## CONTENTS

### Preface

### Chapter 1  Product Overview

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is GDDR?</td>
<td>20</td>
</tr>
<tr>
<td>Major features</td>
<td>20</td>
</tr>
<tr>
<td>Situational awareness</td>
<td>20</td>
</tr>
<tr>
<td>Survivor recognition</td>
<td>21</td>
</tr>
<tr>
<td>Restart coordination</td>
<td>21</td>
</tr>
<tr>
<td>Additional capabilities</td>
<td>21</td>
</tr>
<tr>
<td>Types of environment</td>
<td>21</td>
</tr>
<tr>
<td>Supported business continuity configurations</td>
<td>22</td>
</tr>
<tr>
<td>SRDF/A configuration</td>
<td>23</td>
</tr>
<tr>
<td>GDDR fundamentals</td>
<td>24</td>
</tr>
<tr>
<td>Control systems</td>
<td>24</td>
</tr>
<tr>
<td>Workload location</td>
<td>25</td>
</tr>
<tr>
<td>Managed workloads</td>
<td>25</td>
</tr>
<tr>
<td>External workloads</td>
<td>25</td>
</tr>
<tr>
<td>Excluded systems</td>
<td>26</td>
</tr>
<tr>
<td>HMC-only systems</td>
<td>26</td>
</tr>
<tr>
<td>HMC Bypass feature</td>
<td>26</td>
</tr>
<tr>
<td>GDDR processes</td>
<td>26</td>
</tr>
<tr>
<td>GDDR components</td>
<td>27</td>
</tr>
<tr>
<td>Parameters</td>
<td>27</td>
</tr>
<tr>
<td>User interface</td>
<td>27</td>
</tr>
<tr>
<td>Events</td>
<td>27</td>
</tr>
<tr>
<td>Monitors</td>
<td>28</td>
</tr>
<tr>
<td>Message interception rules</td>
<td>28</td>
</tr>
<tr>
<td>GDDR-managed system types</td>
<td>29</td>
</tr>
<tr>
<td>Production or test systems and their optional contingency systems</td>
<td>29</td>
</tr>
<tr>
<td>Excluded systems</td>
<td>30</td>
</tr>
<tr>
<td>HMC only systems</td>
<td>30</td>
</tr>
<tr>
<td>External workload systems</td>
<td>30</td>
</tr>
<tr>
<td>Coupling facility LPARs</td>
<td>30</td>
</tr>
<tr>
<td>GDDR scripts</td>
<td>31</td>
</tr>
<tr>
<td>Planned event management</td>
<td>31</td>
</tr>
<tr>
<td>Test event management</td>
<td>32</td>
</tr>
<tr>
<td>Unplanned event management</td>
<td>32</td>
</tr>
<tr>
<td>Resumption of replication after SRDF link outages</td>
<td>33</td>
</tr>
<tr>
<td>Special actions</td>
<td>33</td>
</tr>
<tr>
<td>Parameter Load wizard: Telling GDDR what to manage</td>
<td>33</td>
</tr>
</tbody>
</table>

### Chapter 2  Installing GDDR

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware and software requirements</td>
<td>36</td>
</tr>
<tr>
<td>Mainframe environment requirements</td>
<td>36</td>
</tr>
<tr>
<td>Minimum software requirements</td>
<td>36</td>
</tr>
<tr>
<td>Minimum hardware requirements</td>
<td>37</td>
</tr>
<tr>
<td>DASD support</td>
<td>37</td>
</tr>
<tr>
<td>Required installation information</td>
<td>38</td>
</tr>
</tbody>
</table>
Chapter 3  Integrating GDDR

Integration tasks ................................................................. 48
Update system parameter files ................................................... 48
  SYS1.PARMLIB(IKJTSOxx) customization ...................................... 48
  TSO logon customization .......................................................... 50
  APF authorization ...................................................................... 50
  LINKLIB and REXX parameter file installation .............................. 50
Configure BCPIi ....................................................................... 51
  Security changes required to use BCPIi........................................ 52
  SYS1.PARMLIB changes required for BCPIi ................................. 53
Create parameter members for SRDF Host Component on C-Systems 53
Edit SCF initialization parameters ............................................. 54
Specify GDDR security ............................................................. 54
  Define GDDR RACF functional groups ....................................... 54
  Summary of RACF permissions .................................................. 55
  Define GDDR user interface security ......................................... 57
  Authorize RACF for HMC LPAR actions ................................. 58
  Authorize the Mainframe Enablers EMCSAFI security interface .... 59
  Verify the module and RACF-protected resource authorization ... 60
Define GDDR datasets ................................................................ 64
  Define global variable datasets ............................................... 64
  Allocate the parameter management datasets .......................... 64
Install GDDR started procedures .................................................. 65
  Install GDDR Licensed Feature Code ...................................... 66
Customize GDDRMAIN parameters .............................................. 66
  Specify GDDR data in virtual dataset names .............................. 72
  Modify worker parameters ...................................................... 73
Configure GDDR ...................................................................... 79
  Step 1: Update your personal GDDR ISPF profile ....................... 79
  Step 2: Define initial site, system, storage, utility, and GDDR option parameters .......................................................... 80
  Step 3: Modify GDDR user exits (optional) ............................... 83
Optional configuration features ................................................... 84
  Configure GDDR support for external devices ......................... 84
  CPC Recovery - LPAR Recovery features ................................. 84

Chapter 4  Performing GDDRMAIN Operations

GDDRMAIN subtasks ............................................................. 88
GDDRMAIN dependent address spaces ........................................ 88
GDDRMAIN console commands ............................................... 89
  Stop command (P) ................................................................. 89
  Modify commands (F) ........................................................... 89
GDDRMAIN EXEC parameters .................................................. 96
Chapter 5  Using GDDR Online Facilities

Primary Options menu ................................................................. 106
Option P: Profile—Update Personal GDDR ISPF Profile ................. 108
  Change your GDDR profile ...................................................... 108
Option M: Maintenance—GDDR Setup and Maintenance ............... 109
  Option P: Manage GDDR Parameters ..................................... 110
  Option D: Message, Debug and Trace Options ......................... 188
  Option Q: Manage GDDR Internal Command Queue ................. 189
  Option H: Perform HMC Discovery ......................................... 190
  Option R: Refresh GDDR Message Table .................................. 191
  Option S: Manage GDDR System Variables ............................. 192
  Option T: Transfer Master C-System ...................................... 194
Option G: GDDR Config—View GDDR Configuration ..................... 195
Option C: Checkup—Perform Pre-script Checkup ......................... 196
  Health Check monitoring ...................................................... 201
  GDDR Event monitoring exception notification ....................... 202
  Additional pre-script environment checks ............................... 207
Option S: Scripts — Run GDDR Scripts ...................................... 208
Option T: Timing—View GDDR Script Statistics ............................ 213
Option A: Actions—Perform GDDR Actions ............................... 215
  Option H: Perform HMC Discovery ........................................ 215
  Option L: Perform HMC LPAR Actions ................................. 216
  Option CBU: Perform HMC CBU actions ............................... 224
  Option S: Manage Couple Datasets ....................................... 224
  Option CF: Manage CF Structures ........................................ 226
Option ST: eSTEM - Run EMC Started Task Execution Manager ........ 227
  Adding a new procedure ..................................................... 227
  Adding tasks to the procedure ............................................. 228
  Executing a procedure locally .............................................. 230
  Executing a procedure remotely ......................................... 231
  Executing a procedure within a procedure ............................ 234

Chapter 6  Using GDDR Utilities

Automated Configuration Discovery for DASD (GDDRACDD) ............ 240
  Operating modes ............................................................... 240
  Prerequisites ................................................................. 240
  Procedure ................................................................. 241
## Chapter 7  Performing Script Operations

| Running scripts | 330 |
| Call overrides | 333 |
| Recovering from script errors | 335 |
| Rerunning a script | 336 |
| WTOR messages | 337 |
| Planned script operations | 337 |
| Automated Configuration Check - DASD - GDDRPCCD | 337 |
| Abandon Site DC1 (site swap) - GDD2P17A | 338 |
| Abandon Secondary site (DC2) - GDDRPA60 | 338 |
| Restart production at DC2 after site swap - GDD2P18A | 338 |
| Test operations | 339 |
| Perform test IPL from BCVs at DC3 - GDD2P01A | 339 |
| Resume after test IPL from BCVs at DC3 - GDD2P02A | 339 |
| Unplanned script operations | 339 |
| Recover after loss of DC1 (RDR) - GDD2U12A | 339 |
| Resume replication after loss of DC1 - GDD2PA0A | 340 |
| Resumption operations | 340 |
| Resume SRDF/A after link loss - GDDRPA29 (internally known as GDDRP29) | 340 |
| Reclaim Secondary site (DC2) - GDDRPA65 | 340 |
| Special operations | 340 |
| Transfer Master C System to <DCx> - GDDRPXMC | 340 |
| Global Variable Backup - GDDRPGVB | 341 |
| Move systems to alternate CPC - GDDRMPCP | 341 |

## Chapter 8  Handling Unplanned Events

| Introduction | 344 |
| Regional disaster operations | 344 |
| Confirm loss of DC1 | 344 |
| System failure operations | 345 |
| EMC GDDR C-System failure | 345 |
| Production system failure | 346 |
| GDDR Master Function transfer | 347 |

## Chapter 9  Performing Maintenance Procedures

| Setting up a new GDDR C-System | 350 |
| Renaming an existing GDDR C-System | 350 |
| Changing the GDDR C-System or GDDR managed system IP address | 351 |
| Changing the GDDR C-System or GDDR managed system IP port | 351 |
| Adding a new system or sysplex to GDDR | 352 |
| Changing the MSC group name | 353 |
| Adding new RDF groups to GDDR | 354 |
| Adding new devices to GDDR | 357 |
| Removing an RDF group from GDDR control | 358 |
| Removing devices from GDDR control | 359 |
| Removing a system or a sysplex from GDDR | 359 |
| Special cases | 361 |
| Page datasets | 361 |
| Non-LOGR couple datasets | 361 |
| Standalone dump considerations | 362 |
Chapter 10 Troubleshooting
Detecting and resolving problems ............................................................. 364
Using the GDDRXCMD batch utility ............................................................ 364
To print the current queue ................................................................... 364
To clear the current queue ................................................................... 364
Troubleshooting ....................................................................................... 364

Appendix A User Exits
User exit programming considerations ...................................................... 366
Sample procedure to use interpreted REXX user exits ...................... 366
Sample procedure to use CA-OPS/MVS compiled REXX user exits .... 366
Built-in routines available to exits ....................................................... 366
Exit specifications ..................................................................................... 367
GDDRUX01 .......................................................................................... 367
GDDRUX02 .......................................................................................... 367
GDDRUX03 .......................................................................................... 368
GDDRUX04 .......................................................................................... 368

Appendix B BCPii Interface
Introduction to the BCPii interface............................................................. 372
BCPii HMC Networking capabilities and requirements ...................... 372
GDDMPARM CPC parameter entries ....................................................... 373
CPC parameter requirements ............................................................... 373
CPC parameter SE/HMC control requirements .................................. 374
GDDRBPCI BCPii test job for B-Systems .................................................. 376
<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 SRDF/A environment ......................................................................</td>
<td>23</td>
</tr>
<tr>
<td>2 EMC JCL customization utility ..................................................</td>
<td>42</td>
</tr>
<tr>
<td>3 EMC JCL customization utility completed panel ............................</td>
<td>44</td>
</tr>
<tr>
<td>4 Validation of GDDR$ADM group access by facility ..........................</td>
<td>63</td>
</tr>
<tr>
<td>5 Validation of specific calls from GDDR modules ...........................</td>
<td>63</td>
</tr>
<tr>
<td>6 Primary Options menu ....................................................................</td>
<td>106</td>
</tr>
<tr>
<td>7 Change GDDR ISPF Profile Variable Values panel ............................</td>
<td>108</td>
</tr>
<tr>
<td>8 Setup and Maintenance Menu .......................................................</td>
<td>109</td>
</tr>
<tr>
<td>9 Parameter Management Options Menu .............................................</td>
<td>110</td>
</tr>
<tr>
<td>10 GDDR parameter management .....................................................</td>
<td>112</td>
</tr>
<tr>
<td>11 Parameter Load wizard work flow — step 1 ...................................</td>
<td>114</td>
</tr>
<tr>
<td>12 Parameter Load wizard work flow — step 2 ...................................</td>
<td>114</td>
</tr>
<tr>
<td>13 Parameter Load wizard work flow — step 3 ...................................</td>
<td>115</td>
</tr>
<tr>
<td>14 Parameter Load wizard work flow — step 4 ...................................</td>
<td>115</td>
</tr>
<tr>
<td>15 Parameter Load wizard work flow — step 5 ...................................</td>
<td>116</td>
</tr>
<tr>
<td>16 Parameter Load wizard work flow — step 6 ...................................</td>
<td>117</td>
</tr>
<tr>
<td>17 Select Parameter Input Dataset for parameter review ....................</td>
<td>118</td>
</tr>
<tr>
<td>18 Reviewer’s version of the Parameter Management Options Menu ............</td>
<td>118</td>
</tr>
<tr>
<td>19 Manage GDDR Parameter Backups panel .........................................</td>
<td>119</td>
</tr>
<tr>
<td>20 Select Dataset for GDDR Parameter Backup ....................................</td>
<td>121</td>
</tr>
<tr>
<td>21 GDDRACDD - Automated Configuration Discovery for DASD panel ............</td>
<td>123</td>
</tr>
<tr>
<td>22 Dataset name to Backup panel ...................................................</td>
<td>124</td>
</tr>
<tr>
<td>23 Select Parameter Input Dataset panel .........................................</td>
<td>124</td>
</tr>
<tr>
<td>24 Prepare Work Dataset for Parameter Load confirmation panel ............</td>
<td>126</td>
</tr>
<tr>
<td>25 Prepare Work Dataset status panel .............................................</td>
<td>127</td>
</tr>
<tr>
<td>26 Parameter Management Options Menu with parameter load input selection</td>
<td>127</td>
</tr>
<tr>
<td>27 Select Parameter Input Dataset panel with Edit-in-Progress serialization lock - User 1</td>
<td>128</td>
</tr>
<tr>
<td>28 Select Parameter Input Dataset panel with Edit-in-Progress FORCE authorization - User 2</td>
<td>129</td>
</tr>
<tr>
<td>29 GDDR Parameter Wizard panel after FORCE of Edit-in-Progress serialization lock - User 1</td>
<td>130</td>
</tr>
<tr>
<td>30 Select Parameter Input Dataset panel after FORCE of Edit-in-Progress serialization lock - User 1</td>
<td>130</td>
</tr>
<tr>
<td>31 Define Configuration Basics panel .............................................</td>
<td>131</td>
</tr>
<tr>
<td>32 Define GDDR Configuration Features ...........................................</td>
<td>132</td>
</tr>
<tr>
<td>33 Define Configuration Features panel, configuration-specific options</td>
<td>133</td>
</tr>
<tr>
<td>34 Define C-Systems panel ............................................................</td>
<td>134</td>
</tr>
<tr>
<td>35 Define GDDR Datasets panel .......................................................</td>
<td>136</td>
</tr>
<tr>
<td>36 Define Site Roles and Groups panel ............................................</td>
<td>137</td>
</tr>
<tr>
<td>37 Define Data Storage Objects panel .............................................</td>
<td>138</td>
</tr>
<tr>
<td>38 Define SRDF Device Ranges panel ..............................................</td>
<td>139</td>
</tr>
<tr>
<td>39 Define TimeFinder Device Ranges panel (with STDBCV parameters) ....</td>
<td>141</td>
</tr>
<tr>
<td>40 Define TimeFinder Device Ranges panel (no STDBCV parameters) .......</td>
<td>141</td>
</tr>
<tr>
<td>41 Define GDDR Snapshot Names .....................................................</td>
<td>142</td>
</tr>
<tr>
<td>42 Define SDDF Clean Utility Gatekeepers panel ..................................</td>
<td>143</td>
</tr>
<tr>
<td>43 Define Dlm Devices panel ..........................................................</td>
<td>143</td>
</tr>
<tr>
<td>44 Define Dlm Devices - ACP Details panel .......................................</td>
<td>144</td>
</tr>
<tr>
<td>45 Define Host Objects panel ..........................................................</td>
<td>146</td>
</tr>
<tr>
<td>46 Define Managed Systems panel ....................................................</td>
<td>147</td>
</tr>
<tr>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Define System IPL Priorities panel</td>
<td>149</td>
</tr>
<tr>
<td>Define Managed LPARs panel</td>
<td>151</td>
</tr>
<tr>
<td>Define System Recovery Attributes panel</td>
<td>153</td>
</tr>
<tr>
<td>Define Managed CPCs panel</td>
<td>155</td>
</tr>
<tr>
<td>Define IPL Parameters panel</td>
<td>156</td>
</tr>
<tr>
<td>Define HMC Load Activation Profiles</td>
<td>159</td>
</tr>
<tr>
<td>Define Managed HMCs panel</td>
<td>159</td>
</tr>
<tr>
<td>Define HMC Community Names panel</td>
<td>160</td>
</tr>
<tr>
<td>Define Managed Couple Datasets panel 1 of 2</td>
<td>161</td>
</tr>
<tr>
<td>Define Managed Couple Datasets panel 2 of 2</td>
<td>162</td>
</tr>
<tr>
<td>Define Managed CF Structures panel 1 of 2</td>
<td>164</td>
</tr>
<tr>
<td>Define Managed CF Structures panel 2 of 2</td>
<td>164</td>
</tr>
<tr>
<td>Define External Workloads panel</td>
<td>166</td>
</tr>
<tr>
<td>Define EMC Mainframe Enablers STCs panel</td>
<td>167</td>
</tr>
<tr>
<td>Specify GDDR Options panel</td>
<td>168</td>
</tr>
<tr>
<td>Specify Default Script Call Overrides panel</td>
<td>170</td>
</tr>
<tr>
<td>Script Sysplex Options panel</td>
<td>171</td>
</tr>
<tr>
<td>Script JCL Parameters panel</td>
<td>173</td>
</tr>
<tr>
<td>Utility Parameters panel</td>
<td>175</td>
</tr>
<tr>
<td>Messaging Options panel</td>
<td>176</td>
</tr>
<tr>
<td>Specify GDDR Tuning Values panels</td>
<td>177</td>
</tr>
<tr>
<td>Define GDDR User Labels panel</td>
<td>180</td>
</tr>
<tr>
<td>Validate GDDR Parameter Set panel</td>
<td>181</td>
</tr>
<tr>
<td>Activate GDDR Parameter Set panel</td>
<td>184</td>
</tr>
<tr>
<td>Parameter Load Activation status panel 1 of 2</td>
<td>187</td>
</tr>
<tr>
<td>Parameter Load Activation status panel 2 of 2</td>
<td>187</td>
</tr>
<tr>
<td>Set Output Message Levels by Program panel</td>
<td>188</td>
</tr>
<tr>
<td>Add Program to MsgLevel/Debug/Trace List panel</td>
<td>189</td>
</tr>
<tr>
<td>Manage GDDR Internal Command Queue panel</td>
<td>189</td>
</tr>
<tr>
<td>HMC object discovery panel</td>
<td>190</td>
</tr>
<tr>
<td>HMC Discovery Results panel</td>
<td>191</td>
</tr>
<tr>
<td>Message table refresh indicator</td>
<td>191</td>
</tr>
<tr>
<td>Manage GDDR System Variables panel</td>
<td>193</td>
</tr>
<tr>
<td>Transfer Master C-System panel</td>
<td>194</td>
</tr>
<tr>
<td>View GDDR Configuration panel</td>
<td>195</td>
</tr>
<tr>
<td>Perform Health Check panel</td>
<td>196</td>
</tr>
<tr>
<td>GDDRMAIN System Details panel</td>
<td>200</td>
</tr>
<tr>
<td>Select Script to Run panel (at DC1)</td>
<td>209</td>
</tr>
<tr>
<td>Select Script to Run panel (following the Abandon Site DC1 script at DC3)</td>
<td>209</td>
</tr>
<tr>
<td>Select Script to Run panel — unavailable scripts displayed</td>
<td>210</td>
</tr>
<tr>
<td>Select Script to Run panel — unavailable scripts hidden</td>
<td>211</td>
</tr>
<tr>
<td>Script Selection for Status panel</td>
<td>213</td>
</tr>
<tr>
<td>GDDR Actions Menu</td>
<td>215</td>
</tr>
<tr>
<td>HMC object discovery panel</td>
<td>215</td>
</tr>
<tr>
<td>HMC Discovery Results panel</td>
<td>216</td>
</tr>
<tr>
<td>Perform HMC LPAR Actions panel</td>
<td>216</td>
</tr>
<tr>
<td>Perform CBU Actions panel</td>
<td>224</td>
</tr>
<tr>
<td>Manage Couple Datasets panel</td>
<td>224</td>
</tr>
<tr>
<td>Manage CF Structures panel</td>
<td>226</td>
</tr>
<tr>
<td>EMC Started Task Execution Manager Procedure Member List panel</td>
<td>227</td>
</tr>
<tr>
<td>SRDF/A GDDRCACDD parameter example</td>
<td>259</td>
</tr>
<tr>
<td>GDDR SRDF directors overview for ALL RDF groups, 1 of 2</td>
<td>282</td>
</tr>
<tr>
<td>GDDR SRDF directors overview for ALL RDF groups, 2 of 2</td>
<td>283</td>
</tr>
</tbody>
</table>
Figures

100 RDF groups by director by VMAX by site, Site: DC1 .....................................................   284
101 RDF groups by director by VMAX by site, Site: DC2 .....................................................   285
102 RDF groups by director by VMAX by site, Site: DC3 and Site UNK..............................   286
103 Specify Parameters for Initial Script Run panel ..........................................................   330
104 Specify Call Overrides panel ...................................................................................   331
105 Confirm Job Submission panel ..................................................................................   332
### TABLES

<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Mainframe environment requirements</td>
<td>36</td>
</tr>
<tr>
<td>2 Minimum hardware requirements</td>
<td>37</td>
</tr>
<tr>
<td>3 RIMLIB library contents</td>
<td>41</td>
</tr>
<tr>
<td>4 SRDF Host Component parameter members</td>
<td>54</td>
</tr>
<tr>
<td>5 RACF functional groups</td>
<td>54</td>
</tr>
<tr>
<td>6 RACF permissions</td>
<td>55</td>
</tr>
<tr>
<td>7 RACF permissions, OPERCMDS class</td>
<td>56</td>
</tr>
<tr>
<td>8 Summary of GDDR ISPF RACF permissions</td>
<td>57</td>
</tr>
<tr>
<td>9 SAMPLIB inventory</td>
<td>64</td>
</tr>
<tr>
<td>10 GDDRMAIN subtasks</td>
<td>88</td>
</tr>
<tr>
<td>11 Dependent address spaces</td>
<td>88</td>
</tr>
<tr>
<td>12 GDDRMAIN console command summary</td>
<td>89</td>
</tr>
<tr>
<td>13 Possible lock states</td>
<td>98</td>
</tr>
<tr>
<td>14 Script keywords</td>
<td>101</td>
</tr>
<tr>
<td>15 Control keywords</td>
<td>103</td>
</tr>
<tr>
<td>16 Defining Managed Couple Datasets</td>
<td>162</td>
</tr>
<tr>
<td>17 Monitored events</td>
<td>201</td>
</tr>
<tr>
<td>18 Software state analysis messages</td>
<td>202</td>
</tr>
<tr>
<td>19 Script generation status messages</td>
<td>212</td>
</tr>
<tr>
<td>20 GDDGATEK return codes</td>
<td>304</td>
</tr>
<tr>
<td>21 GDDR call overrides</td>
<td>333</td>
</tr>
<tr>
<td>22 Resiliency Expert global variables</td>
<td>335</td>
</tr>
</tbody>
</table>
Tables
As part of an effort to improve its product lines, EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your EMC representative if a product does not function properly or does not function as described in this document.

Note: This document was accurate at publication time. New versions of this document might be released in EMC Online Support. Check EMC Online Support to ensure that you are using the latest version of this document.

Purpose

This guide describes the basic concepts of EMC Geographically Dispersed Disaster Restart (EMC GDDR), how to install it, and how to implement its major features and facilities.

Audience

This document is part of the EMC GDDR documentation set, and is intended for use by GDDR systems administrators and computer operators.

Readers of this document are expected to be familiar with the following topics:
- IBM z/OS operating environments
- IBM parallel sysplex
- EMC software: SRDF, ResourcePak Base, Consistency Group, and AutoSwap

Related documentation

The following publications provide additional information:
- *EMC GDDR Release Notes*
- *EMC GDDR Message Guide*
- *EMC Mainframe Enablers Installation and Customization Guide*
- *EMC ResourcePak Base for z/OS Product Guide*
- *EMC SRDF Host Component for z/OS Product Guide*
- *EMC AutoSwap for z/OS Product Guide*
- *EMC Consistency Group for z/OS Product Guide*
- *EMC TimeFinder SnapVX Product Guide*
- *EMC TimeFinder/Mirror for z/OS Product Guide*
- *EMC TimeFinder/Clone Mainframe Snap Facility Product Guide*
Conventions used in this document

EMC uses the following conventions for special notices:

⚠️ **CAUTION**

A caution contains information essential to avoid data loss or damage to the system or equipment. The caution may apply to hardware or software.

---

**IMPORTANT**

An important notice contains information essential to software or hardware operation.

---

**Note:** A note presents information that is important, but not hazard-related.

---

**Typographical conventions**

EMC uses the following type style conventions in this document:

**Normal**

Used in running (nonprocedural) text for:

- Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus
- Names of resources, attributes, pools, Boolean expressions, buttons, DQL statements, keywords, clauses, environment variables, functions, and utilities
- URLs, pathnames, filenames, directory names, computer names, links, groups, service keys, file systems, and notifications

**Bold**

Used in running (nonprocedural) text for names of commands, daemons, options, programs, processes, services, applications, utilities, kernels, notifications, system calls, and man pages

Used in procedures for:

- Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus
- What the user specifically selects, clicks, presses, or types

**Italic**

Used in all text (including procedures) for:

- Full titles of publications referenced in text
- Emphasis, for example, a new term
- Variables

**Courier**

Used for:

- System output, such as an error message or script
- URLs, complete paths, filenames, prompts, and syntax when shown outside of running text

**Courier bold**

Used for specific user input, such as commands

**Courier italic**

Used in procedures for:

- Variables on the command line
- User input variables

<> Angle brackets enclose parameter or variable values supplied by the user

[] Square brackets enclose optional values

| Vertical bar indicates alternate selections — the bar means “or”

{} Braces enclose content that the user must specify, such as x or y or z

... Ellipses indicate nonessential information omitted from the example
Where to get help

EMC support, product, and licensing information can be obtained through EMC Online Support as described next.

**Note:** To open a service request through EMC Online Support, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or to answer any questions about your account.

Product information

For documentation, release notes, software updates, or for information about EMC products, licensing, and service, go to EMC Online Support (registration required) at:

https://support.EMC.com

Technical support

EMC offers a variety of support options.

**Support by Product** — EMC offers consolidated, product-specific information on the Web at:

https://support.EMC.com/products

The Support by Product web pages offer quick links to Documentation, White Papers, Advisories (such as frequently used Knowledgebase articles), and Downloads, as well as more dynamic content, such as presentations, discussions, relevant Customer Support Forum entries, and a link to EMC Live Chat.

**EMC Live Chat** — Open a Chat or instant message session with an EMC Support Engineer.

**eLicensing support**

To activate your entitlements and obtain your VMAX license files, visit the Service Center on https://support.EMC.com, as directed on your License Authorization Code (LAC) letter emailed to you.

For help with missing or incorrect entitlements after activation (that is, expected functionality remains unavailable because it is not licensed), contact your EMC Account Representative or Authorized Reseller.

For help with any errors applying license files through Solutions Enabler, contact the EMC Customer Support Center.

If you are missing a LAC letter, or require further instructions on activating your licenses through EMC Online Support, contact EMC’s worldwide Licensing team at licensing@emc.com or call:

- North America, Latin America, APJK, Australia, New Zealand: SVC4EMC (800-782-4362) and follow the voice prompts.
- EMEA: +353 (0) 21 4879862 and follow the voice prompts.

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to:

VMAXContentFeedback@emc.com
CHAPTER 1
Product Overview

This chapter presents an overview of GDDR and its capabilities.

◆ What is GDDR? ................................................................. 20
◆ Major features ................................................................. 20
◆ Supported business continuity configurations ................................................. 22
◆ GDDR fundamentals .......................................................... 24
◆ GDDR components ............................................................ 27
◆ GDDR-managed system types ...................................................... 29
◆ GDDR scripts ................................................................. 31
◆ Parameter Load wizard: Telling GDDR what to manage ............................. 33
What is GDDR?

EMC® Geographically Dispersed Disaster Restart (GDDR) automates business recovery following both planned outages and disaster situations, including the total loss of a data center. GDDR achieves this goal by providing monitoring, automation, and quality controls to many EMC and third-party hardware and software products required for business restart.

Because GDDR restarts managed systems following disasters, it does not reside on the same z/OS systems that it is seeking to protect. GDDR resides in separate logical partitions (LPARs) from the host z/OS systems that run application workloads.

GDDR is installed on a control z/OS system at each site. Each GDDR node is aware of the other GDDR nodes through network connections between each site. This awareness allows GDDR to:
- Detect disasters
- Identify survivors

To achieve the task of business restart, GDDR automation extends well beyond the disk level and into the host operating system level. It is at this level that sufficient controls and access to third party software and hardware products are implemented to enable EMC to perform automated recovery.

GDDR’s main activities include:
- Managing planned site swaps (workload and DASD) between the primary and secondary sites.
- Managing the recovery of the SRDF environment and restarting SRDF®/A (asynchronous remote replication) in the event of an unplanned site swap.
- Active monitoring of the managed environment and responding to exception conditions.
- Reset/IPL of z/OS systems at remote site.
- Testing disaster recovery from BCVs at remote site.
- Testing disaster recovery from R2 at remote site.

Major features

GDDR provides the following major features:
- Situational awareness
- Survivor recognition

Situational awareness

GDDR can distinguish normal operational disruptions from disasters and respond accordingly. For example, GDDR is able to distinguish between network outages (SRDF link drop) and real disasters. This awareness is achieved by periodic exchange of dual-direction heartbeats between the GDDR Control Systems (C-Systems).
Survivor recognition

GDDR can determine which sites and systems have survived a disaster. Unlike the EMC foundation mainframe software products (such as TimeFinder®/Mirror or TimeFinder/Clone Mainframe Snap Facility), GDDR has built-in intelligence to monitor other GDDR systems. GDDR constantly checks for disaster situations and ensures that other GDDR systems are “healthy.” This checking allows GDDR to recognize and act on potential disaster situations, even if only one GDDR system survives.

“Split brain” problems associated with cluster technologies are avoided through operator prompts. Upon the initial recognition stage, GDDR issues messages to the operator console seeking confirmation of the event and confirmation of restart actions required.

Restart coordination

If a primary site disaster occurs, the GDDR Master Control System (C-System) located at the secondary site will execute the recovery. The GDDR Master C-System operates in a Master Owner/ No-Owner role for other GDDR C-Systems.

Changes to GDDR configuration information can only be made on the GDDR Master Control System (C-System). GDDR propagates these changes to the GDDR-managed systems using the GDDR inter-system communications feature.

Restart procedures following disasters are coordinated from the GDDR Master C-System. GDDR coordinates and executes predetermined processes to:

- Restart the enterprise at the desired surviving site in the event of a disaster
- Automate a planned site swap

Additional capabilities

As part of the planned site swap process and the recovery process after an unplanned site swap, GDDR can optionally perform the following tasks:

- Trigger stopping or starting distributed workloads
- Trigger stopping or starting z/OS workloads in multiple sysplexes in parallel

Types of environment

GDDR can manage environments that are comprised of the following elements:

- Multiple z/OS systems
- Multiple sysplexes
- Multiple VMAX® controllers
- Intermix of CKD and FBA/FBAM DASD and BCVs
Supported business continuity configurations

A GDDR site is a physical location, housing CPU or DASD or both, where:

- Data Center 1 (DC1) is part of all supported GDDR configurations
- DC3 is a site connected to DC1 with SRDF/A, either actively or as a recovery connection

GDDR is available in the following configurations:

**SRDF/S with ConGroup** — The 2-site SRDF/S with ConGroup configuration provides disaster restart capabilities at site DC2.

**SRDF/S with AutoSwap** — The 2-site SRDF/S with AutoSwap configuration provides for near-continuous availability through device failover between DC1 and DC2.

**SRDF/A** — The 2-site SRDF/A configuration provides disaster restart capabilities at site DC3.

**SRDF/Star** — The 3-site SRDF/Star configuration provides disaster restart capabilities at either DC2 or DC3. Concurrent and cascaded SRDF support further minimize the DC3 recovery time objective. The 2-site SRDF/Star configuration has DASD at 3 sites in a concurrent topology, but has no C-System at DC2, and provides disaster restart at DC3 only.

**Note:** All SRDF/Star configurations must have R22 devices at DC3.

**SRDF/Star with AutoSwap** — The 3-site SRDF/Star with AutoSwap configuration provides for near-continuous availability through device failover between DC1 and DC2 as well as disaster restart capabilities at DC3. Concurrent and cascaded SRDF support further minimize the DC3 recovery time objective.

**Note:** Cascaded SRDF/Star configurations, with or without AutoSwap, can be dynamically reconfigured back and forth between concurrent and cascaded data flow.

