P1282 NDLII - "WAITFORIDLE" IMPLEMENTED

The adaptor control function WAITFORIDLE has been implemented.

P1347 NDLII - ALGORITHM TRANSLATE TABLE LIST

The translate table list of the ALGORITHMDEC record in the NIFII was not being created. This problem has been corrected.

P3222 NDLII - DUPLICATE ADAPTOR ASSIGNMENT

If two adaptors with the same number were specified for an LSP, no syntax error was given. A "DUPLICATE FR" fault would occur at NSP initialization. This problem has been corrected.

P3223 NDLII - "SYSTEMWAIT" CODE

The line control pre-declared variable SYSTEMWAIT no longer generates the wrong code.

P3256 NDLII - CONTROL BLOCK ARRAYS

If CB were declared to be a CONTROL BLOCK ARRAY in an algorithm, the following references generated incorrect code:

```
CB[I].TEXT,
CB[I].<predeclared variable>
CB[I].<reply variable>
```

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

These problems have been corrected.

P3285 NDIII - "IDSTORE" SIZE INCREASED

The size of the IDSTORE array has been increased to 16K words to avoid INVALID INDEX faults while compiling very large NDIII source programs.

P3310 NDIII - COPYSTRING WITH "<CB VARIABLE>.TEXT"

If a COPYSTRING statement used <cb-variable>.TEXT as the <string-exp> to be assigned to the <cb-variable>, an INVALID OP would occur. This problem has been corrected.

P3311 NDIII - TRANSMIT STATEMENT

The compiler was generating incorrect code if the <forced-DLE> option were specified in a TRANSMIT statement. The problem has been corrected.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

DOCUMENT CHANGES NOTES (D NOTES)

NEWP

D4115 NEWP - SYNTAX DELETED

The following syntax has been deleted from NEWP:

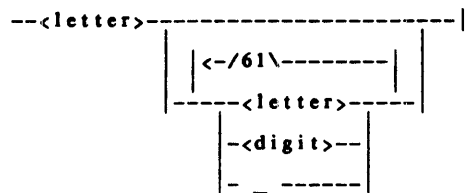
<procedure id> ... 0 [<expression>]

D4166 NEWP - ALLOW UNDERScores IN IDENTIFIERS

The definition of identifiers has been changed to include underscore ("_") characters.

The syntax diagram for an <identifier> is now the following:

<identifier>



Underscores are treated as significant characters in an identifier.

INTERACTIVEXREF has been changed to allow it to reference these new identifiers.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

NEWP

P1022 NEWP - INVALID INDEX FOR "CASE" EXPRESSION

NEWP will now generate code which will cause an invalid index when the index of a CASE expression is out of range. Previously, an incorrect branch was generated.

P1023 NEWP - "XREF" OF PROCEDURES IN "MODULE" EXPORT LIST

NEWP was not entering a procedure id in a MODULE export list into the XREF Refs file. This problem has been corrected.

P1106 NEWP - STRING PARAMETERS TO "STOP"

String parameters to the unsafe intrinsic STOP are now flagged as errors.

P1167 NEWP - MAKE SCALERIGHTS RETURN AN INTEGER

The quotient returned by the SCALERIGHTS function is now an integer.

DOCUMENT CHANGES NOTES (D NOTES)

NSP DUMP ANALYZER

D4175 NSPDUMPANALY - RECOGNIZE "MARK 7 NSP" FIRMWARE

The NSPDUMPANALYZER now recognizes and analyzes NSP dumps created under the Mark 7 NSP firmware. The NSPDUMPANALYZER still properly analyzes dumps created under pre-Mark 7 NSP firmware.

D4176 NSPDUMPANALY - ADD FAULT TOLERANCE

"Fault tolerance" has been added to the NSPDUMPANALYZER. When a fault occurs, a program dump is taken using the options supplied in the option control card, and further analysis of the current structure type is stopped. It will then attempt to continue analyzing the remaining structure types.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

NSP DUMP ANALYZER

P1204 NSPDUMPANALY - ELIMINATE FIRMWARE CODE

The NSPDUMPANALYZER no longer includes the NSP firmware area in the analyzed dump when analyzing a pre-Mark 7 NSP dump, thus decreasing the amount of output produced.

P3251 NSPDUMPANALY - DIFFERENT LEVEL FIRMWARE HANDLING

The NSPDUMPANALYZER now correctly handles all levels of NSP firmware, including level 562.

In addition, memory usage statistics are now reported in bytes rather than in granules.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

PATCH

P1139 PATCH - NO "BCL" ON "EBCDIC" SYSTEM

Creation of a BCL NEWTAPE on an B5900 system has been disallowed.

DOCUMENT CHANGES NOTES (D NOTES)

PATCHCONTROLWARE

D3128 PATCHCW - INITIAL RELEASE OF "PATCHCONTROLWARE"

SYSTEM/PATCHCONTROLWARE is a utility to patch Controlware files. It may be initiated from an ODT or remote terminal. The internal file name CONTROLWARE may be label-equated to the Controlware file to be patched. If the Controlware file is not found, the utility will request the file name to be entered.

Patches are entered, one at a time, using the following format:

```
CC PATCH [/DISP] [/PREV] <FILENAME> <SEGMENT> <ADDRESS-1>
      <ADDRESS-2> <LENGTH> [UA] [UN] <DATA>
```

where:

```
[...] denotes optional input
/DISP causes <DATA> to be displayed/printed
/PREV causes patched data to be displayed/printed
<FILENAME> ::= name of file to be patched
<SEGMENT> ::= zero relative segment address
<ADDRESS-1> ::= zero relative memory address
<ADDRESS-2> ::= zero relative digit address in segment
<LENGTH> ::= length of patch (maximum value of 30)
UA ::= alpha data follows
UN ::= numeric/hex data follows
<DATA> ::= the patch itself
<SEGMENT>, <ADDRESS>, and <LENGTH> are in decimal
```

Enter STOP to terminate the program.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

PLI

P3224 PLI - INCORRECT SYNTAX ERROR

A structured item defined to overlay another item sometimes caused an incorrect Level-1 syntax error: "LENGTH, PRECISION OR INITIALIZATION OF GROUP ITEM IGNORED". This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

PLISUPPORT

P1103 PLIS'PP - ISAM, "END-OF-FILE"

On the Mark 32 system software release, ISREADNEXT failed to return the correct results to COBOL programs for END-OF-FILE. This error caused programs which sequentially read an ISAM file to loop on the last record. This problem has been corrected.

P1109 PLISUPP - "ISAM DELETE," LARGE "FINE, COARSE" TABLES

If the length of an ISAM file's FINE or COARSE tables was greater than or equal to 2**16 bytes, record keys were not properly deleted from the tables by DELETE operations. This problem has been corrected.

P1233 PLISUPP - "GET LIST, GET DATA"

On the Mark 32 release, GET LIST operations required a double quote (") as the opening delimiter of character string items. GET DATA operations required double quotes for both opening and closing delimiters of character string items. Now, single quotes (') are required for these delimiters, as on previous releases.

P1256 PLISUPP - BAD PARAMETERS TO "DELTA"

If an attribute error were encountered, the PL/I LIBRARY would pass a bad parameter to DELTA, causing the program to be DSed. This problem has been corrected.

P1304 PLISUPP - "INVALID INDEX" IN CONDITION INFO TABLE

An INVALID INDEX was being generated referencing the 4096th unhandled condition in a PL/I program. This error is a downstream problem of not completing the handling of certain conditions. Condition handling in those cases is now done correctly.

P1305 PLISUPP - "PUT EDIT," BIT FORMATS

An error in the PLISUPPORT library caused it to sometimes get an INVALID OP when handling bit formats for PUT EDIT statements. This problem has been corrected.

P1330 PLISUPP - "ISAM" HARDWARE ERRORS

The ISAM procedures of the PLISUPPORT library sometimes returned values which erroneously indicated an end-of-file because of hardware errors. Now, such hardware errors no longer appear in returned values.

P3225 PLISUPP - "ISAM, DELETE"

Deleting records from an ISAM file in a COBOL program with "\$SET ANSI74" sometimes caused the structure of the file to become corrupted. This problem has been corrected.

P3226 PLISUPP - "ISAM" , "COBOL START" STATEMENT

COBOL statements of the following form caused an INVALID OP in PLI SUPPORT when the key K existed in the file F:

```
START F KEY > K INVALID . . .
```

Now, the statement executes correctly.

P3259 PLISUPP - "ISAM," PREMATURE END OF FILE

On the Mark 32 release, an end-of-file condition incorrectly occurred when records were added to the file-overflow area of an ISAM file. This problem has been corrected.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

DOCUMENT CHANGES NOTES (D NOTES)

STANDALONE PRINTER DUMP (SAD)

D4246 PRINTERDUMP - "PRINTERDUMP" IMPLEMENTATION

SYSTEM/PRINTERDUMP is a free-standing program, used to produce a raw hex dump of main memory. Before control is transferred to SYSTEM/PRINTERDUMP, the operator must note the contents of several critical registers, in order to help with later analysis. The registers to be recorded are the following: BOSR, LOSR, S, F, D[0], D[11], PBR, PIR or PWI, PSR or PSI, PDR or SDI, and SNR.

Once the contents of these registers have been recorded, the operator should load SYSTEM/PRINTERDUMP as described in the documents for the particular system. B5900 users should see the B5900 System Reference Manual (Form No. 5011034). B6900 users should follow the documented procedure for loading UTILoader from the MDP, but type "T PRINTERDUMP" instead of "T UTILoader".

The symbolic file for this program is SYMBOL/PRINTERDUMP.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

PTDMCP

P1336 PTDMCP - "PTD" "SPO" HANDLING

The following problems have been corrected:

1. PTD, when requesting the device for the test, was displaying residual text from the previous display.
2. PTD was not clearing the command line before rolling the screen, so that if two ETXs had been on the command line when the command was transmitted, on some terminals the entire line would appear at the bottom of the display, while on others, only the text preceding the first ETX would appear.

P1339 PTDMCP - "PTD" PATH DEASSIGNMENT, "I/O" DEBUG OPTIONS

The following problems have been corrected:

1. With some test cases having multiple DEVICESTRUCTURES, PTD would sometimes go to EOJ leaving the path assigned to it.
2. The DEVICESTRUCTURE debugging options associated with opening and closing DEVICESTRUCTURES (*BOPEN, *AOPEN, etc.) had been broken previously such that a segmented array error would occur when any of these options were set.

DOCUMENT CHANGES NOTES (D NOTES)

PERIPHERAL TEST DRIVER (PTD)

D4197 PTDTST - IMPROVED "PTD" PATH SELECTION

The Peripheral Test Driver, as documented in the Mark 32 system notes, has been revised. The complete documentation for PTD follows; PCN marks in the right margin indicate changes since the Mark 32 system software release.

1. PTD EXECUTION

On B6900 and B5900 (referred to as "MLIP") systems, the Peripheral Test Driver is a selectable module whose equivalent on the B6700 and B6800 is SCR. "PTD" is recognized by the WFL compiler, running on an MLIP system, as the command to start SYSTEM/MAINTENANCE as an independent runner. "PTD" is not recognized by old (pre-Mark 29) WFL, which means that either option 29 (CONTROLOLDWFL) must be set or the request to initiate PTD must be preceded by "BEGIN JOB;".

PTD is an MCP procedure that interprets op-codes ("S-ops") found in the test case S-code file selected by the user. Thus, to initiate a test on a particular type of peripheral/DLP, the user must label-equate the appropriate Burroughs-provided test case code file to PTD's internal SCODE file.

In the Large Systems implementation, PTD can be operated from either an ODT or a datacom terminal. In either case, all PTD input and displays are through a logical file titled PTDSPO; the use of the "<mix #>AX" construct is eliminated. With the ODT, however, every message input to PTD must be preceded in column one by the triangular group-separator character (GS) so that the entry can be directed to PTD instead of to the CONTROLLER. To direct the operator dialogue to a datacom station, the user need only change the KIND of PTDSPO to REMOTE and the TITLE of PTDSPO to the station name; the group-separator is not used with this medium.

PTD can be executed with a VALUE clause that sets certain of its options immediately at run time. This feature is useful primarily in debugging test cases or PTD itself. The relevant bits are:

[2:1]	TRACE	(test case S-ops in execution)
[1:1]	DUMP	(test case virtual memory dump)
[0:1]	PGMDUMP	(program dump of PTD)

Hence, executing PTD with VALUE = 2 causes the DUMP option to be set, VALUE = 3 sets both DUMP and PGMDUMP, etc.

GENERAL EXECUTION SYNTAX:

In general, the Large Systems PTD is executed by entering a command of the form:

PTD; FILE SCODE(TITLE = <S-code file name>)

Other items are entered depending on the setting of option 29, the user's choice of dialogue device, and the user's (rare) decision to add the VALUE clause. Actual SCODE file names are used in the following examples. See Mark 32 PR1 PTDTST note D4240, "PTDTSTTS Tape", for a detailed description of the release tape, file naming conventions, and individual test case documentation.

ODT EXECUTION EXAMPLES:

PTD is executed at an ODT and all subsequent operator dialogue occurs at that ODT. All three forms are acceptable if CONTROLOLDWFL (option 29) is set; only the third form ("BEGIN JOB; . . .") is acceptable if CONTROLOLDWFL is reset.

PTD; FILE SCODE(TITLE = PTD/MAINT/CP)

PTD; FILE SCODE(TITLE = PTD/CONF/CR ON DMS); VALUE = 1

BEGIN JOB; PTD; FILE SCODE(TITLE = PTD/MAINT/PE); VALUE = 2

ODT INITIATION, REMOTE (DATACOM) DIALOGUE DEVICE EXAMPLES:

PTD is executed at an ODT but all further dialogue occurs at the specified datacom terminal. All three forms are acceptable if CONTROLOLDWFL (option 29) is set; only the third form ("BEGIN JOB; . . .") is acceptable if CONTROLOLDWFL is reset.

PTD; FILE SCODE(TITLE = PTD/MAINT/HTS ON DISK);
FILE PTDSPO(KIND = REMOTE, TITLE = TD450365)

PTD; FILE SCODE(TITLE = PTD/CONF/PSS); STATION = 55; VALUE = 3

BEGIN JOB; PTD; FILE SCODE(TITLE = PTD/CONF/PE); STATION = 55

REMOTE EXECUTION EXAMPLES:

PTD is executed at a logged-on CANDE terminal and all subsequent operator dialogue occurs at that terminal. Notice that "BEGIN JOB;" is required, irrespective of the setting of CONTROLOLDWFL.

WFL BEGIN JOB; PTD; FILE SCODE(TITLE = PTD/MAINT/PE);
VALUE = 1

WFL BEGIN JOB; PTD; FILE SCODE (TITLE=PTD/MAINT/PE)

WFL BEGIN JOB; PTD; FILE SCODE(TITLE = PTD/MAINT/HTS
ON TESTPACK);

2. PTD DIRECTIVES

When executed, PTD first performs some internal initialization steps that may take a few seconds to accomplish depending on the size of the test case code file and the number of other jobs in the mix. As noted elsewhere, PTD displays the testcase file name and release-level data and then enters its idle state after displaying:

AWAITING DIRECTIVE

Any valid PTD directive can now be entered, one directive per transmission. PTD recognizes any number of correctly spelled characters from the minimum (underlined>) set to the fully formed word; i.e., PRI, PRIN, and PRINT are acceptable, while PR and PRINNT are not. The directives fall into four general categories and are summarized below.

Remember that if the ODT is used, the group-separator character (GS) must appear in column one.

A. TOGGLES

These are options that are either on or off. To set a toggle, the user enters the option name only; to reset a toggle, the user enters the option name followed by "O", "OF", or "OFF". Hence, "PRI" sets the PRINT toggle; "PRI O" or "PRINT OFF" resets it.

DISPLAY - Messages are displayed to the ODT or remote terminal.
 --- When the medium is an ODT, resetting DISPLAY causes PTD to close its PTDSPO file so that normal system display traffic may resume.