**SRDF/SQAR with AutoSwap** — The 4-site SRDF/SQAR with AutoSwap configuration provides for near-continuous availability through device failover between DC1 and DC2, within Region 1; as well as disaster restart capabilities at Region 2 with DC3 and DC4 located an extended geographical distance away from Region 1. SRDF concurrent or cascaded replication protects data originating from the recovery site following a primary region outage.

GDDR can be customized to operate in any of these configurations. GDDR functionality is controlled by a parameter library. During GDDR implementation, this parameter library is customized to reflect:

- The prerequisite software stack
- The desired data center topology (two, three, or four sites, synchronous or asynchronous). The data centers are referred to as sites DC1 and DC3.

GDDR is able to control multiple sysplexes from a single GDDR Control System.

This document discusses the GDDR SRDF/A configuration. Documentation for other GDDR configurations is available on the EMC Online Support site.
SRDF/A configuration

The 2-site SRDF/A configuration provides disaster restart capabilities at site DC3.

**Note:** In the description below, the primary and secondary site roles are interchangeable.

*Figure 1* illustrates GDDR operation in the SRDF/A environment.

As *Figure 1* shows, sites DC1 and DC3 are the primary and secondary data centers of critical production applications and data. DC1 is the primary site, with SRDF/A data replication to the secondary site, DC3. These sites are considered fully equivalent for strategic production applications, connected with highly redundant direct network links. Both open systems (FBA) and mainframe (CKD) disk images can be replicated.

Each SRDF/A environment can manage one Multi-Session Consistency (MSC) group. An MSC group is a named group, consisting of multiple RDF groups operating in SRDF/A mode, managed by the EMC MSC control software feature as a single unit. These groups can be on multiple VMAX units. *Figure 1* shows the two GDDR C-Systems with their heartbeat communication paths, separate from the production disk and computer facilities.

GDDR does not have a requirement to “freeze” I/O to obtain a point of consistency. Multi-Session Consistency and SRDF/A provide the mechanism. At the point that EMC GDDR receives notification of an unplanned or failure event, a point of consistency is already achieved through these foundation technologies.

In this environment, GDDR can do the following:

- Manage planned site swaps
- Restart processing at the secondary site following unplanned primary site events
- Perform standard operational tasks:
  - IPL, system reset, activate, deactivate
Product Overview

- Trigger stop/start of business workloads
- Actively monitor for unplanned/failure events:
  - Sites
  - Systems
  - Loss of SRDF/A
  - Inter-site communication failure

GDDR fundamentals

This section discusses:
- Control systems
- Workload location
- Managed workloads
- GDDR processes

Control systems

The GDDR control systems are more commonly referred to as GDDR C-Systems. One GDDR C-System is located at each site (DC1 and DC3).

C-Systems must be configured as standalone systems by specifying either XCFLOCAL or MONOPLEX in PARMLIB’s IEASYS PLEXCFG parameter (XCFLOCAL is recommended). This enables the C-Systems to avoid SFM sysplex timer failure recovery operations and allows them to continue operations during sysplex timer recovery operations. Each GDDR C-System runs as a standalone z/OS system from local DASD.

Note: It is recommended to locate the C-System DASD on separate controllers from the production DASD. Because the EMC software applications run from local C-System volumes, this separation ensures that the C-Systems are not affected by any events that may impact the availability of the managed systems.

GDDR C-Systems do not run any production workload. The main functions of a GDDR C-System are to:
- Control the recovery after an outage
- Control a planned site swap

One of the C-Systems is the Master C-System. During normal operations, the Master C-System is the central control point for all GDDR activities. The Master C-System is located at the primary DASD site. In the event of the loss of the primary DASD site, GDDR transfers the Master C-System to the secondary site, for completion of the restart coordination.

Some GDDR functions can only be carried out by the Master C-System, for example:
- Running planned processes
- Updating GDDR parameters

All GDDR C-Systems are potential candidates to takeover as the Master C-System.
Workload location

In a GDDR Complex, the business or production workload runs at a single site; that is, one side of the sysplex. This is the same location as the primary DASD site.

Production system

A production system is a managed system that normally runs the site's workload and updates the primary DASD. Production systems and primary DASD must always be at the same site in the configuration.

Contingency or standby system

A contingency or standby system is a system that replaces production system capacity in the event of a loss of use of the primary site. A contingency system:

- May be used for expendable workload which is displaced by business workload following the loss of a primary site
- May be cold (not powered up), or warm (powered up but not IPLed) systems reserved for business workload restart and testing of restart processes, but not in support of any meaningful day-to-day workload

Contingency or standby systems are located at an appropriate distance from the primary systems to minimize risks from geographic and infrastructure exposures which may negatively impact primary systems availability.

Recovery LPAR

As previously described, contingency systems provide a way to move a workload from one system to a different system. Recovery LPARs provide a way to run the same system in two different locations at different times. A recovery LPAR is located on the same CPC, or on a different CPC, at the same site or at a different site. If a system is defined with a recovery LPAR, an additional recovery option is presented to the operators when such a system is lost. Managed systems can have a contingency system as well as a recovery LPAR.

Managed systems

Any production or contingency/standby system defined to GDDR is known as a GDDR-managed system.

Managed workloads

GDDR can trigger the stop and restart of production workloads on:

- z/OS systems
- Distributed systems

External workloads

External workloads run in mainframe systems which do not have their DASD in the managed VMAX units.

GDDR can coordinate Stop and Start of the workload on these "non-managed" mainframe systems with the workload Stop and Start for managed systems.
Excluded systems

GDDR can be configured to exclude certain systems from workload management, although these systems have their DASD in the managed VMAX units.

HMC-only systems

GDDR can be configured to limit IPL and CBU actions for certain systems to the online interface. No other actions or automation are performed for these systems.

HMC Bypass feature

If the site where GDDR is running is under management of a third-party facility provider, GDDR offers the HMC Bypass feature, by site and by LPAR to prevent GDDR HMC interaction with all or selected LPARs at that site.

GDDR processes

A GDDR process is a predetermined sequence of function calls. Generally one function call corresponds to one action. A GDDR process is started by calling GDDR-provided routines, either from a batch job or as a result of specific messages being issued.

There are two types of GDDR processes — planned and unplanned.

Planned process

A GDDR planned process is initiated through the GDDR interface to perform a planned task. The planned process encompasses planned swap, reconfiguration, resumption, and test processes.

Unplanned process/takeover process

The GDDR unplanned process or takeover process can only be initiated following an error that results in a possible takeover situation. Takeover processes are initiated as a result of certain messages being issued or specific events occurring.

The messages or events that trigger an unplanned or takeover process can originate on any system, either a C-System or a production system. They only take place on the current Master C-System.

They are invoked following operator confirmation of any of the following types of failure or loss:

- Sites
- DASD
- Systems
- Loss of SRDF link
- Loss of host channels
Process restart

The return codes from the function calls that make up a GDDR process are saved in GDDR global variables. For functions that issue EMC SRDF Host Component commands, the return code of the commands is also saved. If multiple commands are issued from one function, the return codes from each command are saved in GDDR global variables.

After the cause of the original failure has been identified and resolved, the GDDR process can be rerun. GDDR uses the saved return codes to establish the point of restart; that is, the point of the previous failure. This ensures that no modifications to the supplied GDDR process jobs are required in order to rerun after a failure.

GDDR components

GDDR consists of the following components:

- Parameters
- User interface
- Events
- Monitors
- Message rules

Parameters

GDDR parameters define the environment and configuration that it manages. The parameters can modify the sequence of function calls that is a GDDR process.

User interface

The GDDR user interface is an ISPF application. It is available only on GDDR C-Systems.

Events

A GDDR event is a change in state of a component part of the environment that GDDR is actively monitoring. Examples of GDDR events include:

- SRA — SRDF/A link is down
- MHB — missing C-System heartbeat

The event can have a state of either TRUE or FALSE. If the event has a state of TRUE, it has occurred or is currently occurring. If the event has a state of FALSE, it is no longer occurring. An event that is TRUE is considered an exception.

GDDR events are used by the GDDR event monitor and GDDR processes to determine environment state. A change in state can then:

- Request operator confirmation of the event and present the relevant actions
- Prevent a planned process from running
Monitors

There are two monitors on each GDDR C-System:

- The GDDR event monitor
- The GDDR heartbeat monitor

Event monitor

The GDDR event monitor runs on each C-System and is used to analyze event state changes in which GDDR is interested. On detecting the occurrence of selected events, the event monitor determines what action to take and prompts operators with the appropriate choices.

The Event Monitor verifies the status of SRDF operation on a user-defined interval. GDDR produces messages for integration with user automation that indicate when a GDDR event changes state. (OFF to ON, or ON to OFF). Table 17, “Monitored events,” on page 201 provides a detailed description of GDDR-monitored events.

Certain software operating states are monitored and communicated solely through messages. Message rules enable certain messages of interest to be forwarded to managed systems where user automation can then react to the problem. Table 18, “Software state analysis messages,” on page 202 provides detailed descriptions of state analysis messages. Examples of the usage of messages for the monitored operating states are in applicable configurations:

- MSC analysis
- SRDF/A analysis
- RDF group and link analysis
- Loss of DASD access
- Loss of site

Heartbeat monitor

The GDDR heartbeat monitor aids the event monitor in determining the status of the GDDR managed environment. The lack of a heartbeat from a particular C-System is used to determine the state of a C-System and the site.

Message interception rules

GDDR is supplied with message interception rules to be installed on the GDDR C-Systems and GDDR-managed systems.

The message interception rules have two primary functions:

- To detect events that GDDR is interested in and set the appropriate GDDR event TRUE or FALSE.
- To detect events that GDDR processes have to wait for (WTOR), and reply as to the success or failure of the waited for event. This will determine if a GDDR process proceeds or terminates.

GDDR uses the z/OS MCSOPER facility to monitor the GDDR-managed systems for messages of interest. GDDRMAIN is the main GDDR address space. The GDDRMAIN tasks which are installed on the GDDR C-Systems and the GDDR-managed systems perform the communication function to route message traffic to or from production systems. The
arrival of a message at the target production system can be used to trigger an automation rule (for example using Computer Associates OPS/MVS® Event Management and Automation, IBM Tivoli NetView®, or BMC Control-M®). Such rules can be used to start or shut down workloads on the appropriate systems.

**DYNAPI interface**

The GDDR interface to EMC DYNAPI allows GDDR to run dynamic SRDF commands in parallel.

**GDDR-managed system types**

In addition to C-Systems, there are four different types of systems defined to GDDR:

- Production or test systems and their optional contingency systems
- Excluded systems
- HMC only systems
- External workload systems

These systems are referred to as GDDR-managed systems within this document.

**Production or test systems and their optional contingency systems**

These systems benefit from the full set of GDDR systems management features, and require that GDDRMAIN be running on these systems at all times to fully benefit from the following features:

- These systems run from the GDDR-managed DASD.
- GDDR triggers the start and stop of the business workload during GDDR scripts.
- GDDR performs Stop/Start ConGroup operations on these systems if Call_Override bytes 08/09 are true.
- These systems can be IPLed, reset, activated, or deactivated during certain scripts.
- These systems can be protected using the LPAR Recovery feature and can participate in CPC (Central Processor Complex) swaps.
- These systems can be managed from the Perform HMC LPAR Actions panel.
- GDDR can perform CBU actions, in scripts or in panels, for the CPCs on which these systems run.
- GDDR manages or simulates the management of couple datasets and CF Structures for these systems, if they are in a sysplex.

Note the following requirements:

- These systems must be specified in GDDMPARM COMM parameters.
- The Define Managed Systems panel must have an entry for each system.
- These systems must have CPC and LPAR defined for them.
- These systems must have IPL parameters defined for them.
- These systems must have Mainframe Enabler STCs defined for them.
Excluded systems

These are mainframe systems running from the GDDR-managed DASD.

- GDDR performs Stop/Start ConGroup on these systems if Call Override bytes 08/09 are true.
- GDDR does not trigger Start/Stop of workload for these systems.
- GDDR does not perform HMC LPAR actions for these systems during scripts, but they are listed on the GDDR HMC LPAR Actions panel.

The excluded systems appear on the Define Managed Systems panel as “Manage Workload=NO”.

- These systems can be specified in GDDMPARM COMM parameters.
- These systems can have CPC and LPAR defined for them.
- These systems can have Mainframe Enabler STCs defined for them.

Note: The above three items are required if you intend to use the ConGroup Stop/Start method (Call Override bytes 08/09 = 1).

You can add or delete these systems from the Define Managed Systems panel, unless they are specified in GDDMPARM COMM parameters.

HMC only systems

The only functionality for these systems is ACT/DEACT/LOAD/RESET actions performed from within the GDDR HMC LPAR Actions panel. To facilitate auto-discovery, EMC recommends that these systems be defined in GDDMPARM.

- These systems must have CPC and LPAR defined.
- These systems must have IPL parameters defined.
- You can add or delete these systems from the Define Managed Systems panel, unless they are specified in GDDMPARM COMM parameters.

External workload systems

This category of systems appears on the Define External Workloads panel. These systems have very limited support. These are mainframe systems; they are not running from the GDDR-managed DASD. However, when GDDR triggers stop/start workload for managed systems, these systems will be included as well.

Coupling facility LPARs

The only functionality for these systems is ACT/DEACT actions performed from within scripts and the GDDR HMC LPAR Actions panel. For activation, coupling facility (CF) LPARs come first. During deactivation, CF LPARs are processed following all others.

- Exactly one CF LPAR is included on the Define Managed LPARs panel associated with the system flagged CF LPAR ‘YES’ on the Define Managed Systems panel, for the Home site of the system.
GDDR scripts

GDDR provides scripts that allow you to perform any of the following actions:

- Planned event management
- Test event management
- Unplanned event management
- Resumption of replication after SRDF link outages
- Special actions

Planned event management

Operations personnel can handle planned event management scenarios by running any of the following scripts.

**Note:** DC1 and DC3 represent the current primary DASD site or current secondary DASD site. When these representations are shown in italic type in script titles, this indicates the values are interchangeable. The descriptions assume that DC1 is the Primary DASD site and Primary site at the beginning of the script.

Automated Configuration Check - DASD - GDDRPCCD

Use this script as part of the pre-script checkup before any GDDR script is run. Review the GDDRPCCD script joblog for GDDP4** E** level messages, and resolve reported discrepancies before starting a planned, test or resumption script. This script performs the following actions:

- Discovers VMAX devices in a set of defined VMAX units and RDF groups as well as TimeFinder devices, and validates that discovered devices are defined to GDDR in RDF.DEVICES, DLM.DEVICES and STDBCV parameters.
- Validates existing RDF.DEVICES and DLM.DEVICES parameters, as well as PARMS.STDBCV parameters and other configuration global variables against the discovered DASD configuration and against GDDMPARM information.
- If discrepancies are found between the discovered configuration and the defined configuration, the CFG event is set to true. This prevents all scripts from running, except the GDDRPCCD, GDDRPEDD and GDDRPGVB scripts. To be able to run other scripts, either the DASD configuration must be changed to match the defined configuration, or new parameters must activated using the GDDR Parameter Wizard to match the existing DASD configuration.

Abandon Site **DC1** (site swap) - GDD2P17A

- Stops the business workload at the primary DASD site
- Waits for the stop of all business applications
Product Overview

- Resets clear all production systems managed by GDDR

Abandon Secondary Site (DC2) - GDDRPA60 (Site maintenance)

The GDDRPA60 scripts allow operations personnel to take down any site in the configuration for maintenance purposes.

Restart production at DC3 after site swap - GDD2P18A

This script performs the following actions after the loss of the primary site:

- Attempts reset clear of all systems at the primary DASD site
- Activates CBU (if required)
- Activates all needed LPARs at the secondary DASD site
- Creates a consistency point at the secondary DASD site
- Prepares the SRDF environment
- IPLs all needed production systems

Test event management

Perform test IPL from BCVs at DC3 - GDD2P01A

- Splits BCVs, makes them R/W
- Activates LPARs and loads test z/OS systems using BCV volumes

Resume after test IPL from BCVs at DC3 - GDD2P02A

- Stops test business workload, if applicable
- Reset clears test system LPARs
- Reestablishes the BCVs

Unplanned event management

Operations personnel can manage unplanned events in one of two ways:

- The GDDR Event Monitor prompts the operator for management confirmation of trigger events which indicate a site or DASD outage. The operator replies affirmative to the prompt and the GDDR recovery script starts.
- The operator may start the appropriate unplanned script and respond to prompts. The script initiates and validates that the state of the current host and storage environments matches the script prerequisites before proceeding.

Recover after loss of DC1 (RDR) - GDD2U12A

- Confirms that an MSC drop occurred
- Confirms that SRDF links failed
- Confirms that a regional disaster (RDR) event occurred
- Shuts down applications at the primary site, if applicable
Product Overview

- Splits BCVs and conditions R2s at secondary site for restart
- Activates contingency systems
- Restarts applications

Resume replication after loss of DC1 - GDD2PA0A
- Confirms SRDF/A links are down
- Splits BCVs at the secondary site, if applicable
- Issues MSC cleanup and SRDF/A restart commands
- Reestablishes BCVs at the secondary site

Resumption of replication after SRDF link outages
Operations personnel can resume operations after planned or unplanned outages by running any of the following scripts.

Resume SRDF/A after link loss - GDDRPA29
This script restores the SRDF/A links after a planned or unplanned stop of SRDF/A.

Reclaim Secondary site (DC2) - GDDRPA65
The GDDRPA65 scripts allow operations personnel to restore normal operations after a site has been abandoned for maintenance.

Special actions
- Transfer Master C System to <DCx> - GDDRPXMC
- Global Variable Backup - GDDRPGVB
- Move systems to alternate CPC - GDDRMCPA

Parameter Load wizard: Telling GDDR what to manage
The environment that GDDR manages is described to GDDR through a collection of common variables. The GDDR Parameter Load wizard groups these variables in a series of ISPF panels, each backed by a member in a PDS. For the initial setup of GDDR, it is strongly recommended that you go through the entire series of panels at least once to become familiar with all the required and optional features of GDDR, and to ensure that all defined elements are in agreement with the desired behavior of the product. The variable groups include the following:
- Configuration-defining variables
  These variables define the type of managed configuration, the C-systems, the initial role for each site, the consistency group names and the MSC group names.
- Storage object variables
  These variables define the actual SRDF and TimeFinder devices, SRDF groups, SnapVX snapshot names, DLm devices, and gatekeeper devices that form the configuration that GDDR will manage.
Product Overview

- Host object variables
  These variables define the managed, external and HMC-only systems, and their LPARs, CF LPARs, system recovery attributes, IPL-parameters, IPL-priorities, and CPCs. Host object variables also define HMC consoles, sysplex objects and EMC Mainframe Enablers started tasks.

- GDDR option variables
  These variables define user-selectable values for a variety of actions taken in the course of GDDR automation sequences. GDDR option variables also define site defaults for JCL and utilities used by GDDR, messaging options, and tuning values.
CHAPTER 2
Installing GDDR

This chapter describes the GDDR installation procedure.

- **Hardware and software requirements**................................................................. 36
- **Required installation information**................................................................. 38
- **Installing GDDR**............................................................................................. 39
- **Post-installation tasks**.................................................................................. 45
Hardware and software requirements

Before installing GDDR, review the hardware and software requirements listed next.

Mainframe environment requirements

The basic infrastructure must support SRDF/A. In addition to this, EMC GDDR has the following specific infrastructure requirements:

- There must be network connectivity between all C-Systems.
- An HMC (Hardware Management Console) must be available at each site that can be accessed from each C-System (access to these HMCs can be protected by means of a private VLAN).

GDDR has the mainframe environment requirements listed in Table 1. Before you install GDDR, make sure your environment meets these requirements.

<table>
<thead>
<tr>
<th>Item</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor hardware configuration</td>
<td>Any system that supports current IBM mainframe operating systems</td>
</tr>
<tr>
<td>DASD hardware configuration</td>
<td>Any supported VMAX DASD model at an Enginuity level specified in the EMC GDDR Release Notes</td>
</tr>
<tr>
<td>Software</td>
<td>Any currently supported IBM operating system</td>
</tr>
</tbody>
</table>

Minimum software requirements

The minimum software prerequisites needed to run GDDR 5.0 are as follows:

- z/OS
- SRDF/Host Component
- ResourcePak® Base with SRDF/A multi-session consistency (MSC)
- BCPIi

The IBM Base Control Program internal interface (BCPIi) is supported if the GDDR C-Systems are using z/OS 1.10 or a later release. In addition, the CPC must be a z9 or higher (BC or EC). Appendix B, “BCPIi Interface,” provides additional information.

The z/OS level of the managed systems is not a consideration for the use of BCPIi for HMC operations. BCPIi operations are conducted on the CPC named in GDDRMAIN control statements. That CPC may or may not host a C-system, but it must host some system running GDDRMAIN (C-System or production system).

Note: The MCL levels that must be met are explained in the BCPIi chapter of the MVS Programming Callable Services for High Level Languages document (SA22-7613).

Note: The EMC GDDR Release Notes provide information regarding supported software release levels for the previous items.
Installation procedures for the EMC software products are provided in the *EMC Mainframe Enablers Installation and Customization Guide*.

Additional configuration requirements

**SRDF/A** — Please refer to the *EMC SRDF Host Component for z/OS Product Guide* for information on configuring an SRDF/A environment.

**Note:** GDDR is compatible with SRDF Automated Recovery functionality.

SRDF/A MSC has the following additional gatekeeper requirement:

- There must be one or more gatekeeper devices for each MSC-controlled RDF group. These gatekeeper devices must be in OS configuration as OFFLINE at IPL- as regular local devices (not BCV, SRDF, SAV, and so forth).

Minimum hardware requirements

*Table 2* describes the recommended minimum processor and I/O configuration for a GDDR C-System.

<table>
<thead>
<tr>
<th>Item</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical processors</td>
<td>1 (2 are recommended)</td>
</tr>
<tr>
<td>MSU</td>
<td>15 on a IBM 2084-306 (or equivalent)</td>
</tr>
<tr>
<td>Storage</td>
<td>512 MB</td>
</tr>
<tr>
<td>Logical paths to own local DASD devices</td>
<td>4</td>
</tr>
<tr>
<td>Logical paths to managed DASD devices</td>
<td>4</td>
</tr>
</tbody>
</table>

*Note:* EMC recommends separate channels for GDDR-managed storage gatekeepers and production gatekeeper functions.

DASD support

GDDR supports and can manage the following combinations of DASD:

- Single EMC VMAX controllers configured with any of the following:
  - All CKD devices
  - All FBA and FBA-META devices
  - Any combination of CKD, FBA and FBA-META devices
- Multiple EMC VMAX controllers configured with any of the following:
  - All CKD devices
  - All FBA and FBA-META devices
  - Any combination of CKD, FBA and FBA-META devices
Management and monitoring of both CKD and FBA/FBA-META devices is performed from the z/OS platform where the GDDR application resides. From the GDDR point of view, CKD and FBA/FBA-META VMAX devices are the same; that is, each is treated no differently than the other. They are all command targets of SRDF Host Component configuration commands using local, remote or GNS syntax.

GDDR requires that if even only one device in an RDF group is defined to GDDR, then all devices in that group must be defined to GDDR. Most GDDR actions are directed at the RDF group level (although in some cases, GDDR will act on device ranges if that is appropriate).

GDDR has no limitations on the number of EMC VMAX controllers/devices that can be managed. Any limitations are subject to restrictions in EMC hardware and software.

## Required installation information

Before beginning the GDDR installation, you need to gather information in preparation. Identify or decide upon the following items:

- **CLIST library and EDIT macro**
  
  Determine a name for the edit macro created by the installation dialog. You also need to determine the name of a CLIST library where you can store the edit macro.

- **Product dataset name prefix**
  
  Choose the dataset prefix you will use to install GDDR. Names for the product datasets consist of a final qualifier, such as LINKLIB, and a dataset prefix. For example, if you choose a dataset prefix of EMC.GDDRvrm, the LINKLIB dataset will be named EMC.GDDRvrm.LINKLIB.

  Ensure that you have RACF ALTER authority (or the equivalent from another security manager) for the datasets created with this dataset prefix.

  **Note:** Throughout this guide, datasets created using this dataset prefix are referred to as if they had been created with the suggested value. The actual `fmid` for your installation may be different.

- **ResourcePak Base dataset name prefix**
  
  Specify the dataset name prefix you used when you install ResourcePak Base. EMC recommends that you use `EMC.fmid` if it agrees with your site standards.

- **SMP/E dataset name prefix**
  
  Choose the name prefix for the SMP/E datasets into which you installed GDDR. If you have installed another EMC product using SMP/E, you should install GDDR into the same CSI. If you are installing an EMC SMP/E maintained product for the first time, EMC recommends using “EMC.SMPE.”

- **SMP/E datasets volser**
  
  Choose the disk volume onto which you will install the distribution libraries required by SMP/E. This may be the same volume you use for the product libraries. However, many customer sites prefer to keep SMP/E-related datasets on separate volumes from product libraries. An amount of space similar to that needed for the product libraries is required.
Installing GDDR

This section describes how to install GDDR. The GDDR installation kit is provided as an electronic download from EMC Online Support.

Install GDDR

The GDDR kit consists of a PDS containing TSO TRANSMIT images of files needed to perform an SMP/E indirect-library installation. This PDS is packaged on CD or as an electronic download from EMC Online Support.

To install GDDR on an GDDR control system, take the following steps:

1. Load the TSO TRANSMIT file, GDDR vrm.XMITLIB, to the mainframe disk.
2. Run GDDR vrm.XMITLIB(#EXTRACT) to extract ds-prefix.RIMLIB and the SMP/E indirect libraries.
3. Customize the RIMLIB JCL.
4. Run the installation jobs.
5. Perform cleanup.
6. Apply maintenance updates.

The following sections describe these steps in more detail.

Load GDDR vrm.XMITFILE to disk

1. Take one of the following steps:
   - If you are installing GDDR from a CD, complete the following steps:
     a. Mount the CD on an open system host.
     b. Allocate a working directory on the open system for the installation.
     c. Copy the contents of the CD to the working directory.
   - If you are installing GDDR from an EMC Online Support download, complete the following steps:
     a. Log in to a privileged account on an open systems host (root on UNIX or administrator on Windows).
     b. Allocate a working directory on the open system for the installation.
     c. Log on to EMC Online Support.
d. Navigate to Downloads > Geographically Dispersed Disaster Restart (GDDR).

   **Note:** If you are not able to access this location, you may not have registered your software or registered it incorrectly. Follow the prompts to register your software, correct your registration, or contact EMC in the event of a problem.

e. Click the product version you want to download. The product version consists of a zip file that contains the installation kit and the installation instructions.

f. Download the installation kit into the working directory.

2. If your current host is a Windows system, unzip the file in the working directory. If your current host is a UNIX system, unzip and untar the file into the working directory.

3. Locate GDDR\*vrm.XMITFILE.

   This file is in TSO TRANSMIT format and contains a flattened copy of GDDR\*vrm.XMITLIB, a PDS that holds other TRANSMIT images, the JCL to extract them, and necessary SMP/E installation files.

4. On the target mainframe, allocate a file to which you can FTP GDDR\*vrm.XMITFILE.

   Use the dataset name prefix you intend to use for product installation. The final qualifier must be XMITFILE. For example, if you intend to install the product with a dataset name prefix of EMC.GDDR\*vrm, name the file EMC.GDDR\*vrm.XMITFILE.

   Allocate the dataset with the following characteristics:

   
<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRECL</td>
<td>80</td>
</tr>
<tr>
<td>BLKSIZE</td>
<td>3120</td>
</tr>
<tr>
<td>DSORG</td>
<td>PS</td>
</tr>
<tr>
<td>SPACE</td>
<td>(CYL,(44,2))</td>
</tr>
</tbody>
</table>

   **Note:** The SPACE parameter assumes that you are allocating the dataset on a 3390 device.

5. FTP the file to the mainframe in binary format.

   Your FTP session may look something like the following:

   ```
   ftp hostname
   (username and password prompts)
   cd..
   25 """" is working directory name prefix
   binary
   200 Representation type is image
   put GDDR\*vrm.XMITFILE EMC.GDDR\*vrm.XMITFILE
   ```

6. Use TSO RECEIVE to receive the file into a PDS.
The PDS is created by the RECEIVE command and does not have to be pre allocated. However, you must specify a dataset name using the DA[taset] parameter or the file will be allocated using your TSO prefix (usually your logonid). The dataset name specified must have the final qualifier of XMITLIB.

For example:

```
receive indataset('EMC.GDDRvrm.XMITFILE')
INMR901I Dataset EMC.GDDRvrm.XMITLIB from userid on nodename
INMR906A Enter restore parameters or 'DELETE' or 'END' +
da('EMC.GDDRvrm.XMITFILE')
```

If you did not specify “DA(...)” as above, the dataset would be allocated as `userid.XMITLIB`.

Run GDDRvrm.XMITLIB(#EXTRACT)

Now run GDDRvrm.XMITLIB(#EXTRACT) to extract `ds-preface.RIMLIB` and the SMP/E indirect libraries. Take the following steps:

1. Edit the #EXTRACT member of the newly RECEIVED library.

   You can edit the #EXTRACT job by running the SETUP REXX program you can find in the XMITLIB dataset. The SETUP REXX program prompts you for all of the information needed to edit the job.

   If you wish to edit the job manually, make the following changes:
   - Change the JOB card to one that conforms to your standards.
   - Globally change `ds-prefix` to the dataset prefix of this library (which will be the dataset prefix for the product libraries).
   - Globally change DVOL to the disk volser onto which you want to place the extracted libraries.
   - Globally change DISK-UNIT to an esoteric unit name such as “SYSDA” that is appropriate for your site.

2. Submit #EXTRACT. Step completion codes should be 0, except for the DELETE step, which will have a step completion code of 8 unless the job is a rerun.

Customize the RIMLIB JCL

The RIMLIB library (`ds-prefix`.RIMLIB) is a PDS containing JCL to install the product. After you extract the RIMLIB PDS, you find that RIMLIB has the contents shown in Table 3.

<table>
<thead>
<tr>
<th>File</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>#01ALLOC</td>
<td>Allocate target and distribution libraries</td>
</tr>
<tr>
<td>#02DDDEF</td>
<td>Add or replace product library DDDEFS to SMP/E CSI</td>
</tr>
<tr>
<td>#03RECEV</td>
<td>SMP/E RECEIVE function into global zone</td>
</tr>
<tr>
<td>#04APPLY</td>
<td>SMP/E APPLY function into target zone</td>
</tr>
<tr>
<td>#05ACCP</td>
<td>SMP/E ACCEP product sysmods into distribution zone</td>
</tr>
<tr>
<td>#06CLEAN</td>
<td>Deletes indirect libraries and DDDEFS used for them</td>
</tr>
</tbody>
</table>
Installing GDDR

Complete the following steps to customize the installation JCL using the automated dialog:

1. Edit the RIMLIB library (ds-prefix.RIMLIB).
2. Locate the member named SETUP on the member selection list and type EX in the selection column next to it and press Enter.

Menu  Functions  Confirm  Utilities  Help
----------------------------------------------------------------------------------
EDIT              EMC.GDDR          Row 00001 of 00013
Command ===>                                         Scroll ===> CSR
Name     Prompt       Size   Created          Changed          ID
_________ #01ALLOC                 45      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ #02DDDF                 51      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ #03RECEV                 22      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ #04APPLY                 22      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ #05ACCPT                 22      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ #06CLEAN                 53      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ #99MAINT                 27      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ GDRJCL                   206     yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
_________ GDRWIN1                  51      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
ex_________ SETUP                   13      yyyy/mm/dd yyyy/mm/dd hh:mm:ss idstring
**End**

Result: The panel shown in Figure 2 appears.

| COMMAND ==> ____________________________ |
| Type EXEC on the command line and press ENTER to proceed, or PF3 to exit. |
| CLIST library ==> 'hlq.GDDRvrm.RIMLIB' |
| Edit macro name ==> GDR |
| Product dsname prefix ==> hlq.GDDRvrm |
| Mainframe Enablers dsname prefix ==> hlq.MFEvrm |
| SMP/E dsname prefix ==> EMC.SMPE |
| SMP/E datasets volser ==> |
| Install-to disk volser ==> Disk unit name ==> SYSDA |

Enter your job card below ('%MEMBER%' will be replaced by member name):
-> //%MEMBER% JOB MSGCLASS=A,CLASS=A,MSGLEVEL=(1,1)

---

Figure 2 EMC JCL customization utility

3. Enter or change the following information on the panel shown in Figure 2 to customize your installation:
   a. The CLIST library field is set by default to the name of the RIMLIB library. This field should contain the name of a library in which you want the edit macro created by this dialog to be stored.

The default value is fine for most users and need not be changed.
Installing GDDR

b. In the Edit macro name field, either:
   - Accept the default name displayed.
   or
   - If necessary, change the name of the edit macro.
   
   **Note:** Normally, you should not have to change the name.

   **Result:** The edit macro is created in the CLIST library from the data entered on this panel and applied to all members of RIMLIB that start with a # character.

c. In the Product dsname prefix field, enter the dataset name prefix you want to use for the target datasets. EMC suggests *hlq.GDDRvrm*.

d. In the SMP/E dsname prefix field, enter the dataset name prefix of the SMP/E datasets into which you installed EMC GDDR.

   For example, if you called the SMPSCDS dataset EMC.SMPE.SMPSCDS, enter EM.C.SMPE.

e. In the SMP/E datasets volser field, enter the six-character volume serial number of the disk volume on which you want to allocate the SMP/E distribution libraries for GDDR.