Default: DISPLAY IS SET.

DUMP - Prior to EOJ, the test case's stack and data memory are dumped by PTD to a printer file. Note: If any fatal execution error occurs, the data is dumped unconditionally by PTD; the output should be submitted when reporting any such test case or PTD problems.

Default: DUMP is RESET.

IOTRACE - I/O information is displayed before and after every test case I/O operation. Prior to initiating the I/O, the MLI REQUEST and IOLENGTH are shown. After the I/O has completed, PTD displays the MLI RESULT, the number of data-bytes transferred, and an "MLIP" result.

The user should be aware that whenever the words "MLIP RESULT" are used in the context of PTD or a Burroughs test case, they refer to a value constructed strictly for the purposes of a programmatic, host independent result descriptor that summarizes errors that occur within the host to MLI interface. On Large Systems, the true MLIP result descriptor can be obtained only by inspecting the test case IOCB(s) as they complete. The address of the IOCB is made available through the use of the special I/O options, described in Section 5 below.

Default: IOTRACE is RESET.

MONITOR - All display traffic is written to a printer file and is formatted such that all operator entries are flagged with an asterisk ("*") in column one. (The MONITOR and PRINT functions are to be merged into the PRINT toggle on a future release; for now, the user should set both MONITOR and PRINT.)

Default: MONITOR is RESET.

PGMDUMP - Prior to EOJ, PTD dumps itself (via PROGRAMDUMP) to a printer file. Note: If any fatal execution error occurs, PROGRAMDUMP is taken unconditionally; the output should be submitted when reporting any such test case or PTD problems.

Default: PGMDUMP is RESET.

PRINT - PTD prints test-case-specified text to a printer file
 --- when the PRNT S-op is encountered. (See the note in the
 description of MONITOR, above.)

Default: PRINT is RESET.

TRACE - PTD writes each test case S-op to a printer file after
 --- the S-op is executed. It is used for test case and PTD
 debugging. Since vast numbers of S-ops are executed in
 the usual test case, tracing them may waste storage
 space and paper.

DEFAULT: TRACE is RESET.

B. I/O ERROR CONTROL

Generally, the test case issues the I/O (via the EXIO S-op), waits for it to complete (IOCW S-op), and then checks the results. If it finds a result descriptor, data, or length error, it notifies PTD of the error via the EROP S-op.

When PTD encounters the EROP S-op, it either stops and idles or ignores it, depending on what the user has directed. By default, PTD halts on the EROP S-op.

ERROR HALT

--- - - PTD halts on any non-fatal I/O error and displays:

STOPPED ON ERROR

At this point, any valid PTD directive may be entered.
 (The GO directive, explained below, causes PTD to
 continue execution at the very next S-op.)

ERROR IGNORE

--- - - PTD treats all non-fatal test case I/O errors as
 no-ops; execution continues uninterrupted.

Note: The test case may retry an I/O as part of its testing algorithm, but no test I/O is ever automatically retried by PTD or the MCP.

C. PTD STATUS

The current status of PTD is obtained via the STATUS directive.

STATUS - PTD displays the number of the section currently in
 --- execution, the toggles that are set, and the setting
 of the ERROR option.

D. SECTION CONTROL DIRECTIVES

This class of directives controls the actual execution of the test case. As described elsewhere, test cases are organized into numbered "sections" that perform certain operations. There are groups of sections that fall into logical categories called "blocks".

For example, block 100 may consist of sections 101, 102, and 103, where those particular sections test the TESTID, the READBUFFER, and the WRITEBUFFER operations respectively. Other blocks may consist of sections designed to perform complex sequences of data transfer that drive a peripheral or DLP at a certain level of tolerance. Sections from 3000 above are "special" sections which must be explicitly invoked by the user; section 4000 lists what the other sections do and is particularly useful. Refer to the Burroughs-provided documentation for precise descriptions of the organization of each test case.

The RUN, REPEAT, and GO directives place PTD into immediate execution. When execution of the specified section(s) has been completed, PTD displays:

AWAITING DIRECTIVE

and idles until a new directive is entered.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

RUN <section list>

PTD executes those sections or ranges of sections contained in the <section list>. The section list may be empty, or the list may contain a single section number, or a range of sections, or it may contain up to ten sections and ranges separated by commas.

If only the word "RUN" is entered, PTD ALWAYS executes sections 100 to 2999. (Recall that sections from 3000 above must be deliberately specified.)

Examples: RUN
RUN 3999
RUN 200
RUN 101, 201
RUN 203 TO 207, 301-302, 4000-END

- Rules:
- The maximum number of pairs of section numbers that can occur in a <section list> is 10. (A single section number, 305 say, is treated as the pair: 305-305.)
 - In any section pair, the second number must be greater than or equal to the first. For example: "RUN 300-101" is an error.
 - The word END can appear only as the last item in a <section list> and must be preceded by "<number> -" or "<number> TO". For example: "RUN END" and "RUN 101, END" are both erroneous entries.
 - The word END is interpreted to mean "the very last section of the test case". When END appears in a RUN directive, PTD goes to end of job when execution of the final section has completed.

REPEAT <section list>

or

REPEAT <section list> FOR <count>

- The <section list> is repeated (looped on) either an infinite number of times or <count> times. As in the RUN directive, if no list follows the word REPEAT, sections 100-2999 are automatically executed.

Examples: REPEAT
REP F 3
REPEAT 102 FOR 10
REPEAT 101, 301, 102, 302
REPEAT 207-209, 101, 305 - 400

- Rules:
- The maximum allowable value of <count> is 999999999.
 - The RUN <section list> rules also apply to the REPEAT directive, with the single exception that the occurrence of END does not cause PTD to go to EOJ.

GO

- When PTD either stops on an error (see ERROR directive) or is interrupted by the user (see item 5, PTD OPERATOR INTERRUPTION), the GO directive causes PTD to resume execution from the point where it was stopped. If the user had entered, say, "RUN 304", PTD would continue running section 304. Similarly, if PTD had been REPEATING a list, GO would cause it to continue exactly where it left off. However, if PTD has finished executing a section list, it responds to "GO" by displaying:

SECTION LIST COMPLETED. USE 'RUN' OR 'REPEAT'.

Hence, "GO" always means "continue", and there has to be a function which PTD can continue.

QUIT - Terminates PTD.

3. SPECIAL LARGE SYSTEMS PTD DIRECTIVES

There are Large Systems PTD directives that are intended primarily as aids in developing and debugging PTD and test cases. They may be useful in observing the flow of test case I/Os and have certain value (see IOSTOP) in stepping an I/O through the processor.

With the exception of IOSTOP and LINES, the setting of a special option results in the analysis by PTD of the test case I/O currently in process at a selectable point in the execution of the test. Relevant DEVICESTRUCTURE information is displayed on the ODT or remote terminal. (A DEVICESTRUCTURE is the programmatic I/O interface between the test case and PTD and is analogous to a FIB, or File Information Block. Up to ten separate DEVICESTRUCTURES may be declared by the test case writer in the PTL (Peripheral Test Language) test case program; each carries all information relating to the I/O device, the I/O request, and the result of the operation.)

In every case, an asterisk must appear before the non-standard option name; they are set and reset like all other PTD flags (see item 2A, TOGGLES above).

- *HELP IODEBUG -

 - PTD displays explanation of special I/O options.
- *BOJ - Display each DEVICESTRUCTURE the first time it is
---- touched after this option has been set.
- *BOPEN - Display before each DEVICESTRUCTURE is opened.

- *AOPEN - Display after each DEVICESTRUCTURE is opened.

- *BIIO - Display before the I/O is initiated out of each
---- DEVICESTRUCTURE.
- *AIIO - Display after the I/O has completed.

- *BCLOSE - Display before each DEVICESTRUCTURE is closed.

- *ACLOSE - Display after each DEVICESTRUCTURE is closed.

- *EOJ - Display each DEVICESTRUCTURE immediately prior to EOJ.

- *IOSTOP - At each user-selected analysis point, PTD
---- programmatically stops after the DEVICESTRUCTURE
display; entering any input or null reactivates PTD.
If *BIIO has been set in addition to *IOSTOP, then
after the I/O information display and attendant
stop/start, PTD executes a conditional halt prior
to firing the I/O (CHLT must be set on the machine).
4*IOCBIOCBIOCB" is placed in the B-register and the
address of the IOCB to be fired is placed in the A-reg.
- *ALL - Sets all of the above flags.

- *IODEBUG- Setting any of the above sets this toggle. Resetting
---- this toggle disables all analysis, but the settings of
all other toggles are preserved. If IODEBUG is
subsequently set again, the flags are restored
and the analysis resumes.

In addition to the above toggles, the operator may specify the number of lines of data to be displayed in the analysis output, allowing the user to view an entire data buffer on request.

*LINES <n> - sets the number of lines of data to be displayed,
---- 40 HEX digits (10 MLI words) per line. Default
is one line.

All non-standard options are reset by default.

4. TEST DEVICES

A test case is written with either a physical device or a baddisk file as its target. The connection between the test case and test object is established by way of:

- a. A programmatic test case data structure called a DEVICESTRUCTURE,
- b. The OPEN S-op,
- c. A device mnemonic or disk file name entered by the user.

Once running, PTID eventually encounters an OPEN S-op, whereupon it displays:

ENTER DEVICE FOR <peripheral designation>

or

ENTER FILE NAME FOR <baddisk designation>

where the <peripheral designation> or the <baddisk designation> is merely a string of characters that PTID finds in the DEVICESTRUCTURE and displays on the screen. For example, in the cardreader test, the display looks like:

ENTER DEVICE FOR CARDREADER

In the case of a peripheral/DLP test, PTID expects the user to enter a Large Systems unit mnemonic and optional path specification. The general form of the response is the following:

<unit> VIA PATHID <pathid>

Examples:

CR12
PK57 VIA PATHID 17

For baddisk testing, PTID expects the title of a valid Large Systems baddisk file, including family name, of the form

<file title> ON <family name>
--

Example:

BADDISK/FMLYINX1/UNIT192/AD405000 ON PACK

UNIT RESERVATION RULES

With few exceptions, the unit being tested must have been reserved through the "UR <unit>" ODT command. The exceptions are:

- The pack scanner confidence test (the unit need not be reserved).
- The baddisk confidence test (the unit cannot be reserved).

See the PATH SELECTION AND RESERVATION RULES, below, for related restrictions.

PATH SELECTION AND RESERVATION RULES

On the Large Systems PTID, selection of a path (DLP) through which test case I/Os are channeled occurs in one of three ways:

- Explicit specification by the user (through the "VIA" keyword).
- Automatic selection by PTID.
- Dynamic selection by the MCP.

Some test cases issue operations that are clearly risky in an on-line environment unless the path has been reserved through the

"UR <unit> PATHID <pathid>"

ODT command. Other test cases, the baddisk test for example, are by nature milder in their behavior and do not require a reserved path. Furthermore, explicit path selection in the on-line environment may interfere with the MCP's path selection algorithms for other users of the unit, but if the unit is reserved (i.e., no other users) then no such problem exists. Therefore, path selection rules are enforced by PTID on the basis of the type of test being executed.

It is stressed that whenever a reserved path is selected by PTID or is specified by the user, PTID assigns itself to that path -- this

assignment will be reflected in the response to a subsequent 'OL' ODT command, where the PATHSTATUS for that path will be either "ASSIGNED RESERVED ONLINE" or "ASSIGNED RESERVED OFFLINE". As long as at least one test case DEVICESTRUCTURE is open to a unit through that path, the path will remain assigned to PTD, and no other invocation of PTD will be able to use it until the path is unassigned. However, in a given invocation of PTD, one path can be used by more than one DEVICESTRUCTURE (see SELECTION EXAMPLE 2, below).

A. PACKSCANNER (PTD/CONF/PS)

Path selection by either the operator or PTD is allowed only if the unit has been reserved.

- If the operator attempts path selection to a non-reserved unit; e.g., "PK 65 VIA PATHID 3" or "PK 65 VIA", PTD will respond:

"PATH SELECTION NOT ALLOWED UNLESS UNIT IS RESERVED."

(Note: if the user then reserves the unit, path selection will henceforth be allowed.)

- If the operator attempts path selection to a reserved unit, the specified path must be on-line and not assigned to any other stack. If the specified path is off-line, PTD will respond:

"PATH IS OFFLINE - TEST REQUIRES AN ONLINE PATH."

If the specified path has been assigned to another stack, PTD will respond:

"PATH IS IN USE."

- If no path selection is attempted and the unit is not reserved, the path will be selected by the MCP. PTD will, however, ensure that at least one path is available -- on-line and not reserved. If an available path cannot be found, PTD will display:

"AT LEAST ONE PATH MUST BE ONLINE AND NOT RESERVED."

- If no path selection is attempted and the unit is reserved, PTD will automatically search for an on-line path in this order:
 - i. the on-line, non-reserved path with the highest PATHID value; e.g., if PATHIDs 11 and 10 are available, then 11 will be selected.
 - ii. the on-line, reserved, unassigned path with the highest PATHID value.

If no suitable path is found, PTD will respond:

"NO PATHS TO THE UNIT ARE ONLINE."

B. BADDISK TESTS (PTD/CONF/BD20X and PTD/CONF/BD2X5)

Path selection is never allowed and an on-line, non-reserved path must be available.

Since the test case requires a valid BADDISK file, the disk file header must be in memory so that the specified file can be located. Therefore, the unit cannot be reserved and path selection is always left to the MCP. PTD will ensure that at least one path is on-line and not reserved.

- If the operator attempts to specify a path, PTD responds:

"PATH SELECTION NOT ALLOWED FOR THIS TEST."

- If PTD can see no available paths to the unit holding the file, the following message is issued:

"AT LEAST ONE PATH MUST BE ONLINE AND NOT RESERVED."

C. ALL OTHER CONFIDENCE TESTS (PTD/CONF/IVR, PTD/CONF/MT, etc.)

The path does not necessarily have to be reserved. Path selection by the user is always allowed and an on-line, unassigned path is required. (Recall that the unit must have been reserved.)

- If the user specifies an off-line path, PTD will respond:

"PATH IS OFFLINE - TEST REQUIRES AN ONLINE PATH."

- If the user specifies an on-line, reserved path but the path is assigned to another stack, PTD will respond:

"PATH IS IN USE."

- If no path selection is attempted and the unit is reserved, PTD will automatically search for an on-line path in this order:
 - i. the on-line, non-reserved path with the highest PATHID value; e.g., if PATHIDs 11 and 10 are available, then 11 will be selected.
 - ii. the on-line, reserved, unassigned path with the highest PATHID value.
- If no suitable path is found, PTD will respond:

"NO PATHS TO THE UNIT ARE ONLINE."

D. ALL MAINTENANCE TESTS (PTD/MAINT/...)

Path selection by the user and PTD is allowed; however, the path must be reserved and not assigned to another stack. (Recall that the unit must have been reserved.)

- If a non-reserved path is specified, PTD will respond:

"PATH IS NOT RESERVED."

- If a reserved path is chosen, but that path is assigned to another stack (e.g., another invocation of PTD), PTD will respond:

"PATH IS IN USE."

- If no path selection is attempted, PTD will search for a path in this order:
 - i. the path with the highest PATHID value that is already assigned to this invocation of PTD. In this way, test cases with more than one DEVICESTRUCTURE will always use the same path unless the user specifies otherwise.
 - ii. the reserved, on-line, unassigned path with the highest PATHID value.
 - iii. the reserved, off-line, unassigned path with the highest PATHID value.

If no reserved path is found, PTD will respond:

"NO PATHS TO THE UNIT ARE RESERVED."

If all paths are reserved and assigned to other stacks, PTD will report:

"ALL PATHS TO THE UNIT ARE IN USE."

PATH SELECTION EXAMPLE 1

The following are valid responses for the device-request in the 206/207 diskpack maintenance test, PTD/MAINT/HT20X, where there are four hypothetical paths to unit PK49:

PATHID	PATHSTATUS
11	RESERVED OFFLINE
10	RESERVED ONLINE
09	ONLINE
08	RESERVED ONLINE

PTD:
ENTER DEVICE FOR DISK

USER:
PK 49 % PTD would automatically select PATHID 10
 % since it is the reserved, on-line path
 % with the highest PATHID value.

- or -

PK 49 VIA % PTD would display all paths to the unit in
 % a form similar to the response to an
 % "OL" ODT command and then ask the user
 % to select a reserved path from the list.

- or -

PK 49 VIA PATHID 11 % Since this is a maintenance test and both
 % the unit and path are reserved, this would
 % be an acceptable specification.

PATH SELECTION EXAMPLE 2

Suppose that, in the maintenance test of the above example, PATHID 10 has been selected as the path for the first DEVICESTRUCTURE to be opened. The following information would be displayed for a subsequent "OL PK49" ODT command:

PATHID	PATHSTATUS
11	RESERVED OFFLINE
10	ASSIGNED RESERVED ONLINE
09	ONLINE
08	RESERVED ONLINE

Suppose that a second DEVICESTRUCTURE is opened.

PTD:

ENTER DEVICE FOR DISK

USER:

PK 49 % PTD would automatically select PATHID 10,
 % since it is the highest PATHID value that
 % PTD has already assigned to an open
 % DEVICESTRUCTURE.

- or -

PK 49 VIA % PTD would display all paths to the unit in
 % a form similar to the response to an
 % "OL" ODT command and then ask the user
 % to select a reserved path from the list.

- or -

PK 49 VIA PATHID 11 % Since this is a maintenance test and both
 % the unit and path are reserved, this would
 % be an acceptable specification.

5. PTD OPERATOR INTERRUPTION

PTD operates in a "fetch S-op/execute S-op" cycle and is sensitive to an operator interrupt request before each fetch operation. An interrupt request is serviced only when all outstanding test case I/Os have been completed. When the operator-dialogue device is an ODT, PTD is interrupted by entering:

```
<SYSTEM/MAINTENANCE mix number> HI
```

Example:

```
1127 HI
```

When the operator dialogue is through a REMOTE station, the "HI" statement must be preceded by a question mark in column one.

Example:

```
?9142 HI
```

Once interrupted, PTD responds by displaying:

```
AWAITING DIRECTIVE
```

At this point, any valid PTD directive is accepted.

6. TESTCASE EXAMPLE

The following is an example of the printed output of the PE tape DLP maintenance test showing the use of the RUN, REPEAT, and GO directives, and illustrating one of the path selection techniques. PTD was operated at an ODT and, with CONTROLOLDWFL reset, was executed by entering:

```
BEGIN JOB; PTD; FILE SCODE(TITLE = PTD/MAINT/PE)
```

After the AWAITING DIRECTIVE message appeared, MONITOR was set -- all subsequent operator entries are flagged with the asterisk in column one. When "MT17" was first entered, the unit was found not to be reserved; the operator reserved it with the "UR MT17" ODT message (directed to the system by omitting the GS character). Next, "MT 17 VIA" was entered to PTD, resulting in the path information display and path selection query. When the device was successfully open, the test case later reported errors that were due to the operator not having mounted a tape on the drive. With the exception of the operator dialogue, all messages are produced by the test case.

```
LISTING FOR TESTCASE=PTD/MAINT/PE
AWAITING DIRECTIVE
*PRINT
AWAITING DIRECTIVE
*RUN 101
STARTING SECTION 0101
--> ECHO OP - ALL BITS OFF
ENTER DEVICE FOR MAGTAPE
*MT 17
UNIT MT 17 : NOT RESERVED.
ENTER DEVICE FOR MAGTAPE
[COMMENT: operator reserved MT 17]
*MT 17 VIA

UNIT 'MT17':
DLP ID =04
BASE NUMBER = 000
RELATIVE UNIT (W.R.T. DLP) = 1
PATH INFORMATION
-----

PATHID  PROC  MLIPPORT  LEMPORT  DLPNUM  PATHSTATUS
17      3      1      0      6      RESERVED, ONLINE

ENTER DESIRED PATHID
*17
AWAITING DIRECTIVE
```

```

*REPEAT 102 FOR 3
  STARTING SECTION 0102
  --> ECHO OP - ALL BITS ON
  STARTING SECTION 0102
  --> ECHO OP - ALL BITS ON
  STARTING SECTION 0102
  --> ECHO OP - ALL BITS ON
  AWAITING DIRECTIVE
*RUN 200-END
  STARTING SECTION 0200
  --> OP CODES TEST
  ### TEST SEL = 0200 TEST RUN = 0201 #####
  OPCODE+VAR = 2F1000
  OPCODE = 2F0000 = TEST
  + UNIT NUMBER = 1(01)
  & IDLENGTH = 0006 CHARACTERS
  CYCLE=001 I/O=00013 ERR:MLIP=000 L=000 RD=001 DATA=000
  RESULT DESCRIPTOR IS 41100000
  RD WD1 4000 = DESCRIPTOR ERROR
  RD WD1 0100 = TAPE UNIT NOT READY
  RD WD1 0010 = WRITE LOCKOUT
  RESULT DESCRIPTOR EX 00800000
  RD WD1 0080 = BOT (BEGINNING OF TAPE)
  ### RD ERROR ##### RD ERROR #####
  STOPPED ON ERROR
*GO
  ### TEST SEL = 0200 TEST RUN = 0201 #####
  OPCODE+VAR = 2F1000
  OPCODE = 2F0000 = TEST
  + UNIT NUMBER = 1(01)
  & IDLENGTH = 0006 CHARACTERS
  CYCLE=002 I/O=00014 ERR:MLIP=000 L=000 RD=002 DATA=000
  RESULT DESCRIPTOR IS 41100000
  RD WD1 4000 = DESCRIPTOR ERROR
  RD WD1 0100 = TAPE UNIT NOT READY
  RD WD1 0010 = WRITE LOCKOUT
  RESULT DESCRIPTOR EX 00800000
  RD WD1 0080 = BOT (BEGINNING OF TAPE)
  ### RD ERROR ##### RD ERROR #####
  STOPPED ON ERROR
*QUIT
*** EOJ PTD ***

```

D4240 PTDTEST - "PTDTESTS" TAPE

A tape, "PTDTESTS", is included in the B5900 and B6900 (referred to as "MLIP" systems) support release. It contains files required by hardware support personnel for maintenance and confidence testing of the new I/O facilities on MLIP systems.

The files on the PTDTESTS tape may be categorized as follows:

1. Confidence tests, PTD/CONF/=
2. Documentation for the confidence tests, PTD/DOC/CONF/=
3. Maintenance tests, PTD/MAINT/=
4. Documentation for the maintenance tests, PTD/DOC/MAINT/=
5. A summary of the confidence and maintenance files, PTD/DOC/INSTRUCTIONS.
(Any reference to files containing 'NS' in their titles should be ignored by MLIP systems customers.)
6. A users guide for the PTD (Peripheral Test Driver), PTD/DOC/USERS/GUIDE, which is augmented for MLIP systems by PTD note D4197, "Improved PTD Path Selection".
7. The symbolic and object files (LISTNOTES, OBJECT/LISTNOTES) of a program for processing the 'DOC' files.

All the files, except LISTNOTES, are generated by the I/O Systems Organization and copied to the PTDTESTS tape.

Processing the 'DOC' Files

A 'DOC' file (e.g., PTD/DOC/INSTRUCTIONS) may be printed by one of the following methods:

1. Batch Use

```

RUN *OBJECT/LISTNOTES; FILE IN(TITLE=PTD/DOC/INSTRUCTIONS);
VALUE N

```

2. Remote Use

R *LISTNOTES; FILE IN(TITLE=PTD/DOC/INSTRUCTIONS); VALUE N

The default value N=0 will cause the document to be printed, without translation, on an EBCDIC96 printer.

Value N=1 will allow the document to be printed, with translation to upper case, on an EBCDIC72 printer.

Detailed operating instructions for LISTNOTES appear in GENERAL note D3396. The operating instructions differ from those on releases prior to Mark 32. The instructions are also contained at the start of the symbolic file LISTNOTES.

LIBRARY TAPE: PTDTESTS HAS 37 FILES, 27700 SEGMENTS

* AFTER SEGS - FILE IS UNCRUNCHED

FILE TITLE	SEGS	TIMESTAMP	FILEKIND
*LISTNOTES,	%	360 07-05-80	ALGOLSYMBOL
*OBJECT/LISTNOTES,	%	126 07-05-80	ALGOLCODE
*PTD/CONF/BD2X5,	%	300*02-27-81	MDLCODE
*PTD/CONF/BD20X,	%	400*02-27-81	MDLCODE
*PTD/CONF/CP,	%	200*09-04-80	MDLCODE
*PTD/CONF/CR,	%	200*09-05-80	MDLCODE
*PTD/CONF/IVR,	%	300*12-16-80	MDLCODE
*PTD/CONF/PE,	%	200*09-04-80	MDLCODE
*PTD/CONF/PS,	%	300*09-11-80	MDLCODE
*PTD/CONF/TP,	%	300*09-04-80	MDLCODE
*PTD/DOC/CONF/BD,	%	966 06-18-80	CDATA
*PTD/DOC/CONF/CP,	%	266 05-27-80	CDATA
*PTD/DOC/CONF/CR,	%	266 05-30-80	CDATA
*PTD/DOC/CONF/IVR,	%	784 12-15-80	CDATA
*PTD/DOC/CONF/PE,	%	266 06-18-80	CDATA
*PTD/DOC/CONF/PS,	%	644 06-18-80	CDATA
*PTD/DOC/CONF/TP,	%	182 06-18-80	CDATA
*PTD/DOC/INSTRUCTIONS,	%	252 02-27-81	CDATA
*PTD/DOC/MAINT/CP,	%	2576 02-27-81	CDATA
*PTD/DOC/MAINT/CR,	%	2086 02-27-81	CDATA
*PTD/DOC/MAINT/HT,	%	1456 06-18-80	CDATA
*PTD/DOC/MAINT/LSP,	%	938 02-18-81	CDATA
*PTD/DOC/MAINT/NSP,	%	910 02-06-81	CDATA
*PTD/DOC/MAINT/ODT,	%	714 11-26-80	CDATA
*PTD/DOC/MAINT/PE,	%	2366 08-29-80	CDATA
*PTD/DOC/MAINT/TP,	%	1694 01-18-81	CDATA
*PTD/DOC/USERS/GUIDE,	%	448 08-08-80	CDATA
*PTD/MAINT/CP,	%	600*02-27-81	MDLCODE
*PTD/MAINT/CR,	%	600*02-27-81	MDLCODE
*PTD/MAINT/HT2X5,	%	500*09-04-80	MDLCODE
*PTD/MAINT/HT20X,	%	500*09-04-80	MDLCODE
*PTD/MAINT/LSP,	%	500*02-18-81	MDLCODE
*PTD/MAINT/NSP,	%	2100*01-21-81	MDLCODE
*PTD/MAINT/NSP/1,	%	2000*09-11-80	MDLCODE
*PTD/MAINT/ODT,	%	300*01-21-81	MDLCODE
*PTD/MAINT/PE,	%	600*09-11-80	MDLCODE
*PTD/MAINT/TP,	%	500*01-20-81	MDLCODE

DOCUMENT CHANGES NOTES (D NOTES)

REMOTE JOB ENTRY

D4117 RJE - "02" AND "04" CONTROL MESSAGE UPDATE

The 02 (Buffer Size) and 04 (Buffer Size Reply) control messages have been modified to allow for an optional four bytes of information at the end of the messages. These four bytes contain the File Transfer Block Size ("FTBLK") in decimal notation. The "FTBLK" can be changed by use at the *SF and *SM SF commands and interrogated with the *TF and *SM TF commands.

The new format of the messages is as follows:

```

----- 02 ---<buffer size>-----|
| - 04 - |                |-<ftblk>-|

```

The <buffer size> and "FTBLK" are four digit integers.

D4118 RJE - "CLEAR" <LSN> VS VIRTUAL STATIONS

The use of *SM CLEAR <LSN> will now cause the clearing of all virtual station information for the requested <LSN>.

D4163 RJE - REMOVE "BCL" CONSTRUCTS

All use of BCL constructs has been removed, leaving the program functionally the same.

D4164 RJE - FILE TRANSFER FLOODS MEMORY

The RJEAFTS and RJESEFTS terminal statements in the DCP section of SYMBOL/SOURCENDL have been modified to include MSGSPACE and MAXMSGSPACE. Assigning a value to MAXMSGSPACE will cause the DCP not to continue to allocate additional GETAREAs for the RJE file transfer terminals. This will cause memory not to be flooded as only MAXMSGSPACE GETAREAs can be used. For further information, refer to the Message-Oriented Data Comm Information Manual.

Note: The values of MSGSPACE and MAXMSGSPACE in SYMBOL/SOURCENDL are examples only and can be changed to meet an individual site's requirements.

D4185 RJE - BACKUP AND PRINT QUEUE REBUILDING

While rebuilding the PRINTQUEUE during initialization, RJE would incorrectly queue backup files of active CANDE tasks. Such files could then be prematurely removed by AUTOBACKUP of RJE with no lines printed. This problem has been corrected. Also, any manual request (*PB <job number>) that includes active CANDE tasks will only print those files whose tasks have completed.

To accomplish the above, the SYSTEMSTATUS mix information call (Group 3) has been modified to return a result of TRUE with field [12:1]=1 and [11:8]=33 if the mix number is that of an active session. Please note that the field [12:1] is valid only if [11:8] is 33.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

REMOTE JOB ENTRY

P1024 RJE - SESSION INITIALIZATION

RJE will now initialize a session upon receipt of an 09 control message if the effected station is a HOST or LOGON is reset. This will allow a cleaner File Transfer Link initialization.

P1025 RJE - TERMINAL TRANSFER COMPATIBILITY WITH "SYCOM"

This patch corrects several terminal transfer compatibility problems with B900/B1900 SYCOM.

P1026 RJE - "RJE HOST" TO "HOST" LINK LOSING FIRST MESSAGE

In some cases the first non-control (non-device address 00) message between two HOST systems would be lost by RJE. This problem has been corrected.

P1027 RJE - FILE TRANSFER VALID CHAR LIST UPDATE

System/RJE will no longer consider a 4*15" character as valid for file transfer. Any record containing a 4*15" will be expanded because of no corresponding ASCII character.

P1110 RJE - USERCODE WORD NOT ZEROED OUT

RJE now zeroes out the usercode word in log records before computing the usercode standard-form name length, thus correcting a possible dump from JOBFORMATTER.

P1168 RJE - CONTIGUOUS BLANKS

RJE would incorrectly deblank file transfer records with 200 or more contiguous blanks. This problem has been corrected.

P1169 RJE - "FTS" RECORD SIZE, BLOCK SIZE

The put/get record (control messages 10 and 11) format only allows four digits (maximum 9999) for MAXRECSIZE and BLOCKSIZE in characters. Now, the 9999-character limit is only enforced when the RJE file transfer is to a non-peer host (B1800, B800, etc.). Transfer to a peer host (B6700, B6800, B5900, etc.) uses the value of MAXRECSIZE and BLOCKSIZE found in the symbolic header record that precedes the data.

The error message displayed when MAXRECSIZE or BLOCKSIZE is greater than 9999 characters and transfer is to a non-peer host is the following:

```
#RECORD OR BLOCKSIZE OF FILE <file title> > 9999 CHARACTERS
```

P1191 RJE - STATION "ID" RECONFIGURATION

During reconfiguration, it was possible for a user to enter the name of a station belonging to another MCS or a station which was not in RJE's tables as an RJE terminal. This would cause either RJE to fault with an INVALID INDEX or the user's line to hang. RJE now recognizes these situations and does not reconfigure to such stations.