   This volume may be the same as the volume you specify in the next step, or you may elect to keep these datasets on a separate volume.

f. In the Install-to disk volser field, enter the six-character volume serial number of the disk volume to which you want to install the GDDR libraries.

g. In the Disk unit name field, you can specify an esoteric disk name that is appropriate to your site. SYSDA is the default, but you can overtype it with another esoteric disk name.

h. Enter a site-appropriate job card.

   The job card is initially set to a value which may be suitable to many users. The first seven characters of the job name is set to your TSO userid, plus “X.”

   You can set the job name to %MEMBER%. This causes the edit macro to set each job name equal to the JCL member name (that is, #01ALLOC, #02DDDEF, and so forth).

   Do not use any parameter that contains an ampersand (&), such as NOTIFY=&SYSUID. An ampersand in the job card can cause edit macro errors.
Installing GDDR

**Figure 3** shows an example of a completed panel as the user is about to press **Enter** and complete the dialog.

![EMC JCL customization utility completed panel](image)

**Figure 3** EMC JCL customization utility completed panel

4. When you are satisfied with your entries, type **exec** on the command line and press **Enter**.

   **Result:** If the dialog completes successfully, you see something similar to the following:

   ```
   BUILDING AN EDIT MACRO(GD) IN 'EMC.GDDRvrm.RIMLIB'
   PROCESSING MEMBER: #01ALLOC
   PROCESSING MEMBER: #02DDDEF
   PROCESSING MEMBER: #03RECEV
   PROCESSING MEMBER: #04APPLY
   PROCESSING MEMBER: #05ACCPT
   PROCESSING MEMBER: #06CLEAN
   ***
   ```

   Run the installation jobs

   Carefully examine each job before you submit it to make sure that it was customized the way you intended.

   Submit the customized jobs in the following order, making sure that each job completes successfully before submitting the next one:

   1. #01ALLOC
   2. #02DDDEF
   3. #03RECEV
   4. #04APPLY

   You should expect completion codes of 0 (zero) for all jobs except for #02DDDEF, where 04 is acceptable if this is a new installation rather than an upgrade.

   If your testing results are positive, run #05ACCPT to update the distribution libraries and zone. The #05ACCPT job completes with an RC=04. This is normal for the SMP/E ACCEPT process. You can ignore it.

   SMP/E installation is now complete.
Post-installation tasks

Having completed the SMP/E installation steps, several more tasks remain to complete the installation of GDDR. These tasks are described in detail in Chapter 3, “Integrating GDDR.”

Cleanup

After you are satisfied that GDDR is correctly installed and functioning properly, run the #06CLEAN job to delete datasets and DDDEFs used during the installation process that are no longer needed.

Apply maintenance updates

If you have received maintenance cover letters from EMC or have instructions to apply maintenance from EMC support personnel, use the supplied job #99MAINT. This job receives and applies APARs and PTFs. This job may require further customization before you run it, depending on the nature of the maintenance.

**Note:** Do not attempt to apply maintenance until the GDDR ACCEPT job has completed successfully and then only if instructed to do so by EMC Customer Service.

Post-installation tasks
Installing GDDR
This chapter describes how to customize and configure GDDR.

⚠️ CAUTION ⚠️

These changes, with the exception of the GDDR-managed system security definitions and started task installation, are made on the GDDR C-Systems only.

- Integration tasks .......................................................... 48
- Update system parameter files ........................................ 48
- Create parameter members for SRDF Host Component on C-Systems ............ 53
- Edit SCF initialization parameters ...................................... 54
- Specify GDDR security ..................................................... 54
- Define GDDR datasets ..................................................... 64
- Install GDDR started procedures ....................................... 65
- Configure GDDR ........................................................... 79
- Optional configuration features ......................................... 84
Integration tasks

Once you have completed the SMP/E installation steps, complete the tasks described in the following sections before using GDDR:

- “Update system parameter files” on page 48
- “Configure BCPii” on page 51
- “Create parameter members for SRDF Host Component on C-Systems” on page 53
- “Specify GDDR security” on page 54
- “Define GDDR datasets” on page 64
- “Install GDDR started procedures” on page 65
- “Configure GDDR” on page 79

⚠️ CAUTION

Unless noted otherwise, these changes are made on the GDDR C-Systems only.

Update system parameter files

Perform the following system parameter file updates.

SYS1.PARMLIB(IKJTSOxx) customization

1. Confirm that the following entries exist in AUTHCMD, AUTHPGM, and AUTHTSF lists. Add any missing entries to the IKJTSOxx member of SYS1.PARMLIB of each C-System.
2. Activate this change using an IPL or dynamically make the change using the TSO PARMLIB UPDATE(xx) command.
Integrating GDDR

TSO logon customization

1. You may need to increase the region size of TSO logon procedures that use the GDDR user interface and batch jobs that run GDDR scripts. EMC recommends allocating a TSO logon proc region of 2,100,000, as a starting point.

2. Ensure that TSO logon procs of all TSO users who wish to run the GDDR ISPF user interface contain a SYSEXEC DD which points to hlq.GDDRvrm.RCXFE. This library is allocated during installation.

APF authorization

◆ APF-authorize the hlq.MFEvrm.LINKLIB and hlq.GDDRvrm.LINKLIB libraries.

LINKLIB and REXX parameter file installation

Note: If your site prefers to reference hlq.GDDRvrm.LINKLIB and hlq.MFEvrm.LINKLIB by using STEPLIB DD statements, then you can skip the LINKLIST and REXX parameter file customization procedures.

GDDR uses a REXX function package named GDDFUSER with an alias of IRXFUSER. IRXFUSER is a placeholder in SYS1.LINKLIB, providing flexibility for customization of REXX function packages. You have several choices for installation, depending on whether you have applications using previously customized REXX function packages on GDDR C-Systems and GDDR-managed systems. These instructions apply to each C-System and each GDDR-managed system on which GDDRMAIN is to be installed.

◆ If no applications use previously customized REXX function packages, or there are applications which use STEPLIB DD statements to reference a customized REXX function package module, then follow the directions provided with “Customize LINKLIST” below.

◆ If customized REXX function packages named IRXFUSER accessed via the LINKLIST exist on one or more GDDR-managed systems or C-Systems, perform the steps listed in “Customize LINKLIST” below and “Customize REXX parameter files” on page 51.

Customize LINKLIST

1. Delete the placeholder IRXFUSER module from SYS1.LINKLIB.

2. Add hlq.MFEvrm.LINKLIB and hlq.GDDRvrm.LINKLIB to the LINKLIST using one of the following methods:

   • Add the following LNKLST entries in a PROGxx member:

     ```
     LNKLST ADD NAME(LNKLST) DSN(hlq.MFEvrm.LINKLIB)
     LNKLST ADD NAME(LNKLST) DSN(hlq.GDDRvrm.LINKLIB)
     ```

   or

   • Add the following entries in a LNKLSTxx member:

     ```
     hlq.MFEvrm.LINKLIB(vvvvvv)
     hlq.GDDRvrm.LINKLIB(vvvvvv)
     ```
In these entries, \textit{vrm} is the current GDDR version, release, modification identifier and \textit{vvvvvv} is the volser where the \texttt{hlq.GDDRvrm.LINKLIB} dataset resides. The volser specification is only required if the dataset is not cataloged in the master catalog.

3. Replace \texttt{hlq.GDDRvrm.LINKLIB} with the dsname of the GDDR LINKLIB SMP/E target library allocated and filled during the installation process. Activate this change using one of the following methods:
   - IPL
   - Issue the SET PROG=xx command
   - Issue the SETPROG LINKLIST,ADD command

### Customize REXX parameter files

This procedure is used when there is an application using previously customized REXX function packages not accessed by a STEPLIB DD statement. The application has already taken advantage of the “placeholder” aspect of IRXFUSER in SYS1.LINKLIB, so additional steps are required to ensure the GDDR uses the proper function package named GDDFUSER, without the alias IRXFUSER.

1. Add GDDFUSER to the three REXX parameter files used in establishing the REXX processing environment. Before you begin, create a backup copy of the files.

   The REXX parameter files to be edited are:
   - IRXPARMS for REXX under MVS
   - IRSTSPRM for REXX under TSO
   - IRXISPRM for REXX under ISPF

2. Place the newly assembled versions of IRXPARMS, IRXISPRM, and IRXTSPRM into SYS1.LPALIB overlaying the default members.

For more information, refer to the \texttt{TSO/E REXX Reference SA22-7790}, chapter Language Processing Environments, subtopic ‘Changing the default values for initializing an environment’.

### Configure BCPii

The Base Control Program internal interface (BCPi\texttt{ii}) is the interface to the IBM Mainframe Service Element (SE) and the Hardware Management Console (HMC) used to communicate with the Central Processing Complex (CPC) SE. BCPii support is available starting with GDDR 4.1 with PTF GD41091 or higher maintenance. Additional information regarding BCPii is provided in Appendix B, “BCPi\texttt{ii} Interface.”

\textbf{Note:} BCPii configuration is required on all systems where GDDRM\texttt{AIN} is expected to support BCPii actions.

The following requirements must be met to use BCPii:

- The z/OS operating system be at level 1.11 or higher (BCPi\texttt{ii} can run under z/OS 1.10 with the installed BCPii deliverable; however, not all functionality is available at the 1.10 level).
 Integrating GDDR

- The CPC must be a z9 or higher (BC or EC). There are certain MCL levels that must be met which are explained in the BCII chapter of the *MVS Programming Callable Services for High Level Languages* document (SA22-7613).
- Specific actions must be performed on the SE/HMC for each CPC in order to enable BCII to run on that CPC. This is referred to as LIC Enabled status for the HMC in reference to the SE/CPC.

**Security changes required to use BCII**

BCII security is controlled both at the SE for each CPC holding z/OS Images that will participate in BCII communications and also within the security product running on each of the participating z/OS systems. Examples provided here illustrate configuration using only the IBM Security Server (RACF) product.

At the SE side, follow the instructions in the *MVS Programming Callable Services for High Level Languages* document (SA22-7613) in the Callable Service chapter section "Base Control Program Internal Interface" under the heading BCII Setup and Installation. For each z/OS Image on a CPC that will participate, cross-partition authority must be enabled through the SE. Additionally, an SNMP Community Name for each CPC must be established. Due to restrictions with the security products on z/OS, the Community Name cannot contain any lowercase characters. Refer to the documentation for specific steps to follow to establish the connection between authorized applications and the Service Element.

At the z/OS Security Product side, the section entitled “Setting up Authority to Use BCII” outlines the steps required to update your security product to allow the connection to BCII.

The HWIBCII address space under z/OS is the pathway between the authorized applications and the Service Element when communicating via BCII. If everything is configured properly, this address space comes up automatically during IPL and remains available for the life of the image. There are instructions for shutting this address space down and restarting if necessary.

**IMPORTANT**

It is important that all of the setup and requirements outlined in the BCII chapter in *MVS Programming Callable Services for High Level Languages* are understood. This ensures the security configuration meets your specific needs in a safe and straightforward fashion.

**BCII facility classes**

This section provides an overview of the pertinent BCII facility classes related to GDDR.

**HWI.APPLNAME.HWISERV**

HWI.APPLNAME.HWISERV in the Facility Resource Class controls which applications can use BCII services. A minimum of READ access is required and BCII requires that the Facility Resource Class be RACLISED. See the Callable Services Guide for an example of creating, using and refreshing this profile.
HWI.TARGET.netid.nau

HWI.TARGET.netid.nau in the Facility Resource Class controls access to a specified CPC using the full format name comprised of network ID netid followed by a period followed by the CPC name nau (network addressable unit). netid and nau can be up to 8 characters in length each, so that the netid.nau field can be from 3 to 17 characters total length. The APPLICATION DATA field for this facility class must match exactly the community name entered in the SE for the CPC during BCPIi setup. This is the connection between the security product and access to the desired CPC. This facility class is also used to control access to any Activation Profiles stored on the CPC (image, load or reset).

HWI.TARGET.netid.nau.imagename

HWI.TARGET.netid.nau.imagename in the Facility Resource Class controls access to the specified image on the specified CPC. imagename can be up to 8 characters in length.

Note: RACF allows for the use of generic facility classes. This can facilitate setup for the site. Keep in mind that RACF determines which profile to use when more than one applies by using the more specific class. For example, you define HWI.TARGET.IBM390PS.C.* as a generic Facility class to cover access to all LPARs on the C CPC. You also define HWI.TARGET.IBM390PS.* as a generic Facility class to cover access to all LPARs regardless of CEC. If you are working with an LPAR on the C CPC, RACF will use HWI.TARGET.IBM390PS.C.* ; otherwise, HWI.TARGET.IBM390PS.* will be used.

SYS1.PARMLIB changes required for BCPIi

Authority to run the GDDR BCPIi API functions requires changes to the AUTHTSF section of IKJTSS00 in SYS1.PARMLIB. Add the following entries on all GDDR B-Systems:

```
GDDBCPC2 /* BCPII AUTH CONNECT */ +
GDDBCPC3 /* BCPII AUTH COMMAND */ +
GDDBCPC2 /* BCPII AUTH DISCONNECT */ +
GDDBCPC2 /* BCPII AUTH EVENT */ +
GDDBCPL2 /* BCPII AUTH LIST */ +
GDDBCPL2 /* BCPII AUTH QUERY */ +
GDDBCPS2 /* BCPII AUTH SET */ +
```

Create parameter members for SRDF Host Component on C-Systems

During planned and unplanned Swap scripts, GDDR uses a utility named in the Utility.IEBGENER GDDR parameter to copy the appropriate parameter member over the currently used SRDF Host Component RDFPARM member.

Note: The hlq.GDDR.vrm.SAMPLIB members with the names listed below must be created in the PDS pointed to by the RDF entries within the Define EMC Mainframe Enablers STCs panel, “Option E: Define EMC MF Enablers STCs” on page 167. These members contain complete and identical SRDF Host Component initialization parameters, but are different
Integrating GDDR

with respect to MSC group definition, as described in Table 4. Many of these members define MSC gatekeeper devices. Avoid using these gatekeepers elsewhere in the GDDR-managed configuration.

Table 4  SRDF Host Component parameter members

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SITEUDC1</td>
<td>Used during a swap from DC3 to DC1. Defines an MSC group in MSC-only mode, with the R1 devices at DC1 and WF=0. Note: WF indicates the MSC weight factor. The primary server is the server running with MSC_WEIGHT_FACTOR = 0.</td>
</tr>
<tr>
<td>SITEUDC3</td>
<td>Used during a swap from DC1 to DC3. Defines an MSC group in MSC-only mode, with the R1 devices at DC3 and WF=0.</td>
</tr>
</tbody>
</table>

Note: The members listed above must be created on each C-system, in a PDS pointed to on the Define Mainframe Enablers STCs GDDR Parameter Load wizard panel. This panel defines one additional member (default: SRDFSTAR) which is actively used by the Host Component instance on the specified C-system.

Edit SCF initialization parameters

On each GDDR-managed host and C-System, make sure the following SCF parameter is set correctly:

- Ensure that SCF.CSC.VERBOSE is set to YES. This setting enables additional messaging required for the proper detection of the loss of an EMC GDDR-managed system.

Specify GDDR security

This section describes how to define the security environment required by EMC GDDR.

Define GDDR RACF functional groups

- Define the following RACF groups to grant the appropriate access based upon job function.

Table 5  RACF functional groups

<table>
<thead>
<tr>
<th>Functional group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDR$ADM</td>
<td>For systems programmers who will install and configure GDDR. For GDDR administrators who will configure GDDR.</td>
</tr>
<tr>
<td>GDDR$USR</td>
<td>For operators and operations support staff who will operate GDDR.</td>
</tr>
<tr>
<td>GDDR$REV</td>
<td>For management or interested parties who require the GDDR Parameter Review capability.</td>
</tr>
<tr>
<td>GDDR$STC</td>
<td>For the GDDR monitors, planned and unplanned processes.</td>
</tr>
</tbody>
</table>
## Summary of RACF permissions

Table 6 provides an overview of the RACF profiles and permissions required to protect GDDR resources.

**Table 6  RACF permissions**

<table>
<thead>
<tr>
<th>GDDR resource owning group</th>
<th>GDDR STC's user group</th>
<th>GDDR user group</th>
<th>GDDR reviewer group</th>
<th>Admin/Sysprog user group</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dataset profile</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>hlq:GDDRvm..LINKLIB</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>hlq:GDDRvm..ISPMLIB</td>
<td>READ</td>
<td>-</td>
<td>READ</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm..OPSEXEC</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm..ISPLLIB</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm..PROCLIB</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvrm.REXX</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm..IPSMLIB</td>
<td>READ</td>
<td>READ</td>
<td>-</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm..PARMLIB</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm.csys.DIV.*</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm..*</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>workhlq.qualifier.*</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm..BKUPVARS.CNTL</td>
<td>ALTER</td>
<td>READ</td>
<td>-</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm.PARMS.BKUP</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm.PARMS.WORK</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
<td>ALTER</td>
</tr>
<tr>
<td>hlq:GDDRvm.PARMS.LAPD</td>
<td>ALTER</td>
<td>READ</td>
<td>ALTER</td>
<td>ALTER</td>
</tr>
</tbody>
</table>

| **FACILITY profile**      |                       |                |                     |                         |
| GDDR.HMC.LISTOBJECTS      | READ                  | READ           | READ                | READ                    |
| GDDR.HMC.GETMSGS          | READ                  | READ           | READ                | READ                    |
| GDDR.GLOBAL.VARIABLE.ACCESS | UPDATE               | READ           | READ                | UPDATE                  |
| HWI.APPLNAME.HWISERV      | READ                  | READ           | -                   | READ                    |
| HWI.TARGET.netid.nau#1    | ALTER                 | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.nau#2    | ALTER                 | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.nau#n    | ALTER                 | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.nau#1.image#1 | ALTER              | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.nau#1.image#2 | ALTER              | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.nau#2.image#1 | ALTER              | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.nau#2.image#2 | ALTER              | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.ima*.*   | ALTER                 | ALTER          | -                   | ALTER                   |
| HWI.TARGET.netid.nau.*    | ALTER                 | ALTER          | -                   | ALTER                   |

| **TSOAUTH profile**       |                       |                |                     |                         |
| OPER                      | ACCESS NEEDED         | ACCESS NEEDED  | ACCESS NEEDED      | ACCESS NEEDED           |
| userid.SUBMIT             | ACCESS NEEDED         | ACCESS NEEDED  | ACCESS NEEDED      | ACCESS NEEDED           |

| **SURROGAT profile**      |                       |                |                     |                         |
| user.id.SUBMIT            | ACCESS NEEDED         | ACCESS NEEDED  | ACCESS NEEDED      | ACCESS NEEDED           |

| **JESSPOOL profile**      |                       |                |                     |                         |
| Jes2node.GDDR.*.*.*.*     | ACCESS NEEDED         | ACCESS NEEDED  | ACCESS NEEDED      | ACCESS NEEDED           |

---

a. hlq is any dataset high level qualifier, if one is used.

b. workhlq.qualifier.* is defined using Option J: Script JCL Parameters panel, field 'Work HLQ' as shown on page 173. The default value for workhlq is "GDDR". The Userids assigned to GDDR started tasks, as well as Userids authorized to submit GDDR scripts must be authorized with ALTER to this HLQ.
Integrating GDDR

c. **userid** is the Surrogate User ID defined on the Script JCL Parameters panel.

d. **Jes2node** is the JES2 node name of the GDDR C-System. The JES2 node name can be determined by issuing the JES2 console command $DNODE,OWNNODE=YES on the appropriate GDDR C-System. The output of the JES2 command is as follows:

```bash
$HASP826 NODE(1)
$HASP826 NODE(1) NAME=MFSYS3,STATUS=(OWNNODE),AUTH=(DEVICE=YES,
$HASP826 JOB=YES,NET=NO,SYSTEM=YES),TRANSMIT=NEITHER,
$HASP826 RECEIVE=NONE,HOLD=NONE,PENCRIPT=NO,
$HASP826 SIGNON=COMPAT,DIRECT=NO,ENDNODE=NO,REST=0,
$HASP826 SREENTRY=ACCEPT,COMPACT=0,LIN=0,LOGMODE=,
$HASP826 LOGON=0,NETSRV=0,OWNNODE=YES,
$HASP826 PASSWORD=(VERIFY=(NOTSET),SEND=(NOTSET)),
$HASP826 PATHMGR=YES,PRIVATE=NO,SUBNET=,TRACE=NO
```

The actual JES2 node name is identified on the NAME=output statement.

All GDDR RACF non-generic profiles should have a universal access (UACC) of NONE.

**Note:** Use the JCL provided in hlq.GDDRvrm.SAMPLIB member GDDCRACJ to define RACF authorizations for GDDR C-Systems using hlq.GDDRvrm.SAMPLIB member GDDCRACF. For managed systems, GDDPRACF. SAMPLIB members GDDCRACD and GDDPRACD are provided for ease of deleting RACF authorization definitions from systems.

**OPERCMDS class resource definitions (optional)**

If command authorization checking is in place, refer to the following OPERCMDS class resource definitions in Table 7. Sample commands are supplied in &hlq.GDDR&vrm.SAMPLIB(GDDCRACF).

**Table 7 RACF permissions, OPERCMDS class**

<table>
<thead>
<tr>
<th>GDDR resource owning group</th>
<th>Command/keyword</th>
<th>GDDRSST C</th>
<th>GDDRUS R</th>
<th>GDDRUS AD M</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVS.REPLY</td>
<td>REPLY</td>
<td>READ</td>
<td>READ</td>
<td></td>
</tr>
<tr>
<td>MVS.MODIFY.STC.<em>.</em></td>
<td>MODIFY jobname MODIFY jobname.id MODIFY id</td>
<td>UPDATE</td>
<td>UPDATE</td>
<td>-</td>
</tr>
</tbody>
</table>

Apply the RACF RDEFINE and PERMIT commands for the MVS.MCSOPER facility class which are contained in hlq.GDDRvrm.SAMPLIB(GDDCRACF).

IBM z/OS apar OA26369, which is contained in PTF UA48307 for z/OS 1.9, UA48308 for z/OS 1.10 and UA48309 for z/OS 1.11 enforces the authorization check on facility class MVS.MCSOPER.
Define GDDR user interface security

You may choose to implement role-based access controls with GDDR user interface facility profiles. Your site’s security authorization product is used to control access to the entire GDDR user interface, to selected menus, and to selected actions within menus.

Table 8  Summary of GDDR ISPF RACF permissions (1 of 2)

<table>
<thead>
<tr>
<th>Facility profile</th>
<th>Panel-ID[Function-Description]</th>
<th>GDDR$ADM group</th>
<th>GDDR$USR group</th>
<th>GDDR$REV group</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRISPF.ACCESS</td>
<td>GDDRPRIM [GDDR Primary Options Menu]</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>GDDRISPF.SETUP.*</td>
<td>GDDRMNT0 [All Setup and Maintenance Actions]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.ACCESS</td>
<td>GDDRMNT0 [Setup and Maintenance Submenus]</td>
<td>READ</td>
<td>READ</td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.PARMS.*</td>
<td>Authorizes all Parameter Edit, Validate and Activate actions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.PARMS.ACCESS</td>
<td>GDDRMNT0 [Perform GDDR Setup and Maintenance Functions]</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>GDDRISPF.SETUP.PARMS.BACKUP</td>
<td>GDDRPRM0 [Manage GDDR Parameter backups]</td>
<td>READ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.PARMS.FORCE</td>
<td>Authorizes the override of the parameter edit-in-progress serialization lock.</td>
<td>READ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.PARMS.LOAD</td>
<td>Authorizes Parameter Edit and Validation actions</td>
<td>READ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.PARMS.REVIEW</td>
<td>Authorizes Parameter Review actions</td>
<td></td>
<td></td>
<td>READ</td>
</tr>
</tbody>
</table>
Integrating GDDR

Table 8  Summary of GDDR ISPF RACF permissions (2 of 2)

<table>
<thead>
<tr>
<th>Facility profile</th>
<th>Panel-ID[Function-Description]</th>
<th>GDDR$ADMM group</th>
<th>GDDR$USR group</th>
<th>GDDR$REV group</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRISPF.SETUP.PARMS.ACTIVATE</td>
<td>GDDRPRM0 [Activate GDDR Parameter Set]</td>
<td>READ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.DEBUG</td>
<td>GDDRMNT0 [Message, Debug and Trace options]</td>
<td>READ</td>
<td>READ</td>
<td></td>
</tr>
<tr>
<td>GDDRISPF.SETUP.QUEUE</td>
<td>GDDRMNT0 [Manage GDDR Internal Command Queue]</td>
<td>READ</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| GDDRISPF.SETUP.DISCOVERHMC | GDDRMNT0 [Perform HMC Discovery]  
GDDRACT0 [Perform HMC Discovery] | READ            |                |                |
| GDDRISPF.SETUP.REFRESH | GDDRMNT0 [Refresh GDDR Message Table]                                                          | READ            |                |                |
| GDDRISPF.SETUP.AUTO | GDDRPRIM [GDDR ON / OFF Automation]  
GDDRMNT0 [GDDR ON / OFF Automation] | READ            |                |                |
| GDDRISPF.SETUP.STATE | GDDRMNT0 [Manage GDDR System variables]                                                        | READ            |                |                |
| GDDRISPF.SETUP.SITEROLES | GDDRRMFXR [Transfer Master C-System]                                                            | READ            |                |                |
| GDDRISPF.SCRIPTS.* | Authorizes all Script management actions                                                        | READ            |                |                |
| GDDRISPF.SCRIPTS.CHECKUP | GDDRPRIM [Perform pre-script checkup]                                                            | READ            | READ           |                |
| GDDRISPF.SCRIPTS.RUN.ACCESS | GDDRPRIM [Run GDDR Scripts]                                                                    | READ            |                |                |
| GDDRISPF.SCRIPTS.MANAGE,JCL | GDDR0PRJ [Jobcards for your user]                                                                    | READ            | READ           |                |
| GDDRISPF.SCRIPTS.VIEWSTATS | GDDRPRIM [View GDDR Script Statistics]                                                            | READ            |                |                |
| GDDRISPF.VIEW.CONFIG | GDDRPRIM [View GDDR configuration]                                                                | READ            | READ           |                |
| GDDRISPF.ACTIONS.* | Authorizes all GDDR CBU, HMC and SYSPLEX actions                                                                 | READ            |                |                |
| GDDRISPF.ACTIONS.ACCESS | GDDRPRIM [Perform GDDR Actions]                                                                   | READ            | READ           |                |
| GDDRISPF.ACTIONS.CBU.ACCESS | GDDRACT0 [Perform HMC CBU actions]                                                                 | READ            | READ           |                |
| GDDRISPF.ACTCBU.site.cpc | Authorizes CBU actions by SITE and CPC                                                              | READ            | READ           |                |
| GDDRISPF.ACTIONS.HMC.ACCESS | GDDRACT0 [Perform HMC LPAR actions]                                                              | READ            | READ           |                |
| GDDRISPF.ACTIONS.HMC._system_name | Authorizes HMC actions by system name                                                              | READ            | READ           |                |
| GDDRISPF.ACTIONS.SYSPLEX | GDDRACT0 [Manage Couple Datasets]                                                                 | READ            | READ           |                |

Member GDDRACF in hlq.GDDRvrm.SAMPLIB list the RACF commands used for EMC GDDR ISPF permissions. These commands are optionally used to protect GDDR functions.

Authorize RACF for HMC LPAR actions

The facility profile GDDRISPF.ACTIONS.HMC._system_name provides for site-specific authorization of GDDR LPAR processing. Authorization is enabled through the GDDR panel “Option L: Perform HMC LPAR Actions” shown on page 216. Following your specification of GDDRISPF.ACTIONS.HMC._system_name facility profiles, the Perform HMC LPAR Actions panel will only list the system names for actions by authorized GDDR users. Unprotected system names will display for any EMC GDDR user.
The value of `system_name` corresponds to the name found in the system field of the Define Managed LPARs panel shown on page 151. You can use generic or discrete system names in the facility profile.

The following rules apply to generic profile names, where valid generic characters are *, %, and **:

- Specify % in the profile name to match any single non-blank character (except a period) in the same position of the system ID name.
- Specify * or ** in the profile name to match more than one character in the same position of the system ID name.

The z/OS Security Server RACF Command Language Reference, SA22-7687, provides detailed descriptions and examples that illustrate how to specify generic profile names.

**Note:** The sample GDDRISPF.ACTIONS.HMC facility profile definition, supplied with hlq.GDDRVrm.SAMPLIB member GDDIRACF authorizes all systems to users who are connected to the GDDR$USR group.

**Examples**

The following examples demonstrate the use of the GDDRISPF.ACTIONS.HMC facility profile:

```plaintext
RDEFINE FACILITY GDDRISPF.ACTIONS.HMC.** UACC(NONE) OWNER(GDD$ADM)
PERMIT GDDRISPF.ACTIONS.HMC.ZOSESYS* CLASS(FACILITY) ACCESS(READ) ID(GDDR)
```

In the above example, the GDDR HMC LPAR Actions panel at a given GDDR C-System will display all system names defined in the System field of the Define Managed LPARs panel that match the generic name ‘ZOSESYS*’ for all users connected to the group ‘GDDR’. System names which do not match ‘ZOSESYS*’ are protected and will not be displayed on any GDDR HMC LPAR Actions panel.

```plaintext
RDEFINE FACILITY GDDRISPF.ACTIONS.HMC.ZOSESYS1 UACC(NONE) OWNER(GDD$ADM)
PERMIT GDDRISPF.ACTIONS.HMC.ZOSESYS1 CLASS(FACILITY) ACCESS(READ) ID(SYSPGM1)
```

In the above example, the GDDR HMC LPAR Actions panel at a given GDDR C-System will display only system name ZOSESYS1 for user SYSPGM1. All other systems defined in the System field of the Define Managed LPARs panel are not protected and therefore will be displayed on the GDDR HMC LPAR Actions panel.

**Authorize the Mainframe Enablers EMCSAFI security interface**

If you have protected access to Consistency Group and SRDF Host Component commands, authorize the GDDR userid and the GDDR$STC group with SAF profiles described in the configuration chapters of the EMC Consistency Groups for z/OS Product Guide and EMC SRDF Host Component for z/OS Product Guide.

If TimeFinder/Mirror for z/OS and TimeFinder/Mirror clone emulation are in use and commands are protected, the GDDR userid and the GDR$STC group also require authorization.

Similarly, if devices to be managed by GDDR have been protected using the EMCSAFI security interface of EMC ResourcePak Base for z/OS, authorization must be added for the GDDR userid and the GDDR$STC group.
Integrating GDDR

The *EMC Mainframe Enablers Installation and Configuration Guide* provides information about the security interface and the class and resource names used.

Verify the module and RACF-protected resource authorization

After completing the steps listed in “SYS1.PARMLIB(IKJTSOxx) customization” on page 48 and in “Specify GDDR security” on page 54, check that GDDR load modules are properly authorized and accessible and that RACF resources have been properly defined and authorized using member GDDRECHK in *hlq.GDDRvrm.SAMPLIB*.

The RACF-protected resource authorization verification utility (jobname GDDRECHK) provides a means for the RACF administrator to verify that the site's security rules have been properly defined for support of GDDR processing. The utility validates authorizations using a control statement specifying group, (GDDR$STC, GDDR$ADM, and GDDR$USR), site, CPC, and LPAR. The GDDRECHK job validates authorization of the user associated with the job against the RACF definitions on the system where the job is run. Therefore, the GDDRECHK job must be submitted on each C-System and managed system. EMC recommends use of the USERID jobcard parameter with the GDDRECHK job when the security administrator validates authorization for GDDR administrators and operations personnel.

“RACF-protected resource authorization and verification SAMPLIB inventory” on page 64 lists the members provided for use in defining and validating security authorization.

Perform the following steps to use the GDDRECHK job:

1. Customize the jobcard to match your site's requirements. If the verification is to be performed for a userid other than the submitter, include jobcard parameter USERID=, to specify the desired userid.

2. Specify a value for GRP= from those documented in Table 6, “RACF permissions,” on page 55.

   For example:
   
   GRP=GDDR$ADM for GDDR administrators
   
   GRP=GDDR$USR for GDDR operations users
   
   GRP=GDDR$STC for GDDR started tasks
   
   GRP=GDDR$REV for GDDR review users (optional)
   
   GRP= for non-GDDR users to check all GDDR resources have been properly protected.

   **Note:** Steps 3, 4, 5, and 6 and the substitution value for MFEPFX in step 7 are not applicable to managed systems.

3. Specify the CP1= value that was used from SAMPLIB(GDDIRACF), or specify a list of CPC names.

4. Specify the CP2= value that was used from SAMPLIB(GDDIRACF), or specify a list of CPC names.

5. Specify the CP3= value that was used from SAMPLIB(GDDIRACF), or specify a list of CPC names.
6. Specify the SYS= value that was used from SAMPLIB(GDDIRACF), or specify a list of system names.