P1234 RJE - "CP-9500 (B800)" APPEARS AS PEER HOST

The method for determining if a remote host is a peer has been corrected to not report a CP-9500 (B800) as a peer. The only CPU models considered as peers are the following: B5900, B6700, B6800, B6900, B7700, B7800.

P1236 RJE - "OFFLINE" REQUEST BLANKS STATIONNAME ARRAY

An OFFLINE request will cause the stationname array entry for that unit number to be nulled for the following items: CPU model, protocol level and version, station ID, terminal ID and station type.

This change corrects conflicts at restart time when a different station (HOST or terminal) is attempting to use the line.

P1237 RJE - "RUNTIMEOPTIONS" STORAGE IN LINKFILE

It was possible for the RUNTIMEOPTIONS word (57) in the linkfile to be corrupted. This problem has been corrected.

P1348 RJE - "EBCDIC" TO "ASCII" FOR "B800" SYSTEMS

The following characters have been added to the valid character list for file transfer between B6000 and B800 systems:

	EBCDIC	ASCII
\ = BACK SLASH	E0	5C
_ = UNDERSCORE	D6	5F
˘ = CIRCUMFLEX (NOT)	5F	5E
{ = OPENING BRACE	C0	7B
} = CLOSING BRACE	D0	7D
The first non-blank character in the previous line is a representation of the closing brace.		
˜ = OVERLINE (TILDE)	A1	7E
The first non-blank character in the previous line is a representation of the overline (tilde).		
⏏ = VERTICAL LINE	6A	7C
The first non-blank character in the previous line is a representation of the vertical line.		

P1349 RJE - "FS1, FS2" RECORDS IGNORED

RJE now ignores the FS1 record at the start of a card (device address=02) input file and the FS2 record at the end. The FS1 and FS2 records are optional, and not all systems send them.

P3227 RJE - "SEG ARRAY" ERROR

RJE no longer DSEs with a SEG ARRAY error when expanding a terminal transfer record with a control character.

P3230 RJE - CONNECT REPLY FOR INACTIVE STATION

RJE now handles a Connect Reply (control message 21) for an inactive virtual station. A Disconnect (control message 22) is sent and corresponding arrays are cleared. A possible INVALID INDEX at MCS restart time has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

READER SORTER - RSMCP

P3255 RSMCP - "RSPIO" PARAMETER

In the Mark 31 release of SCR, RS SCANIN DATA and RS SCAN STATUS returned incorrect results.
This problem has been corrected.

DOCUMENT CHANGES NOTES (D NOTES)

SCRMCP

D3586 SCRMCP - "ICMD" MAINTENANCE

In order to qualify the Industry Compatible Mini-Disk (ICMD), the Maintenance and Test (MAT) language has been modified to do I/Os to ICMDs.

The following should be added to the MAT Language Information Manual (Form No. 5000169).

Section 7:

READ DT STATEMENT

SYNTAX:

```

<READ DT STATEMENT> ::= READ <UNIT SPECIFIER>
  <READ DT MODIFIER LIST> <ERROR HANDLING I/O MODIFIER>
<READ DT MODIFIER LIST> ::= <READ DT MODIFIER> /
  <READ DT MODIFIER LIST> <SEPARATOR> <READ DT MODIFIER>
<READ DT MODIFIER> ::= <STANDARD I/O MODIFIER> /
  <DISKETTE READ MODIFIER> / <DISKETTE ADDRESS I/O MODIFIER>

```

EXAMPLES:

1. READ DT58 SECTOR 1000 IOLENGTH = 1280 CHARACTERS INTO BUFFER A;
2. READ DT59 DELETED SECTOR X IOLENGTH = Z WORDS INTO BUFFER A ON ERROR();
3. READ UNIT U ALL SECTOR 0 IOLENGTH = 26 SECTORS INTO BUFFER A;
4. READ DT58 STATUS INTO BUFFER A;
5. READ DT58 EXTENDED INTO BUFFER A;

SEMANTICS:

THE VARIANTS OF READING A DISKETTE ARE EXPLAINED IN <DISKETTE READ MODIFIER>.

WRITE DT STATEMENT

SYNTAX:

```

<WRITE DT STATEMENT> ::= WRITE <UNIT SPECIFIER>
  <WRITE DT MODIFIER LIST> <ERROR HANDLING I/O MODIFIER>
<WRITE DT MODIFIER LIST> ::= <WRITE DT MODIFIER> /
  <WRITE DT MODIFIER LIST> <SEPARATOR> <WRITE DT MODIFIER>
<WRITE DT MODIFIER> ::= <STANDARD I/O MODIFIER> /
  <DISKETTE WRITE MODIFIER> /
  <DISKETTE ADDRESS I/O MODIFIER>

```

EXAMPLES:

1. WRITE DT58 SECTOR 1000 IOLENGTH = 1280 CHARACTERS FROM BUFFER A;
2. WRITE DT58 CHECK SECTOR X IOLENGTH = Z WORDS FROM BUFFER A ON ERROR();
3. WRITE UNIT U DELETED SECTOR 0 FROM BUFFER A;
4. WRITE DT59 CHECK DELETED SECTOR X IOLENGTH = Z SECTORS FROM BUFFER A;
5. WRITE DT58 FORMAT SECTOR 1924 FROM BUFFER A;
6. WRITE DT58 CHECK FORMAT SECTOR 0 FROM BUFFER A;

SEMANTICS:

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

THE VARIANTS OF WRITING A DISKETTE ARE EXPLAINED IN <DISKETTE WRITE MODIFIER>.

THE <TEST I/O STATEMENT> IS USED TO ISSUE THE TEST OPERATOR TO THE DISKETTE. FOR EXAMPLÉ:

TEST DT58;

SECTORSIZE I/O STATEMENT

SYNTAX:

```

<SECTORSIZE I/O STATEMENT> ::= SECTORSIZE <UNIT SPECIFIER>
    SIZE = <PRIMARY> BYTES <OPTIONAL SECTORSIZE MODIFIERS>
    <ERROR HANDLING I/O MODIFIER>
<OPTIONAL SECTORSIZE MODIFIERS> ::= <SECTORSIZE OPTIONS> /
    <OPTIONAL SECTORSIZE MODIFIERS> <SEPARATOR>
    <SECTORSIZE OPTIONS>
<SECTORSIZE OPTIONS> ::= <ATTENTION I/O MODIFIER> /
    <RESULT I/O MODIFIER>
  
```

EXAMPLES:

1. SECTORSIZE DT58 SIZE = 100 BYTES;

SEMANTICS:

THE <SECTORSIZE I/O STATEMENT> IS ONLY APPLICABLE TO DISKETTES AND SETS THE SECTOR SIZE OF THE DATA TRANSMITTED TO AND FROM A SPECIFIED DISKETTE ON SUCCEEDING I/O STATEMENTS TO THAT DISKETTE. IT MAY BE INVOKED ANY NUMBER OF TIMES, BUT THE MOST RECENT INVOCATION DETERMINES THE SECTOR SIZE TRANSFERRED IN LATER I/O STATEMENTS FOR THAT SPECIFIED DISKETTE.

IT IS RECOMMENDED TO SET THE SECTOR SIZE BEFORE DOING ANY READS OR WRITES TO THE DISKETTE. IF THE SECTOR SIZE FOR A DISKETTE IS NOT SET BEFORE THE FIRST I/O TO THAT UNIT, MAINTENANCE WILL ASSUME, FOR PURPOSES OF IOLENGTH CALCULATIONS, THAT THE SECTOR SIZE IS 128 BYTES. THEN IF THE ACTUAL SECTOR SIZE IS LESS THAN 128 BYTES, THE IOLENGTH CALCULATED WILL BE WRONG.

ANY I/O'S TO TRACK ZERO (SECTORS 0 THROUGH 25) USE A SECTOR SIZE OF 128 BYTES, EVEN IF THE SECTOR SIZE FOR THE SPECIFIED UNIT HAS BEEN SET TO LESS THAN 128 BYTES. MAKE SURE THE SECTOR SIZE IS SET TO 128 BYTES BEFORE READING OR WRITING TRACK ZERO OR THE IOLENGTH CALCULATIONS COULD BE WRONG.

<PRIMARY> MUST BE AN EVEN NUMBER BETWEEN 4 AND 128, INCLUSIVE.

RESTORE I/O STATEMENT

SYNTAX:

```

<RESTORE I/O STATEMENT> ::= RESTORE <UNIT SPECIFIER>
    <OPTIONAL RESTORE MODIFIERS> <ERROR HANDLING I/O MODIFIER>
<OPTIONAL RESTORE MODIFIERS> ::= <RESTORE OPTIONS> /
    <OPTIONAL RESTORE MODIFIERS> <SEPARATOR> <RESTORE OPTIONS>
<RESTORE OPTIONS> ::= <ATTENTION I/O MODIFIER> /
    <RESULT I/O MODIFIER>
  
```

EXAMPLES:

1. RESTORE DT59;

SEMANTICS:

THE <RESTORE I/O STATEMENT> IS ONLY APPLICABLE TO DISKETTES AND CAUSES THE READ/WRITE HEAD TO RETURN TO THE PHYSICAL TRACK ZERO POSITION.

Section 8:

DISKETTE READ MODIFIER

SYNTAX:

```
-----
<DISKETTE READ MODIFIER> ::= <EMPTY> / DELETED / ALL / STATUS /
EXTENDED
```

EXAMPLES:

- ```

1. ...
2. ...DELETED...
3. ...ALL...
4. ...STATUS...
5. ...EXTENDED...
```

## SEMANTICS:

```

ONLY ONE <DISKETTE READ MODIFIER> MAY BE USED IN A <READ DT STATEMENT>.
```

IF THE <DISKETTE READ MODIFIER> IS EMPTY, A READ/COUNT IS DONE. THE SECTORS ARE READ IN AND THE FIRST N BYTES OF EACH, AS SPECIFIED IN THE PREVIOUS <SECTORSIZE I/O STATEMENT>, ARE TRANSFERRED UNTIL THE NUMBER OF SECTORS SPECIFIED IN THE <IOLENGTH I/O MODIFIER> IS REACHED. DELETED/FAULTY SECTORS ARE NOT TRANSFERRED BUT THEY ARE COUNTED AND THEIR OCCURRENCE IS NOTED IN THE RESULT DESCRIPTOR. FOR EXAMPLE, A REQUEST OF 10 SECTORS WILL RESULT IN 9 SECTORS GETTING TRANSFERRED IF ONE DELETED SECTOR IS ENCOUNTERED.

IF "DELETED" IS SPECIFIED, THE SAME ACTION OCCURS AS READ/COUNT EXCEPT THAT THE OPERATION IS ALSO TERMINATED AFTER READING AND TRANSFERRING THE FIRST DELETED/FAULTY SECTOR ENCOUNTERED.

IF "ALL" IS SPECIFIED, THE SAME ACTION OCCURS AS READ/COUNT EXCEPT THAT ALL SECTORS (BOTH NORMAL AND DELETED/FAULTY) ARE READ AND TRANSFERRED.

IF "STATUS" IS SPECIFIED, THE STATUS INFORMATION IS RETURNED. THE 16 BITS OF STATUS INFORMATION INCLUDE THE FIRMWARE LEVEL OF THE CONTROLLER AND THE READY STATUS OF THE UNIT. THE <DISKETTE ADDRESS I/O MODIFIER> IS NOT SPECIFIED WITH THE STATUS MODIFIER.

IF "EXTENDED" IS SPECIFIED, THE EXTENDED RESULT DESCRIPTOR IS RETURNED. THE 32 BITS OF EXTENDED RESULT DESCRIPTOR INFORMATION REGARDING THE LAST I/O INCLUDES THE ENDING RELATIVE SECTOR NUMBER AND EXTENDED ERROR BITS. A NON-RECOVERABLE ERROR WILL CAUSE THE ADDRESS OF THE SECTOR BEING READ OR WRITTEN AT THE SAME TIME OF THE ERRORS TO BE STORED IN THE EXTENDED R/D SECTOR ADDRESS FIELD. THE <DISKETTE ADDRESS I/O MODIFIER> IS NOT SPECIFIED WITH THE EXTENDED MODIFIER.

## DISKETTE WRITE MODIFIER

## SYNTAX:

```

<DISKETTE WRITE MODIFIER> ::= <DISKETTE WRITE CHECK>
<DISKETTE WRITE VARIANT>
<DISKETTE WRITE CHECK> ::= <EMPTY> / CHECK
<DISKETTE WRITE VARIANT> ::= <EMPTY> / DELETED / FORMAT
```

## EXAMPLES:

- ```
-----
1. ...
2. ...CHECK...
3. ...DELETED...
4. ...CHECK DELETED...
5. ...FORMAT...
6. ...CHECK FORMAT...
```

SEMANTICS:

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

ONLY ONE <DISKETTE WRITE MODIFIER> MAY BE USED IN A <WRITE DT STATEMENT>.

IF "CHECK" IS SPECIFIED, UPON COMPLETION OF A WRITE OPERATION, ALL DATA JUST WRITTEN IS VERIFIED BY CHECKING ALL CRC BYTES. THE WRITE CHECK PHASE WILL NOT BE STARTED ON AN UNSUCCESSFUL WRITE OPERATION. THE COMPLETE BLOCK WILL BE CHECKED FOR ERRORS. THE ADDRESS OF THE FIRST SECTOR IN ERROR WILL BE STORED IN THE EXTENDED R/D SECTOR ADDRESS FIELD. ADDITIONAL ERRORS WILL CAUSE THE MULTIPLE ERROR FLAG TO BE SET. NOTE THAT WRITE CHECK IS NOT USED INDEPENDENTLY, BUT IS AN ADDITION TO THE WRITE OPERATION.

IF <DISKETTE WRITE VARIANT> IS EMPTY, SECTORS ARE WRITTEN NORMALLY. IF THE NUMBER OF BYTES PER SECTOR AS SPECIFIED IN THE PREVIOUS SECTOR SIZE I/O IS LESS THAN 128, THEN THE REMAINDER OF THE SECTOR WILL BE FILLED WITH EBCDIC BLANK (HEX "40") CHARACTERS.

IF "DELETED" IS SPECIFIED, THE ACTION TAKEN IS THE SAME AS WRITE NORMAL EXCEPT THAT THE FIRST LOGICAL SECTOR SPECIFIED IS WRITTEN AS A DELETED/FAULTY SECTOR. IF "CHECK" IS ALSO SPECIFIED, THE WRITE CHECK PHASE IGNORES THE DELETED/FAULTY SECTOR ERROR ON THE FIRST SECTOR CHECKED BUT WILL TERMINATE IF ANY OTHER SECTOR ENCOUNTERED HAS THAT CONDITION. LIKEWISE A DELETED/FAULTY SECTOR ERROR IS RETURNED FOR SECTORS OTHER THAN THE FIRST.

IF "FORMAT" IS SPECIFIED, THE TRACK ADDRESSES AND GAPS FOR THE SPECIFIED PHYSICAL TRACK ARE WRITTEN. THE SECTOR DATA FIELDS ARE FILLED WITH BLANKS. THE SEEK TO THE SPECIFIED PHYSICAL TRACK IS PRECEDED BY A RESTORE. THE <DISKETTE ADDRESS I/O MODIFIER> SHOULD BE THE FIRST SECTOR OF THE TRACK FOR A WRITE FORMAT. THE DATA IN THE BUFFER IS NOT TRANSFERRED AND THE SECTOR SIZE IS IGNORED.

DISKETTE ADDRESS I/O MODIFIER

SYNTAX:

<DISKETTE ADDRESS I/O MODIFIER> ::= SECTOR <PRIMARY>

EXAMPLES:

1. SECTOR 1000
2. SECTOR X

SEMANTICS:

THIS MODIFIER IS ONLY APPLICABLE TO DISKETTES AND SPECIFIES THE STARTING RELATIVE SECTOR FOR AN I/O TO A DISKETTE.

<PRIMARY> MUST BE BETWEEN 0 AND 1949, INCLUSIVE.