7. Specify dataset name prefixes, as used to define dataset profiles in SAMPLIB(GDDCRACF):

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRPFX=</td>
<td>dsname prefix for GDDR installed datasets</td>
<td>EMC.GDDR500</td>
</tr>
<tr>
<td>MFEPFX=</td>
<td>dsname prefix for EMC Mainframe Enablers installed datasets</td>
<td>EMC.MFE800</td>
</tr>
</tbody>
</table>

8. Run the GDDRECHK job on each C-System and managed system to verify RACF definitions.

Returns: 0 - Check the output in SYSTSPRT for RACF error messages

The following is an example of the parameter statements.

```//jobname  JOB (EMC),'GDDRECHK',CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1), NOTIFY=&SYSUID  //GDDCHECK PROC GRP=GDDR$ADM,           <=== CPC NAMES TO CHECK AT DC1  //             CP1=CPC1,                <=== CPC NAMES TO CHECK AT DC2  //             CP2=CPC2,                <=== CPC NAMES TO CHECK AT DC3  //             SYS=VC1B,                <=== SYSTEM NAMES TO CHECK  //             GDDRPFX=EMC.GDDR500,     <=== GDDR DSN PREFIX  //             MFEPFX=EMC.MFE800        <=== MF ENABLERS DSN PREFIX  //**************************************************************************  //```  

GDDRECHK produces a listing which shows the result of GDDR SAF authorizations by OPERCMDS, SURROGAT, TSOAUTH, JESJOBS, and FACILITY classes. The GDDRECHK program is used to check that the security rules (RACF/ACF2) have been set up correctly for a given security ID (specified on the job card USER parameter, or defaulted to the TSO user submitting the job) and a given GDDR security role (specified in the PARM='GROUP(GDDR$xxx)' field). GDDRECHK needs to be run separately for every combination of GROUP(GDDR$xxx) and user ID, to verify all security has been implemented correctly.

Figure 4, “Validation of GDDR$ADM group access by facility,” and Figure 5, “Validation of specific calls from GDDR modules,” on page 63 provide examples of the GDDRECHK program. The following fields are displayed in the output:

- **CLASS**
  This column specifies a valid class name defined in the RACF class descriptor table associated with the RESOURCE NAME. The GDDRECHK utility expects DATASET or FACILITY.

- **RESOURCE NAME**
  For a particular GDDR security role, shown in the report at the top on the line ‘Testing GDDR$xxx group access’, this column specifies the discrete or generic DATASET or FACILITY profile name to be validated.

- **ACCESS**
  The ACCESS column shows what access the user SHOULD have ('READ', 'UPDATE', 'NONE').
**GDDSECUR RESULT**

The GDDSECUR RESULT column shows the response from the security system (a SAF call) to an inquiry to check the indicated ACCESS for the indicated resource (FACILITY and RESOURCE NAME). This can be ‘ALLOW’, ‘DENY’, or ‘RESOURCE NOT PROTECTED’.

**COMMENT**

The comment column shows the value ‘Ok’ if the access is correct (security system correctly allowed or denied access), or ***ERROR*** if the access is not correct (the user has too much or too little access).

If the user has too much access, ***ERROR*** will be reported when either:

1. The resource is not protected at all
2. The particular user ID has more privileges to access the indicated resource than the user should have (defined by the value specified on the PARM='GROUP(GDDR$xxx)' field).

If the user has too little access, ***ERROR*** will be reported when

- The security system returns DENY (for ACCESS=READ or ACCESS=WRITE)

An ***ERROR*** is always returned when a resource that should be protected is not protected by your security system (security system returns: RESOURCE NOT PROTECTED).

All lines flagged with ***ERROR*** should be investigated and fixed, to get a clean report with all ‘Ok’ values in the COMMENT column.

GDDTSECR - Test RACF security access
GDDTSECR version OPT338524 2010/07/14

Parameter passed to GDDTSECR: "GROUP(GDDR$ADM) SITE(DC1) LPAR(VC1B) CPC(CPC1)"

20100714 17:30:16 GDDSECUR test started

Testing GDDR$ADM group access

<table>
<thead>
<tr>
<th>CLASS</th>
<th>RESOURCE_NAME</th>
<th>ACCESS</th>
<th>GDDSECUR RESULT</th>
<th>COMMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.ACCESS</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.*</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.ACCESS</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.PARMS.*</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.PARMS.ACCESS</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.PARMS.BACKUP</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.PARMS.LOAD</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.PARMS.REVIEW</td>
<td>NONE</td>
<td>ALLOW</td>
<td>*** ERROR ***</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.PARMS.ACTIVATE</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.DEBUG</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.QUEUE</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.DISCOVERHMC</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.REFRESH</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.AUTO</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.STATE</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SETUP.SITEROLES</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SCRIPTS.*</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SCRIPTS.CHECKUP</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SCRIPTS.RUN.ACCESS</td>
<td>NONE</td>
<td>ALLOW</td>
<td>*** ERROR ***</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SCRIPTS.MANAGE.JCL</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.SCRIPTS.VIEWSTATS</td>
<td>NONE</td>
<td>ALLOW</td>
<td>*** ERROR ***</td>
</tr>
<tr>
<td>FACILITY</td>
<td>GDDRISPF.VIEW.CONFIG</td>
<td>READ</td>
<td>ALLOW</td>
<td>Ok</td>
</tr>
</tbody>
</table>
Integrating GDDR

Figure 4  Validation of GDDR$ADM group access by facility

Testing specific calls from GDDR modules

<table>
<thead>
<tr>
<th>CLASS</th>
<th>RESOURCE NAME</th>
<th>ACCESS</th>
<th>GDDSECUR</th>
<th>RESULT</th>
<th>COMMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>FACILITY GDDBSP.ACTCISPF.ACTIONS.*</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.ACTCISPF.ACTIONS.ACCESS</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.ACTCISPF.ACTIONS.CBU.ACCESS</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.ACTCISPF.ACTIONS.HMC.ACCESS</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.ACTCISPF.ACTIONS.HMC.VC1B</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.ACTCISPF.ACTIONS.SYSPLEX</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.ACTCISPF.OPS.ACCESS</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.GLOBAL.VARIABLE.ACCESS</td>
<td>UPDATE ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.HMC.LISTOBJECTS</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FACILITY GDDBSP.HMC.GETMSGS</td>
<td>READ ALLOW</td>
<td>Ok</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5  Validation of specific calls from GDDR modules

20100714 17:31:28 GDDSECUR test ended - MAXRC=12 Elapsed=71.785423 seconds

GDDTSECR ending - maximum RC=12
RACF-protected resource authorization and verification SAMPLIB inventory

Table 9 provides an inventory of hlq.GDDRvrm.SAMPLIB members used for defining and validating GDDR security controls.

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDCRACF</td>
<td>Define RACF C-System resources</td>
</tr>
<tr>
<td>GDDCRACD</td>
<td>Delete RACF C-System resources</td>
</tr>
<tr>
<td>GDDCRACJ</td>
<td>JCL to delete/define C-System resources</td>
</tr>
<tr>
<td>GDDIRACF</td>
<td>Define RACF GDDR ISPF resources</td>
</tr>
<tr>
<td>GDDIRACD</td>
<td>Delete RACF GDDR ISPF resources</td>
</tr>
<tr>
<td>GDDIRACJ</td>
<td>JCL to delete/define GDDR ISPF resources</td>
</tr>
<tr>
<td>GDDPRACF</td>
<td>Define RACF P-System resources</td>
</tr>
<tr>
<td>GDDPRACD</td>
<td>Delete RACF P-System resources</td>
</tr>
<tr>
<td>GDDPRACJ</td>
<td>JCL to delete/define P-System resources</td>
</tr>
<tr>
<td>GDDRECHK</td>
<td>Verify GDDR load module authorizations and RACF access authorizations</td>
</tr>
</tbody>
</table>

Define GDDR datasets

GDDR C-Systems utilize permanent global variable data-in-virtual (DIV) and parameter backup and parameter wizard work datasets. Catalog these datasets on each C-System using the jobs supplied in hlq.GDDRvrm.SAMPLIB.

Define global variable datasets

- Define a VSAM linear dataset to manage global variables on each C-system. The recommended dataset name convention is hlq.GDDRvrm.csys.DIV. JCL has been provided in hlq.GDDRvrm.SAMPLIB(GDDIVDEF).

Allocate the parameter management datasets

Note: EMC recommends the use of two parameter backup datasets—one for parameter management functions and the other for backups performed when the Heartbeat Monitor initializes.

1. Customize and run the job in member GDDRABDS in hlq.GDDRvrm.SAMPLIB. Ensure it has run successfully.

- The parameter backup dataset name to be used for parameter management functions is defined to GDDR in your personal GDDR ISPF Profile, Option P on the GDDR Primary Options Menu. This dataset is used when you create a backup of GDDR parameters using the GDDR ISPF Interface, Option B, Manage GDDR Parameter backups, from the GDDR Parameter Management Options menu. It also used for the implicit backups created before and after the update of GDDR parameters during an Activate. Refer to “Option P: Profile—Update Personal GDDR ISPF Profile” on page 108 for assistance.
• The parameter backup dataset to be used for Heartbeat Initialization backups is defined to EMC GDDR in the Define GDDR Datasets panel using the BKUPVARS field type. Refer to “Option D: Define GDDR Datasets” on page 136 for details.

• The parameter wizard work dataset for your userid is a PDS with the same attributes as a parameter backup dataset, which serves as a temporary store for EMC GDDR parameters in preparation of a GDDR parameter activation.

• The parameter wizard work dataset must be allocated as a PDS, with attributes FB and LRECL=80.

• The last activated parameter dataset has the same allocation parameters as the parameter wizard work dataset and contains a copy of the work PDS used during the most recent activation.

2. Allocate or choose a PDS, FB, LRECL=80, to store GDDRMAIN parameters. You could use hlq.GDDRvrm.PARMLIB for this purpose.

### Install GDDR started procedures

The GDDRMAIN and Event Monitor tasks must be started and remain running on the GDDR C-Systems at all times. Also, GDDRMAIN tasks must be started and remain running on all GDDR-managed systems at all times. Before using GDDR, you must customize these started procedures and make them available. On GDDR C-Systems, GDDRMAIN will automatically start GDDRWORK and GDDREV started tasks.

**Note:** Use automation to start the GDDR procedures on the GDDR C-Systems at system startup. No GDDR started tasks are to be started SUB=MSTR.

1. Update members GDDRMAIN, GDDRWORK, and GDDREV in hlq.GDDR.vrm.PROCLIB so the following DD statements point to the datasets resulting from your SMP/E installation: ISPPLIB, ISPMLIB, ISPSLIB, and SYSTSIN.

2. Update member GDDRMAIN in hlq.GDDR.vrm.PROCLIB so the GDDREXEC DD statement points to the hlq.GDDRvrm.REXX dataset resulting from your SMP/E installation, and the GDDMPARM DD points to the GDDRMAIN parameter dataset (as described in step 2 on page 65).

3. Make the GDDR started procedures available to the C-Systems by copying members GDDRMAIN, GDDRWORK, and GDDREV from hlq.GDDR.vrm.PROCLIB to SYS1.PROCLIB or equivalent library for started tasks.

4. Copy member GDDREVMP from hlq.GDDRvrm.PARMLIB to the parmlib you have referenced in your GDDREVMP started procedure member.

5. Make the GDDRMAIN started procedure available to the GDDR-managed systems by copying member GDDRMAIN from hlq.GDDRvrm.PROCLIB to SYS1.PROCLIB or equivalent library for started tasks on the respective system.

6. If you plan to use customized versions of GDDR user exits, include a SYSEXEC DD statement referencing your user RCXFE library in the GDDRPROC procedure member.

**Note:** “Sample procedure to use interpreted REXX user exits” on page 366 provides more information.
7. Ensure GDDR started tasks connect to the appropriate SCF instance.

The GDDRMAIN started task must be associated with a specific SCF instance. This is accomplished by providing an identical \$SCF$xxx DD DUMMY statement in the started procedure of the SCF started task and each of the GDDR started tasks. The value substituted for xxx is a user-supplied value that is unique to the SCF instance and the GDDR started procedures.

Note: The default procs supplied for GDDRWORK and GDDRPROC refer to the hlq.GDDRvrm.ISPPROF library allocated during installation. Even though GDDR does not use or save any ISPF profile variables, this ISPPROF library is necessary to be able to run ISPF in batch, as ISPF will not start without a //ISPPROF DD statement pointing to a PDS.

Install GDDR Licensed Feature Code

To use GDDR, you must install a Licensed Feature Code (LFC). An LFC is a 16-character alphanumeric string that is attached to product or a feature within a product. The LFC is provided on the Licensed Feature Authorization Card included with GDDR.

- Enter the key in the initialization parameters file of ResourcePak Base as described in the ResourcePak Base for z/OS Product Guide. For additional support, email licensekeys@emc.com.

Customize GDDRMAIN parameters

The GDDRMAIN parameters are the global variable data in virtual (DIV) datasets, C-System system names and sites, communication parameters, CPC and SYMM parameters, and worker tasks. These parameters are defined in GDDMPARM members, and propagated to be available to C-Systems and GDDR-managed systems at installation.

GDDRMAIN parameters must also be maintained as environment changes occur. Environment changes which impact GDDRMAIN are IP address and IP port changes (COMM parameters), adding or removing storage controllers (SYMM parameters), and adding or removing a system (CSYSSITE and CPC parameters). See Chapter 9, “Performing Maintenance Procedures,” for assistance with these changes.

IMPORTANT

GDDMPARM consistency—GDDMPARM contents are crucial to GDDRMAIN operation. If systems have different values for system names or IP addresses, communication between GDDRMAIN instances may be impossible. Therefore, GDDRMAIN takes steps to ensure that the GDDMPARM dataset in use by all systems is identical.

Install GDDMPARM

You must complete GDDMPARM installation before attempting to use the GDDR ISPF interface. In general, GDDR-managed systems do not have DASD which is shared by all systems. Some systems may have common DASD (such as production systems at one site), but none are required to. In a typical installation, C-Systems may not share DASD with managed systems within a site. Therefore, you will maintain multiple copies of GDDMPARM.
The following procedure describes how to define and maintain systems so that they all reference consistent copies of GDDMPARM.

1. Create a GDDMPARM member in the designated parameter library on one system by editing the provided sample.

   Include a DIV dataset definition for each C-System (GVDIVDSN), a definition of each C system (CSYSSITE), an IP communications definition for every system (COMM), CPC parameters for every CPC where C-systems or GDDR managed systems may run, SYMM parameters for all GDDR-managed VMAX systems, and optionally, any MSG and WORKER statements needed. Refer to the requirements for each of these parameters in "Enable or disable message processing" on page 69, "Automate messages" on page 69, "Modify worker parameters" on page 73, “Update COMM statements” on page 74, “CPC parameters” on page 76, and “SYMM parameters” on page 77.

2. When the initial dataset is ready, start GDDRMAIN on the system where you created the dataset.

3. Check for any parameter error messages at startup. If there are any messages, stop GDDRMAIN and correct the dataset.

4. Once GDDRMAIN starts without problem, issue the F GDDRMAIN,COMM command and check the list of systems and IP addresses and ports displayed to be sure the list is complete and all addresses are correct.

   The following example shows COMM command output:

   GDDM127I Communications Status
   Sys PRD1, IP nnn.nnn.nnn.nnn,9876 Last *NEVER*
   Sys PRD2, IP nnn.nnn.nnn.nnn,9876 Last *NEVER*
   Sys PRD3, IP nnn.nnn.nnn.nnn,9876 Last *NEVER*
   Sys SYS3 (DC3), IP nnn.nnn.nnn.nnn,9876 Last *NEVER*
   Sys SYS1 (DC1), IP nnn.nnn.nnn.nnn,9876 Last *NEVER*

5. When the COMM command output is correct, propagate the GDDMPARM dataset to all managed systems (C and P systems).

   If you have systems which have common DASD, you do not have to propagate the dataset to each system, but each system must have a copy that is accessible. EMC recommends that you use an FTP job or exec to accomplish this, and keep the job or exec for future use.

6. Start GDDRMAIN on all the other systems.

7. Check for parameter error messages at startup. If there are error messages, stop GDDRMAIN on all systems and go back to step 2.

8. Once all GDDRMAIN instances start without error, issue F GDDRMAIN,MPARM on any system.

   - Check the command output to be sure that a valid data line is shown for each system (no occurrences of “unable to communicate”). On each line, be sure that the values shown for in-use and dataset are the same. Check that those values are the same on each line for every system.
Integrating GDDR

- If any values are different, or if any show “unable to communicate”, you have not propagated the GDDMPARM member correctly to the systems with different values. You must copy the original GDDMPARM to those systems and restart the COMM task of GDDRMAIN on those systems. After doing this, repeat this step.

Note: You only need to execute the MPARM command on one C-System or managed system. If the system list is complete and all systems respond, the output would be the same on every system.

The following example shows MPARM command output:

```
GDDM140I GDDMPARM Status
Sys PRD1 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
Sys SYS3 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
Sys SYS2 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
Sys SYS1 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
```

At this point, all GDDRMAIN instances are up and are consistent.

Timed GDDMPARM consistency checks

GDDRMAIN checks the consistency of the GDDMPARM data in use every time the COMM subtask is started. In addition, you can perform a check at any time by using the F GDDRMAIN,MPARM,CHECK command.

GDDRMAIN also checks for consistency every eight hours and checks that the GDDMPARM data matches the data in use by the COMM subtask. If the data does not match, message GDDM144W is issued and the checks are made every hour until they are found to be equal, at which time the interval reverts to eight hours.

Note: Message GDDM144W may not represent a problem if you are in the process of propagating an updated GDDMPARM, but have not yet activated it by restarting COMM.

Correct any inconsistencies in GDDMPARM

During operation, GDDRMAIN checks that the GDDMPARM data remains consistent. If a discrepancy is found, message GDDM141E will be issued and Degraded Mode is set. If this occurs, complete the following steps:

1. Issue the F GDDRMAIN,MPARM command to identify systems that are out of synchronization.
2. Correct the GDDMPARM data on those systems, and restart the COMM subtask on those systems.
3. If this action does not solve the problem, issue F GDDRMAIN,MPARM,CHECK from any GDDR-managed system.

The MPARM,CHECK command retrieves the in-use data from each system and compares them all. If all are equal, consistency is restored and Degraded Mode is turned off. The MPARM,CHECK command will tolerate no response from production systems, but must receive a response from each C-System in order to declare consistency. If any system (C-System or production system) responds with a different in-use value, consistency will not be declared and error messages will be issued.

The warning for inconsistent GDDMPARM data, indicated by message GDDM141E is:
Enable or disable message processing

Specify your message status preference for selected message by adding a statement in the GDDRMAIN parameters (GDDMPARM) to enable or disable the messages. This statement is processed when GDDRMAIN is restarted, thus keeping the desired enable/disable state over GDDRMAIN restarts.

The syntax for the MSG/MESSAGE statement in GDDMPARM is as follows:

```
MSG  system  GDDR  msgid,ENABLE|DISABLE
```

Where:
- `system` is a specific system name or an asterisk (*) indicating all systems.
- `msgid` is the ID of the message to be manipulated.
- `ENABLE` indicates the message should be enabled.
- `DISABLE` indicates the message should be disabled.

**Note:** ENABLE and DISABLE may be abbreviated to any length (for example, ENABLE can be specified as E, EN, or ENA).

For example, you may want to enable processing of messages GDDX191I and GDDX291I to trigger action on your production systems (these messages are disabled by default):

```
MESSAGE  *        GDDR GDDX191I,ENABLE
MESSAGE  *        GDDR GDDX291I,ENABLE
```

Automate messages

You can define message automation rules by adding a statement in the GDDRMAIN parameters (GDDMPARM). In this statement, you can either define a message ID for GDDRMAIN to intercept and process or you can define an additional action for a message that GDDRMAIN already intercepts.

For example, you could add a new message that forwards the message to other systems in the GDDR sysplex for processing, or runs a specified REXX routine. Or you could modify an existing GDDR-defined message to specify a REXX routine to be executed after all GDDR-defined actions have been taken for the message.

If the user message automation facility is used, a new type of worker will be started in a GDDRWORK address space. This worker (the GDDWCX) processes all user-defined actions. The normal console message worker (GDDWCM) runs as a subtask in the GDDRMAIN address space. Using a worker in a GDDRWORK address space protects GDDRMAIN from the user defined REXX routine and any modules it might execute.

The syntax for the MSG/MESSAGE statement in GDDMPARM is as follows:

```
MSG  system  GDDR  msgid,ADD|MODIFY[,,actions]
```

Where:
- `system` is a specific system name or an asterisk (*) indicating all systems.
- `msgid` is the ID of the message to be manipulated.
**ADD** indicates the message is being added to the table of messages that GDDRMAIN intercepts.

**MODIFY** indicates the message is already defined (either already present in the GDDRMAIN-supplied table or added by a previous ADD statement).

**Note:** ADD and MODIFY may be abbreviated to any length (for example, MODIFY can be specified as M, MO, or MOD).

Available MODIFY **actions** are as follows. If you need more actions than will fit on one statement, you can use additional MODIFY statements to add more.

<table>
<thead>
<tr>
<th>action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROCESS=ONCSYS</td>
<td>ONPSYS</td>
</tr>
<tr>
<td>ONCSYS</td>
<td>The message can be processed on any C-System.</td>
</tr>
<tr>
<td>ONPSYS</td>
<td>The message can be processed on any production system.</td>
</tr>
<tr>
<td>MASTONLY</td>
<td>The message can be processed only on the master C-System.</td>
</tr>
<tr>
<td>LOCAL</td>
<td>Process the message wherever it occurs. This is the default if PROCESS is not specified or is implied by FORWARD.</td>
</tr>
<tr>
<td>FORWARD=TOCSYS</td>
<td>TOPSYS</td>
</tr>
<tr>
<td>TOCSYS</td>
<td>If the message occurs on a production system, it will be forwarded to every C-System, including the master. If the message occurs on a C-System, it will not be forwarded.</td>
</tr>
<tr>
<td>TOPSYS</td>
<td>If the message occurs on a C-System or Master C-System, it will be forwarded to every production system. If the message occurs on a production system, it will not be forwarded.</td>
</tr>
<tr>
<td>TOMAST</td>
<td>If the message occurs on a C-System which is not the master, it will be forwarded to the Master C-System. If the message occurs on a production system, it will be forwarded to every C-System, including the master. If the message occurs on the Master C-System, it will not be forwarded.</td>
</tr>
<tr>
<td>NONE</td>
<td>Do not forward the message. This is the default unless PROCESS is specified and implies a value above.</td>
</tr>
</tbody>
</table>
Integrating GDDR

Use the UMA (user message automation) console command to set the message automation state. The syntax is as follows:

```bash
F GDDRMAIN,UMA[,option]
```

Where `option` can be one of the following:

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null</td>
<td>Display current user message automation status (ON or OFF).</td>
</tr>
<tr>
<td>ON</td>
<td>Turn on user message automation (allow processing of user-defined actions).</td>
</tr>
<tr>
<td>OFF</td>
<td>Turn off user message automation (do not allow processing of user-defined actions).</td>
</tr>
<tr>
<td>RELOAD</td>
<td>Turn on user message automation and restart the MCSOPER subtask of GDDRMAIN with indication that the message table should be reloaded and MSG parameters should be processed. Note that this reloads all messages, not just user-defined messages.</td>
</tr>
</tbody>
</table>

Broadcast console commands

GDDRMAIN allows you to broadcast an operator command from any EMC GDDR-managed system (where GDDRMAIN is running and in communication with other instances of GDDRMAIN) to all or selected GDDR systems.

The syntax is as follows:

```bash
F GDDRMAIN,BC[BR],<to>,<cmd>
```

Where `<to>` is one of the following:

- `System-name` The system on which `<cmd>` is to be executed
- `Site-name` The site on which `<cmd>` is to be executed
- `*` All GDDR-managed systems
- `*C` All C-Systems
- `*P` All production systems
- `*1` All GDDR systems, this system first
- `*X` All GDDR systems except this system
- `*Z` All GDDR systems, this system last

---

Note: FORWARD and PROCESS must be consistent with each other. For example, you cannot forward a message to production systems and specify process on C-Systems. PROCESS=LOCAL is not allowed with any value of FORWARD except NONE.
<cmd> is the command to be executed on the system.

BC (Broadcast) does not return any output of the executed command. It also does not indicate if the system actually received or processed the command. It does provide a message for each system to which it sends the command.

BR (Broadcast and Response) returns the output of the executed command.

In the case of IP port change above, COMM restart can be handled via the F GDDRMAIN,BC,*Z,F GDDRMAIN,RESTART COMM command. (If GDDRMAIN is the name of the GDDRMAIN address space on all systems.) Note that *Z is required, because the command cannot be broadcast from the system where it is issued if that system is already using the new ports and the other systems are not.

When adding a new system, COMM restart can be handled by issuing F GDDRMAIN,BC,*X,F GDDRMAIN,RESTART,COMM from the new system.

When deleting a system, COMM restart can be handled by issuing F GDDRMAIN,BC,*F GDDRMAIN,RESTART COMM. Note that *1 or *Z could be used here as well. For example:

```
BC,*D T
GDDM145I Command sent to system VC1D
GDDM145I Command sent to system VC19
GDDM145I Command sent to system 001E
GDDM145I Command sent to system VC1A
GDDM145I Command sent to system 0016
GDDM149I Issuing D T for system VC1A Job GDDRMAIN (STC05841)
D T
IEE136I LOCAL: TIME=04.25.36 DATE=2010.033 UTC: TIME=09.25.36 DATE=2010.033
```

On each system receiving a command via BC, message GDDM149I is issued to identify the source of the broadcast. The following is the result on system 0016:

```
GDDM149I Issuing D T for system VC1A Job GDDRMAIN (STC05841)
D T
IEE136I LOCAL: TIME=04.25.36 DATE=2010.033 UTC: TIME=09.25.36 DATE=2010.033
```

Specify GDDR data in virtual dataset names

The DIV dataset names are conveyed to GDDRMAIN via the GVDIVDSN parameter statement of the GDDMPARM member. The DIV datasets serve as a permanent store of GDDR global variables.

If you need to change the DIV dataset, refer to the instructions in paragraph “Changing the global variable DIV dataset or WORKER parameters” on page 360.

- Add a line to the GDDMPARM member of hlq.GDDRvrm.PARMLIB defining the DIV dataset name for the system and GDDR instance.
The format of the lines in GDDMPARM is as follows:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>The 8-character GVDIVDSN constant. Statements with an asterisk (*) in column 1 are comments and are ignored.</td>
</tr>
<tr>
<td>10-17</td>
<td>The -character MVS system name of a GDDR C-System which is specified using the SYSTYPE=system-name statement in SYS1.PARMLIB(EASYSO0) or equivalent parameter file. Note that this is not the SMF id of the system (although it could be the same).</td>
</tr>
<tr>
<td>19-22</td>
<td>The 1 to 4-character left-justified name of the GDDR instance to which this applies. This supports multiple GDDR instances on a single C-System. This field will always contain GDDR.</td>
</tr>
<tr>
<td>24-67</td>
<td>The left-justified dataset name of the DIV.</td>
</tr>
</tbody>
</table>

For example:

```
GVDIVDSN -SYSTEM- GDDR -DIV DATA SET NAME--------------------------
GVDIVDSN SYS1     GDDR HLQ.GDDRVRM.SYS1.DIV
GVDIVDSN SYS2     GDDR HLQ.GDDRVRM.SYS2.DIV
GVDIVDSN SYS3     GDDR HLQ.GDDRVRM.SYS3.DIV
```

Modify worker parameters

Worker parameters supplied in GDDMPARM specify the names of the worker tasks and the minimum and maximum number of worker tasks per system.

If you need to change any worker parameters, refer to the instructions in “Changing the global variable DIV dataset or WORKER parameters” on page 360.

Worker parameter names and their functions are as follows:

- **GDDWCM**  Console message processing
- **GDDWCO**  Command processing
- **GDDWCX**  User message automation worker
- **GDDWDV**  Storage device worker
- **GDDWGV**  Global variable processing
- **GDDWST**  Status or information retrieval
- **GDDWSX**  Restricted status processing
- **GDDWXH**  REXX hardware management console interaction
- **GDDWXQ**  Host Component command queue processing
- **GDDWXR**  Message interception rule processing

- MIN is the minimum number to start (blank for no change).
- MAX is the maximum number to allow (blank for no change).
- MIN and MAX must be three digits.
- MIN must be >= 1.
- MAX must be >= MIN.
- On a production system, only GDDWCM, GDDWCO, GDDWST, GDDWSX, and GDDWCX are allowed. GDDWXH may be allowed if the managed system handles BCPII requests.
Integrating GDDR

**Note:** EMC recommends that you do not specify these parameters in your GDDMPARM member. This maintains the minimum and maximum numbers for all workers at their default values, which should be appropriate for most configurations.

**Update COMM statements**

GDDR inter-system communication is defined via COMM statements in GDDMPARM. The COMM parameters specify the IP addresses and ports for GDDRMAIN to use.

You can specify two IP addresses per statement and multiple COMM statements per system. This allows GDDRMAIN to communicate with LPARs that are not at their designated location.

Having multiple COMM statements for a system also allows for situations where the same system may present different IP addresses to other systems based on firewalls or other network considerations. For example, system A may appear to have IP 1.2.3.4 to system B, but may appear to have IP 5.6.7.8 to system C. Communication with systems not at their designated location is also a consideration of the DRTCOMM parameter statement.

**Note:** To take advantage of the new features, you must update the COMM statements in GDDMPARM. However, existing COMM definitions from releases prior to GDDR V4.2 can be used without change.

The COMM statement in GDDMPARM is as follows:

```
COMM system GDDR ip-address, port[,recv-ip-address]
```

Where:

- **system** The z/OS system name of a GDDR-managed system.
- **ip-address** The “send to” address. If **recv-ip-address** is not provided, this address is used for both send and receive.
- **port** The port number.
- **recv-ip-address** The “receive from” address (optional).

**Note:** You can select any port number from the site’s unregistered ports, but the selected port number must be the same for all systems.

The address used is based on incoming communication. When a message is received, it is matched to a “receive from” address of a system. If that is not the current “receive from” system address, communication with that system is switched to use this “receive from” address and the associated “send to” address. Thus IP address selection is dynamic and requires no intervention.

When the COMM subtask initializes, it sends a null message to each “send to” address of every other system. When received, this message causes the receiving system to send back a different null message, thus informing the original system of what IP address is in use for the system. If the IP changes while COMM is up, it will recognize the change and update its control blocks when it receives a message from the system which changed. Communication can always be forced via operator command (MPARM or BC/BR commands).
Integrating GDDR

Specify a DR test COMM statement

A DRTCOMM statement is allowed in GDDMPARM. This statement allows communication with an LPAR that is participating in a disaster recovery (DR) test. The syntax is as follows:

DRTCOMM system GDDR site,ip-address,port[,recv-ip-address]

*site* is the site at which this system would be running during a DR test. The remainder of the statement values are identical to the COMM statement values.

At least one COMM statement for the system must precede the DRTCOMM statement.

Specify the system name

You can use a COMM console command to specify an optional system name. The syntax of the COMM command is as follows:

F GDDRMMAIN,COMM[,system]

- If *system* is not specified, the command output is unchanged from GDDR releases prior to V4.2.
- If *system* is specified, all input IP addresses are shown along with any DRTCOMM definitions.

**Note:** If you need to change any COMM parameters, refer to the instructions in “Changing the GDDR C-System or GDDR managed system IP port” on page 351.

CSYSSITE parameters

CSYSSITE parameters specify the system names of the C-Systems and the site where they are located. The required system name is the system name specified on the SYSNAME=*system-name* statement in SYS1.PARMLIB(EASY00) or equivalent parameter file, in columns 10-17, left-justified.

For example:

CSYSSITE -SYSTEM- GDDR SITE
CSYSSITE SYS1 GDDR DC1
CPC parameters

The primary purpose of the GDDMPARM CPC parameter is to specify the fully qualified CPC name in the format netid.nau for GDDR managed systems, and to provide the ‘NETID’ literal associated with each GDDR C-System. The syntax of the CPC parameter statement is as follows:

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>CPC</td>
</tr>
<tr>
<td>10-17</td>
<td>System, left-justified, uppercase characters. The 8-character MVS system name of a GDDR C-System which is specified using the SYSNAME=system-name statement in SYS1.PARMLIB(EASY500) or equivalent parameter file. Note that this is not the SMF ID of the system (although it could be the same). The system named here is usually a GDDR C-System, but can be any system where GDRRMAIN is running and has BCPIi connectivity to the named CPC.</td>
</tr>
<tr>
<td>19-22</td>
<td>GDDR. The 1 to 4-character left-justified name of the GDDR instance to which this applies. This field will always contain GDDR.</td>
</tr>
<tr>
<td>24-26</td>
<td>This site is the physical location of the named CPC. DC1 or DC3</td>
</tr>
<tr>
<td>27</td>
<td>,</td>
</tr>
<tr>
<td>28-nn</td>
<td>CPC name. The full CPC name in the format netid.nau: netid (1-8 characters), nau (1-8 characters), separated by a ‘.’. The full CPC name is followed by a comma. BCPII HMC Interface, literal ‘BCPII’. Add at least one CPC parameter for each CPC where GDDR C-Systems or GDDR-managed systems can run.</td>
</tr>
</tbody>
</table>

For example:

```
12345678
CPC SYSTEM GDDR DCx,CPC Name, HMC Interface
CPC SYS1 GDDR DC1,nnnnnnnn.nnnnnnnn,BCPII
CPC SYS2 GDDR DCn,nnn.nnnnnn,BCPII
CPC SYS3 GDDR DC3,nnnnnnnn.nnnnnnnn,BCPII
```

**Note:** If a change is required to CPC parameters, refer to “GDDMPARM CPC parameter entries” on page 373 for detailed instructions.
SYMM parameters

SYMM parameters define VMAX controllers to GDDRMAIN. The syntax of the statement is as follows:

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>SYMM</td>
</tr>
<tr>
<td>10-12</td>
<td>This is the site from where the listed gatekeepers provide access to the listed VMAX controller. DC1 or DC3</td>
</tr>
<tr>
<td>13-15</td>
<td>(H) optional, indicates that the listed site is the home site or physical location of the listed VMAX controller.</td>
</tr>
<tr>
<td>19-22</td>
<td>GDDR. The 1 to 4-character left-justified name of the GDDR instance to which this applies. This field will always contain GDDR.</td>
</tr>
<tr>
<td>24-35</td>
<td>12-digit VMAX controller serial number.</td>
</tr>
<tr>
<td>36</td>
<td>,</td>
</tr>
<tr>
<td>37</td>
<td>A list of gatekeepers (individual unit addresses, ranges of unit addresses, or a combination).</td>
</tr>
</tbody>
</table>

For example:

```
SYMM     SITE(H)  GDDR     Symm S/N     GateKeeper(s)
SYMM     DC1      GDDR  nnnnnnnnnnn,aaaa,bbbb-cccc,dddd
SYMM     DC3      GDDR  nnnnnnnnnnn,oooo,pppp-qqqq,rrrr
```

**Note:** The same VMAX controller can be listed at different sites if it is channel connected to each of the sites. The (H) indicates the site at which that VMAX controller is physically located. Each controller must have only one home site.

If the list of gatekeepers is so long that it will not fit on one statement, you can repeat the statement up to the gatekeeper list and put more gatekeepers on subsequent lines.

For example:

```
SYMM     DC1 (H)  GDDR  012345678912,4202,3456-346F,1234,9876-9877
SYMM     DC1 (H)  GDDR  012345678912,8202,8456-846F,8234,8876-8877
```

There must be at least one SYMM parameter for each VMAX unit to be managed by this GDDR instance. This SYMM parameter defines the physical location (home site) for the VMAX unit.

**Note:** All GDDR command gatekeepers used on GDDR RDF.DEVICES, DLM.DEVICES, and STDBCV parameters as defined using the GDDR Parameter Wizard must first be defined on a SYMM parameter in GDDMPARM. Do not specify MSC gatekeepers in the population of GDDR command gatekeepers.

When adding or removing RDF groups or SRDF devices from GDDR, it is possible that changes may be required to the SYMM parameters in GDDMPARM. This will be the case if you add or remove VMAX units, or if the changed configuration requires less or more gatekeeper devices. These changes, if necessary, must be done before performing the procedures described in Chapter 9, "Performing Maintenance Procedures."
To change SYMM parameters:

1. Update the relevant SYMM parameters locally, and perform a RESTART WORKMGR. If no error messages occur, propagate the updated GDDMPARM to all systems in the GDDRPLEX.

2. Restart the COMM subtask of GDDRMAIN on all GDDR systems, even though the COMM parameters have not changed. If you do not do this, you will receive message GDDM144W informing you that the dataset and the in-use values are possibly different. You can accomplish this by using the BC command of GDDRMAIN if you wish. “Broadcast console commands” on page 71 provides details.

3. Perform a RESTART WORKMGR on all systems in the GDDRPLEX. You can use the BC command for this purpose.

Customize member GDDRPROC

Customize member GDDRPROC in hlq.GDDRvrm.PROCLIB used to run GDDR scripts to your environment.

1. Update the STEPLIB DD statement to include the following load libraries:
   • hlq.GDDRvrm.LINKLIB resulting from your GDDR SMP/E installation
   • Your EMC Mainframe Enablers load library

2. Make sure the following DD statements refer to the GDDR datasets resulting from your GDDR SMP/E installation:
   • ISPPLIB
   • ISPMPLIB
   • ISPSLIB

3. If you plan to use customized versions of GDDR user exits, refer to “Sample procedure to use interpreted REXX user exits” on page 366.

Customize the GDDR ISPF user interface invocation REXX exec

Customize the GDDR ISPF user interface invocation REXX exec for your installation. A sample REXX exec is supplied in hlq.GDDRvrm.SAMPLIB, member GDDREXC.

- Set ggdrpfx to the initial qualifiers of your GDDR installation’s dataset names.

An extract from the sample GDDREXC member is shown below:

```plaintext
/***************************************************************
****/ /* Initialize custom variables */
/***************************************************************
****/
ggdrpfx = "EMC.GDDR500." 
```

Call the GDDR ISPF user interface from an ISPF selection panel

To start the GDDR ISPF user interface by entering an option character from the ISPF primary options menu panel (ISR@PRIM) or from any other ISPF selection menu panel, add an option in the )PROC section for that panel that issues the following SELECT service call:

```plaintext
'CMD(GDDREXC) NEWAPPL(GDDR) MODE(FSCR)' 
```
Call the GDDR ISPF panel from any ISPF panel with TSO GDDR command

To start the GDDR ISPF user interface from any ISPF panel, by issuing the command TSO GDDR from any ISPF panel's primary command line, you must create a new a REXX exec by the name GDDR with the following contents:

```/* REXX */
address ISPEXEC "SELECT CMD(GDDREXC) NEWAPPL(GDDR) MODE(FSCR)"
```

It is important that you use the NEWAPPL(GDDR) keyword to ensure that all profile variables are stored properly in the same ISPF application profile.

Configure GDDR

Use the GDDR online facilities to complete the tasks described in this section.

**Note:** Review the introductory procedures described in Chapter 5, “Using GDDR Online Facilities” prior to performing the tasks listed in this section.

The Parameter Management Options Menu shown in Figure 9 on page 110 presents parameter input panels which conditionally prompt for required parameter values based on your particular disaster restart topography and EMC SRDF configuration. The systems, devices, and EMC GDDR options defined during the parameter input task specify how GDDR handles conditions requiring the use of EMC recovery and business continuity facilities.

Step 1: Update your personal GDDR ISPF profile

When you complete the GDDR software installation, dependent software installation, and related user-authorization definitions, the GDDR Primary Options Menu displays:

```
---------- GDDR - Primary Options Menu ----------
Option =>
Profile     Update personal GDDR ISPF Profile
                This System: SYS2
Maintenance  GDDR Setup and Maintenance       This Site: DC2
                Master-C: SYS2
                Primary Site: DC1
                Primary DASD: DC1
                Automation: ON
                Planned script: None
                Unplanned script: None

EMC Geographically Dispersed Disaster Restart  05.00.00
Copyright © 2007-2016 EMC Corporation.
Select an option and press <Enter>
Press <F3> to Terminate GDDR ISPF
```

**Note:** This panel may vary based on relevant user authorizations.

Before you can perform any GDDR Setup and Maintenance actions, you must populate your ISPF profile dataset with basic information.
1. On the GDDR Primary Options Menu, select **Option P, Update Personal GDDR ISPF Profile**.

   The Change GDDR ISPF Profile Variable Values panel displays as follows:

   ![Change GDDR ISPF Profile Variable Values Panel](image)

   2. Complete the panel fields as follows:

   - **GDDR Subsystem Name** — No action required. This field defaults to GDDR and is not editable.
   - **JCL dataset** — Specify the name of the PROCLIB dataset holding the GDDRPROC customized during the GDDR integration procedure “Customize member GDDRPROC” on page 78.
   - **ISPF skeleton dataset** — Specify the name of a dataset containing the GDDR file tailoring skeletons created when you downloaded the install cartridge.
   - **Global variable backup** — Specify the name of the parameter backup dataset to be used for parameter management functions, defined during the GDDR integration procedure “Define GDDR datasets” on page 64.
   - **Jobcards** — This field displays if you are authorized to activate parameter changes and to manage GDDR scripts. The JOBCARD information you enter here applies only to the current user and is used for parameter activate and script submission tasks. Always include the REGION=0M jobcard parameter.

     For example:

     ```
     ===>//JOBNAME JOB (acct-number),gddr-job,CLASS=A,REGION=0M,
     ===>// MSGCLASS=A,USER=GDDR,NOTIFY=GDDR
     ===>/*JOBPARM LINES=999999
     ===>/*
     ```

   3. Press **F3** to save the changes and return to the GDDR Primary Options Menu.

**Step 2: Define initial site, system, storage, utility, and GDDR option parameters**

The GDDR parameter management process uses an existing backup of parameter values as the starting point for subsequent updates. To facilitate the parameter customization associated with a new GDDR implementation, initial parameter definitions are populated from the GDDMPARM parameters supplied during installation as described in “Install
GDDMPARM” on page 66.

Note: Subsequent parameter changes will be applied to a copy of your parameters, created as a backup dataset at the time the existing parameters were applied, or any later time.

To perform an initial parameter activation:

1. From the GDDR Primary Options Menu, select Option M, Setup and Maintenance, then select Option P, Manage GDDR Parameters.

Because there are no existing parameters, Option P displays a Select Parameter Input Dataset panel as follows:

2. To begin parameter definition with an initial parameter set, type S in the Action field associated with the entry described 'GDDR parameter load from scratch', member name '--NONE--' and press Enter.

The panel displays a message requesting that you provide a work dataset name and a parameter load description:

3. Supply a parameter load work dataset and parameter load description.

3. Supply a parameter load work dataset and parameter load description.
Enter the parameter load work dataset that was defined earlier in “Allocate the parameter management datasets” on page 64. This dataset contains your 'work-in-process', enabling you to assemble a complete parameter set by saving your input data from each of the parameter definition panels, and returning to the task at a later time.

**Note:** The parameter input dataset and the parameter load work dataset must be two different datasets. This is because the contents of the parameter load work dataset are overwritten when exiting the Select Parameter Input Dataset panel in step 6.

The parameter load work dataset must also be different from the dataset that is defined to GDDR as the "Last Activated Parameter Dataset".

4. Press **F3** to exit.

Upon exiting from the Select Parameter Input Dataset panel, the work dataset is initialized with PDS members which are associated with each of the parameter load function panels. Messages associated with the work dataset initialization are returned to your TSO session as follows:

```
OPS0996I Preparing Work Dataset
OPS0996I OPS0996I Validating Input variables:
OPS0996I ===> CONFIG
OPS0996I OPS0996I Writing work members:
OPS0996I OPS0996I ===> A01FEATR
OPS0996I OPS0996I ===> A02CSYST
OPS0996I OPS0996I ===> A03GDDS
OPS0996I OPS0996I ===> C01ROLES
OPS0996I OPS0996I ===> E01CNTRL
OPS0996I OPS0996I ===> E02GDDGK
OPS0996I OPS0996I ===> E03MSCGK
OPS0996I OPS0996I ===> E04SRDFD
OPS0996I OPS0996I ===> E05TFDEV
OPS0996I OPS0996I ===> E06SDDF0
OPS0996I OPS0996I ===> E07GNS00
OPS0996I OPS0996I ===> H01SYSTS
OPS0996I OPS0996I ===> H02LPARS
OPS0996I OPS0996I ===> H03CPC00
OPS0996I OPS0996I ===> H04IPLO0
```
5. Upon selection of the initial parameter backup member and completion of the work dataset initialization process, the Parameter Management Options Menu displays with the complete list of parameter load functions as follows:

```
Option -->
B ackup       Manage GDDR Parameter backups             This System: SYS2
            This Site: DC1
.- Parameter Load functions -.            Master-C: SYS2
  I nput       Select Parameter Input Dataset |        Primary Site: DC1
  C onfig      Define Configuration basics      Primary DASD: DC1
  D ASD        Define Data Storage objects     Automation: ON
  H ost        Define Host objects             Planned script: None
  O ptions     Specify GDDR Options           Unplanned script: None
  V alidate    Validate GDDR Parameter Set     
  A ctivate    Activate GDDR Parameter Set      
```

You can complete customizing the contents of the parameter work dataset with your site-specific values using the Parameter Load functions on the Parameter Management Options Menu. This process is identical for an initial parameter customization or for any subsequent customization. “Option P: Manage GDDR Parameters” on page 110 provides details on this process.

**Step 3: Modify GDDR user exits (optional)**

GDDR provides exit points that you may use to augment GDDR facilities. Sample exits are provided in the `hlq.GDDRvrm.SAMPLIB` distribution library. You may modify these exits, or write your own, following the guidelines presented in Appendix A, “User Exits.” The exit points are the following:

- **GDDRUX01** is called from planned or unplanned event scripts at a point appropriate for starting production mainframe workloads.
- **GDDRUX02** is called from planned or unplanned scripts at a point appropriate for shutting down production mainframe workloads.
- **GDDRUX03** is called from planned or unplanned scripts at a point appropriate for starting distributed production workloads.
- **GDDRUX04** is called from planned or unplanned scripts at a point appropriate for stopping distributed production workloads.
Optional configuration features

The following topics describe optional features which may be configured.

Configure GDDR support for external devices

GDDR always manages devices under consistency protection. This can be ConGroup protection, MSC protection, or SRDF/Star protection. GDDR support for external devices adds appropriate commands to GDDR scripts to manage RDF groups outside of the GDDR-managed consistency protection.

GDDR support for external devices enhances automated restart at a site which is an SRDF/A or SRDF/S target by adding management of various types of volumes (for example, page volumes and temp pool volumes) which are not typically replicated with consistency protection, due to various considerations such as bandwidth constraints or the fact that they simply do not have the consistency requirement and should not trigger Swap or Trip events. The SRDF device actions to be performed on the external devices are “make R2 R/W” or “make R1 TNR”, depending on the script and the site role.

The addition of external devices to GDDR typically requires changes to SYMM parameters in GDMPARM, as additional gatekeeper devices will usually be needed. You must make the updates to the SYMM parameters before completing the implementation procedure described below. Refer to “SYMM parameters” on page 77 for more information on how to change SYMM parameters in GDMPARM.

1. Run utility GDDRACDD to discover all devices to be managed by GDDR. Refer to “Automated Configuration Discovery for DASD (GDDRACDD)” on page 240 for detailed instructions. Alternatively, define the external devices and set the External field on the Define SRDF Devices panel, as described on page 140 to YES.

2. From “Option O: Default Script Call Overrides” on page 170, set Manage External devices to Y.

GDDR support for the external devices is as follows:

- **Planned Site Swap (GDD2P17A / GDD2P18A)**
  The external devices are RDF-SUSPended in GDD2P17A and in GDD2P18A they are SWAPped and RDF-RSUMed.

- **Resume replication after loss of primary (GDD2PA0A)**
  The non-SRDF/A devices are SWAPped, set to ADCOPY-DISK, RDF-RSUMed.

CPC Recovery - LPAR Recovery features

CPC Recovery and LPAR Recovery features deliver system recovery capabilities for Central Processor Complexes (CPC), protecting a single system or all systems within the complex. CPC Recovery is enabled for all or selected processor complexes within a site as soon as LPAR Recovery is defined for one of the systems which has an LPAR defined with the CPC. Protection is accomplished through monitoring for system outage trigger messages indicating a system has unregistered from a common communication facility, in this case, EMC’s Symmetrix Control Facility (SCF).

The CPC Recovery functionality complements the production/contingency protection mechanism and entails the following:
Contingency systems are no longer required to be defined for GDDR-managed systems.

A GDDR-managed system can have either a contingency system or a recovery LPAR defined, or both.

A C-System can have a recovery LPAR defined.

A system protected with LPAR Recovery is specified with a "home" and "recovery" LPAR location ("away").

An LPAR within a CPC can be either the normal LPAR or a recovery LPAR for a system, or both (normal LPAR for system A, recovery LPAR for system B).

The recovery LPAR can be in the same CPC hosting the system at home.

The recovery LPAR can be in a different CPC in the system's "home" data center.

If the LPAR location of a system A is the recovery LPAR of a system B, then system A cannot have a recovery LPAR defined.

If a system A has an LPAR defined for it which is the recovery LPAR for a system B, then system A cannot be a contingency system.

If a system is protected with a recovery LPAR at an "away" site different from its "home" site, then it cannot have a second LPAR parameter for that "away" site. Once this restriction is honored, LPAR-recovery protected systems can have one LPAR per site in the configuration.

If a system is not protected with LPAR recovery, then it can have one LPAR parameter defined for each site in the configuration.

The SITE.<sysname> parameters for systems which do not have a recovery LPAR defined change when these systems are IPL'd in an LPAR on a different site when they are being displaced by the recovery of an LPAR-recovery protected system. For systems protected with LPAR Recovery, the site does not change, but a separate DCN.LOCATION.<sysname> parameter is toggled between values "home" and "away".
Integrating GDDR
CHAPTER 4
Performing GDDRMAIN Operations

This chapter describes the use of GDDRMAIN.

- GDDRMAIN subtasks ................................................................. 88
- GDDRMAIN dependent address spaces ........................................ 88
- GDDRMAIN console commands .................................................. 89
- GDDRMAIN EXEC parameters ....................................................... 96
- GDDRGVX utility ........................................................................ 97
- GDDR system variable integrity and access ....................................... 98
- Remote command processing ...................................................... 100
- GDDR script submission operator command .................................... 100
Performing GDDRMAIN Operations

GDDRMAIN subtasks

GDDRMAIN is the main GDDR address space. GDDRMAIN controls GDDR global variable management, message interception, and communication between C-Systems and managed systems, and manages remote command processing using the subtasks listed in Table 10.

<table>
<thead>
<tr>
<th>GDDRMAIN subtask name</th>
<th>C-Systems</th>
<th>Managed systems</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSOLE</td>
<td>Yes</td>
<td>Yes</td>
<td>Console command processing</td>
</tr>
<tr>
<td>COMM</td>
<td>Yes</td>
<td>Yes</td>
<td>Inter-system communication</td>
</tr>
<tr>
<td>GVT</td>
<td>Yes</td>
<td>No</td>
<td>Global variable data management — load from and save to DIV</td>
</tr>
<tr>
<td>HBM</td>
<td>Yes</td>
<td>No</td>
<td>Heartbeat Monitor</td>
</tr>
<tr>
<td>HBMDLM</td>
<td>Yes</td>
<td>No</td>
<td>Subtask if DLM is configured</td>
</tr>
<tr>
<td>MCSOPER</td>
<td>Yes</td>
<td>Yes</td>
<td>Console message interception</td>
</tr>
<tr>
<td>MCSOCART</td>
<td>Yes</td>
<td>Yes</td>
<td>Console command response processor</td>
</tr>
<tr>
<td>MISC</td>
<td>Yes</td>
<td>Yes</td>
<td>Recurring timed functions</td>
</tr>
<tr>
<td>MISCBSnn</td>
<td>Yes</td>
<td>Yes</td>
<td>BCPii console message scanning</td>
</tr>
<tr>
<td>WORKMGR</td>
<td>Yes</td>
<td>Yes</td>
<td>Worker task management</td>
</tr>
<tr>
<td>GDDWSTnn</td>
<td>Yes</td>
<td>Yes</td>
<td>Status or information retrieval</td>
</tr>
<tr>
<td>GDDWDVnn</td>
<td>Yes</td>
<td>No</td>
<td>Storage device status processing</td>
</tr>
<tr>
<td>GDDWGVnn</td>
<td>Yes</td>
<td>No</td>
<td>Global variable functions</td>
</tr>
<tr>
<td>GDDWCO.nn</td>
<td>Yes</td>
<td>Yes</td>
<td>Issue console commands</td>
</tr>
<tr>
<td>GDDWCMnn</td>
<td>Yes</td>
<td>Yes</td>
<td>Process console messages</td>
</tr>
<tr>
<td>GDDWSXnn</td>
<td>Yes</td>
<td>Yes</td>
<td>Restricted status worker</td>
</tr>
</tbody>
</table>

GDDRMAIN dependent address spaces

Dependent address spaces are started, restarted, and stopped/canceled by GDDRMAIN. The GDDR Event Monitor and GDDRWORK dependent address spaces are described in Table 11.

<table>
<thead>
<tr>
<th>Dependent address spaces</th>
<th>C-Systems</th>
<th>Managed systems</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRWORK</td>
<td>Yes</td>
<td>No</td>
<td>Worker task management in separate address spaces</td>
</tr>
<tr>
<td>GDDWXHnn</td>
<td>Yes</td>
<td>No</td>
<td>HMC worker in separate address space</td>
</tr>
<tr>
<td>GDDWQXnn</td>
<td>Yes</td>
<td>No</td>
<td>Command queue worker in separate address space</td>
</tr>
<tr>
<td>GDDWXQnn</td>
<td>Yes</td>
<td>No</td>
<td>REXX (external address spaces)</td>
</tr>
<tr>
<td>GDDWCMnn</td>
<td>Yes</td>
<td>Yes</td>
<td>User message automation worker</td>
</tr>
<tr>
<td>GDDWCMQnn</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>GDDWCMXnn</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>GDDWCMXQnn</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>EVM</td>
<td>Yes</td>
<td>No</td>
<td>Event Monitor</td>
</tr>
</tbody>
</table>

a. GDDWXH can run on a managed system if that system runs BCPii work.
Performing GDDRMAIN Operations

**Note:** Certain scripts perform actions on a C-System other than the Master C-System. When script actions are performed on another C-System, the output for those steps appears in the joblog of a GDDRWORK address space on the targeted system.

**Event Monitor administration**

If the Event Monitor is running at midnight, it is automatically restarted. The EVM output (SYSTSPRT) will be spun off and purged to avoid filling up the spool.

**GDDRMAIN console commands**

GDDRMAIN accepts the z/OS modify command (F) and stop command (P).

**Stop command (P)**

To stop GDDRMAIN, issue a P GDDRMAIN command (or use the current jobname, if different).

⚠️ **CAUTION**

Do not cancel GDDRMAIN except in an emergency. Although GDDRMAIN has a TERM=YES ESTAE, actions by the ESTAE exit routine are limited when a cancel is used.

When a P command is issued, GDDRMAIN will stop all subtasks except GVT, then stop GVT. If GVT is not running or if it is unable to write the DIV for any reason, a WTOR will be issued to allow saving the data space if that is desired (see GDDRGVX DSPSAVE).

**Modify commands (F)**

GDDRMAIN accepts z/OS console commands in the form “F GDDRMAIN,verb operand”, where verb operand is one commands listed in Table 12.

<table>
<thead>
<tr>
<th>Command</th>
<th>Usage</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>START</td>
<td>Start subtasks</td>
<td>F GDDRMAIN,START name</td>
</tr>
<tr>
<td>STOP</td>
<td>Stop subtasks</td>
<td>F GDDRMAIN,STOP name</td>
</tr>
<tr>
<td>RESTART</td>
<td>Restart subtasks</td>
<td>F GDDRMAIN,RESTART name</td>
</tr>
<tr>
<td>CANCEL</td>
<td>Cancel subtasks</td>
<td>CANCEL of GDDRMAIN internal subtasks now requires a previous STOP command. CANCEL works immediately for WORKERS in external address spaces. Cancels worker subtasks in external address spaces individually using the 8-character worker name, or all external worker subtasks which match a 1-7 character name mask.</td>
</tr>
<tr>
<td>TASKS</td>
<td>Display subtasks</td>
<td>F GDDRMAIN,TASKS</td>
</tr>
<tr>
<td>BCPii</td>
<td>Display the MISCBSnn subtasks</td>
<td>F GDDRMAIN,BCPII</td>
</tr>
</tbody>
</table>
### Table 12: GDDRMAIN console command summary (page 2 of 3)

<table>
<thead>
<tr>
<th>Command</th>
<th>Usage</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LOCK</strong></td>
<td>Administer locks</td>
<td>F GDDRMAIN,LOCK&lt;br&gt; F GDDRMAIN,LOCK,&lt;lock-name&gt;&lt;br&gt; F GDDRMAIN,LOCK,&lt;lock-name&gt;,cmd&lt;br&gt; where lock-name is INDEX, UPDATE, or BOTH and cmd is DISPLAY or CLEAR. The default value for lock-name is BOTH and the default command is DISPLAY.</td>
</tr>
<tr>
<td><strong>COMM</strong></td>
<td>Communication status display</td>
<td>F GDDRMAIN,COMM</td>
</tr>
<tr>
<td><strong>MPARM</strong></td>
<td>MPARM checksum display, resynchronization</td>
<td>F GDDRMAIN,MPARM&lt;br&gt; F GDDRMAIN,MPARM,CHECK</td>
</tr>
<tr>
<td><strong>WORKER</strong></td>
<td>Worker display and customization (min and max limits)</td>
<td>F GDDRMAIN, WORKER[,name[,MIN=min</td>
</tr>
<tr>
<td><strong>MSGS</strong></td>
<td>Message rule administration</td>
<td>F GDDRMAIN, MSGS[,msgid[,ENABLE</td>
</tr>
<tr>
<td><strong>UMA</strong></td>
<td>Control user message automation</td>
<td>F GDDRMAIN,UMA[ON</td>
</tr>
<tr>
<td><strong>GATEK</strong></td>
<td>Invoke the GDDGATEK utility</td>
<td>F GDDRMAIN,GATEK [parms]&lt;br&gt; Where parms are optional. Refer to &quot;Parameter syntax&quot; on page 303 for details.</td>
</tr>
<tr>
<td><strong>BC</strong></td>
<td>Broadcast GDDRMAIN console commands</td>
<td>/F GDDRMAIN,BC,*C,START EVM&lt;br&gt; /F GDDRMAIN,BC,*C,START HBM&lt;br&gt; F GDDRMAIN,BC,&lt;to&gt;,&lt;cmd&gt;&lt;br&gt; Where &lt;to&gt; is one of:&lt;br&gt;  - System-name - where cmd is to be executed&lt;br&gt;  - Site-ID - where cmd is to be executed&lt;br&gt;  - * - all GDDR managed systems&lt;br&gt;  - *C - all C-Systems&lt;br&gt;  - *P - all production systems&lt;br&gt;  - *1 - all GDDR systems, this system first&lt;br&gt;  - *X - all GDDR systems except this system&lt;br&gt;  - *Z - all GDDR systems, this system last&lt;br&gt;  - &lt;cmd&gt; is the command to be executed on the system(s)</td>
</tr>
<tr>
<td><strong>BR</strong></td>
<td>Broadcast with response</td>
<td>Syntactically identical to BC</td>
</tr>
<tr>
<td><strong>SCRIPT</strong></td>
<td>Execute a script</td>
<td>F GDDRMAIN,SCRIPT,&lt;name&gt;,&lt;option1,option2,optionn&gt;&lt;br&gt; Provides externally accessible script execution capability.</td>
</tr>
<tr>
<td><strong>DLM</strong></td>
<td>Display status of DLm ACP systems</td>
<td>F GDDRMAIN,DLM</td>
</tr>
</tbody>
</table>
Performing GDDRMAIN Operations

START

Use START to start a subtask which is not currently running. The syntax is “F GDDRMAIN,START name” where name is the name of the subtask. Issue the TASKS command to see the names of the subtasks.

STOP

Use STOP to stop a subtask which is currently running. The syntax is “F GDDRMAIN,STOP name” where name is the name of the subtask. STOP will post the subtask to end, but if it does not end, you can use CANCEL to stop it.

RESTART

Use RESTART to stop a subtask which is running and restart it immediately. The syntax is “F GDDRMAIN,RESTART name” where name is the name of the subtask. RESTART will post the subtask to end, but if it does not end, you can use CANCEL to stop it. If you use CANCEL, the automatic restart will not occur, but you can use START to restart the subtask.

CANCEL

Use CANCEL to stop one or more running subtasks. The syntax is “F GDDRMAIN,CANCEL name” where name is a 1-8 character name mask.

For example, F GDDRMAIN,CANCEL GDDWXQ will cancel every GDDRWXQ worker (GDDWXQ00, GDDWXQ01, GDDWXQnn)

CANCEL works immediately for WORKERS in external address spaces. For GDDRMAIN internal subtasks, CANCEL is not accepted without a prior STOP command.

TASKS

Use TASKS to display the status of each subtask. There are no operands. The syntax is “F GDDRMAIN,TASKS”. For each subtask, a line is displayed with the subtask name and its status, ACTIVE or ENDED. If the status is ENDED, the last return code from the subtask is displayed in parentheses after ENDED.
LOCK

Use LOCK to display or clear the index, or update locks, or both. The syntax is “F GDDRMAIN,LOCK” or “F GDDRMAIN,LOCK,lock-name” or “F GDDRMAIN,LOCK,lock-name,cmd” where lock-name is INDEX, UPDATE, or BOTH and cmd is DISPLAY or CLEAR. The default value for lock-name is BOTH and the default command is DISPLAY. The status of the requested lock will be displayed (CLEAR, SET, IN USE, PENDING, or BROKEN).

- If the status is SET, the job name and ID of the locker will be displayed along with the number of seconds since the lock was set.
- If the status is IN USE or PENDING, the number of seconds since the shared lock was changed will be displayed.

Refer to “GDDR system variable integrity and access” on page 98 for lock descriptions, status definitions, and administrative considerations of the locks.

COMM

Use the COMM command to display the list of systems, and IP addresses and ports comprising the GDDR complex. The syntax is F GDDRMAIN,COMM.

You can specify two IP addresses per statement and multiple COMM statements per system. This allows GDDRMAIN to communicate with LPARs that are not at their designated location. “Update COMM statements” on page 74 provides more details.

The following example shows COMM command output:

```
GDDM127I Communications Status
Sys PRD1 ,      IP nnn.nnn.nnn.nnn,9876 Last 12/04/09 17:27:44.50
Sys PRD2 ,      IP nnn.nnn.nnn.nnn,9876 Last 12/04/09 17:27:44.50
Sys PRD3 ,      IP nnn.nnn.nnn.nnn,9876 Last 12/04/09 17:27:44.50
Sys SYS3 (DC3), IP nnn.nnn.nnn.nnn,9876 Last 12/04/09 17:27:44.51
Sys SYS2 (DC2), IP nnn.nnn.nnn.nnn,9876 Last 12/04/09 17:27:44.51
Sys SYS1 (DC1), IP nnn.nnn.nnn.nnn,9876 Last 12/04/09 17:27:44.51
```

DLM

Use the DLM command to determine the status of DLm systems managed by GDDR. The DLM command issues a heartbeat request to every DLm system defined at the GDDR - Define DLm Devices panel, and reports whether the DLm is alive or dead.

The following example shows DLM command output when all GDDR managed DLm systems have responded to a GDDRMAIN heartbeat request.

```
DLM
GDDM280I DLM status  347
DC1 DLM 11486269 (M, IP nnn.nnn.nnn.nnn,9050) Status ALIVE
DC3 DLM 20486269 (M, IP nnn.nnn.nnn.nnn,9050) Status ALIVE
DC2 DLM 21117450 (M, IP nnn.nnn.nnn.nnn,9050) Status ALIVE
```

GVB

Global variable backup is accomplished by running a script (GDDRPGVB) in a local REXX worker (GDDWXR). The backup is made into the dataset named in the global variable GLOBAL.GDDR.PARMS.GDDRVAR_BACKUP.<system>.
The GVB command is used to schedule a regular backup of global variables or to perform a backup on demand. If entered with no value, the current global variable backup scheduled day and time is displayed. A value of NONE causes backups to not be scheduled. A value of NOW causes an immediate backup.

When the scheduled time comes, the backup will be postponed if a script is in progress.

The default value for GVB is NONE. This default can be overridden via the JCL PARM of GDDRMAIN. The syntax for GVB in the GDDRMAIN parameter is the same as above, except that NOW cannot be specified and a value must be specified.

### MPARM

Use the MPARM command to display GDDMPARM checksums which indicate when the copies of GDDMPARM referenced by GDDRMAIN on each system are consistent. The MPARM command output lists each system with which GDDRMAIN can possibly communicate. The display shows each system name, the checksum of the GDDMPARM data in use, the checksum of the contents of the GDDMPARM dataset, and whether or not the system has found different GDDMPARM data from other systems. If GDDRMAIN cannot communicate with the system, the "unable to communicate" variant is displayed.

The following example shows MPARM command output when all GDDRMAIN tasks are consistent and capable of communicating:

```
GDDM140I GDDMPARM Status
Sys PRD1 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
Sys PRD2 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
Sys SYS3 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
Sys SYS2 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
Sys SYS1 : In-use 0C63D749, Dataset 0F3F7372, Consistency Y
```

Use MPARM,CHECK command to force a return to synchronization if the GDDMPARM data in use everywhere is synchronized. The MPARM,CHECK command retrieves the GDDMPARM in-use checksum from each system and compares them all. If all are equal, consistency is restored and degraded status is turned off. MPARM,CHECK will tolerate no response from GDDR-managed systems, but must get a response from each C-System in order to declare consistency. If any system (C-System or production system) responds with a different in-use checksum value, consistency will not be declared.

Sample output:

```
GDDM142I GDDMPARM Consistency set globally
GDDM142I GDDMPARM Consistency set locally
```

The ‘GDDMPARM Consistency set locally’ version of the message will appear in the system log of each system with successful communication.

If any C-System cannot be reached the output is:

```
GDDM141E GDDMPARM Inconsistency detected, C-System SYS1 is down
```

If any system has a different in-use checksum, the output is:

```
GDDM141E GDDMPARM Inconsistency detected, System SYS1 is inconsistent
```
Performing GDDRMAIN Operations

RDFREFR

GDDRMAIN creates a map of RDF links between VMAX systems when the WORKMGR subtask starts. If an RDF group is added or removed, this map can be out of sync with reality. The RDFREFR command reconstructs this map. Previously, it was necessary to restart WORKMGR to reconstruct the RDF link map.

WORKER

Use the WORKER command to display WORKER subtasks and dynamically change the minimum and maximum number of worker tasks of a particular type.

The syntax is F GDDRMAIN, WORKER, name and F GDDRMAIN, WORKER[, name[, MIN=min|MAX=max]]

Sample output of WORKER command is as follows:

GDDM128I Worker Status
GDDWCO Type S, Act 1, Min 1, Max 1
   Name GDDWCO00, Status Waiting
GDDWCM Type S, Act 1, Min 1, Max 1
   Name GDDWCM00, Status Waiting
GDDWST Type S, Act 1, Min 1, Max 1
   Name GDDWST00, Status Waiting
GDDWXR Type A, Act 1, Min 1, Max 1
   Name GDDWXR00, Status Waiting
GDDWGV Type S, Act 1, Min 1, Max 1
   Name GDDWGV00, Status Waiting

You can restrict the display to a single worker type by using WORKER,GDDWST:

GDDM128I Worker Status
GDDWST Type S, Act 1, Min 1, Max 1
   Name GDDWST00, Status Waiting

You can modify the minimum and maximum settings for a worker type as follows:

WORKER, GDDWST, MIN=2
GDDM128I Worker Status
GDDWST Type S, Act 1, Min 2, Max 2
   Name GDDWST00, Status Waiting

(Note that maximum value was also set to 2.)

WORKER, GDDWST, MAX=1
GDDM128I Worker Status
GDDWST Type S, Act 2, Min 1, Max 1
   Name GDDWST01, Status Waiting
   Name GDDWST00, Status Waiting

(Note that the minimum value was also set to 1.)

**Note:** WORKMGR monitors the backlog of requests to the Storage device worker and the Status worker. When the backlog grows, additional tasks are started. When the backlog has been processed, idle worker tasks are stopped.
MSGS

Use the MSGS command to enable or disable GDDR message rules which are message IDs that indicate changes in storage or system status, delivered in hlq.GDDRvrm.LINKLIB(GDDRMMSG). The syntax is F GDDRMAIN, MSGS[,msgid[,ENABLE|DISABLE]].

Sample MSGS command output:

GDDM130I Message Status
BPXM023I, E/D E, Int 0 *NEVER*, Proc 0 *NEVER*
CGRP005E, E/D E, Int 0 *NEVER*, Proc 0 *NEVER*
CGRP051I, E/D E, Int 0 *NEVER*, Proc 0 *NEVER*
CGRP056I, E/D E, Int 0 *NEVER*, Proc 0 *NEVER*
CGRP093E, E/D E, Int 0 *NEVER*, Proc 0 *NEVER*
CGRP095E, E/D E, Int 0 *NEVER*, Proc 0 *NEVER*
CGRP100E, E/D E, Int 0 *NEVER*, Proc 0 *NEVER*

◆ The value of E/D indicates if interception of the message is enabled (E) or disabled (D).
◆ Int indicates the number of times and the last date and time (GMT) when the message was intercepted on this system.
◆ Proc indicates the number of times and the last date and time (GMT) when the message was processed on this system.

To display a particular message, use MSGS, EMCGM9EE:

GDDM130I Message Status
EMCGM9EE, E/D E, Int 1 01/07/10 20:44:53.61, Proc 1 01/07/10 20:44:53.

To enable or disable interception of a particular message, use MSGS, EMCGM9EE, DISABLE:

GDDM130I Message Status
EMCGM9EE, E/D D, Int 1 01/07/10 20:44:53.61, Proc 1 01/07/10 20:44:53.

UMA

Use the UMA command to control user message automation. See “Automate messages” on page 69 for more information.

GATEK

Use the GATEK command to invoke the GDDGATEK utility. See “GDDR Gatekeeper Validation utility (GDDGATEK)” on page 302 provides parameter syntax and descriptions and example command output.

BC and BR

BC (broadcast) and BR (broadcast with response) send a command to any, all, or some GDDR-managed systems. The syntax is F GDDRMAIN,BC[BR],<to>,<cmd>.

to can be the name of a specific system, or one of the following:

* — send to all systems (including the local system)
*C — send to all C-Systems
*P — send to all production systems
*1 — send to all systems, but send to local system first
Performing GDDRMAIN Operations

*Z — send to all systems, but send to local system last
*X — send to all systems except local system

cmd is the command to be sent to the indicated systems.

For example, BC,*,D T sends a D T command to every GDDR system:

GDDM145I Command sent to system X19
GDDM145I Command sent to system X118
GDDM145I Command sent to system X99
GDDM145I Command sent to system X117
GDDM149I Issuing D T for system X117 Job ABCDE (J0003662)
D T

The GDDM145I messages show which systems were sent the command. The GDDM149I message is displayed on every system which receives the command.

If BR is used to issue an F GDDRMAIN command to systems which include the system on which the BR command was issued, there will be no response shown from the system on which the BR was issued. This is because the command processor on that system is busy with the BR command and will not execute the BR’d command until the BR completes. In practical terms, this is of no consequence, because the command will be executed right after the broadcast processing, so the command output will be seen when it executes on the host system.

SCRIPT

Use SCRIPT to submit scripts externally for execution. The syntax is F GDDRMAIN,SCRIPT,<name>,<option> where:

◆ <name> is the name of a GDDR script supported through this interface.
◆ <option1,option2,optionn> is comma-separated string of keywords and/or keywords=values.

“GDDR script submission operator command” on page 100 provides details on SCRIPT command implementation and usage.

GDDRMAIN EXEC parameters

GDDRMAIN accepts the EXEC statement parameters listed below. The parameters may be specified in any order and any combination, separated by commas. The valid parameters are NOEVM and NOHBM.

The NOEVM and NOHBM parameters cause GDDRMAIN to not start the Event Monitor and Heartbeat Monitor tasks. This is useful prior to the initial parameter Activation or to correct some global variable problem which needs to be rectified prior to starting these tasks.
GDDRGVX utility

The GDDRGVX utility program is available in hlq.GDDRvrm.SAMPLIB(GDDRGVX) and provides global variable analysis and management functions for the DIV dataset. Its functions are requested by the following parameters:

- DSPLIST
- DIVLIST
- DSPSAVE
- RELOAD

DSPLIST

The DSPLIST parameter produces a formatted list of the contents of the global variable data space.

DIVLIST

The DIVLIST parameter produces a formatted list of the contents of the DIV. This parameter may only be used when GDDRMAIN is not active. The output format is identical to DSPLIST. The utility JCL requires a GDDRGVDS DD pointing to the DIV dataset.

DSPSAVE

The DSPSAVE parameter copies the contents of the global variable data space to a sequential dataset. The dataset is pointed to by a //GVSAVE DD statement. The output dataset can be used by IDCAMS REPRO to populate a new DIV dataset.

RELOAD

The RELOAD parameter copies global variables from an old DIV to a new DIV. The reason to use reload instead of IDCAMS REPRO is that reload will not copy deleted variables. Also, any "gas" in the DIV will be removed. To use the RELOAD parameter, GDDRMAIN must be running. The old DIV is pointed to by a //GDDRGVDS DD statement in the utility JCL, while the //GDDRGVDS DD statement in GDDRMAIN must be pointing to another (empty) DIV.
GDDR system variable integrity and access

Two locks are used when managing GDDR system variable integrity: the index lock and the update lock. Both locks allow you to record, without serialization, when you are using the index or doing an update, and allow exclusive control by a single user when no use is in progress. If either lock becomes "stuck", it can have negative implications for global variable access.

The lock consists of two words; the first is the exclusive lock, the second is the shared lock. If both words are zero, the lock is completely open. This is the normal state. The lock words can have the combinations of values listed in Table 13.

Table 13 Possible lock states

<table>
<thead>
<tr>
<th>First word - exclusive value</th>
<th>Second word - shared value</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>The lock is open. There is no shared or exclusive use of the resource.</td>
</tr>
<tr>
<td>0</td>
<td>positive value</td>
<td>The resource is in use by one or more non-exclusive users. The shared value is the number of users. No one is requesting exclusive use.</td>
</tr>
<tr>
<td>1</td>
<td>positive value</td>
<td>The resource is in use by one or more non-exclusive users, but one user is waiting for exclusive use. If the shared word does not go to zero within a very short period of time, the lock is stuck. This is referred to as 'shared lock stuck'.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>The lock is locked, or set. One task has exclusive use of the resource and any other task which wants to use it must wait or take alternative action. If the exclusive word does not go to 0 within a very short period of time, the lock is stuck. This is referred to as 'exclusive lock stuck'.</td>
</tr>
<tr>
<td>other than 0 or 1</td>
<td>any value</td>
<td>The lock is broken. The exclusive word should only have the value 0 or 1.</td>
</tr>
<tr>
<td>0 or 1</td>
<td>negative value</td>
<td>The lock is broken. The shared word should only have non-negative values.</td>
</tr>
</tbody>
</table>

You can display the locks with GDDRGVX DSPLIST. GDDRGVX DSPLIST displays the time when the exclusive lock was set, as well as the job which set it. There is no display of jobs which set the shared lock, but GDDRGVX DSPLIST indicates the last time the shared lock was changed. You can also use the LOCK command to display this information.

You can clear a stuck or broken lock by using the LOCK command.

Index lock

The index lock indicates that the global variable index is in use or is being rebuilt. If the index is being rebuilt, it cannot be used by anyone.

If the exclusive lock becomes stuck, no one will be able to use the index and performance will suffer greatly. However, no functions will fail. If the exclusive lock is set for more than 5 seconds and a task tries to use the index, a message is issued.
If the shared lock becomes stuck, the index will not be able to be rebuilt when it needs to be done and performance will suffer, but not as severely as a stuck exclusive lock. Again, no functions will fail. A message is issued if a reindex is desired but cannot be done because the shared lock is not zero. If this message occurs with any frequency, investigate and consider manually clearing the lock.

**Update lock**

The update lock is used (shared) when updates to global variables are made which change the structure of the data space. The exclusive lock is used by GVT when it writes the data space out to the DIV.

If the exclusive lock becomes stuck, no updates to global variables which require structural changes can be made (adding a new variable, changing a value to a longer value). If this occurs, requests will fail and messages will be issued.

If the shared lock becomes stuck, GVT will not be able to write the data space out to the DIV. However, if GVT is unable to obtain the update exclusive lock, it will force it (that is, set the lock words to 1,0) after some period of trying to obtain the lock. A stuck shared update lock will automatically be unstuck if GVT is active and if a DIV write needs to be done. Note that there is no impact of a stuck shared update lock except in these conditions.

**Dynamic LPA**

When GDDRMAIN starts, it loads module GDDRXG1A into dynamic LPA. If the module is already loaded, GDDRMAIN does not reload it. Therefore, if maintenance occurs to that module, it has to be deleted from LPA by using operator command SETPROG LPA,DELETE,MODNAME=GDDRXG1A,FORCE=YES.

After the module is deleted, you can either restart GDDRMAIN so that it will reload it, or add it to LPA manually by issuing operator command SETPROG LPA,ADD,MODNAME=GDDRXG1A,DSNAME=gddr-linklib.

**IMPORTANT**

Before deleting the module you should be sure that the dynamic exit GDDREXG1GDDR (the last four characters are the GDDR subsystem name) is inactive as described in “Dynamic exits” below.

**Dynamic exits**

When GDDRMAIN starts, it creates a dynamic exit called GDDREXG1GDDR (the final four GDDR characters are variable and represent the GDDR subsystem name). LPA module GDDRXG1A (see “Dynamic LPA” above) is associated with this exit.

If consistency is requested for global variable backup or restore, the exit is activated. Should the backup or restore end abnormally (for example, by being cancelled), it is possible that the exit will be left active and may block or complain about global variable...
Performing GDDRMAIN Operations

updates. If this occurs, the exit can be manually deactivated by issuing SETPROG
EXIT,MODIFY,EXITNAME=GDDREXG1GDDR,MODNAME=
GDDRXG1A,STATE=INACTIVE.

Remote command processing

Remote command processing issues commands to GDDR-managed systems in parallel
and asynchronously acknowledges the completion status of each command. When the
WTOR associated with a remote command is issued, the command processor waits for a
response to the WTOR and also waits for the command to end. If the command ends, the
command processor cancels the associated WTOR.

GDDR script submission operator command

The GDDR script submission operation command is F GDDRMAIN,SCRIPT,name,option.

◆ <name> is the name of a GDDR script supported through this interface.

◆ <option1,option2,optionn> is comma-separated string of keywords that are described
  in detail in “Keywords for options” on page 101.

All keywords after SCRIPT are optional. However, if options are specified, then name must
first be specified. If script name is going to be supplied via a LOAD keyword, the script
name can be specified as *.

Use of the SCRIPT command results in work being sent to the GDDWXR worker to run
module GDDRMCSS under ISPF under the TMP. This is so that file tailoring is available. No
JCL modifications are required for any GDDR procs. File tailoring (if done) uses the ISPSLIB
DD in the GDDRWORK proc. The skeleton to be used (GDDRXSUB) is very similar to
GDDRXACT. It can be in a different dataset (still in the ISPSLIB DD concatenation) so that
the contents can be user-controlled. However, it does not have to be in a different dataset
and will ship in ISPSLIB.

The general flow (which can be modified by various keywords) is as follows:

F GDDRMAIN,SCRIPT,GDDRPA99
GDDM251I Script command work submitted

This is followed by a WTOR issued by a GDDRWORK address space:

GDDM252A Enter script parameters

The response to this is a comma-separated string of keywords as shown below. If CANCEL
is entered, the process ends with no script generation or submission. The message is
repeated until either END or CANCEL is entered or implied.

When END is entered, the following message is displayed showing all the options in effect:

GDDM253I Script GDDRPA99 parameters:
  Site   - DC1
  System - *
  ...
  Submit - YES

This is followed by the following message:

GDDM255A Confirm script generation (Y/N/CANCEL)
Performing GDDRMAIN Operations

- If N is entered, return to the GDDM252A message.
- If CANCEL is entered, the process ends at this point with no further action.
- If Y is entered, the script is generated (file tailored). The resultant JCL is saved if JCLSAVE was specified, and is submitted if SUBMIT was specified.

If a job is submitted, the following message occurs:

GDDM266I Script GDDRPA99 submitted, JOB jobname (jobid)

Authorization

Skeleton member GDDRXSUB supports secure access to GDDR scripts run through the SCRIPT console command. You may concatenate a PDS containing customized JCL skeleton GDDRXSUB to the GDDRMAIN ISPSLIB DD statement. This skeleton must provide a job-card.

As scripts will be submitted by GDDRMAIN, they would normally run with GDDRMAIN authorizations. The job-card provided in skeleton GDDRXSUB may have a USER= keyword. The named user would then be the only one with all the authorizations that are required to run a GDDR script. GDDRMAIN would only need SURROGAT authority to that Userid.

Keywords for options

<table>
<thead>
<tr>
<th>Table 14  Script keywords (page 1 of 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALLOWHMC</td>
</tr>
<tr>
<td>ALTPARMS</td>
</tr>
<tr>
<td>BGCLONE</td>
</tr>
<tr>
<td>CALLOVER</td>
</tr>
<tr>
<td>DRTPARMS</td>
</tr>
<tr>
<td>EXTBCV</td>
</tr>
<tr>
<td>HMCPRIM</td>
</tr>
<tr>
<td>HMCSECN</td>
</tr>
<tr>
<td>HMCWTOR</td>
</tr>
<tr>
<td>HMCLOADC</td>
</tr>
<tr>
<td>NONSTAR</td>
</tr>
</tbody>
</table>
### Table 14  Script keywords (page 2 of 2)

<table>
<thead>
<tr>
<th><strong>PRISWAP</strong></th>
<th>Name of primary AutoSwap group. Default taken from global.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PARM</strong></td>
<td>Allow specification of the script parameter. In general, there is no script parameter. However, in some cases, a parameter is required in order to identify that a 4-site script is being requested or to convey additional information. If the configuration is 4-site, the default parameter is set to 4SITE; if the configuration is 2-site, the default parameter is set to 2SITE.</td>
</tr>
<tr>
<td></td>
<td>• GDDRPA0A  Value is UP31 if the last run unplanned script was GDDRUP31; null otherwise.</td>
</tr>
<tr>
<td></td>
<td>• GDDRPA05  Value is SRDFS if in a 4-site configuration and SRDF/S replication to the other site in the recovery region is desired; null otherwise.</td>
</tr>
<tr>
<td></td>
<td>• GDDRPA06  In 4-site configurations, the value is CON if concurrent SRDF is desired and CAS if cascaded SRDF is desired. In other than 4-site configurations, the value is null.</td>
</tr>
<tr>
<td></td>
<td>• GDDRPA07  In 4-site configurations, the value is CON if concurrent SRDF is desired and CAS if cascaded SRDF is desired. In 2-site configurations, the value is 2SITE (default). In 3-site configurations, the value is null (default).</td>
</tr>
<tr>
<td></td>
<td>• GDDRPA36  RESTART if an MSC server is being restarted, null otherwise.</td>
</tr>
<tr>
<td></td>
<td>• GDDRPF29  In configurations without AutoSwap, must be NCAX; null (default) in all other cases.</td>
</tr>
<tr>
<td></td>
<td>• GDDRPM29  In 2-site configurations, the value is 2SITE (default); in all other configurations, the value is null (default).</td>
</tr>
<tr>
<td><strong>RRSKIP</strong></td>
<td>Value is YES or NO (any substring of YES or NO beginning with the first letter is allowed). No value implies YES. Default is NO.</td>
</tr>
<tr>
<td><strong>SCRIPT</strong></td>
<td>Value is the script name (same as first positional parameter on SCRIPT command).</td>
</tr>
<tr>
<td><strong>SDDFCLN</strong></td>
<td>Value is YES or NO (any substring of YES or NO beginning with the first letter is allowed). Default taken from global.</td>
</tr>
<tr>
<td><strong>SECSWAP</strong></td>
<td>Value is name of secondary AutoSwap group. Default taken from global.</td>
</tr>
<tr>
<td><strong>SITE</strong></td>
<td>Value is site name (DCx). Default is *.</td>
</tr>
<tr>
<td><strong>SYSTEM</strong></td>
<td>Value is system name. Default is *.</td>
</tr>
<tr>
<td><strong>TESTBCV1</strong></td>
<td>Value is Y or N, use TEST-set BCV devices at DC1.</td>
</tr>
<tr>
<td><strong>TESTBCV2</strong></td>
<td>Value is Y or N, use TEST-set BCV devices at DC2.</td>
</tr>
<tr>
<td><strong>TESTBCV3</strong></td>
<td>Value is Y or N, use TEST-set BCV devices at DC3.</td>
</tr>
<tr>
<td><strong>TESTBCV4</strong></td>
<td>Value is Y or N, use TEST-set BCV devices at DC4.</td>
</tr>
<tr>
<td><strong>USER</strong></td>
<td>Value is user ID. Can have no value, implying no user ID is to be set in script job. Default is blank (no value).</td>
</tr>
</tbody>
</table>
Performing GDDRMAIN Operations

Table 15  Control keywords

<table>
<thead>
<tr>
<th>AUTO</th>
<th>Value is ON or OFF. No value implies ON. ON means that no further WTORs should be issued and the process should proceed as if every WTOR were answered in the affirmative. Default is OFF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CANCEL</td>
<td>No value. Indicates that the script submission process should end immediately.</td>
</tr>
<tr>
<td>END</td>
<td>No value. Indicates that the entering of arguments is complete and the process should proceed to confirmation.</td>
</tr>
<tr>
<td>JCLSAVE</td>
<td>Value is DSN(MEMBER). This is the dataset to save the file tailored JCL. Can have no value to indicate that file tailored JCL is not be saved. Default is no value (no JCL saved). The value (if any) must be a partitioned dataset name and member; save to a sequential dataset is not allowed (ISPF restriction).</td>
</tr>
<tr>
<td>JCLSUB</td>
<td>Value is DSN(MEMBER). This is the dataset to be submitted without file tailoring. This parameter supercedes all script parameters except RRSKIP. Implies AUTO, END, SUBMIT.</td>
</tr>
<tr>
<td>LOAD</td>
<td>Value is DSN[[MEMBER]]. This is a dataset containing arguments which are to be loaded into current arg values. Any dataset/member can be used for LOAD only 1 time per LOAD. That is, dataset A cannot contain a LOAD of dataset A; dataset A cannot contain a LOAD of dataset B if B contains a LOAD of dataset A.</td>
</tr>
<tr>
<td>SAVE</td>
<td>Value is DSN[[MEMBER]]. This is a dataset into which the current script (not control) argument values (plus JCLSAVE if specified) will be saved.</td>
</tr>
<tr>
<td>SUBMIT</td>
<td>Value is YES or NO (any substring of YES or NO beginning with the first letter is allowed). No value implies YES. This indicates the resultant JCL should or should not be submitted. Default is YES.</td>
</tr>
</tbody>
</table>

Examples

User input is shown in italics.

F GDDRMAIN, SCRIPT, GDDRPA99, JCLSUB=USERID1.WORK.CNTL(PA21)
GDDM251I Script command work submitted
GDDM266I Script GDDRPA99 submitted, JOB GDDRPA21 (J1234567)
GDDM267I GDDR Script command for GDDRPA99 ended

F GDDRMAIN, SCRIPT, GDDRPA99
GDDM251I Script command work submitted
GDDM252A Enter script parameters
SITE=DC1, CALLOVER=01010101010101010101, SUBMIT=YES, USER=USERID1
GDDM252A Enter script parameters
JCLSAVE=USERID1.WORK.CNTL(PA21), NONSTAR=YES, ALLOWHMC=YYN, END
GDDM253I Script GDDRPA99 Parameters
SITE=DC1
SYSTEM=* PRISWAP=SWAP1 SECSWAP=SWAP2 CALLOVER=01010101010101010101 USER=USERID1 SDDFCLN=NO NONSTAR=YES ALLOWHMC=YYN RRISK=NO SUBMIT=YES JCLSAVE=USERID1.WORK.CNTL(PA21)
GDDM255A Confirm script generation (Y/N/CANCEL)
Y
Performing GDDRMAIN Operations

GDDM266I Script GDDRPA99 submitted, JOB GDDRPA99 (J1234567)
GDDM267I GDDR Script command for GDDRPA99 ended

F GDDRMAIN,SCRIPT,GDDRPA99
GDDM251I Script command work submitted
GDDM252A Enter script parameters
LOAD=USERID1.WORK.CNTL(PA21),SITE=DC2,END
GDDM253I Script GDDRPA99 Parameters
  SITE=DC2
  SYSTEM=*  
  PRISWAP=SWAP1
  SECSWAP=SWAP2
  CALLOVER=0101010101010101010101
  USER=USERID1
  SDDFCLN=NO
  NONSTAR=YES
  ALLOWHMC=YYN
  RRSKIP=NO
  SUBMIT=YES
GDDM255A Confirm script generation (Y/N/CANCEL)
  Y
GDDM266I Script GDDRPA99 submitted, JOB GDDRPA21 (J1234567)
GDDM267I GDDR Script command for GDDRPA99 ended

F GDDRMAIN,SCRIPT,GDDRPA21,LOAD=USERID1.WORK.CNTL(PA21),SITE=DC2,AUTO
GDDM251I Script command work submitted
GDDM253I Script GDDRPA99 Parameters
  SITE=DC2
  SYSTEM=*  
  PRISWAP=SWAP1
  SECSWAP=SWAP2
  CALLOVER=0101010101010101010101
  USER=USERID1
  SDDFCLN=NO
  NONSTAR=YES
  ALLOWHMC=YYN
  RRSKIP=NO
  SUBMIT=YES
GDDM266I Script GDDRPA99 submitted, JOB GDDRPA21 (J1234567)
GDDM267I GDDR Script command for GDDRPA99 ended

F GDDRMAIN,SCRIPT,GDDRPA99
GDDM251I Script command work submitted
GDDM252A Enter script parameters
SITE=DC1,CALLOVER=0101010101010101010101,SUBMIT=YES,USER=USERID1
GDDM252A Enter script parameters
CANCEL
GDDM257I GDDR Script command for GDDRPA99 cancelled
CHAPTER 5
Using GDDR Online Facilities

This chapter describes the GDDR online interface.

- Primary Options menu .......................................................... 106
- Option P: Profile—Update Personal GDDR ISPF Profile .................. 108
- Option M: Maintenance—GDDR Setup and Maintenance ............... 109
- Option G: GDDR Config—View GDDR Configuration .................. 195
- Option C: Checkup—Perform Pre-script Checkup ........................ 196
- Option S: Scripts — Run GDDR Scripts .................................... 208
- Option T: Timing—View GDDR Script Statistics .......................... 213
- Option A: Actions—Perform GDDR Actions ............................... 215
- Option ST: eSTEM - Run EMC Started Task Execution Manager ... 227
Primary Options menu

The GDDR online facilities permit z/OS system programmers to perform configuration and administration tasks and operators to perform operator functions. To use the GDDR online facilities interface, invoke the GDDR invocation REXX exec which was customized in “Customize the GDDR ISPF user interface invocation REXX exec” on page 78. The following panel appears:

--- GDDR - Primary Options Menu for GDDR_50 ---

Option -->
P  Profile       Update personal GDDR ISPF Profile        This System: GA2LB148
M  Maintenance   GDDR Setup and Maintenance                 This Site: DC1
G  GDDR Config   View GDDR configuration                  This region: RG1
C  Checkup       Perform pre-script checkout                Master-C: GA2LB148
S  Scripts       Run GDDR Scripts                           Primary Site: DC1
T  Timing        View GDDR Script Statistics                Primary DASD: DC1
A  Actions       Perform GDDR Actions                      Automation: ON
ST eSTEM         Run EMC Started Task Execution Mgr.        Planned script: None
                 Unplanned script: None                           Appl ID.: GDDR

EMC Geographically Dispersed Disaster Restart 5.0.0
Copyright © 2007-2016 EMC Corporation.

Select an option and press <Enter>
Press <F3> to Terminate GDDR ISPF

Control fields

At the right side of the panel, the GDDR Primary Options menu displays the current settings for several GDDR control values.

Note: The fields described below also appear on other GDDR panels.

Figure 6 Primary Options menu

◆ This System
   Specifies the current system.

◆ This Site
   Specifies the current site.

◆ This Region
   Specifies the current region.

◆ Master-C
   Shows the name of the C-System that currently owns the GDDR master function. In environments supported by cross-site host-DASD channels, the master C-System is located at the opposite site from the business applications. Where cross-site host-DASD channels are not available, the master C-System will be the C-System at the site where the business applications are hosted. Thus, if the business applications are running at site DC1, the master C-System will be the C-System at site DC1. Most GDDR actions are allowed only when logged onto the master C-System.
Primary Options menu

● Primary Site
  Indicates the site where the business applications are currently running.

● Primary DASD
  Indicates the site at which the R1 DASD currently reside. Generally, the sites indicated by Primary Site and Primary DASD should be the same.

● Automation
  Indicates the current availability of GDDR automation functionality. The automation state can be changed from the Setup and Maintenance Menu. Refer to "Changing the automation state" on page 109 for details.

● Planned Script
  Indicates which Planned script, if any, is currently in progress. A Planned script is considered to be in progress if it has been started but has not yet completed successfully; in this case, it is eligible for restart. If no Planned script is currently in progress, 'None' displays in this field.

● Unplanned Script
  Indicates which Unplanned script, if any, is currently in progress. An Unplanned script is considered to be in progress if it has been started but has not yet completed successfully; in this case, it is eligible for restart. If no Unplanned script is currently in progress, 'None' displays in this field.

Note: In the version number displayed after EMC Geographically Dispersed Disaster Restart at the bottom of the panel, nn reflects the latest maintenance level and pfname is the name of the latest PTF, when the GDDR SMP/E Post-Maintenance procedure is used.

Options

At the left of the panel, a series of options are listed. To invoke one of the following actions, enter the appropriate option on the command line and press Enter. These options are described in the following sections.

● "Option P: Profile—Update Personal GDDR ISPF Profile" on page 108
● "Option M: Maintenance—GDDR Setup and Maintenance" on page 109
● "Option G: GDDR Config—View GDDR Configuration" on page 195
● "Option C: Checkup—Perform Pre-script Checkup" on page 196
● "Option S: Scripts — Run GDDR Scripts" on page 208
● "Option T: Timing—View GDDR Script Statistics" on page 213
● "Option A: Actions—Perform GDDR Actions" on page 215
● "Option ST: eSTEM - Run EMC Started Task Execution Manager" on page 227
Option P: Profile—Update Personal GDDR ISPF Profile

Specify option P in the GDDR Primary Options menu (on page 106) to access the GDDR ISPF profiles. The following panel appears:

--- GDDR - Change GDDR ISPF Profile Variable Values ---
Command ===> GDDR Subsystem Name ===> GDDR
JCL dataset ===> ISPF skeleton dataset ===> Global variable backup ===> Jobcards for your user:
// // // // Press ENTER to apply updates
Press <F3> when finished
Enter CANCEL to return without changing any profile variable values
Enter CLEAR to set all values to null and exit
Enter RESBT to restore the values as they were upon entry

Figure 7 Change GDDR ISPF Profile Variable Values panel

Change your GDDR profile

Use this panel to specify GDDR-related variables pertaining to your Userid.

Note: The GDDR Subsystem Name field is not enabled for editing in GDDR Version 4.0.

1. Specify the JCL dataset.
   Type the name of the PROCLIB dataset holding GDDRPROC and GDDRWORK members customized during the GDDR integration procedure “Customize member GDDRPROC” on page 78.

2. Specify the ISPF skeleton dataset.
   Type the name of a dataset containing the GDDR file tailoring skeletons created when you downloaded the install cartridge. This dataset is used to retrieve ISPF file tailoring skeletons for the parameter activation, when done in background mode and when scripts are submitted.

3. Specify the global variable backup.
   Type the name of the parameter backup dataset to be used for parameter management functions, defined during the GDDR integration procedure “Allocate the parameter management datasets” on page 64.

   Note: Whenever a dataset name is required, you must specify a fully-qualified dataset name. TSO prefixing does not apply to any dataset name specified within GDDR.

4. Enter jobcard information.
Using GDDR Online Facilities

This field is displayed conditionally for EMC GDDR users who are authorized to activate parameter changes and to manage GDDR scripts. The jobcard information entered here applies only to the current user and is used for parameter activate and script submission tasks. Always include the REGION=0M jobcard parameter. For example:

```plaintext
 ===>//JOBNAME JOB (acct-number),gddr-job,CLASS=A,REGION=0M,
 ===>// MSGCLASS=A,USER=GDDR,NOTIFY=GDDR
 ===>/*JOBPARM LINES=999999
 ===>*/
```

5. Press F3 to save the values and return to the Primary Options Menu.

Option M: Maintenance—GDDR Setup and Maintenance

Specify option M on the GDDR Primary Options menu (on page 106) to access the GDDR setup and maintenance functions. The following panel appears:

![Setup and Maintenance Menu]

**Changing the automation state**

At the right side of the panel, the menu displays the current settings for several GDDR control values. The Automation field indicates the current availability of GDDR automation functionality. To change the automation state, type GDDR ON or GDDR OFF in the Option line of the Setup and Maintenance Menu. The displayed value of the automation setting on the right of the panel reflects the change immediately.

**Note:** The GDDR ON/OFF command is valid only on the master C-System.

- When you specify GDDR OFF to turn GDDR automation off, the EMC GDDR Event Monitor does not respond to events that would normally indicate a storage or system failure. Therefore, the OFF automation status should be used only when system availability may be impacted by scheduled maintenance activities.

After you enter GDDR OFF, messages similar to the following appear:
When you specify **GDDR ON** to turn GDDR automation on, GDDR operators will be able to run both Planned and Unplanned scripts.

The following sections describe the Setup and Maintenance Menu options.

- **“Option P: Manage GDDR Parameters” on page 110**
- **“Option D: Message, Debug and Trace Options” on page 188**
- **“Option Q: Manage GDDR Internal Command Queue” on page 189**
- **“Option H: Perform HMC Discovery” on page 190**
- **“Option R: Refresh GDDR Message Table” on page 191**
- **“Option S: Manage GDDR System Variables” on page 192**
- **“Option T: Transfer Master C-System” on page 194**

**Option P: Manage GDDR Parameters**

Specify option **P** in the Setup and Maintenance Menu (Figure 8 on page 109) to access the GDDR parameter management functions. The following panel appears:

```
------------------- GDDR - Parameter Management Options Menu ------------------
Option ===>     
  B ackup       Manage GDDR Parameter backups             This System: GA2LB148
  G DDRACDD     Manage GDDRACDD                             This Site: DC1
  P ARMLIB      Manage PARMLIB DSN backups                   Master-C: GA2LB148
  .---------- Parameter Load functions ----------.        Primary Site: DC1
  |  I nput       Select Parameter Input Dataset |        Primary DASD: DC1
  |  C onfig      Define Configuration Basics    |          Automation: ON
  |  D ata        Define Data Storage Objects    |      Planned script: None
  |  H ost        Define Data Host Objects       |    Unplanned script: None
  |  O ptions     Specify GDDR Options           |
  |                                              |
  |______________________________________________|

Current work data set SYSU.GDDR500.PARMS.WORK.M38
EMC Geographically Dispersed Disaster Restart 5.0.0
Copyright © 2007-2016 EMC Corporation.