IOLENGTH I/O MODIFIER (Page 8-24)

Change to syntax:

<IOLENGTH UNITS> ::= CHARACTERS / SEGMENTS / WORDS / SECTORS

Add to examples:

4. IOLENGTH = 26 SECTORS

Add to semantics:

THE <IOLENGTH I/O MODIFIER> HAS SPECIAL SEMANTICS FOR DISKETTES. IF "SECTORS" IS SPECIFIED, <PRIMARY> IS PUT DIRECTLY INTO THE "NUMBER OF SECTORS" FIELD OF THE IOCW AND THE WORD-COUNT TO BE USED IS CALCULATED USING THE BYTES PER SECTOR SPECIFIED IN THE PREVIOUS <SECTORSIZE I/O STATEMENT> FOR THE DISKETTE. THE "SECTORS" MODIFIER MAY ONLY BE USED IF THE <I/O STATEMENT> REFERENCES A DISKETTE. IF "CHARACTERS" IS SPECIFIED, THE NUMBER OF SECTORS TO PUT INTO THE IOCW IS CALCULATED USING THE BYTES PER SECTOR SPECIFIED AND THE WORD-COUNT TO BE USED IS THE <PRIMARY> MULTIPLIED BY 6 CHARACTERS PER WORD. IF "WORDS" IS SPECIFIED, THE NUMBER OF SECTORS TO PUT INTO THE IOCW IS CALCULATED USING THE BYTES PER SECTOR SPECIFIED AND <PRIMARY> IS THE WORD-COUNT. IF NO <IOLENGTH I/O MODIFIER> IS SPECIFIED, THE NUMBER OF SECTORS IS CALCULATED USING THE LENGTH OF THE BUFFER REFERENCED. THE "NUMBER OF SECTORS" FIELD OF THE IOCW MUST BE BETWEEN 1 AND 99, INCLUSIVE. IF THE NUMBER OF SECTORS IS SPECIFIED OR CALCULATED TO BE LESS THAN 1 OR GREATER THAN 99, 99 IS PLACED IN THE IOCW IF IT IS A WRITE STATEMENT AND AN ERROR OCCURS IF IT IS A READ STATEMENT.

TESTOP I/O MODIFIER (Page 8-41)

Add to semantics:

6. <DISKETTE ADDRESS I/O MODIFIER>

UNIT DECLARATION (Page 4-11)

Change the syntax:

<UNIT MNEMONIC> ::= CP / CR / DK / DT / HC / LP / MT / PK /
PP / PR / SC
<RESERVED UNIT TYPE> ::= CP / CR / DT / HC / LP / MT / PP / PR

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SCRMCP

P1195 SCRMCP - "SCR TEST11" RECOGNIZES PACK

The MAKETESTPACK of DISKPACKCONFIDENCE in MAINTENANCE was writing the label into the wrong segment. Now the label is written to the correct segment.

P1196 SCRMCP - TAGS MATCH "TEST7"

Tags were not compared in the COMPARE routine of TEST7 of SCR. Now, the tag is compared.

DOCUMENT CHANGES NOTES (D NOTES)

SORTMCP

D4133 SORTMCP - SORT CHANGES VALUE OF "NEWFILE"

The sort changes the value of the NEWFILE attribute if appropriate and the NEWFILE attribute has been previously set TRUE or FALSE for any file passed into the sort.

DOCUMENT CHANGES NOTES (D NOTES)

SOURCENDL

D4164 SOURCENDL - FILE TRANSFER FLOODS MEMORY

The RJEAFTS and RJESFTS terminal statements in the DCP section of SYMBOL/SOURCENDL have been modified to include MSGSPACE and MAXMSGSPACE. Assigning a value to MAXMSGSPACE will cause the DCP not to continue to allocate additional GETAREAs for the RJE file transfer terminals. This will cause memory not to be flooded as only MAXMSGSPACE GETAREAs can be used. For further information, refer to the Message-Oriented Data Comm Information Manual.

Note: The values of MSGSPACE and MAXMSGSPACE in SYMBOL/SOURCENDL are examples only and can be changed to meet an individual site's requirements.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SOURCENDL

P1257 SOURCENDL - "RJE" HOST LINK REESTABLISHMENT

If two RJE host systems were communicating over a private link and one system went down, when that system came back up, its RJE request set running in its DCP would have reinitialized all its link-level variables. However, the variables in the RJE request set on the other system's DCP would not have changed. This could cause a problem in link reestablishment between those two systems. Since one of the variables contains a module-two transmission number, 50% of the time the first new message from the system reestablishing would look like a retransmission of a previously-accepted message and would therefore be discarded. This problem has been eliminated by performing the same variable reinitialization in the NDL at link reestablishment as was previously done when a switched link disconnected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SOURCENDLII

P1028 SOURCENDLII - FULL DUPLEX LINE HANDLING CORRECTED

These changes correct the operation of the Full Duplex TTY Line handling. Problems corrected include:

1. Line Control and Line Output Control Process could be left hung in the event of a BRK or DISCONNECT detected by Line Input.
2. False starts of LINE INPUT and OUTPUT CONTROLLERS whenever something set LINE.STATUSCHANGED to TRUE.
3. Failure of ADAPTOR CONTROL to RESET BREAK DETECTED with the result that no output messages would be subsequently transmitted.

P1283 SOURCENDLII - ELIMINATE "STRUCTURE PROTECT" FAULTS

Under certain conditions, the TD algorithm could fault terminate with a STRUCTURE PROTECT error. This problem, caused by mistakenly initiating the adaptor input process with a null-length poll list, has been corrected.

P3228 SOURCENDLII - "TTY/FULLDUPLEXTTY" PAGE, BREAK HANDLING

Formerly, if a station using the TTY algorithm was waiting on a page boundary, an attempt to make the station not ready would fail, leading to problems in clearing the station. This problem has been resolved.

In addition, "?BRK" handling in the FULLDUPLEXTTY algorithm has been changed to work the same as in the TTY algorithm. Also, the means of cancelling input in both algorithms has been improved.

P3229 SOURCENDLII - UPPER/LOWER CASE HANDLING, EDIT BUFFER SIZE

The LOGICALIO Editor had the implementation of "?+" (lower-case enabled) and "?-" (upper-case only) exactly reversed. Also, a translate table was applied to TD830 input in addition to sending the escape sequence to set upper/lower-case in the terminal. Both these problems have been resolved.

Also, the size of the DESTINATION buffer in the output process of the LOGICALIO Editor is now determined based on the kind of output (normal/sequence mode), on whether scrolling is or is not enabled, and on whether the skip/space carriage control is being used. The effect of these calculations is to reduce buffer memory usage in most cases.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SUSPENDER

P1006 SUSPENDER - RESTORE "SHOW" COMMAND

The SHOW command, accidentally omitted between the Mark 30 release and the Mark 31 release, has been restored.

P1007 SUSPENDER - "SUSPENDER" ON MONOLITHIC SYSTEMS

A problem has been corrected where, on a monolithic system, nothing was being suspended.

DOCUMENT CHANGES NOTES (D NOTES)

SYSTEMSTATUS

D4291 SYSTEMSTATUS - DOCUMENTATION CHANGE

The example given on Page 1-10 of the SYSTEMSTATUS Reference Manual (Form No. 5011786) shows a quaternary string code in front of the string of ones instead of a binary string code. The example should read as follows:

SYSTEMSTATUS(A,2&1"11111"[29:5],0)

DOCUMENT CHANGES NOTES (D NOTES)

UDSTRUCTURE TABLE

D4192 UDSTRCTTAB - "IDENTITY"

A new item, IDENTITY, has been associated with each usercode. IDENTITY has no semantic meaning; it is intended to be a comment entry to be used by operations management. IDENTITY is of userdata type TEXT.

D4232 UDSTRCTTAB - "CLASS=0"

The description of CLASS=0, paragraph G on Page 9-29-9 of SOG Reference Manual, Volume 2 (Form No. 5001688), is correct (it should not be revised as stated in Mark 32 UDSTRUCTURE TABLE note D3388).

Paragraph G is repeated below for clarity:

*g. CLASS - CLASS is used to specify into which queue a job initiated by this user will go. (CLASS=0 indicates no specification; CLASS=4"80" forces CLASS to zero.)"

D4259 UDSTRCTTAB - "SYSTEMUSER" REDEFINED

On the Mark 31 release, a new bit was added to the userdatafile for Host Services. This bit, SYSTEMUSER, is used to allow or deny privileges when entering inter-system ODT commands. The location of this bit conflicted with a bit used on B7000 systems (see Mark 32 UDSTRUCTURE TABLE note D3221) and was moved. The new location created another conflict with B7000 systems. In order to resolve this conflict the bit has been moved again. Sites that have not set the SYSTEMUSER bit for any of their users should disregard the remainder of this note.

For sites that have set the SYSTEMUSER bit, the following procedure should be followed:

- a. Recompile any software that may reference the SYSTEMUSER bit. SYSTEM/BNA is the only standard software that uses it.
- b. For users that currently have the SYSTEMUSER bit set, reset the CHARGERREQ bit and remove the CAPABILITIES node.
- c. Set the SYSTEMUSER bit for those users that should be given that distinction.

Example:

Steps b and c can be accomplished for user X with the following MAKEUSER statement:

```
USER X -CHARGERREQ -CAPABILITIES +SYSTEMUSER;
```

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

UTILITY LOADER

P1261 UTILOADER - "UTILOADER VIA" SYNTAX VS. "MCP"

When specifying "PK <nn> VIA <path address>" for the HALTLOAD command on MLIP systems, certain Mark 32 level MCPs would take a printer dump. This problem has been corrected.

P1266 UTILOADER - SAVE SPACE

To reduce the size of UTILOADER, all SAVE ARRAYS have been converted to regular arrays. In the beginning of the outer block, memory allocation for all global arrays is done, starting down from the end of MOD 0 (4'3FFF'). UTILOADER will stop if it tries to allocate space below 4'2300' (it is assumed UTILOADER will run below 4'2300').

DOCUMENT CHANGES NOTES (D NOTES)

WORK FLOW LANGUAGE

D4165 WFL - "BEND CARDS" EXAMPLE

Replace the WFL job on Page 6-12 of the WFL Reference Manual (Form No. 5001555) with the following:

```
<i>BEGIN JOB;  
  COMPILER MAKE/BEND/CARDS WITH ALGOL GO;  
  COMPILER DATA CARD  
  BEGIN  
    FILE P(KIND=PUNCH, EXTMODE=BINARY);  
    THRU 20 DO  
      BEGIN  
        P.OPEN :=TRUE;  
        CLOSE(P);  
      END;  
    END.  
<i>END JOB.
```

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

WORK FLOW LANGUAGE

P1083 WFL - BY REFERENCE "TASK" FOR "COMPILE" AND "GO"

When a TASK, which was passed by reference to a WFL subroutine, was used on the GO part of a COMPILE and GO, the WFL compiler did not handle it correctly.

This problem has been corrected.

P1087 WFL - ELIMINATE "BCL" CONSTRUCTS FROM "WFL"

All uses of BCL pointers have been removed from WFL.

P1242 WFL - USERDATA CLASS=4'80"

When CLASS=4'80" was specified for a user in USERDATA, WFL was not forcing the user to use CLASS=0 (as indicated in the SOG Reference Manual, Volume 2). This problem has been corrected.

P1243 WFL - "\$INCLUDE * <FILENAME>"

WFL was ignoring the asterisk on file names for INCLUDEs. The following example would be treated as (<usercode>)A/B:

```
$INCLUDE * A/B
This problem has been corrected.
```

P1244 WFL - MULTIPLE "INCLUDE" STATEMENTS

WFL did not reset the FILETYPE to 8 for the include file for jobs with multiple INCLUDEs. This could cause a problem when the attributes of the include files were different. This problem has been corrected.

P3709 WFL - NAMES WITH HYPHENS AND UNDERSCORES

WFL will now recognize hyphens (-) and underscores (_) in <name>s which are represented by #<string primary>.

Example:

```
STRING S;
S:="A B-C/D ON DISK"
COMPILE X WITH ALGOL;
  ALGOL FILE CARD(TITLE=#S);
```

DOCUMENT CHANGES NOTES (D NOTES)

DOCUMENTS

D4152 DOCUMENTS - MARK "31" COMPILATION

The first sentence in paragraph 5 of the Mark 31 GENERAL note D2465 should read as follows:

"The Mark 31 ALGOL, DCALGOL and NEWP can be compiled on a Mark 30 MCP using a Mark 30 ALGOL compiler. The Mark 31 MCP, CONTROLLER and WFL can be compiled on a Mark 30 MCP, but only with Mark 31 compilers."

D4153 DOCUMENTS - "??RJ" ODT MESSAGE

The ??RJ primitive ODT message marks the JOBDESC file as "to be removed" at the next Halt/Load. The file is not reinitialized immediately; the Halt/Load is required.

D4188 DOCUMENTS - RECORD NUMBER VS. LINE NUMBER

Page 1-4-2 of the SOG Reference Manual, Volume 1 (Form No. 5011661, which describes the "File Format" of backup files, should be corrected, as follows:

Word 298 contains the number of lines in the block (not the number of records).

Word 299 contains the line number of the first record (not the record number of the first record).

(A record that specifies double space after PRINT would be counted as two print lines.)

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SYSTEST - IO/IOTEST

P1274 IOTEST - REMOVE "BCL" CONSTRUCTS

All uses of BCL constructs have been removed for use on EBCDIC systems.

The option to print out miscompares in BCL has been deimplemented; any miscompares will be printed in hex.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SYSTEST - IO/IOINTERACT

P1274 IOINTERACT - REMOVE "BCL" CONSTRUCTS

All uses of BCL constructs have been removed for use on EBCDIC systems.

The option to print out miscompares in BCL has been deimplemented; any miscompares will be printed in hex.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SYSTEST - LANG/ALGOLSORT

P3276 LNG-ALGOLSORT - PRINT MONTH PROPERLY

A problem existed which caused the month in the "date run" to always be printed as "1" (i.e., January), regardless of the actual date. This problem has been corrected.

SOFTWARE IMPROVEMENTS NOTES (P NOTES)

SYSTEST - SCR/SC01

P3274 SC01 - SYNTAX ERRORS CORRECTED

Syntax errors that existed in SYSTEST/SCR/SC01 have been corrected.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

DOCUMENT CHANGES NOTES (D NOTES)

SYSTEST - SCR/ICMD

D4271 ICMD - "SCR/ICMD"

SYSTEST/SCR/ICMD has been added to the SYSTESTSMPX tape.

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
ACR	32.1.0157*	42323	P1004	Incorrect Key/Data Mismatch, W
ACR	32.1.0158*	42702	P1039	LIMITERROR Problems
ACR	32.1.0159*	43042	P1040	Ordered Data Set Records Out o
ACR	32.1.0160	43218	P1094	Visible DBS, INQUIRY Return So
ACR	32.1.0161*	43058	P1101	Population Too High
ACR	32.1.0162*	43223	P1115	Linearsearch Not Finding All R
ACR	32.1.0163	43236	P1226	Abort Failure
ACR	32.1.0164*	42702	P1039	LIMITERROR Problems
ACR	32.1.0165*	43228	P1156	Spurious NOTFOUND After Abort
ACR	32.1.0166*	43242	P1218	FIND PRIOR AT KEY
ACR	32.1.0167*	43243	P1219	Usage Statistics Inconsistency
ACR	32.1.0168	43568	P1227	Opening of Data Base Results i
ACR	32.1.0169	43248	P1241	Ordered Data Set Corruption
ACR	32.1.0170*	43259	P1245	Direct Key Corrupted
ACR	32.1.0171*	42323	P1004	Incorrect Key/Data Mismatch, W
ACR	32.1.0172	44036	P1374	DS of Linear Search
ACR	32.1.0174	43583	P3194	Bad Open
ACR	32.1.0175	43569	P1375	Limit Errors
ACR	32.1.0176	43962	P1376	NO FILE on RECOVERYINFO
ACR	32.1.0178	43969	P3250	Abort Overlooks Duplicated Aud
ACR	32.1.0179	44560	P3257	Rebuild Across Reorganization
ACR	32.1.0180	43066	P3327	Excessive Memory
ACR	32.1.0181	44801	P3328	FORGETSPACE Timing Window
ALGOL	32.1.0135	42242	P3813	"REAL (<pointer exp>,<arithmet
ALGOL	32.1.0136	42245	D4108	BCL Constants Allowed
ALGOL	32.1.0137	42246	P1017	Erroneous Attribute Error Mess
ALGOL	32.1.0138	42247	D4109	"<fault stack history> ARRAY"
ALGOL	32.1.0139	42248	D4110	Truthset ALPHA6 Allowed
ALGOL	32.1.0140	42249	D4111	BCL Constants
ALGOL	32.1.0141	42243	D4112	Dope Vectors can be Resized
ALGOL	32.1.0142	42243	D4112	Dope Vectors can be Resized
ALGOL	32.1.0143	42243	D4112	Dope Vectors can be Resized
ALGOL	32.1.0146*	43010	P1124	"IF <boolean exp> THEN ELSE" S
ALGOL	32.1.0150*	43649	D4162	Call By Name Parameters
ALGOL	32.1.0151	32937	D4166	Allow Underscores in Identifie
ALGOL	32.1.0152*	43651	P1217	Incorrect Code for "Pointer-Va
ALGOL	32.1.0155*	43656	P1277	LINEINFO
ALGOL	32.1.0156*	43651	P1217	Incorrect Code for "Pointer-Va
ALGOL	32.1.0157*	43655	P1287	"IPC-Capable", RUN, CALL State
ALGOL	32.1.0158*	43657	P1300	Incorrect Branch Using SWITCH
ALGOL	32.1.0160	43662	D4260	ALGOL BATCH and Arrays
ALGOL	32.1.0170	44656	P3306	SEG ARRAY Error for Large Data
ALGOL	32.1.0172	44658	D4312	Double Vs. Single "By-Name" Pa
ALGOL	32.1.0174	45024	D4325	Repeat Count, Size Field "<403
ALGOLTABLE	32.1.0026	42243	D4112	Dope Vectors can be Resized
ALGOLTABLE	32.1.0027*	43651	P1217	Incorrect Code for "Pointer-Va
ALGOLTABLE	32.1.0028	45024	D4325	Repeat Count, Size Field "<403
ALGOLTABLE	32.1.0029	44658	D4312	Double Vs. Single "By-Name" Pa
ATTABLEGEN	32.1.0021	42907	D4131	Spelling Corrected
ATTABLEGEN	32.1.0022	43167	D4138	New Values for SUBFILEERROR
ATTABLEGEN	32.1.0023	43618	D4230	New Attributes Implemented
BACKUP	32.1.0015*	42114	P3743	Drop Spaces in File Title
BACKUP	32.1.0016*	42531	P3759	Requires LP After Punching Fil
BACKUP	32.1.0018*	42540	P3796	Report EQUAL Correction
BACKUP	32.1.0019*	43668	P1201	DLBACKUP Correction
BACKUP	32.1.0021	43697	P1014	ND REPORT Corrected
BACKUP	32.1.0022*	43749	P1292	Restore Printing Speed
BACKUP	32.1.0023	44291	P1371	Double Printing
BARS	32.1.0014*	43704	P1015	SPO Mode
BARS	32.1.0015*	43705	P1258	SPO Mode UNRECOGNIZED REQUEST
BARS	32.1.0016*	44050	P1293	Display of Overlay Traffic Per
BASIC	32.1.0009*	41586	D4155	Flag Length Specification for
BASIC	32.1.0010*	41589	P1162	Compiler Capacity for String L
BASICSUPP	32.1.0023*	41585	P1127	Allow Six Characters in File T
BASICSUPP	32.1.0025*	41587	P1163	RESTORE Statement
BASICSUPP	32.1.0026*	41590	P1193	Flush Random Binary File Buffe
BASICSUPP	32.1.0027*	41591	P1192	Allow Access to File with Secu
BASICSUPP	32.1.0029*	41594	P1299	Matrix Inversion
BASICSUPP	32.1.0039	43813	P3307	Array of Strings Declaration
BDMSALGOL	32.1.0148*	43011	P1161	Interlanguage Binding
BDMSALGOL	32.1.0153*	43653	P1231	Segmented Transaction Records,
BDMSALGOL	32.1.0154*	43654	P1252	Signed Numbers, Hex Pointers
BDMSCOBOL	32.1.0107	43481	P1326	Selection Expression Truncatio
BDMSCOBOL	32.1.0120	42756	P3305	SIB Size Increase for Large Da
BDMSCOBOL74	32.1.0140	43481	P1326	Selection Expression Truncatio
BDMSCOBOL74	32.1.0151	42756	P3305	SIB Size Increase for Large Da
BINDER	32.1.0026	44026	P1289	Bound Code Release Levels
BINDER	32.1.0027	44237	P3186	BINDER Loses Sharing Class
BINDER	32.1.0028	44241	P3187	Deimplement Installation Intri
BINDER	32.1.0029	44374	P3213	Syntax Errors
BINDER	32.1.0030	44449	P3232	Installation One Intrinsic
BNA	32.1.0480	39735	D4134	Implement Logging for BNA

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

B5000/B6000/B7000 SERIES PATCH TABLE (PATCH CLOSING FTNS MARKED WITH *)				
SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
BNA	32.1.0486	39735	D4134	Implement Logging for BNA
BNA	32.1.0494	39735	D4134	Implement Logging for BNA
BNA	32.1.0523	43397	D4150	ODT CANDIDATE Message
BNA	32.1.0564	39735	D4134	Implement Logging for BNA
BUILDING	32.1.0015	43262	P1310	Use, Enforce Correct Item Fiel
CANDE	32.1.0041*	41912	P3760	DELETE ALL Syntax
CANDE	32.1.0043	43473	P1190	INVALID MODE Error for BCL Fil
CANDE	32.1.0045*	42957	D4174	Destination Filename Crunched
CANDE	32.1.0046	42519	P1284	Stack Size for PROGRAMDUMP
CANDE	32.1.0047	33191	P1285	Changing Tankfile Problem
CANDE	32.1.0048	42519	P1284	Stack Size for PROGRAMDUMP
CANDE	32.1.0049	43950	D4222	HN ODT Message
CANDE	32.1.0050	42488	P1372	Usage of RENEW
COBOL	32.1.0097*	42029	P3761	SEG ARRAY with DATADICTINFO Se
COBOL	32.1.0099*	42023	D3655	Level 01 Data Item
COBOL	32.1.0103*	39808	P1117	Library Template Splits Across
COBOL	32.1.0104	43517	P1136	OPEN, CLOSE Statements for Por
COBOL	32.1.0105*	42750	P1164	Global ISAM Files
COBOL	32.1.0106	43483	P1325	Truncation Warning for Report
COBOL	32.1.0108	43482	P1327	COMPUTE Statement in WHEN Clau
COBOL	32.1.0109	43484	P1328	BACKUPKIND Attribute
COBOL	32.1.0110	43487	P1329	Erroneous Right Parenthesis
COBOL	32.1.0111*	39808	P1117	Library Template Splits Across
COBOL	32.1.0112	44183	P3190	Intrinsic BINDINFO Mismatch
COBOL	32.1.0113	44186	P1373	Commas in OPEN Statements
COBOL	32.1.0116	44176	P3217	MAXRECSIZE of Report Files
COBOL	32.1.0119	43800	P3265	BLOCKSIZE Always Set
COBOL	32.1.0121	42761	P3313	Timeout Condition
COBOL	32.1.0122	43844	P3308	PORT Files, Perform Terminus C
COBOLTABLE74	32.1.0005	42751	D4202	USAGE IS ASCII Deleted
COBOLTABLE74	32.1.0006	42752	D4203	STACK SIZE Clause Added
COBOL74	32.1.0122*	42029	P3761	SEG ARRAY with DATADICTINFO Se
COBOL74	32.1.0128	42017	P1018	OPEN REVERSE
COBOL74	32.1.0130	43516	P1118	Status Key for Indexed I/O
COBOL74	32.1.0131*	39808	P1117	Library Template Splits Across
COBOL74	32.1.0132	43517	P1136	OPEN, CLOSE Statements for Por
COBOL74	32.1.0135	43845	D4193	Accept from Timer, 10 Digits
COBOL74	32.1.0136	42751	D4202	USAGE IS ASCII Deleted
COBOL74	32.1.0137	42752	D4203	STACK SIZE Clause Added
COBOL74	32.1.0138	42751	D4202	USAGE IS ASCII Deleted
COBOL74	32.1.0139	43483	P1325	Truncation Warning for Report
COBOL74	32.1.0141	43482	P1327	COMPUTE Statement in WHEN Clau
COBOL74	32.1.0142	43487	P1329	Erroneous Right Parenthesis
COBOL74	32.1.0143*	39808	P1117	Library Template Splits Across
COBOL74	32.1.0146	44176	P3217	MAXRECSIZE of Report Files
COBOL74	32.1.0148	42765	D4173	Segment Number
COBOL74	32.1.0149	43799	P3264	OPEN EXTEND with Tape Files
COBOL74	32.1.0150	43800	P3265	BLOCKSIZE Always Set
CONFIGURATOR	32.1.0010	44089	P1333	Unknown Group ID
CONTROLLER	32.1.0094*	42537	P3762	CU Display Correction
CONTROLLER	32.1.0105	43174	D4136	HOSTNSP Handling
CONTROLLER	32.1.0106*	42860	P1052	HARDCOPY Correction
CONTROLLER	32.1.0107*	42904	P1053	Resource Corrections
CONTROLLER	32.1.0113	43397	D4150	ODT CANDIDATE Message
CONTROLLER	32.1.0114*	42937	P1130	ADM EVENT PER Correction
CONTROLLER	32.1.0118	43554	P1165	Correct Misspelling
CONTROLLER	32.1.0119*	43665	P1203	WAIT Correction
CONTROLLER	32.1.0121	43696	P1259	Halt/Load Reason Check
CONTROLLER	32.1.0123	43202	D4199	Interval Logging for Files
CONTROLLER	32.1.0126	43950	D4222	HN ODT Message
CONTROLLER	32.1.0127	43933	D4245	LT ODT Message Deimplemented
CONTROLLER	32.1.0128	43937	D4221	TRAINID Field in UNITTABLE
CONTROLLER	32.1.0131	44312	P3236	TERM FULLPAGE Corrected
COPYAUD-II	32.1.0006	44828	P3343	Misleading "IOERRORs"
DASDL	32.1.0069*	43239	P1157	NOFILE on WFL Compile Job
DASDL	32.1.0070	43572	D4212	INITPARTITIONS Option Removed
DASDL	32.1.0072	44504	P3258	Deleting, Adding Global Data
DATA COM	32.1.1542	43444	P1184	HOST NSP Shutdown
DATA COM	32.1.1543	43445	P1185	Datacom Error Message Correcti
DATA COM	32.1.1544	43446	P1186	Error Handling in FIREOFFDCP
DATA COM	32.1.1545	43447	P1187	Unsuccessful I/O Message
DATA COM	32.1.1652	44003	P1273	B5900 Compatibility
DATA COM	32.1.1686	43448	P1358	FIREOFFDCP Problem
DATA COM	32.1.1687	43449	P1359	Recall Timing Window
DATA COM	32.1.1688	43451	P1360	Dialout Problem
DATA COM	32.1.1689	43450	D4226	Change Name of HOSTNSP Stack
DATA COM	32.1.1690	43452	P1361	BLOCKEXIT Mom Problem
DATA COM	32.1.1692	43454	P1341	FIREOFFDCP Problem
DATA COM	32.1.1693	44197	P1342	Datacom Table Problems
DATA COM	32.1.1694	44198	P1343	INVALID OP in DCINITIAL
DATA COM	32.1.1695	44199	P1344	Messages for DC Initialization
DATA COM	32.1.1696	44200	P1345	Loop Timeout

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

B5000/B6000/B7000 SERIES PATCH TABLE (PATCH CLOSING FTRS MARKED WITH **)

SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
DATAKOM	32.1.1705	44005	P1340	FORGETCHECK out of STARTSYSTEM
DATAKOM	32.1.1711	44201	P1338	Change NSP Autorecovery
DATAKOM	32.1.1713	43404	P1335	LINESWAP , REINITIALIZATION Co
DATAKOM	32.1.1716	44202	P1334	Retry Counts on Output Results
DATAKOM	32.1.1743	44205	P3205	Change NSP Initialization Fail
DATAKOM	32.1.1744	44203	P3206	Rejected Request
DATAKOM	32.1.1745	44008	P3207	DCRECON Timing Problem
DATAKOM	32.1.1746	44007	P3208	Call BLASTUNIT
DATAKOM	32.1.1767	44204	P3242	Queueing between DCINITIAL/DCC
DATAKOM	32.1.1768	44206	P3243	Initialization Aborted Message
DATAKOM	32.1.1769	44207	P3244	Override Local Optimization in
DATAKOM	32.1.1770	44208	P3245	Mark Line Dead for LINEDELETED
DATAKOM	32.1.1771	44209	P3246	Unit Left Assigned for NSP
DATAKOM	32.1.1772	44210	P3247	STATION NOT READY Handling
DATAKOM	32.1.1778	44009	P3249	Improve Tests
DATAKOM	32.1.1804	44464	P3260	DCINITIATEMCS Stack Overflow
DATAKOM	32.1.1805	44207	P3244	Override Local Optimization in
DATAKOM	32.1.1806	44213	P3261	Miscellaneous HOSTNSP Correcti
DATAKOM	32.1.1809	44466	P3262	Add Station Without Line Assig
DCALGOL	32.1.0147*	43009	P1123	Erroneous Syntax Error on DCAL
DCPPROGEN	32.1.0032*	42009	P3763	DCP Fault 8
DCPPROGEN	32.1.0035*	42680	P1001	Call Designation when Station
DCPPROGEN	32.1.0036*	42748	P1000	Three Address Characters
DCPPROGEN	32.1.0037*	42747	P1068	ASL Clears MAO
DCPPROGEN	32.1.0038	37310	D3371	Priority Output
DCPPROGEN	32.1.0039	44428	P3218	Priority Output Handling
DCPPROGEN	32.1.0040	43101	P3219	Set Dial Data Bit on Each Dial
DIAGNOSTMCS	32.1.0009	42372	D4114	Automatic DCWRITE ERROR Report
DIAGNOSTMCS	32.1.0010	43100	P1301	SS ALL Station not Attached
DUMPALL	32.1.0007*	42944	D4209	Hex and Decimal Record Numbers
DUMPANALY	32.1.0082	42093	D4103	Timestamp at BOT is Recorded i
DUMPANALY	32.1.0088	42539	D4104	New Words in CLOSE Log Record
DUMPANALY	32.1.0098	43494	P1175	Firmware Level Recorded on Dum
DUMPANALY	32.1.0099	43493	P1166	Path Name with Multiple Paths
DUMPANALY	32.1.0100	43627	P1213	Display MLIP Unit Status
DUMPANALY	32.1.0102	43202	D4199	Interval Logging for Files
DUMPANALY	32.1.0104	43937	D4221	TRAINID Field in UNITTABLE
DUMPANALY	32.1.0105	44311	P3191	INVALID INDEX Interrupt
DUMPANALY	32.1.0106	44308	P3193	Fault on Misspelled HELP Query
DUMPDIRLIB	32.1.0015	44038	P1308	SEG-ARRAY Error on "WRITE=","L
ESPOL	32.1.0005	43466	D4269	Cannot Run on "Non-BCL" System
ESPOL	32.1.0006	44381	D4270	Deimplementation Warning
FILECOPY	32.1.0013*	38562	P1253	Default Usercode
FILECOPY	32.1.0014*	44071	P1302	Listing of Serial Numbers
FORTTRAN	32.1.0063*	42553	P1119	READER, PRINTER Options Fail
FORTTRAN	32.1.0064*	42554	P1194	Variable in ASSIGNMENT Stateme
FORTTRAN	32.1.0066	44182	D4233	MODELI Compiler Option
FORTTRAN	32.1.0068	44698	D4324	Repeat Counts, Size Field "<40
GENERALSUPP	32.1.0063*	43416	P1099	Freefield Attempting to Write
GENERALSUPP	32.1.0071	43463	P1137	Remove BCL
GENERALSUPP	32.1.0108*	43889	P1303	Formatted Read of Strings
GENERALSUPP	32.1.0110	43891	P1331	K Modifier with Run Time Forma
GENERALSUPP	32.1.0113	44234	P1362	Freefield WRITE of Arrays
GENERALSUPP	32.1.0119	44372	P3214	G FORMAT When VALUE=0
GENERALSUPP	32.1.0122	44379	P3215	Read into String Error
GENERALSUPP	32.1.0123	43732	D4259	SYSTEMUSER Redefined
GENERALSUPP	32.1.0139	43809	P3277	F Format Output Suppresses Lea
GENERALSUPP	32.1.0140	43806	P3278	Repeat Counts of Zero in Forma
GENERALSUPP	32.1.0141	43807	P3312	Misspelled Format
GENERALSUPP	32.1.0142	43808	P3309	F Format Output Causes Incorre
GENERALSUPP	32.1.0143	43810	D4311	Repeat Count, Size Field "<403
GENERALSUPP	32.1.0144	43304	P3315	"Free-field" Input of Real, Bo
GENERALSUPP	32.1.0145	43305	P3325	BCL A Format Input
GENERALSUPP	32.1.0146	43306	P3314	Formatted Output with "A*" For
HARDCOPY	32.1.0001*	42530	P3764	HARDCOPY Not Reacting to HI
HARDCOPY	32.1.0002*	42529	P3765	HARDCOPY Files Left on Disk
HOSTLIB	32.1.0016*	42703	D4144	Increased Maximum Number of Ac
HOSTLIB	32.1.0017*	43220	P1108	Incorrect Interval Time
HOSTLIB	32.1.0018	43252	P1228	Transaction Records Exceed 205
HOSTLIB	32.1.0019	43570	P1229	Syntax Error
HOSTLIB	32.1.0020	43571	P1230	Deadlock
HOSTLIB	32.1.0021	43567	P3202	Response Transaction not Saved
HOSTLIB	32.1.0022	43580	P3201	Cannot Save Response Transacti
HOSTLIB	32.1.0023	43581	P3200	Recovery of Journal Data File'
HOSTLIB	32.1.0026	44912	P3339	Error Result not Returned
HOSTNSP	32.1.0002	43598	P1140	HOSTNSP MLIP Error Field Reten
HOSTNSP	32.1.0003	43599	D4158	Internal Efficiency Changes
HOSTNSP	32.1.0004	43600	P1141	HOSTNSP Abort "S-OP" Correctio
HOSTNSP	32.1.0005	43601	P1142	LISTLOOKUP
HOSTNSP	32.1.0006	43602	P1143	Null String Handling
HOSTNSP	32.1.0007	43603	P1144	Ready Queue Process Links Corr
HOSTNSP	32.1.0008	43315	D4141	HOSTNSP Implementation