Select an option and press <Enter>
Press <F3> to return to the GDDR Setup and Maintenance Menu
```

Figure 9 Parameter Management Options Menu

**Note:** The options initially displayed in this panel may vary. The panel does not show options C,D,H,O,V,A until the actions described in “**To prepare to edit the current GDDR parameter set:**” on page 122 are completed.

The menu options allow you to manage GDDR parameter backups and define and maintain GDDR parameters using the Parameter Load wizard.
The following sections describe the Parameter Management Menu functions:

- “Option B: Manage GDDR Parameter Backups” on page 119
- “Option G: Manage GDDRACDD” on page 123
- “Option P: Manage PARMLIB DSN backups” on page 124
- “Option G: Manage GDDRACDD” on page 123
- “Option C: Define Configuration Basics” on page 131
- “Option D: Define Data Storage Objects” on page 138
- “Option H: Define Host Objects” on page 146
- “Option O: Specify GDDR Options” on page 168
- “Option V: Validate GDDR Parameter Set” on page 181
- “Option A: Activate GDDR Parameter Set” on page 183

Understanding the Parameter Load wizard workflow

The maintenance and initial definition of GDDR global variables (parameters) is performed using the Parameter Load wizard, which is based on an iterative change process. Changes are applied initially to a consistent work copy of the active GDDR global variables. These changes are then applied to the active GDDR global variables in an Activation step (preceded by an implicit or explicit Validation).

Figure 10 describes the parameter backup, edit serialization, validation and activation process, and the relationship with GDDRMAIN and its dependent address spaces. These operations are represented in the GDDR Parameter Load wizard work flow beginning with Figure 11 on page 114.
Using GDDR Online Facilities

Figure 10  GDDR parameter management

Using the Auto-Discovery feature

The Parameter Load wizard helps you define and maintain the parameters that describe your particular disaster restart topography and EMC SRDF configuration by populating values available through the Auto-Discovery feature.

System parameter changes which are necessitated by changes to the definitions of C-Systems or GDDR-managed systems are initiated by performing the changes to the CSYSSSITE and COMM parameters in the GDDMPARM member, as described in Chapter 9, “Performing Maintenance Procedures.” After performing the parameter changes required in the GDDMPARM member, the changes are propagated to the active GDDR parameter set by use of the Parameter Load wizard.

The Auto-Discovery feature requires that the GDDRMAIN tasks have been started on each C-System and GDDR-managed system before you start the Parameter Load wizard using the Select Parameter Input Dataset panel, shown on page 123.
Using the Parameter Load function panels

The series of Parameter Load function panels enable you to display or define parameter values organized by object category. The panels use function keys as follows so that fields are presented in data dependency order:

- F5 returns you to the immediate preceding panel in the series.
- F6 takes you to the next panel in the series.
- F3 returns you to the menu panel for objects of the same type.

Each Parameter Load display and definition panel does the following:

- Displays values for named parameters from the selected parameter dataset member subject to the precedence rules that apply to data returned to the wizard from the Auto-Discovery feature.
- Displays values which were saved at a previous Parameter Load panel.

If no value is supplied for a field by the selected parameter dataset member, the first time a value for that field is specified as panel input and saved to the Work dataset, the value is propagated to all succeeding panels which contain the same field.

Changes to field values in parameter display and definition panels are saved to your Work dataset when you type SAVE on the Command line of the panel. If after editing the values in the panel and before you use the SAVE command, you wish to restore the original values displayed in the panel, type LOAD in the panel. If you have ‘Saved’ interim changes for values on the current panel, then the Load command restores the last saved values.

**Note:** Each panel in which you use the SAVE command will be utilized for a partial parameter activation, whether or not you performed changes on the panel.

Parameter Load wizard prerequisites

The following items are required for use of the Parameter Load wizard:

- GDDMPARM has been defined with C-System and production system COMM statements and is consistent on all systems.
- GDDRMAIN tasks must be active and communicating on all C-Systems and optionally also on GDDR-managed systems (recommended).
- Parameter backup and parameter work datasets are allocated on all C-Systems.
  - Parameter backup datasets are populated by HBM on all C-Systems.
  - Parameter work datasets are populated during Parameter Load wizard use.
- One “Last Activated Parm Dataset” is pre-allocated on all C-Systems.
- Parameter Validate/Activate is performed on the C-System which is the intended Master C-System for the configuration being defined (exception for missing C-System communication).
- The following rules are true for confirming your intended Master C-System on initial parameter load:
  - The Master C-System must be on the secondary DASD site for AutoSwap configurations, and on the primary DASD site for non-AutoSwap configurations.
• The Validate operation queries the storage environment using the site, system, gatekeeper, RDF group, and device ranges presented as Parameter Load wizard input. The presence of R1 devices determines the current primary DASD site. If the query is unable to find R1 devices, validation uses the primary DASD site found in the parameter input deck.

**Step 1**

Initial definition of global variables is performed using no prior backup, with automation which populates system parameters available from the GDDMPARM parameters defined during GDDRMAIN installation.

Maintenance of global variables is applied using a current global variable backup as the base.

![Figure 11 Parameter Load wizard work flow — step 1](image)

**Step 2**

Parameter Load wizard usage is limited to one active session at a time.

When you signal the intent to update parameters by performing the Prepare command, access to the Parameter Load wizard panels by other users is prevented. You can review the EMC GDDR configuration parameters at any time using the Review command.

Serialization of the panels is held until you have completed an Activate. (An emergency procedure is available in case this serialization creates a problem.)

![Figure 12 Parameter Load wizard work flow — step 2](image)
Step 3

The Parameter Load wizard updates the parameter work dataset which was specified at the time the Prepare (or Review) command was issued. It does this by storing each global variable found in the selected input dataset to the appropriate work dataset member.

![Diagram of Parameter Load wizard work flow — step 3](image)

**Step 4**

Each Parameter Load Wizard panel allows user input for the GDDR global variables found in the corresponding work dataset member. Selected Parameter Load wizard panels are populated with data from GDDMPARM parameters, either directly or by usage of discovery functions.

When discrepancies exist between GDDMPARM derived data and data which is provided from the ‘base’ global variable backup, the difference is indicated in the panels by displaying this data in yellow font.

![Diagram of Parameter Load wizard work flow — step 4](image)
Step 5
The Validate function assembles panel-bound work PDS members as well as the Z97NOPNL member into the Z00PARMS member and presents that to the validation engine.

When using the FULL option, all panel-bound members are included. When using the PARTIAL option, only updated members are included. When using the EVENT(RESET) option, the Z97NOPNL member is ignored.

The Validate function performs live queries against the storage environment. Error messages are reported in the Z00VOUT member of the parameter work dataset. The VIEW command on the Validate panel can be used to review these messages.

Figure 15 Parameter Load wizard work flow — step 5

Step 6
The Activate function assembles panel-bound work PDS members as well as the Z97NOPNL member into the Z00PARMS member and presents that to the validation engine. If this implicit validation succeeds, it proceeds to the Activation steps, which include taking a global variable backup both before and after the Activation.

When using the FULL option, all panel-bound members are included. When using the PARTIAL option, only updated members are included. When using the EVENT(RESET) option, the Z97NOPNL member is ignored.

The implicit Validate performed during the Activate function performs live queries against the storage environment.

Error messages are reported in the Z00AOUT member of the parameter work dataset. The VIEW command on the Activate panel can be used to review these messages.
The Activate function copies your parameter work dataset to the Last Activated Parm Dataset and makes the new parameter set active on all C-Systems. Activate provides options to reset GDDR event variables and clear the GDDR command queue both locally and on other C-Systems.

If the GDDR Heartbeat Monitor and Event Monitor are running, Activate will automatically stop them, and restart them when Activate is complete.

Using the parameter review functions

GDDR makes the current parameters available for review through the GDDR - Parameter Management Options Menu, without the Validate and Activate capabilities. For example, an audit of your disaster restart operations may include a review of GDDR parameters. The GDDR parameters review capability enables a third-party reviewer to access a current parameter backup with no exposure from unwanted changes to the active parameter values.

Authority to use the GDDR parameters review capability is granted to members of the GDDR$REV RACF group with READ access to the GDDRISPF.ACCESS, GDDRISPF.SETUP.ACCESS, and the GDDRISPF.SETUP.PARMS.REVIEW facility profiles, as described in Table 8 on page 57. When authority has been granted, the GDDR Parameter Management Options Menu is displayed without the Validate and Activate Parameter Load functions as shown in Figure 18 on page 118.

The parameter review is performed by making a backup of the current parameters, using “Option B: Manage GDDR Parameter Backups” on page 119. The reviewer selects the backup member to be used to prepare the reviewer's parameter work dataset as shown in Figure 17 on page 118 and described in detail in “Option G: Manage GDDRACDD” on page 123.
Using GDDR Online Facilities

Figure 17  Select Parameter Input Dataset for parameter review

Within the Select Parameter Input Dataset panel, after you select a backup member which
will be used to populate your parameter work dataset, the following message appears:

Ready to start preparation of the work dataset. Press <F3> to proceed.

To display the reviewer's version of the GDDR - Parameter Management Options Menu,
type review in the Command field of the Select Parameter Input Dataset panel, rather than
pressing <F3> to proceed, as the message directs.

The reviewer's version of the Parameter Management Options Menu is displayed as shown in
Figure 18.

Figure 18  Reviewer's version of the Parameter Management Options Menu

The Parameter load functions listed in the reviewer's version of the Parameter
Management Options Menu display the parameter values of the current backup member.
The parameters are described in the following sections:

- “Option G: Manage GDDRACDD” on page 123
- “Option C: Define Configuration Basics” on page 131
- “Option D: Define Data Storage Objects” on page 138
- “Option H: Define Host Objects” on page 146
- “Option O: Specify GDDR Options” on page 168

Option B: Manage GDDR Parameter Backups

GDDR automatically creates GDDR parameter backups before and after parameters are activated and during Heartbeat Monitor initialization. However, there may be other times when it is necessary to back up the GDDR parameters. For example, you can create a backup to be used as input to your next parameter definition. The Manage GDDR Parameter Backups option allows you to do this.

When you specify option B in the Parameter Management Options Menu shown in Figure 9 on page 110, the following panel appears:

--- Table: GDDR - Manage GDDR Parameter Backups ---
Command ==>

- B Backup Create new GDDR Parameter Backup
  
  Backup Dataset ===> JABCDE1.GDDR500.BACKUP.PARMS
  Backup description ===> __________________________________________
  Backup consistency ===> Y (Y/N/1-9)

Press <F3> to return to the GDDR Parameter Management Options Menu
Line commands: B rowse, E edit, D delete, M modify description
S elect for Parameter Load
--- Table: Previous Backups ---

<table>
<thead>
<tr>
<th>Act</th>
<th>Member</th>
<th>Date</th>
<th>Time</th>
<th>Userid</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>ISPE3025</td>
<td>09/05/25</td>
<td>14:30</td>
<td>PQRSTU1</td>
<td>A:testing</td>
</tr>
<tr>
<td>_</td>
<td>ISPE1024</td>
<td>09/05/25</td>
<td>14:30</td>
<td>PQRSTU1</td>
<td>B:testing</td>
</tr>
<tr>
<td>_</td>
<td>ISLG1924</td>
<td>09/05/21</td>
<td>16:19</td>
<td>OPQRST1</td>
<td>B:ptw test</td>
</tr>
<tr>
<td>_</td>
<td>ISLG1852</td>
<td>09/05/21</td>
<td>16:18</td>
<td>OPQRST1</td>
<td>B:ptw test</td>
</tr>
<tr>
<td>_</td>
<td>ISLG1613</td>
<td>09/05/21</td>
<td>16:16</td>
<td>OPQRST1</td>
<td>B:ptw test</td>
</tr>
<tr>
<td>_</td>
<td>ISLG1152</td>
<td>09/05/21</td>
<td>16:11</td>
<td>OPQRST1</td>
<td>B:ptw test</td>
</tr>
<tr>
<td>_</td>
<td>ISL00444</td>
<td>09/05/19</td>
<td>16:04</td>
<td>OPQRST1</td>
<td>B:ptw test</td>
</tr>
<tr>
<td>_</td>
<td>ISUF4346</td>
<td>09/05/19</td>
<td>15:43</td>
<td>RGHIJK1</td>
<td>B:ptw test</td>
</tr>
</tbody>
</table>

Figure 19 Manage GDDR Parameter Backups panel

The following fields are provided on the Manage GDDR Parameter Backups panel:

- Backup Dataset
  The backup dataset name can be one of the following:
  - The GDDR Variable backup dataset name specified in your GDDR ISPF Profile Variable Values panel shown in Figure 7 on page 108.
  - The dataset name associated with the BKUPVARS dataset type in the Define GDDR Datasets panel shown in Figure 35 on page 136.
  - Any PDS or PDSE dataset name you type in this field.
Using GDDR Online Facilities

- **Backup description**
  Use this field to describe the circumstances regarding the backup.

- **Backup consistency**
  Use this field to enforce the consistency of the parameter set being backed up. GDDR global variable backup consistency is enforced by activating a dynamic exit which protects global variables from updates in the time between the start and the completion of the backup. Refer to "Dynamic exits" on page 99 for more information about this exit.

  EMC recommends that you perform all parameter backups with consistency. If you take a backup without consistency, you should manually quiesce EMC GDDR activity to prevent global variable changes during the time it takes to perform the backup. If backups fail regularly due to concurrent activity, specify a number from 1 to 9 to increase the number of retries.

- **Previous Backups**
  Parameter backup dataset members are used as input by the parameter definition process. New members are created by the Heartbeat Monitor at initialization, by the user using the GDDR ISPF interface, and during Activate to capture a "Before" and "After" image of the existing parameters. In the example shown in Figure 19, there is a list of dataset members that had been created by previous parameter activations, where "B" indicates Before and "A" indicates After.

- **Member**
  For either automatic or requested backups, the member name is generated automatically by the GDDR parameter backup processor. It encodes the time of the backup (including year, month, day, hour, minute, and second).

- **Line commands**
  Line commands enable the following actions in the Action (Act) column in the Previous Backups member list:
  - **B** Browse
    Enter the ISPF browse mode for the dataset and member.
  - **D** Delete
    Delete the dataset member.
  - **E** Edit
    Enter ISPF edit mode for the dataset and member.
  - **M** Modify the member description
    Edit the parameter backup member description.
  - **S** Select for Parameter Load
    Select a member. The selected member is displayed in the Selected Member field of the Select Parameter Input Dataset shown in Figure 23 on page 124.
Using the parameter backup feature

Backups of all parameters are automatically taken during Heartbeat Monitor initialization and as part of GDDR parameter set Activation. A parameter backup is also taken when you specifically request a backup using option B on the GDDR Parameter Management Options menu, or when you use the GDDRMAIN GVB console command with the dw\((hhmm)\), *(hhmm), or NOW operand as described in Table 12, “GDDRMAIN console command summary,” on page 89.

For the automatic and GDDRMAIN GVB initiated backups, the dataset to which the backup is written is obtained from the Dataset Name field of the Define GDDR Datasets panel, dataset type of BKUPVARS. See Figure 35, “Define GDDR Datasets panel,” on page 136. For requested backups, the dataset name of dataset type BKUPVARS is presented as a default, but another backup dataset may be specified.

To create a new GDDR parameter backup:

1. Specify option B Backup on the Parameter Management Options Menu (Figure 9 on page 110). If the Global variable backup dataset name specified in your GDDR ISPF Profile variable values differs from the dataset associated with BKUPVARS in the Define GDDR Datasets panel shown in Figure 35 on page 136, then the Select Dataset for GDDR Parameter Backup panel shown in Figure 20 initially displays. This panel presents you with both datasets so you may choose the one to use for parameter backup:

   ![Figure 20 Select Dataset for GDDR Parameter Backup](image)

   **Figure 20** Select Dataset for GDDR Parameter Backup

2. Select a dataset name and press Enter to proceed to the Manage GDDR Parameter Backups panel shown in Figure 19 on page 119.

3. Specify a dataset name in the Backup Dataset field.

4. Type B (Backup) in the command line in the Manage GDDR Parameter Backups panel and press Enter.

When the parameter backup processing completes, the following messages display:

```
GDDR721I GDDR Starting GDDR Global Variable Backup
GDDP304I Backup successful, 503 lines written to PARMOUT
GDDR639I GDDR Completed GDDR Global Variable Backup with rc 0
```
To prepare to edit the current GDDR parameter set:

The Manage GDDR Parameter Backups panel also lets you prepare to edit the current (most recent) GDDR parameter set. You do this when you select an existing member of the parameter backup dataset which is displayed.

1. Type S (Select) next to the desired member in the Previous Backups list displayed in the Manage GDDR Parameter Backups panel shown in Figure 19 on page 119 and press Enter.

2. The Select Parameter Input Dataset panel displays. Specify the panel options as described in “Option G: Manage GDRACDD”.
Option G: Manage GDDRACDD

When you specify option G in the Parameter Management Options Menu shown in Figure 9 on page 110, the following panel appears.

```
------------ GDDRACDD - Automated Configuration Discovery for DASD ------------
Command ===>
Press <F3> to exit
Enter the required information. Press <Enter> to run GDDRACDD

More: +

Run where? === FORE (FOReground or BACKground)
Hold Class === X SDFS Hold Output class.
Range size === (default: 256)

ALIGN === (Y or N)
DLM === (Y or N)
BCV === (Y or N)
RAID === (Y or N)
META === (Y or N)
SIZE === (Y or N)

GLOBALS === (CHECK or NONE)

CONFIG site === (DC1/DC2/DC3/DC4)
Topology === (CONCurrent or CASCaded)
Sort === (Y or N)
Group break === (Y or N)
Debug === 0 (0, 1, or 2)

------------ GDDRACDD - Automated Configuration Discovery for DASD ------------
Command ===>
Press <F3> to exit
Enter the required information. Press <Enter> to run GDDRACDD

More: -

CONFIG site === (DC1/DC2/DC3/DC4)
Topology === (CONCurrent or CASCaded)
Sort === (Y or N)
Group break === (Y or N)
Debug === 0 (0, 1, or 2)

SCF subsys === SCF (SCFx)
GDDR subsys === GDDR (GDDx)

ACDDPARM dsn ===
GDDR JCLLIB ===
Output files:
E04SRDFD dsn ===
E05TFDEV dsn ===

******************************************************************************
```

Figure 21  GDDRACDD - Automated Configuration Discovery for DASD panel

“Automated Configuration Discovery for DASD (GDDRACDD)” on page 240 describes how to configure the options in this panel.
Option P: Manage PARMLIB DSN backups

When you specify option P in the Parameter Management Options Menu shown in Figure 9 on page 110, the following sub-panel appears.

```
------------------------- GDDR - Parameter Management Options Menu -------------------------
Option ===> P

B ackup         Manage GDDR Parameter backups             This System: GA2LB28
G DDRACDD     Manage GDDRACDD                             This Site: DC2
P ARMLIB      Manage PARMLIB DSN backups                   Master-C: GA2LB29

----- GDDRIPS - Dataset name to Backup. ----- Primary Site: DC1
| SYSU.GDDR500_CUSTOM.LIB.K138              Primary DASD: DC1
| Without Quotes                            Automation: ON
| Enter to process       PF3 to Exit            Planned script: None
|                                               Unplanned script: None
|                                               |
+-----------------------------------------------+

Current work data set SYSU.GDDR410.PARMS.WORK.K138
EMC Geographically Dispersed Disaster Restart 5.0.0
Copyright © 2007-2016 EMC Corporation.

Select an option and press <Enter>
Press <F3> to return to the GDDR Setup and Maintenance Menu
```

Figure 22 Dataset name to Backup panel

Type the desired PARMLIB name in the panel and press Enter to back up the PARMLIB or PF3 to exit without a backup.

Option I: Select Parameter Input Dataset

When you specify option I in the Parameter Management Options Menu shown in Figure 9 on page 110, the following panel appears.

```
------------------------ GDDR - Select Parameter Input Dataset -- Row 1 to 3 of 3
Command ===>

Parameter input dataset ===> HLQ.GDDR500.PARMS.BKUP
Selected Member ===>        Unselect? ===> N (Y/N)
Parameter Load work-dataset ===> HLQ.GDDR500.PARMS.WORK
Parameter Load description ===> Change control #090926

Currently activated GDDR Parameter source:
HLQ.GDDR500.PARMS.BKUP(15LG1852) on SYS1
load 38 (User: OPQRST1 10/03/08 03:25)

Please select a member from the list below.
Press <F3> to return to the Parameter Management Options Menu
Line commands: S elect, U nselect, B rowse, E dit

---------- Parameter Input Member Selection ----------
Act  Member    Date   Time   Userid                  Description
--- -------- -------- ----- -------- ------------------------------------------
_  --NONE-- 10/03/08 06:44 PQRSTU1   GDDR parameter load from scratch
_  J2R90530 10/02/27 09:05 PQRSTU1   A: Change control #100227
_  J2R90501 10/02/27 09:05 PQRSTU1   B: Change control #100227

Figure 23 Select Parameter Input Dataset panel

The following fields are provided on the Select Parameter Input Dataset panel:

- Parameter Input Dataset

Specify the parameter backup dataset name. It can be one of the following:
• The Global Variable backup dataset name specified in your GDDR ISPF Profile Variable Values panel shown in Figure 7 on page 108.

• The dataset name associated with the BKUPVARS dataset type in the Define GDDR Datasets panel shown in Figure 35 on page 136.

• Any PDS or PDSE dataset name you type in this field.

◆ Selected Member (display only)
Indicates the currently selected backup member.

◆ Parameter Load Work-dataset
Enter the parameter load work dataset that was defined in “Allocate the parameter management datasets” on page 64. This dataset contains your in-process work, enabling you to assemble a complete parameter set by saving your input data from each of the parameter definition panels, and returning to the task at a later time.

Note: The parameter input dataset and the parameter load work dataset must be two different datasets. This is because the contents of the parameter load work dataset are overwritten when exiting the Select Parameter Input Dataset panel.

The parameter load work dataset must also be different from the dataset that is defined to GDDR as the "Last Activated Parameter Dataset".

◆ Parameter Load Description
Use this field to describe the circumstances for the parameter load or update.

◆ Currently activated GDDR Parameter source
Indicates which work dataset or parameter backup dataset was used for the last Activate, the system where the last Activate was done, the timestamp for that activation, the Userid, and the circumstances of the update.

◆ Parameter Input Member Selection
Displays the current parameter members.

◆ Line commands
Line commands enable the following actions in the Action (Act) column in the Parameter Input Member Selection list:

• S Select
  Specify the Select command to display the Select Parameter Input Dataset panel for the dataset and member.

• U Unselect
  Specify the Unselect command to remove the member.

• B Browse
  Specify the Browse command to enter the ISPF browse mode for the member.

• E Edit
  Specify the Edit command to enter ISPF edit mode for the member.
Using GDDR Online Facilities

**Note:** The parameter backup member information shown in the Select Parameter Input Dataset panel is maintained in GDDR variables for all parameter backups created by GDDR—either as an impromptu user backup, as a backup created before and after a parameter set activation, or as a backup created by the Heartbeat Monitor during initialization. These actions are referred to as registered backups. Where the backup description starts with B: or A: this reflects parameter backups taken by a previous Activate process, where B indicates "Before" and A indicates "After" the actual update of GDDR parameters.

Unregistered or outside parameter backup members may be copied into your backup dataset containing registered parameter backups to facilitate configuration of additional GDDR complexes or reconfiguration of existing GDDR-managed environments. Unregistered parameter backup members are displayed when you enter the 'showall' command on the command line of the Select Parameter Input Dataset panel. The showall command updates the Parameter Input Member Selection list with members which you have previously copied to your specified parameter input dataset.

**To edit the current GDDR parameter set:**

1. In the Select Parameter Input Dataset panel, press F3 to edit your GDDR parameter set by copying the contents of the selected parameter backup member to your parameter work dataset.

   This operation overwrites the contents of your parameter work dataset.

   A confirmation panel requests you to confirm the operation:

   ![Figure 24 Prepare Work Dataset for Parameter Load confirmation panel](image)

   **Figure 24** Prepare Work Dataset for Parameter Load confirmation panel

2. Enter **CONTINUE** to proceed and overwrite the contents of the work dataset with the selected member of the parameter input dataset.
A panel similar to the following displays the status of the operation:

```
--- ---------- GDDR - Select Parameter Input Dataset -- Row 1 to 5 of 5
C
--- ---------- GDDR - Prepare Work Dataset - Status ----------

Validating Input variables:

- TEMPVAR
- TIMEFINDER
- USEROPT
- UTILITY

*** PLEASE WAIT ***
R
--- ---------- Parameter Input Member Selection ----------

Act Member Date Time Userid Description
--- -------- -------- ----- -------- ------------------------------------------
_  --NONE-- 09/06/02 09:07 JABCDE1  GDDR parameter load from scratch
_  I5PEZ11A 09/05/25 14:23 PQRSTU1  SYS1: Still 2-Site SRDF/A
_  I5CZ2142 09/05/12 16:24 PQRSTU1 SYS1: 2-Site SRDF/A
_  I5841951 09/05/11 04:19 PQRSTU1 GDDR BU 01
_  I59B2428 09/05/09 11:24 PQRSTU1 SYS1: 2-Site SRDF/A

*** Bottom of data ***
```

**Figure 25** Prepare Work Dataset status panel

Following completion of the Prepare Work Dataset process, message GDDI172I confirms that your work dataset has been populated with the parameter dataset member you selected using Option I, Select Parameter Input dataset, as shown in Figure 23 on page 124.

The Parameter Management Options Menu redisplay:

```
--- ---------- GDDR - Parameter Management Options Menu ----------
Option =>

B ackup       Manage GDDR Parameter backups       This System: GA2LB148
O DDRACDD     Manage DDRACDD                      This Site: DC1
P ARMLIB      Manage PARMLIB DSN backups          Master-C: GA2LB148
--- Parameter Load functions ---- Primary Site: DC1
| I nput       Select Parameter Input Dataset |
C onfig      Define Configuration Basics          Primary DASD: DC1
D ata        Define Data Storage Objects          Automation: ON
H ost        Define Data Host Objects             Planned script: None
O ptions     Specify GDDR Options                 Unplanned script: None

Current work data set SYSU.GDDR500.PARMS.WORK.M38
EMC Geographically Dispersed Disaster Restart 5.0.0
Copyright © 2007-2016 EMC Corporation.

Select an option and press <Enter>
Pres +----------------------------------------------------------------+
| GDDI172I Member I6L13905 selected as GDDR Parameter Load Input |
```

**Figure 26** Parameter Management Options Menu with parameter load input selection

**To review the current GDDR parameter set**

Select any of the Parameter Load functions to view or change values which were populated from the selected input parameter dataset member.
Edit in-progress serialization

GDDR Parameter Wizard work dataset serialization starts when you issue the PREPARE command on the Select Parameter Input Dataset panel, and ends after the successful activation of the parameter set, when the processing performed by Activate GDDR Parameter Set panel completes.

While editing of the parameter work dataset is in progress, informational message GDDI010I is issued on entry to the Select Parameter Input Dataset panel as shown in Figure 27.

![Figure 27  Select Parameter Input Dataset panel with Edit-in-Progress serialization lock - User 1](image)

The GDDI010I message indicates the name of the work dataset, the TSO ID of the user editing (if no user ID is shown in the message, the current TSO user is the one with the Edit-in-Progress serialization lock), the C-System name of the system on which the parameter edit is in progress, as well as the site (DCx) and the time and date that the edit started.

The serialization can be overridden by another user with proper authority, when the user issues another PREPARE command. Note that the other user must have ‘READ’ authority to facility profile GDDRISPF.SETUP.PARMS.FORCE, as described in “Specify GDDR security” on page 54.
When issuing a PREPARE while an edit is in progress, the new user with 'FORCE' authority is presented with the FORCE option displayed on the popup panel shown in Figure 28.

Figure 28  Select Parameter Input Dataset panel with Edit-in-Progress FORCE authorization - User 2

If the FORCE command is entered, the TSO user id associated with the serialization lock is changed to the second TSO user's ID, and a new edit session effectively begins. The TSO user who originally held the serialization lock will then receive the GDDB010E message with the prompt to indicate they can no longer proceed with their edit session either with message 'Reverting to REVIEW mode', as shown in Figure 29 on page 130 or with message 'Reverting to initial state', as shown in Figure 30 on page 130. Following the display of message GDDB010E, the original TSO user's session displays the Parameter Management Options Menu in Review mode, shown in Figure 18 on page 118.

**Note:** After a successful FORCE, the previous TSO user's session state will either:

- Revert to the initial state (where only the Backup and Input options are shown)
- Revert to the REVIEW mode state

If the second PREPARE command issued by the other user references the *same* parameter work dataset name that was being edited by the first user, then the first user reverts to initial state. If the second PREPARE command issued by the other user references a *different* parameter work dataset name, then the first user reverts to REVIEW mode on the parameter work dataset. Message GDDI010E will appear on the Parameter Wizard panel currently displayed by first user's TSO session.
To release the Edit-In-Progress serialization lock (same user)

Complete the following steps to release the Edit-in-Progress serialization lock by the TSO user who started it, without completing an Activate of the parameter set:

1. Enter the REVIEW line command on the GDDR Select Parameter Input Dataset panel.
2. The Parameter Wizard responds by prompting you for a FORCE command to abandon your changes.
3. Reply FORCE to abandon the EDIT session and start the REVIEW session, which ends the Edit-In-Progress serialization lock.
Option C: Define Configuration Basics

**IMPORTANT**

This series of Define Configuration Basics panels is used once during initial installation of GDDR, and is unlikely to be used again. Most changes performed here will require a FULL parameter activation as well as edits to GDDMPARM parameters. The following changes can be done with a PARTIAL parameter activation:

- Option F: Setting FBA Devices to Y/N
- Option D: Changing dataset names

When you specify option C in the Parameter Management Options Menu shown in Figure 9 on page 110, the Define Configuration Basics panel appears:

![Figure 31 Define Configuration Basics panel](image)

Complete the following options listed in this panel to define your configuration:

- “Option F: Define Configuration Features” on page 132
- “Option C: Define C-Systems” on page 134
- “Option D: Define GDDR Datasets” on page 136
- “Option R: Define Site Roles and Groups” on page 137
IMPORTANT
As you complete each option panel, save your input to ensure that you can return to
the task with no loss of data. If you attempt to exit a panel before a save has been
completed, you are reminded with the following message.

--- SAVE unsaved changes? ---
Command ===>
Warning: You may have unsaved changes.
Enter SAVE to save your work and continue to the next panel.
Enter CANCEL to return to the panel you were on.
Press <F3> key to continue and lose unsaved changes.

IMPORTANT
Note that the “Save unsaved changes?” confirmation pop-up may occur even when you
have not manually entered data on the current panel. The GDDR Parameter Load wizard is
a series of panels with a specific sequence. Panels earlier in the sequence may
necessitate adding new information to panels later in the sequence. This occurs
frequently in the subset of panels under “Define Host Objects”.

If the GDDR-managed systems are changed (added/deleted/replaced), EMC recommends
that you step through the entire sequence of Host Object wizard panels. Where GDDR
expects additional information, the wizard will insert new lines on the panel. These are
usually shown initialized with underscores, and sometimes with template information or
with discovered information. This creates a difference between the panel content and the
work PDS. When an attempt is then made to step to the next wizard panel, the “Save
unsaved changes?” confirmation pop-up will be displayed as a signal that you need to
confirm or complete the information added by the wizard.

Option F: Define Configuration Features
The Define Configuration Features panel displays the Site list options from the
GDDMPARM member you customized during “Install GDDMPARM” on page 66.

---------- GDDR - Define GDDR Configuration features - Row 1 to 2 of 2
Command ===>
Site list: DC1 DC3 FBA Devices: Y  (Y/N)
DLm support: N  (Y/N)
SNAP-VX SOFTLINK Support:  DC1: N   DC3: N                    (Y/N/blank)
Legacy TimeFinder Method:  DC1: C   DC3: C                    (M/C/blank)

Select a configuration from the list below and Press <Enter> to Validate.
<F3> Return to previous Menu  <F6> Define C-Systems
Type SAVE to save, LOAD to restart from last saved values.

Sel   Configuration
---  ----------------------------------------------
   2-site SRDF/A
   2-site SRDF/Star with R22 support
   Cascaded SRDF/Star with Extended Distance Protection
***End of configuration list***

Figure 32  Define GDDR Configuration Features
To proceed, select a Configuration and press **Enter**. The panel updates to display configuration-specific options:

![Figure 33](image)

**Figure 33** Define Configuration Features panel, configuration-specific options

This panel displays the configurations that are valid selections for the Site list.

1. To proceed, enter:
   a. **Y or N** next to FBA Devices to indicate whether Fixed Block Address (FBA) devices will also be managed by GDDR (CKD devices are always managed by GDDR by default).
   b. **Y or N** next to DLm support to specify whether DLm virtual tape drives will be managed by GDDR.
   c. **M or C** for the Legacy TF Method field. Specify **M** to have GDDR use TimeFinder/Mirror with Clone Emulation for the listed site (default). Specify **C** to have GDDR use TimeFinder/Clone for the listed site.
   d. **Y or N** for the SnapVX SOFTLINK Support field to indicate whether or not SnapVX SOFTLINK support is required in GDDR for the listed site. You must specify **Y** if there is any non-GDDR usage of SnapVX with soft linking (for example, zDP) using the GDDR-managed SRDF devices as source devices. You can also specify **Y** if you prefer GDDR to use SnapVX with soft linking. The default option is **N**. You must specify **N** if the listed site has no EMC VMAX V3 system.

The Legacy TF Method and SnapVX SOFTLINK Support fields are site-specific, and affect each other as follows:

- If SOFTLINK support is specified as **Y**, GDDR will use SnapVX with soft linking on VMAX V3 systems and TF/Clone on V1 and V2 systems. The Legacy TF Method field for the affected site is thus ignored.
- If SOFTLINK support is specified as **N** and Legacy TF Method is **M**, GDDR will use TF/Mirror across V1, V2, and V3 systems.
- If SOFTLINK support is specified as **N** and Legacy TF Method is **C**, GDDR will use TF/Clone on V1 and V2 systems and SnapVX with soft linking on VMAX V3 systems, thus ignoring the soft link specification for the affected site.
Using GDDR Online Facilities

e. **S** in the SEL column of one of the rows to select the name of the configuration which matches your site's storage replication environment.