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

B5000/B6000/B7000 SERIES SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
HOSTNSP	32.1.0009	43605	D4160	"Multiple-Wait S-OP" Improve
HOSTNSP	32.1.0010	43606	P1145	Set Datacount at I/O Complete
HOSTNSP	32.1.0011	43607	P1146	ADD STATION/ADD LINE Correctio
HOSTNSP	32.1.0012	43608	P1147	Remove IOCB Correction
HOSTNSP	32.1.0013	43609	P1148	ZAP Process Correction
HOSTNSP	32.1.0014	43610	P1149	Remove Group to ZAP Line Proce
HOSTNSP	32.1.0015	43611	P1150	"Test-Op" Line Count Correctio
HOSTNSP	32.1.0016	44187	D4223	I/O Initiation Address Saved
HOSTNSP	32.1.0017	44188	P1352	"STATION.OUTPUTCOUNT"
HOSTNSP	32.1.0018	44189	P1353	"ABNORMAL-TERMINATE" RESULT Me
HOSTNSP	32.1.0019	44190	P1354	"IOCB.DATACOUNT"
HOSTNSP	32.1.0020	44191	D4224	Editor Process Initiation Acce
HOSTNSP	32.1.0021	44192	P1355	"LINE-NOT-READY" RESULT Messag
HOSTNSP	32.1.0022	44193	P1356	INDEXNAME "S-Op"
HOSTNSP	32.1.0023	44194	P1357	"LIBERATE-TO-CORRECTLY-QUEUE-W
HOSTNSP	32.1.0024	44195	D4225	Save MLIP Hardware Error Flag
HOSTNSP	32.1.0025	43612	P4265	Line Process Abnormal Terminat
IN-OUTPUT	32.1.1328*	42386	P3815	Remote Backup Files
IN-OUTPUT	32.1.1333	42778	P1013	Fix AREALENGTH File Attribute
IN-OUTPUT	32.1.1339	40319	D4105	MAXSUBFILES Attribute
IN-OUTPUT	32.1.1446	43128	P1065	HOSTNAME of Unassigned File
IN-OUTPUT	32.1.1459*	43199	P1170	PROTECTED Vs. REWIND
IN-OUTPUT	32.1.1461*	43198	P1197	SPO Vs. UNITNO
IN-OUTPUT	32.1.1483*	43193	P1171	Reelswitch Vs. Use
IN-OUTPUT	32.1.1540	43467	D4244	BCL Files
IN-OUTPUT	32.1.1681	43979	D4228	Optional Vs. Direct
IN-OUTPUT	32.1.1846	44534	P3321	Datacom Direct I/O
INQ	32.1.0046	43046	P1077	Boolean Key Data
INQ	32.1.0047	42785	P1080	DISPLAY Formats Output in HEAD
INQ	32.1.0048*	43054	P1069	INVALID OP
INQ	32.1.0049*	43064	P1100	Selection Expression Improve
INQ	32.1.0050*	43217	P1092	SEG ARRAY
INQ	32.1.0051	43067	P1181	Save Prompt for Temporary Sets
INQ	32.1.0052*	43233	P1129	Record Function Information No
INQ	32.1.0053*	43246	P1224	Invalid Addresses in Sort Tags
INQ	32.1.0054*	43254	P1225	Report Title Too Large
INQ	32.1.0055*	43255	P1221	Subscript Value Limit
INQ	32.1.0056*	43256	P1220	Modified Bit for AA List Sorts
INQ	32.1.0057	43263	P1309	Minus Sign on "REAL(SM,N)" lte
INQ	32.1.0058	43262	P1310	Use, Enforce Correct Item Fiel
INQ	32.1.0059	43268	P1321	"SHOW ALL OF <data set name>"
INQ	32.1.0060	43963	P3195	Trailing Blanks on Multistatem
INTERFACE	32.1.0015	43575	P1377	INVALID INDEX
IOINTERACT	32.1.0001	43534	P1274	Remove BCL Constructs
IOTEST	32.1.0001	43534	P1274	Remove BCL Constructs
IXREF	32.1.0006	32937	D4166	Allow Underscores in Identifie
IXREF	32.1.0007	43985	P1370	Wrong Sequence Numbers for Ali
JOBFORMAT	32.1.0019*	42179	P3689	Incorrect Compilation Date
JOBFORMAT	32.1.0020	42093	D4103	Timestamp at BOT is Recorded i
JOBFORMAT	32.1.0022	42539	D4104	New Words in CLOSE Log Record
JOBFORMAT	32.1.0023	42839	D4273	BNA Log Formats
JOBFORMAT	32.1.0024	42839	D4273	BNA Log Formats
JOBFORMAT	32.1.0026	43495	P1240	Implementation of Read Extende
JOBFORMAT	32.1.0028	42839	D4273	BNA Log Formats
JOBFORMAT	32.1.0029	42838	D4272	Interval Logging for Files
JOBFORMAT	32.1.0031*	44051	P1294	Print Unit Number of Pack
LNG-ALGLSORT	32.1.0001	43831	P3276	Print Month Properly
LOADDUMP	32.1.0007*	43249	P1222	Incorrect Code Generated
LOADDUMP	32.1.0008*	43240	P1223	Incorrect Source Code
LOADDUMP	32.1.0009	43578	P1378	Incorrect FD Generated
LOADDUMP	32.1.0010	43578	P1378	Incorrect FD Generated
LOADER	32.1.0041	44056	P1297	Diskcycle
LOADER	32.1.0042	44082	P1365	Setting of Options
LOGANALY	32.1.0015	42840	D4274	BNA LOGANALYZER Changes
LOGANALY	32.1.0016	42840	D4274	BNA LOGANALYZER Changes
LOGANALY	32.1.0017	43495	P1240	Implementation of Read Extende
LOGANALY	32.1.0019	42840	D4274	BNA LOGANALYZER Changes
LOGANALY	32.1.0020	43202	D4199	Interval Logging for Files
LOGGER	32.1.0010*	42186	D3636	SEG ARRAY when Totaling, Avera
LOGGER	32.1.0011*	42185	P3766	Break on JOBQUEUEDTIME Causes
LOGGER	32.1.0012*	42459	P3767	LOGGER Mixes ORGUNIT
LOGGER	32.1.0013*	42179	P3689	Incorrect Compilation Date
LOGGER	32.1.0014*	42899	P1005	Update YTD File with Multiple
LTTABLEGEN	32.1.0001	42380	P1010	Complete TRRAINTABLES
MAKEUSER	32.1.0006	32940	P1102	ACCESSCODELIST
MAKEUSER	32.1.0007*	42946	P1232	SET LIST Problems
MCP	32.1.1189*	42180	P3687	BDNAME SEG ARRAY Fault
MCP	32.1.1203	42378	P1038	Miscellaneous Corrections
MCP	32.1.1209	42093	D4103	Timestamp at BOT is Recorded i
MCP	32.1.1212*	42178	P3710	DMSCLOSE Vs. CONTROLLER
MCP	32.1.1214*	42179	P3689	Incorrect Compilation Date
MCP	32.1.1217	42379	P1030	OPEN vs BACKUP

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

PAC 11/81

B5000/B6000/B7000 SERIES PATCH TABLE (PATCH CLOSING FTNS MARKED WITH **
SOFTWARE PATCH PRI NOTE DESCRIPTION

SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
MCP	32.1.1220	42382	P1011	READALABEL Codestrng
MCP	32.1.1235	42380	P1010	Complete TRRAINTABLES
MCP	32.1.1245	41441	P1126	Restart Vs. Missing Disk, Pack
MCP	32.1.1288*	42538	P3786	STANDARDTODISPLAY Decoding
MCP	32.1.1289	42539	D4104	New Words in CLOSE Log Record
MCP	32.1.1304	42746	P3818	Allow "DCSYSTEMTABLES(6)" With
MCP	32.1.1366*	42991	P1037	DUP FAMILY NAME
MCP	32.1.1367	42865	D4170	Two New Library Parameter Type
MCP	32.1.1368*	42993	P1036	DP Vs. STATUSCHANGE
MCP	32.1.1376	42999	D4130	DMERRORMESSAGE Construct
MCP	32.1.1378	42890	P1033	Timing Hole in DOTASKINITIATIO
MCP	32.1.1381	39735	D4134	Implement Logging for BNA
MCP	32.1.1382	42891	P1016	Library Parameter INVALID OP
MCP	32.1.1406	42898	P1034	Restore Saved TEMPLATE MOM
MCP	32.1.1416	39735	D4134	Implement Logging for BNA
MCP	32.1.1417*	42902	P1008	Stack Overflow in KANGAROO
MCP	32.1.1422*	42869	P1054	SUBSPACES Attribute Erroneousl
MCP	32.1.1423	42871	P1061	Aborted Restart
MCP	32.1.1424	42872	P1055	Restart ABORT Messages
MCP	32.1.1426*	42897	P1066	READALABEL Vs. Foreign Tapes
MCP	32.1.1431	42896	P1067	PAST Continuation
MCP	32.1.1433	42908	D4132	No Room for PROGRAMDUMP
MCP	32.1.1452	42539	D4104	New Words in CLOSE Log Record
MCP	32.1.1453	42894	P3345	Releaseheader
MCP	32.1.1460	42796	P1085	DCHOLD Save-Core Buildup
MCP	32.1.1465	42917	P1084	Correct Cancelling of Nonexist
MCP	32.1.1467	42918	P1095	FLATREADER Readlock Loop
MCP	32.1.1468	43315	D4141	HOSTNSP Implementation
MCP	32.1.1496*	42932	P1286	Bad Graph Edge
MCP	32.1.1501*	42928	P1120	ADM EVENT Logged
MCP	32.1.1502	43397	D4150	ODT CANDIDATE Message
MCP	32.1.1512	43502	P1172	Remove BCL Pointers
MCP	32.1.1514	42938	P1112	Critical Block Exit and Librar
MCP	32.1.1519	43194	P1121	Short Disk I/O
MCP	32.1.1520	42797	D4149	Initialization Stop
MCP	32.1.1524*	42943	P1151	SYSTEMSTATUS Type 4 Processor
MCP	32.1.1527*	42942	P1132	SUSPENDER/WSSHERRIFF Suspension
MCP	32.1.1528*	42942	P1132	SUSPENDER/WSSHERRIFF Suspension
MCP	32.1.1529*	43189	P1133	Volume Library Vs. Rewind
MCP	32.1.1531*	42991	P1037	DUP FAMILY NAME
MCP	32.1.1535*	42947	P1134	Checkpoint/Restart
MCP	32.1.1536*	42948	P1135	Restart Abort, PROCESSKILL, PI
MCP	32.1.1554	41289	P1189	STARTUNIT Readies Units Cleanl
MCP	32.1.1559*	43631	P1174	Programdump Analyzes Interrupt
MCP	32.1.1560	43322	D4167	Standard Tape Labels Vs. New S
MCP	32.1.1566	43533	D4241	"TIME(23)" Indicates BCL Suppo
MCP	32.1.1569	43625	P1177	PTD Releaseheader Call
MCP	32.1.1570	43633	P1178	Change Support Libraries
MCP	32.1.1577*	43664	P1205	SHAREDYALL Libraries Forced t
MCP	32.1.1586*	43752	P1246	HDRVECTORLOCK
MCP	32.1.1591	43754	P1247	NEWP Vs. RLTABLEGEN
MCP	32.1.1592	43672	P1207	Backup MCP Deleted
MCP	32.1.1597*	42883	D4185	Backup and Print Queue Rebuild
MCP	32.1.1601	43756	P1208	TRAINID Vs. Subset
MCP	32.1.1605	43758	P1248	ATGRAB Fault
MCP	32.1.1609	37310	D3371	Priority Output
MCP	32.1.1613	38563	P1260	LIBUSEMCPLOCK Vs. PROCESSCHANG
MCP	32.1.1614	43976	P1262	ALIEN I/O
MCP	32.1.1616	39735	D4134	Implement Logging for BNA
MCP	32.1.1621*	41594	P1299	Matrix Inversion
MCP	32.1.1630*	43706	P1264	GETSTATUS INVALID OP
MCP	32.1.1632	43826	P1267	COBOL BCL User Labels Vs. B590
MCP	32.1.1633	43825	P1268	User Define BCL Tape Labels Vs
MCP	32.1.1634*	43708	P1265	ODT Marked as NOTREADY
MCP	32.1.1651	43727	D4198	B5900 BCL Message
MCP	32.1.1654*	43741	P1288	COBOL Print Files not Released
MCP	32.1.1656	43728	P1317	User Vs. MCP Fault
MCP	32.1.1659*	43745	P1290	Box Number for DBS Stacks
MCP	32.1.1661	43202	D4199	Interval Logging for Files
MCP	32.1.1668	41159	P3448	Checkpoint/Restart for Program
MCP	32.1.1669*	44048	P1296	Bad Restart after ENDF
MCP	32.1.1674	44074	P1318	GETSTATUS LINKF Correction
MCP	32.1.1675	44072	P1319	Library Parameter Type
MCP	32.1.1682	35701	D3265	Industry Compatible Mini Disk
MCP	32.1.1684	44075	P1324	System Hang
MCP	32.1.1701	43933	D4245	LT ODT Message Deimplemented
MCP	32.1.1702	44084	D4220	"Pre-Mark 31" Codefiles on B59
MCP	32.1.1704	43937	D4221	TRAINID Field in UNITTABLE
MCP	32.1.1718	35701	D3265	Industry Compatible Mini Disk
MCP	32.1.1720	44092	P1368	SWAPPER Deadlock
MCP	32.1.1724	43740	P1363	Direct Resize Search
MCP	32.1.1731	44302	P1383	STACK OVERFLOW in GETAROW

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

B5000/B6000/B7000 SERIES PATCH TABLE (PATCH CLOSING FTRS MARKED WITH **)

SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
MCP	32.1.1740	44369	P3199	Move Intrinsic FAULTRECOVERY
MCP	32.1.1760	43734	P3234	Dump by BAD MOM SEARCH
MCP	32.1.1763	44091	P3235	Terminating Scheduled Librarie
MCP	32.1.1764	44315	P3241	Inheriting SUBSYSTEMID
MCP	32.1.1776	44304	P3233	SWAPPER and Data Management
MCP	32.1.1780	43726	P3253	FIBSTACK Binary Card I/O
MCP	32.1.1782	42096	D4264	Catalog Block Version Implemen
MCP	32.1.1801	44304	P3233	SWAPPER and Data Management
MCP	32.1.1820	44524	P3275	Stack Searching Error with SWA
MCP	32.1.1834	44304	P3233	SWAPPER and Data Management
MCP	32.1.1839	44550	P3340	Unit Error Statistics Always Z
MCP	32.1.1844	44553	P3319	"Core-to-Core" Reimplemented
MCP	32.1.1845	44533	P3320	Illegal Swapstate While in REA
MCP	32.1.1858	44743	P3334	Setting the SWAPPER Parameter
MCP	32.1.1861	44746	P3335	ACTIVETIME
MLIP	32.1.1440	42912	P1060	LOADHOST Digit Count
MLIP	32.1.1497	43501	P1111	COBOL, ALGOL Tape SORT
MLIP	32.1.1546	43032	P1160	Exchangeable Units
MLIP	32.1.1547	43031	P1159	Log DLP Errors
MLIP	32.1.1552	43624	P1173	RY a Saved Pack Response
MLIP	32.1.1555	43628	P1153	Checkpoint Vs. B6900 I/O
MLIP	32.1.1564	43495	P1240	Implementation of Read Extende
MLIP	32.1.1565	43494	P1175	Firmware Level Recorded on Dum
MLIP	32.1.1576	43626	P1212	Correct ODT UR
MLIP	32.1.1589	43078	P1215	Card Punching Problem
MLIP	32.1.1595	43080	P1214	System Initialization
MLIP	32.1.1596	43080	P1214	System Initialization
MLIP	32.1.1624	43082	P1263	Readying of Packs
MLIP	32.1.1637	43913	P1269	CANCELIO Corrections
MLIP	32.1.1638	43905	P1270	BLASTUNIT Improvements
MLIP	32.1.1642	43701	P1272	PATHRES Vs. Nonstatus DLP
MLIP	32.1.1644	43698	P1276	Correct Reserve Halt/Load Path
MLIP	32.1.1663	43905	P1270	BLASTUNIT Improvements
MLIP	32.1.1665	43908	P1295	Pack Power Off
MLIP	32.1.1722	43912	P1369	WFL/CONTROLCARD Loop
MLIP	32.1.1723	43721	P1364	DOIOERRORIO Returns Logical R/
MLIP	32.1.1725	43089	P1366	TAKEUNIT Vs. "FREE PK<nn>"
MLIP	32.1.1741	43090	P3203	Logical Unit Number Vs. "PB MT
MLIP	32.1.1742	44109	P3209	UR/UA Maintenance Capabilities
MLIP	32.1.1747	43914	P3237	TEST/WAIT Bad Event Reference
MLIP	32.1.1748	43916	P3238	BLASTUNIT, MYIOSONLY
MLIP	32.1.1752	44396	P3239	Selective Clear Vs. Hung DLP
MLIP	32.1.1756	43724	P3240	"UA SC <unit no>" Corrected
MLIP	32.1.1774	44477	P3248	Correct HOSTNSP Results
MLIP	32.1.1779	43725	P3254	BINARY EOF Sensing
MLIP	32.1.1872	44481	P3341	IOERRORTYPE Vs. "EOT/EOF"
MLIP	32.1.1876	44483	P3342	BLASTUNIT Avoids Outboard Path
NDL	32.1.0031*	43102	P1003	Error for Different Line Contr
NDL	32.1.0033	37310	D3371	Priority Output
NDL	32.1.0034	42682	P1254	Recursive Defines
NDL	32.1.0035	42683	P1255	Allow String Parameters to Def
NDL	32.1.0037	42681	P1307	Setting of User Options
NDL	32.1.0038	44018	P1332	Alpha Labels
NDL	32.1.0039	42681	P1307	Setting of User Options
NDLII	32.1.0074	41679	P1019	Detect String Size Too Large E
NDLII	32.1.0075	41680	P1020	Fix Fault with Use of Special
NDLII	32.1.0076	43232	P1158	Synchronous Line Diagnostic Ra
NDLII	32.1.0077	42615	P1235	Line Attribute Offset
NDLII	32.1.0078	43280	P1282	WAITFORIDLE Implemented
NDLII	32.1.0079	43264	P1278	Events, Interlocks in Include
NDLII	32.1.0080	43561	P1279	Fault in Group INCLUDE Declara
NDLII	32.1.0081	43265	P1280	"INVALID DESCRIPTOR ADDRESS (U
NDLII	32.1.0082	43564	P1281	Invalid Extension Skeletons
NDLII	32.1.0083	43574	P1347	Algorithm Translate Table List
NDLII	32.1.0084	43781	P3222	Duplicate Adaptor Assignment
NDLII	32.1.0086	43896	P3223	SYSTEMWAIT Code
NDLII	32.1.0088	43888	P3256	Control Block Arrays
NDLII	32.1.0089	44568	P3285	IDSTORE Size Increased
NDLII	32.1.0090	43295	P3310	COPYSTRING with <cb variable>
NDLII	32.1.0091	43296	P3311	Transmit Statement
NEWP	32.1.0130	42103	D4115	Syntax Deleted
NEWP	32.1.0132	42659	P1022	Invalid Index for CASE Express
NEWP	32.1.0133	42662	P1023	XREF of Procedures in MODULE E
NEWP	32.1.0135	43278	P1106	String Parameters to STOP
NEWP	32.1.0138	32937	D4166	Allow Underscore in Identifie
NEWP	32.1.0139	43437	P1167	Make Scalerights Return an Int
NEWP	32.1.0142	43437	P1167	Make Scalerights Return an Int
NSPDUMPANALY	32.1.0003	43851	D4175	Recognize "Mark 7 NSP" Firmwar
NSPDUMPANALY	32.1.0004	43852	D4176	Add Fault Tolerance
NSPDUMPANALY	32.1.0005	42137	P1204	Eliminate Firmware Code
NSPDUMPANALY	32.1.0006	44462	P3251	Different Level Firmware Handl
PATCH	32.1.0015	43465	P1139	No BCL on EBCDIC System

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
PLI	32.1.0074	44179	P3224	Incorrect Syntax Error
PLISUPP	32.1.0032	43490	P1103	ISAM, "END-OF-FILE"
PLISUPP	32.1.0033*	43457	P1109	ISAM DELETE, Large FINE, COARS
PLISUPP	32.1.0036*	43479	P1233	GET LIST, GET DATA
PLISUPP	32.1.0037	43773	P1256	Bad Parameters to DELTA
PLISUPP	32.1.0038	44035	P1304	INVALID INDEX in Condition Inf
PLISUPP	32.1.0039*	43480	P1305	PUT EDIT, Bit Formats
PLISUPP	32.1.0040	43488	P1330	ISAM Hardware Errors
PLISUPP	32.1.0041	44180	P3225	ISAM, DELETE
PLISUPP	32.1.0042	44178	P3226	ISAM, COBOL START Statement
PLISUPP	32.1.0043	44174	P3259	ISAM, Premature End of File
PRINTAUDIT	32.1.0010	44828	P3343	Misleading "IOERRORs"
PRINTCOPY	32.1.0001*	42529	P3765	HARDCOPY Files Left on Disk
PROPERTIES	32.1.0042	43954	P1379	Number of Names in DMERRORS Ar
PTDMCP	32.1.1640	43700	D4197	Improved PTD Path Selection
PTDMCP	32.1.1641	43700	D4197	Improved PTD Path Selection
PTDMCP	32.1.1710	43720	P1339	PTD Path Deassignment, I/O Deb
PTDMCP	32.1.1714	43719	P1336	PTD SPO Handling
RECOVERY	32.1.0054	42717	P3821	Rebuild/Reconstruct Abort
RECOVERY	32.1.0056*	43038	P1041	Infinite Loop in RECOVERY
RECOVERY	32.1.0057*	43061	P1091	Write Error
RECOVERY	32.1.0059	44560	P3257	Rebuild Across Reorganization
RECOVERY	32.1.0061	44575	P3329	DATARECOVERY Fails to Reconstr
RECOVERY	32.1.0062	44575	P3329	DATARECOVERY Fails to Reconstr
RECOVERY	32.1.0063	44828	P3343	Misleading "IOERRORs"
REMOVELIB	32.1.0013	44912	P3339	Error Result not Returned
REORG	32.1.0030*	43231	P1128	Checksum on Reorganized Global
REORG	32.1.0031*	43235	P1155	Incorrect BCW on Embedded orde
REORG	32.1.0032	43565	P1291	Corruption of Data
REORG	32.1.0034	43577	P1380	Checksum Error on Interim Tape
REORG	32.1.0035	43582	P3197	Generating Multiple Bitvectors
REORG	32.1.0036	44560	P3257	Rebuild Across Reorganization
REORG	32.1.0037	44828	P3343	Misleading "IOERRORs"
RJE	32.1.0088	42515	P1024	Session Initialization
RJE	32.1.0089	42516	P1025	Terminal Transfer Compatibilit
RJE	32.1.0090	42514	P1026	RJE HOST to HOST Link Losing F
RJE	32.1.0091	42517	P1027	File Transfer Valid Char List
RJE	32.1.0093	42472	D4117	02 and 04 Control Message Upda
RJE	32.1.0094	42874	D4118	CLEAR <LSN> vs Virtual Station
RJE	32.1.0095*	42877	P1110	Usercode Word Not Zeroed Out
RJE	32.1.0096*	42880	P1168	Contiguous Blanks
RJE	32.1.0097*	42881	P1169	FTS Record Size, Block Size
RJE	32.1.0098	43489	D4163	Remove BCL Constructs
RJE	32.1.0099	42882	P1191	Station ID Reconfiguration
RJE	32.1.0100*	42883	D4185	Backup and Print Queue Rebuild
RJE	32.1.0102	42886	P1236	OFFLINE Request Blanks Station
RJE	32.1.0103	42887	P1237	RUNTIMEOPTIONS Storage in Link
RJE	32.1.0104*	43673	P1234	"CP-9500 (B800)" Appears as Pe
RJE	32.1.0105	43675	P1349	"FS1, FS2" Records Ignored
RJE	32.1.0106	43676	P1348	EBCDIC to ASCII for B800 Syste
RJE	32.1.0107	43677	P3227	SEG ARRAY Error
RJE	32.1.0108	43681	P3230	Connect Reply for Inactive Sta
RLTABLEGEN	32.1.0002	43754	P1247	NEWP Vs. RLTABLEGEN
RSMCP	32.1.1721	44134	P3255	RSPIO Parameter
SCRMCP	32.1.1567*	38341	P1195	SCR TESTII Recognizes Pack
SCRMCP	32.1.1568*	38336	P1196	Tags Match TEST7
SCTABLEGEN	32.1.0025	43174	D4136	HOSTNSP Handling
SCTABLEGEN	32.1.0027	43397	D4150	ODT CANDIDATE Message
SCTABLEGEN	32.1.0031	43202	D4199	Interval Logging for Files
SCTABLEGEN	32.1.0032	43933	D4245	LT ODT Message Deimplemented
SC01	32.1.0001	44594	P3274	Syntax Errors Corrected
SORTMCP	32.1.1447	43271	D4133	Sort Changes Value of NEWFILE
SORTMCP	32.1.1495	43271	D4133	Sort Changes Value of NEWFILE
SOURCENDL	32.1.0006*	42878	D4164	File Transfer Floods Memory
SOURCENDL	32.1.0009	44001	P1257	RJE Host Link Reestablishment
SOURCENDLII	32.1.0026	43139	P1028	Full Duplex Line Handling Corr
SOURCENDLII	32.1.0027	44002	P1263	Eliminate STRUCTURE PROTECT Fa
SOURCENDLII	32.1.0028	44460	P3228	TTY/FULLDUPLEXTTY Page, Break
SOURCENDLII	32.1.0029	44461	P3229	Upper/Lower Case Handling, Edi
SUSPENDER	32.1.0001*	42847	P1006	Restore SHOW Command
SUSPENDER	32.1.0002*	42851	P1007	SUSPENDER on Monolithic System
TABLEGEN	32.1.0002	43466	D4269	Cannot Run on "Non-BCL" System
TFL	32.1.0014*	43039	P1116	Duplicated Option
TFL	32.1.0015	43266	P1312	Detection of Syntax Errors
TRINTERFACE	32.1.0005	44039	P1313	Bad Timestamps
TRPROPERTY	32.1.0010*	42703	D4144	Increased Maximum Number of Ac
TRUTILITY	32.1.0010*	43072	P1090	INVALID INDEX
TRUTILITY	32.1.0011*	43234	P1209	Search Function Fault
TRUTILITY	32.1.0012*	43072	P1090	INVALID INDEX
TRUTILITY	32.1.0013	44912	P3339	Error Result not Returned
UDSTRCTTAB	32.1.0006	43922	D4192	IDENTITY
UDSTRCTTAB	32.1.0007	43732	D4259	SYSTEMUSER Redefined

B5000/B6000/B7000 SERIES FIELD RELEASE LEVEL 32 (UPDATE)

B5000/B6000/B7000 SERIES PATCH TABLE (PATCH CLOSING FTRS MARKED WITH *)				
SOFTWARE	PATCH	PRI	NOTE	DESCRIPTION
UTIL	32.1.0068*	43041	P1042	Available Block Numbers Printe
UTIL	32.1.0069	43049	P1081	Syntax Error when Restarting U
UTIL	32.1.0070*	43048	P1070	REBUILD Fails on File Disconti
UTIL	32.1.0071*	43050	P1071	Wrong Numbers of Workers Start
UTIL	32.1.0072	43953	P1381	Data Base Invocation
UTIL	32.1.0075	44584	P3326	UTILITY Recover May Corrupt Da
UTIL	32.1.0077	44828	P3343	Misleading "IOERRORs"
UTILOADER	32.1.0029	43084	P1261	UTILOADER VIA Syntax Vs. MCP
UTILOADER	32.1.0030	43710	P1266	Save Space
WFL	32.1.0044*	41871	P3709	Names with Hyphens and Undersc
WFL	32.1.0051	41681	P1087	Eliminate BCL Constructs from
WFL	32.1.0053	41682	P1083	By Reference TASK for COMPILE
WFL	32.1.0055*	43537	P1243	"\$INCLUDE * <filename>"
WFL	32.1.0056*	43538	P1244	Multiple INCLUDE Statements
WFL	32.1.0058*	43539	P1242	USERDATA CLASS=4"80"