2. Press **Enter** to validate the selected configuration.

3. Type **SAVE** to save your selection.

4. Press **F6** to proceed to the Define C-Systems panel shown in Figure 34 to confirm the C-System management parameters that are specific to your selected configuration.

**Option C: Define C-Systems**

When you specify option C in the Define Configuration Basics panel shown in Figure 31 on page 131, the following panel appears:

```
--------------------------- GDDR - Define C-Systems ---------------------------
Command ==>
Press <F3> to return to the GDDR Define Configuration Basics Menu
<F5> Define Configuration Features    <F6> Define GDDR Datasets
Type SAVE to save, LOAD to restart from last saved values.

<table>
<thead>
<tr>
<th>System</th>
<th>IPL</th>
<th>Site</th>
<th>Name</th>
<th>SMFID</th>
<th>Parameters</th>
<th>CPC (netid.nau)</th>
<th>LPAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC1</td>
<td>SYS1</td>
<td>SYS1</td>
<td>7084</td>
<td>708499M1</td>
<td>IBM390PS.Q3</td>
<td>ZOSESYS1</td>
<td></td>
</tr>
<tr>
<td>DC3</td>
<td>SYS3</td>
<td>SYS3</td>
<td>70B8</td>
<td>709899M1</td>
<td>IBM390PS.C</td>
<td>ZOSESYS3</td>
<td></td>
</tr>
</tbody>
</table>

*-------------------- Auto-Discovery Legend --------------------*
* Green  - Display only. Auto-Discovery had no conflict with input.     *
* Yellow - Display only. Restricted by Auto-Discovery, conflict with input. *
* Red    - User input allowed. Not restricted by Auto-Discovery,          *
*          or impossible to discover at this time.                         *

Figure 34  Define C-Systems panel

1. The System Name, SMFID, IPL parameters, Central Processor Complex (CPC) name, and LPAR fields of the Define C-Systems panel are automatically discovered using CSYSSITE parameter values furnished during installation, as described in “Install GDDMPARM” on page 66.

The color font that displays in these fields indicates the precedence rules that apply to data returned to the Parameter Load wizard from the Auto-Discovery feature when validated against the contents of your parameter work dataset.

- **Green** — Display only. Auto-Discovery had no conflict with input.
- **Red** — User input allowed. Not restricted by Auto-Discovery, or impossible to discover at this time.
- **Yellow** — Display only. Restricted by Auto-Discovery, conflict with input.

**Note:** You can only change values displayed in red font via the panel. Within the Define C-Systems panel, the only values you can change are the IPL parameters.

The values that are displayed may be confirmed or changed using the information provided with each field described below, subject to the Auto-Discovery data precedence rules described above.
Using GDDR Online Facilities

- **Site**
  Indicates the ID of the site location being specified. Site values are populated from the Site List in the Define GDDR Configuration Features panel. Valid values are DC1 or DC3. This field reflects the GDDMPARM CSYSSITE parameter for the listed C-System.

- **System**
  The MVS system name of a GDDR C-System which is specified using the SYSNAME=system-name statement in SYS1.PARMLIB(IEASYS00) or equivalent parameter file. This field reflects the GDDMPARM CSYSSITE parameter for the listed C-System.

- **SMFID**
  The SMFID of the C-System identified by System name. This field cannot be edited, unless Auto-Discovery fails for the listed remote C-System.

- **IPL Parameters**
  Specifies the IPL parameters that GDDR may use to IPL the C-Systems at sites DC1 and DC3 in the following format:

  \[ssss,iiiixxmn\]

  Where
  - \( ssss \) is the Sysres device address
  - \( iiii \) is the IODF device address
  - \( xx \) is the LOADxx member suffix
  - \( m \) is the IMSI Field
  - \( n \) is the IEANUCOn suffix

  You can locate IPL parameters using the D IPLINFO MVS console command, as shown in the following sample output:

  ```
  RESPONSE=SYSB
  IEB254I 21.14.59 IPLINFO DISPLAY 860
  SYSTEM IPLED AT 15.59.20 ON 02/02/2009
  RELEASE z/OS 01.10.00 LICENSE = z/OSe
  USED LOAD02 IN SYS1.IPLPARM ON 0A26
  ARCHLVL = 2 MTLSHARE = N
  IEASYM LIST = B0
  IEASYS LIST = B0 (OP)
  IODF DEVICE 0A26
  IPL DEVICE 0A1D VOLUME RESB14
  ```

- **CPC (Central Processor Complex)**
  Specify the name of the central processor where the C-System z/OS system is defined. This field cannot be edited, as it is fully determined by the GDDMPARM CPC parameter of type NETID for the listed C-System.

- **LPAR (Logical Partition)**
  Specify the name of the LPAR within which this system runs at the specified site. This field cannot be edited, unless Auto-Discovery fails for the listed remote C-System. You can find the CPC and LPAR names using the D M=CPU MVS console command, as shown in the following sample output:

  ```
  RESPONSE=SYSB
  ```
2. Type `Save` on the command line and press `Enter`.

3. Press `F6` to proceed to the Define GDDR Datasets panel shown in Figure 35.

**Option D: Define GDDR Datasets**

When you specify option D in the Define Configuration Basics panel shown in Figure 31 on page 131, the following panel appears:

```
------------------------- GDDR - Define GDDR Datasets ------ Row 1 to 12 of 12
Command --->

Press <F3> to return to the GDDR Define Configuration Basics Menu
<F5> Define C-systems  <F4> Define Site Roles and Groups
Type SAVE to save, LOAD to restart from last saved values.
Line commands: A dd, D elete, R epeat

<table>
<thead>
<tr>
<th>CMD</th>
<th>C-system</th>
<th>DS Type</th>
<th>Seq</th>
<th>Dataset Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>SYS3</td>
<td>BKUPVARS</td>
<td>GDDR.GDDR500.BKUPVARS.CNTL</td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>SYS3</td>
<td>LAPD</td>
<td>GDDR.GDDR500.PARMS.LAPD</td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>SYS3</td>
<td>PROCLIB</td>
<td>1</td>
<td>SYS2.GDDR500.PROCLIB</td>
</tr>
<tr>
<td>_</td>
<td>SYS3</td>
<td>SKELETON</td>
<td>1</td>
<td>GDDR.GDDR500.ISPSLIB</td>
</tr>
</tbody>
</table>
```

--- Bottom of data ---

**Figure 35** Define GDDR Datasets panel

1. **Type Add** in the Option field at the top of the panel to display a new line for entry of values.

2. Insert additional datasets by typing `R` (Repeat) in the command field to the left of an existing entry and overtyping the existing data.

3. **Type D (Delete)** in the command field to the left of an existing entry to remove it.

4. Complete the required information for each dataset type.
• **BKUPVARS**
  Specify the name of the dataset into which the backups produced by EMC GDDR Heartbeat monitor initialization should be saved. This dataset was allocated during the procedure described in “Allocate the parameter management datasets” on page 64.

• **LAPD**
  Specify the name of the last activated parameter dataset. This dataset was allocated during the procedure described in “Allocate the parameter management datasets” on page 64. It cannot be the same dataset as your current work dataset.

• **PROCLIB**
  Specify the name of the library on the C-Systems which contains the GDDRPROC member. This library was selected during the procedure described in “Customize member GDDRPROC” on page 78.

• **SKELETON**
  Specify the name of the GDDR ISPSLIB library that resulted from your SMP/E installation.

• **Seq**
  Specify a sequence number for the dataset. The sequence number enables the Proclib and Skeleton libraries to be assigned sequentially across C-Systems.

5. If you made any changes, type **Save** on the command line and press **Enter**.

6. Press **F6** to proceed to the Define Site Roles and Groups panel shown in Figure 36 on page 137.

**Option R: Define Site Roles and Groups**

When you specify option **R** in the Define Configuration Basics panel shown in Figure 31 on page 131, the Define Site Roles and Groups panel appears.

---
**GDDR - Define Site Roles and Groups**

Option ==>  

Press <F3> to return to the GDDR Define Configuration Basics Menu  
<F5> Define GDDR Datasets  
<F6> Define Data Storage Objects  
Type <SAVE> to save, <LOAD> to restart from last saved values.

Enter the required information. Press <Enter> to Validate.

<table>
<thead>
<tr>
<th>Primary DASD Site:</th>
<th>DC1</th>
<th>Select DC1 or DC3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Site:</td>
<td>DC1</td>
<td></td>
</tr>
<tr>
<td>Consistency Group Name DC1:</td>
<td>Not Applicable</td>
<td></td>
</tr>
<tr>
<td>Consistency Group Name DC2:</td>
<td>Not Applicable</td>
<td></td>
</tr>
<tr>
<td>MSC Group Name DC1:</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>MSC Group Name DC2:</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>MSC Group Name DC3:</td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

-------------------- Bottom of Data --------------------

**Figure 36** Define Site Roles and Groups panel

The Define Site Roles and Groups panel defines your Consistency Group and MSC Group names as GDDR global variables.
**Note:** In SRDF/A environments, Consistency Group names are not used. This is indicated by the 'Not Applicable' note to the right of the Consistency Group Name fields.

1. Complete the following fields in the panel:
   - **Primary DASD Site**
     Specify the site where the source (R1) DASD is located.
   - **Primary Site**
     Specify the site where the production workload is located.
   - **MSC Group Name DC1**
     Specify the name of the managed MSC group when the Primary Site is DC1.
   - **MSC Group Name DC3**
     Specify the name of the managed MSC group when the Primary Site is DC3.

   **Note:** If the RESET option is specified during GDDR parameter activation, GDDR will dynamically determine the correct Primary DASD site based on a discovery process of the managed configuration; this may override the value specified on this panel.

2. If you made any changes, type **Save** on the command line and press **Enter**.
3. Press F6 to proceed to the Define Data Storage Objects panel shown in **Figure 37**.

**Option D: Define Data Storage Objects**

When you specify option **D** in the Parameter Management Options Menu shown in **Figure 9 on page 110**, the Define Data Storage Objects panel appears:

```
---------------------- GDDR - Define Data Storage Objects ---------------------
Option ===>          
S   Define SRDF Device ranges                         This System: GA2LB148
T   Define TimeFinder Device ranges                     This Site: DC1
N   Define GDDR Snapshot Names                           Master-C: GA2LB148
G   Define SDDF Gatekeepers                               Primary DASD: DC1
D   Define DLm Devices                                   Automation: ON
                                Planned script: None
                                Unplanned script: None

Select an option and press <Enter>

<F5> Define Site Roles and Groups               <F6> Define SRDF Device Ranges
Press <F3> to return to the GDDR Parameter Management Options Menu
```

**Figure 37** Define Data Storage Objects panel

Complete the following options listed in this panel to define your configuration:

- “Option S: Define SRDF Device Ranges” on page 139
- “Option T: Define TimeFinder Device Ranges” on page 140
Using GDDR Online Facilities

- “Option N: Define GDDR Snapshot Names” on page 142
- “Option G: Define SDDF Gatekeepers” on page 143
- “Option D: Define DLm Devices” on page 143

**Option S: Define SRDF Device Ranges**

When you specify option S in the Define Data Storage Objects panel, the following panel appears:

```
Figure 38  Define SRDF Device Ranges panel

If the panel is not pre-populated with existing SRDF Device Range entries, run the GDDRACDD utility to discover all SRDF devices to be managed by GDDR, both consistency-protected and 'external' devices. Refer to “Automated Configuration Discovery for DASD (GDDRACDD)” on page 240 for instructions. Alternatively, complete the following steps to manually define the SRDF device configuration to GDDR.

1. **Type Add** in the Command field at the top of the panel to display a new line for entry of SitePair and SRDF Device Range values.
2. Insert additional Site Pairs by typing **R** (Repeat) in the Sel field to the left of an existing SitePair.
3. Complete the fields as follows:
   - **SitePair**
     Specify the GDDR SRDF replication pairs in the format:
     DCm-DCn
     Where \( m \) is the lower site number, and \( n \) is the higher site number. Valid pairings of DCm and DCn are DC1-DC3.
   - **SRDF Device Range**
     Specifies contiguous VMAX device ranges which are common to the specified SitePair for SITE1 and SITE2:
     - **GK**
       SRDF gatekeeper
```
Using GDDR Online Facilities

- RDFgrp
  The RDF group spanning the specified site pair and to which the defined device ranges belong
- Start - End
  A range of contiguous VMAX devices at the corresponding DC site

**Note:**
1. The 4-character gatekeeper masks represent MVS volume addresses used to uniquely identify the site DCm or DCn VMAX control unit where the volume [gatekeeper] resides. The gatekeepers associated with the SRDF device ranges provide information used in the processing of commands: the Site, the VMAX Control Unit at a Site, and the gatekeeper address. All MVS addresses used here MUST be specified on SYMM parameters in GDDMPARM. The addresses must be unique within a site.

2. The VMAX device number ranges at DCm and DCn must contain the same number of devices.

3. If a site is not configured with MVS systems, then dummy MVS addresses must be specified as gatekeeper devices for that site.

- **EXT (External)**
  “Configure GDDR support for external devices” on page 84 provides more information about external devices.
  YES or NO indicates whether these devices are external to the consistency-protected environment.

- **DLM**
  When the Automated Configuration Discovery utility is used with the DLM(Y) argument, GDDRACDD will generate RDF.DEVICES parameters for all devices in all included groups, and DLM.DEVICES parameters for devices in groups included with DLM sub-parameter.
  ‘Yes’ in the DLM column indicates that a given RDF.DEVICES range is also a DLM backend device range. “DLM(Y|N)" on page 243 provides more information about DLM device range selection.

4. If you made any changes, type **Save** on the command line and press **Enter**.

5. Press **F6** to proceed to the Define TimeFinder Device Ranges panel shown in **Figure 40**.

**Option T: Define TimeFinder Device Ranges**
When you specify option **T** as shown in “Define Data Storage Objects panel” on page 138, the Define TimeFinder Device Ranges panel appears.

**Note:** GDDRACDD generates parameters for all BCVs found in selected VMAX systems and generates member E05TFDEV in gatekeeper order. If STDBCV parameters are included in member E05TFDEV, the panel displays as shown in **Figure 39**. If they are not included, the panel displays as shown in **Figure 40**.
Using GDDR Online Facilities

Figure 39  Define TimeFinder Device Ranges panel (with STDBCV parameters)

Figure 40  Define TimeFinder Device Ranges panel (no STDBCV parameters)

1. If the panel is not pre-populated with existing TimeFinder Device Range entries, run the GDDRACDD utility to discover all TimeFinder devices to be managed by GDDR, both consistency protected and ‘external’ devices. Refer to “Automated Configuration Discovery for DASD (GDDRACDD)” on page 240 for detailed instructions.

If only a subset of BCVs are GDDR-managed, use the standard ISPF edit to modify the GDDR Parameter Work PDS, member E05TFDEV. Complete the change with the Parameter Wizard Validation and Activation steps.
Note: Editing of the GDDR Parameter Work PDS member outside of the GDDR Parameter Wizard is best done under guidance of the GDDR Solution Support team.

2. If you made any changes, type Save on the command line and press Enter.

3. Press F6 to proceed to the Define GDDR Snapshot Names panel shown in Figure 42.

Option N: Define GDDR Snapshot Names

When you specify option N in the Define Data Storage Objects panel shown in Figure 37 on page 138, the following panel appears:

--- Define GDDR Snapshot Names ---

Command ===>

Press <F3> to return to the GDDR Define Data Storage objects Menu
<F5> Define TimeFinder Device Ranges  <F6> Define SDDF Gatekeepers
Type <SAVE> to save, <LOAD> to restart from last saved values.

Specify a 1-23 character name for each of the following sets, so that, in combination with the GDDR defined set prefix, a unique identifier is created for each Snapshot

- GOLD-Internal set: (GDDR_INT_)
- GOLD-External set: (GDDR_EXT_)
- TEST-Internal set: (GDDR_INT_)
- TEST-External set: (GDDR_EXT_)

******************************************************************************* Bottom of Data ***********************************************

Figure 41 Define GDDR Snapshot Names

Specify the names to be used by GDDR, one per set.

- Use GDDR_INT_* snapshotname for devices in or paired with consistency-protected RDF groups (ConGroup and/or MSC).
- Use GDDR_EXT_* snapshotname for devices in or paired with ADCOPY-DISK RDF groups.

Note: In SRDF/Star configurations, some devices are consistency-protected in SRDF/S mode between DC1 and DC2, and in ADCOPY-DISK mode between DCn and DC3. Use the GDDR_INT_* prefix for the snapshotname for these devices.
Option G: Define SDDF Gatekeepers

When you specify option G in the Define Data Storage Objects panel shown in Figure 37 on page 138, the following panel appears:

```
------------------------ GDDR - Define SDDF Clean Utility GateKeepers Row 1 to 3 of 3
Command ===>          
Press <F3> to return to the GDDR Define Data Storage Objects Menu
<F5> Define TimeFinder Device Ranges  <F6> Define Host Objects
Type <SAVE> to save, <LOAD> to restart from last saved values.
Line commands: A dd, D delete, R repeat
CMD Site   GK
--- ---- ----

******************************************************************************* Bottom of data *******************************************************************************
```

**Figure 42** Define SDDF Clean Utility Gatekeepers panel

The SDDF Clean Utility GateKeeper parameters are not used in the SRDF/A configuration. Press F6 to proceed to the Define Host Objects panel.

Option D: Define DLm Devices

When you specify option D in the Define Data Storage Objects panel shown in Figure 37 on page 138, the following panel appears:

```
-------------------------- GDDR - Define DLm Devices ---------------- Row 1 of 4
Command ===>          
Press <F3> to return to the GDDR Define Data Storage objects Menu
<F5> Define SDDF Gatekeepers  <F6> Define Host Objects
Type <SAVE> to save, <LOAD> to restart from last saved values.
Line commands: A dd, D delete, R repeat, S serialNos

Vary online ===> AUTOSWITCH (PRIMARY or BOTH or NONE) Autoswitched drives only
Vary offline ===> BOTH (PRIMARY or BOTH or NONE) Any online drives
Stateless? ===> YES (Yes or No)
Number of sites ===> 2 (2, 3 or 4 DLm sites)
Issue DEVSTATUS QUIESCE ===> Y (Yes or No)

Sel Site     IP address (IPv4 or IPv6)       Port    SerialNos Slave?
--- ----  ------------------------------   -----  -------- -----
 DC1 nnn.nnn.nnn.nnn nnnm 12345678+  Y
 DC1 nnn.nnn.nnn.nnn nnnm 23456789+  N
 DC2 nnn.nnn.nnn.nnn nnnm ALL   N

******************************************************************************* Bottom of data *******************************************************************************
```

**Figure 43** Define DLm Devices panel

Each line represents a single DLm ACP (Access Control Point). Each ACP can control multiple VTECs (virtual tape emulation controllers).

Use A, D, R, or S in the Sel column to add, delete, repeat, or select ACP entries.

When selecting an ACP using the S line command, GDDR attempts to contact the DLm ACP station using TCP/IP at the IP address and IP port shown, to query the ACP for the serial numbers of all FICON adapters on all attached VTEs and present the ACP Details panel.
shown in Figure 44.

### Figure 44  Define DLm Devices - ACP Details panel

Note: If there are no entries, type **ADD** on the command line to add the first entry.

DLm FICON adapter serial number auto-discovery usually takes a few seconds to a minute, however, if the ACP is not responding or incorrect TCP/IP information was given (IP address and/or port number) this could take several minutes before a (non) response is detected. If DLm FICON adapter serial number auto-discovery is successful, this list displays on the ACP Details panel shown in Figure 44.

If the auto-discovery fails, or ACP data not reflected by the auto-discovery is desired, the **S** line command also presents the ACP Details panel shown in Figure 44. You can manually maintain the ACP definitions by providing the following information:

- **Site**
  Indicates at which site (DC1 or DC3) this DLm storage controller is located.

- **IP address**
  The IP address of the ACP for this DLm, used to send commands from GDDR to the DLm. The format of the IP address can be either IPv4 or IPv6:
  - `xxx.xxx.xxx.xxx` for IPv4
  where `xxx` are numbers between 0 and 255 inclusive.

- **Port**
  The IP port (at the indicated IP address), to send commands to the DLm.

- **SerialNos**
  The FICON adapter serial numbers associated with the DLm storage controller being described by this entry. If you specify a list of serial numbers here, GDDR scripts will only affect DLm tape drives that are attached to the FICON adapters listed here.

  If you specify **ALL**, then GDDR scripts will affect all DLm tape drives.

  Multiple serial numbers for an ACP are indicated by a `+` suffix. Enter the **S** line command to display all serial numbers or revise the list of serial numbers associated with an ACP using the ACP Details panel shown in Figure 44 on page 144.
To add/edit/delete serial numbers, use the **S** line command, which will cause a popup panel to be displayed where you can update the list.

**Note:** If you use **ALL** in one ACP entry, all the other ACP entries (if any) for the same site MUST also specify **ALL**.

- **Slave**
  Y or N. The default is N. Use this field only if you have MDLs (not DLms), to specify which MDLs are slaves (Y) and which are masters (N). If you have DLms only, always use N (the default), as all DLms should be treated as masters by GDDR.

The following primary commands can be entered on the command line:

- **SAVE** - saves the information listed.
- **CANCEL** - cancels any updates made to this panel (including auto-discovered serial numbers) and returns to the ‘Define DLM Devices’ panel
- **SORT** - sorts the serial numbers in ascending sequence (and removes empty embedded slots)
- **REVERT** - reverts the list of serial numbers back to what they were before auto-discovery or any other manual changes you made on this panel.
- **LOAD** - reloads previously saved sessions.

### GDDR-managed tape drive controls

GDDR controls the state of managed tape drives using the following options.

- **Vary online/offline**
  GDDR controls the state of managed tape drives using the following options.
  - **Vary online**
    PRIMARY causes GDDR scripts to vary online only the managed tape drives set to AUTOSWITCH at the primary site. You must manually vary online GDDR-managed tape drives at the secondary site, if so desired.
    BOTH causes GDDR scripts to vary online managed tape drives set to AUTOSWITCH at the primary and the secondary site.
    AUTOSWITCH causes GDDR scripts to vary online only those managed tape devices that have been set to AUTOSWITCH in z/OS.
  - **Vary offline**
    PRIMARY causes GDDR scripts to vary offline only the managed tape drives at the primary site. You must manually vary offline GDDR-managed tape drives at the secondary site, if so desired.
    BOTH causes GDDR scripts to vary offline managed tape drives at the primary and the secondary site.
    NONE causes GDDR scripts to bypass vary online AUTOSWITCH and vary offline the managed tape drives. These operations are expected to be performed outside of GDDR control.
  - **Stateless?**
The Stateless field specifies whether GDDR scripts will use DLm stateless disaster recovery (Y) or (N). This setting must match the setting used by your DLm hardware installation.

- **Number of Sites**

  Number of Sites specifies the number of GDDR sites that support DLm. The minimum is 2; the maximum is the number of sites configured in the GDDR-plex.

- **DEVSTATUS QUIESCE**

  The DEVSTATUS QUIESCE field specifies whether GDDR scripts will issue a DEVSTATUS QUIESCE to the DLm ACP (Y) or (N).

### Option H: Define Host Objects

When you specify option H in the Parameter Management Options Menu shown in Figure 9 on page 110, the Define Host Objects panel appears:

```
-------------- GDDR - Define Host Objects --------------
Option ===>
 S  Define Managed Systems                        This System: SYS1
 SP Define System IPL Priorities                    This Site: DC1
 L  Define Managed LPARs                           Master-C: SYS1
 R  Define System Recovery Attributes               Primary Site: DC2
 P  Define Managed CPCs                             Primary DASD: DC2
 I  Define IPL Parameters                           Automation: ON
 A  Define HMC Load Activation Profiles             Planned script: None
 H  Define Managed HMCs                             Unplanned script: None
 C  Define HMC Community Names                      Planned script: None
 D  Define Managed Coupling Datasets                Unplanned script: None
 CF Define Managed CF Structures                    Planned script: None
 W  Define External Workloads                       Unplanned script: None
 E  Define EMC MF Enabler STCs

Select an option and press <Enter>

<F5> Define SDDF Clean GateKeepers     <F6> Define Managed Systems
Press <F3> to return to the GDDR Parameter Management Options Menu
```

**Figure 45** Define Host Objects panel

Complete the following options listed in this panel to define your host objects configuration:

- “Option S: Define Managed Systems” on page 147
- “Option SP: Define System IPL Priorities” on page 149
- “Option L: Define Managed LPARs” on page 151
- “Option R: Define System Recovery Attributes” on page 153
- “Option P: Define Managed CPC CBU Options” on page 155
- “Option I: Define IPL Parameters” on page 156
- “Option A: Define HMC Load Activation Profiles” on page 159
- “Option H: Define Managed HMCs” on page 159
- “Option C: Define HMC Community Names” on page 160
Option S: Define Managed Systems

When you specify option S in the Define Host Objects panel, the Define Managed Systems panel appears:

```
----------------------- GDDR - Define Managed Systems ------- Row 1 to 14 of 14
Command ===>
Press <F3> to return to the GDDR Define Host Objects Menu
<F5> Define Host Objects <F6> Define Managed LPARs
Type SAVE to save, LOAD to restart from last saved values.
Line commands: A dd, D elete, R epeat

<table>
<thead>
<tr>
<th>CMD</th>
<th>Site</th>
<th>System</th>
<th>Sysplex</th>
<th>Manage</th>
<th>HMC</th>
<th>Only</th>
<th>LPAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>___</td>
<td>___</td>
<td>------</td>
<td>--------</td>
<td>------</td>
<td>___</td>
<td>___</td>
<td>____</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>PRD1</td>
<td>PROD,PLEX</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST1</td>
<td>__________</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST2</td>
<td>__________</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC2</td>
<td>CNT1</td>
<td>PROD,PLEX</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST5</td>
<td>__________</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST6</td>
<td>__________</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC2</td>
<td>TST3</td>
<td>N/A</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>PRD3</td>
<td>PROD,PLEX</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>TST3</td>
<td>__________</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>TST7</td>
<td>__________</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC4</td>
<td>CNT2</td>
<td>PROD,PLEX</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>TST8</td>
<td>__________</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>TST9</td>
<td>__________</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC4</td>
<td>TSTA</td>
<td>N/A</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
</tr>
</tbody>
</table>

Figure 46 Define Managed Systems panel

Note: Changes to this panel and Figure 48, “Define Managed LPARs panel” are likely to require changes to COMM or CPC parameters in GDDMPARM. If necessary, those changes must be completed before using the GDDR Parameter Wizard.

Use the managed system parameters to define mainframe systems to GDDR and indicate the extent to which they will be managed by GDDR, either using the online interface or during automation sequences. Do not include C-systems here, as those have been defined to GDDR earlier as described in “Option C: Define C-Systems” on page 134. Managed system parameters are defined for every GDDR-managed system to enable HMC functions and workload management.

The first time you use this panel (when you have never issued a SAVE command), the Auto-discovery feature will populate this panel with GDDR-managed system names along with related syplexes, if available. The auto-discovered list of systems is determined by the presence of GDDMPARM COMM parameters for non C-Systems. After you have issued a SAVE command, the previously saved information will be displayed on the panel whenever you return to it.
If changes to GDDMPARM parameters are performed as a result of the various maintenance procedures described in Chapter 9, “Performing Maintenance Procedures,” those changes will be displayed upon re-entry to this panel.

1. If the panel is not pre-populated with existing entries, type Add in the Command field at the top of the panel to display a new line for field entry.

2. Insert additional systems by entering R (Repeat) in the command field to the left of an existing entry and overtype the existing data.

3. Confirm or change the values that are displayed by using the information provided with each field described below, subject to the precedence rules that apply to data returned to the Parameter Load wizard by the Auto-Discovery feature:

   - **Site**
     Specify the ID of the site location being specified. It can have the value DC1 or DC3.

   - **System**
     Specify the z/OS system name of a GDDR-managed system which resides at the specified site. You can find the system name on the SYSNAME=system-name statement in SYS1.PARMLIB(IEASYS00) or the equivalent parameter file.

   - **Sysplex (optional)**
     When defining managed systems to GDDR, specify a sysplex name for those systems where couple facility structure and couple facility dataset management actions are to be performed by GDDR.

     Couple Facility dataset and structure management is controlled for planned and unplanned scripts by the Rebuild CF Dataset, Rebuild CF Structure, and the CF Rebuild Timeout fields described in “Option S: Script Sysplex Options” on page 171, by selections made under “Option D: Define Managed Couple Datasets” on page 161, and under “Option CF: Define Managed CF Structures” on page 164.

     The format of this field is: \(<name>,<type>\), where:

     - \(name\) is either the name of the Sysplex to which the defined system belongs or "NONE"
     - If a name is specified then \(type\) is either "PLEX", "MONO", or "NONE"

     This entry is used to simplify data entry on the Sysplex Object management related panels of the GDDR Parameter Load wizard.

     **Note:** For this release, the Sysplex Object Management features are enabled only for systems which have been defined here with \(<name","PLEX"\).

   - **Manage Workload**
     Indicates whether GDDR will trigger start/stop of the workload for the listed system. Type YES to cause GDDR to trigger the stop and start of applications when EMC GDDR takes planned or unplanned actions that impact the managed systems.
• HMC Only

Indicates whether the specified system is external to the population of systems managed by GDDR scripts. HMC Only ‘YES’ Systems may be managed using the GDDR HMC LPAR Actions panel. Type YES to bypass GDDR parameter validation for the specified system. Type NO to specify that the system is to be managed by GDDR.

• CF LPAR

The CF LPAR setting is used to designate a dummy system name that will be associated with a Coupling Facility LPAR on the Define Managed LPARs panel. Stand-alone coupling facilities are not managed by GDDR.

Note: CF LPARs are not required at any site, and if present at any site, are not necessarily present at the other sites, as stand-alone Coupling Facilities may be defined at the other sites.

4. If you made any changes, type Save on the command line and press Enter.

5. Press F6 to proceed to the Define Managed LPARs panel shown in Figure 48.

Option SP: Define System IPL Priorities

When you specify option SP in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

```
                        --------------------- GDDR - Define System IPL Priorities ----- Row 1 to 14 of 14
Command ==> 
Press <F3> to return to the Define Host Objects Menu
<F5> Define Managed Systems <F6> Define Managed LPARs Type SAVE to save, LOAD to restart from last saved values.
Enter the required information. Press <Enter> to Validate.

IPL Site  System  Priority  IPL Group Description
---      --------    --------    -----------------------------
DC1      PRD1      00        Prod Group 1 - Top Priority
DC1      TST1      90        Test Group 1
DC1      TST2      90        Test Group 1
DC1      TST6      98        Test Group 3
DC1      TST5      98        Test Group 3
DC2      CNT1      02        Prod Group 2
DC2      TST3      95        Test Group 2
DC3      PRD1      00        Prod Group 1 - Top Priority
DC3      TST3      98        Test Group 3
DC3      TST7      98        Test Group 3
DC3      TST8      99        Test Group 4 Lowest Priority
DC3      TST9      99        Test Group 4 Lowest Priority
DC4      CNT2      00        Prod Group 1 - Top Priority
DC4      TSTA      90        Test Group 1

****************************************************************************** Bottom of data ***************
```

**Figure 47** Define System IPL Priorities panel

This panel lists systems eligible for IPL and their home site. This panel is populated from the systems on the Define Managed Systems panel, excluding:

- Excluded systems (MANAGE WORKLOAD column = NO)
- HMC ONLY systems (HMC ONLY column = YES)
- CF LPAR systems (CF LPAR column = YES).
The managed systems shown on this panel are maintained in the Define Managed Systems panel. Each system can be assigned a priority from 00 to 99, with 00 being the highest IPL priority (systems that will be IPLed or ACTIVATEd first), and 99 being the lowest IPL priority (systems that will be IPLed or ACTIVATEd last).

During scripts that perform HMC actions, the list of systems will be divided into groups by priority, and groups are processed in the following manner:

1. Inform the user with a list of target systems.
2. Confirm all actions with a single WTOR.
   Depending on the Confirm GDDR Script HMC actions by WTOR call override setting, other WTORs may be issued.
3. ACTIVATE all systems, then LOAD CLEAR them (or RESET CLEAR all systems, then DEACTIVATE them).

Field descriptions
The following fields are provided on the Define System IPL Priorities panel:

- **Site**
  The home site-id for this system.

- **System**
  The name of the system as defined on the Define Managed Systems panel.

- **IPL Priority**
  A number between 00 (highest) and 99 (lowest) indicating the priority of system during HMC actions.
  - For ACTIVATEs (IPLs), CF LPARs come first, followed by managed systems in order from 00 to 99.
  - For DEACTIVATEs (shutdowns), systems are shutdown in order from 99 to 00, CF LPARs are last.
  - The order of IPL/shutdown for systems with the SAME priority is undetermined.
  You can (re)assign a priority simply by typing a new number into this column, and clearing the Description column.

- **IPL Group Description**
  A 1-30 character description for a given priority group (all systems with the same priority number).
  To change an IPL Group Description, make sure you have entered the correct number in the IPL Priority column, then simply type a new description and press the Enter key.

  Clearing the Description column and pressing the Enter key causes the Description field to be filled in with any existing description for the given IPL Priority value. If you do not clear the Description column when assigning a new priority, you will also be changing the IPL Group Description for the new priority.

Sorting the display
You can sort the display by typing SORT in the Command line at the top of the panel:
SORT [ASC|DESC]colname ...

where colname is the name of the column to sort. Valid column names are:

- SITE
- SYSTEM
- PRIORITY
- DESC

The optional prefixes ASC (default) or DESC cause sorting in ascending or descending order, respectively. For example, the following commands sort the display by ascending site and descending priority and ascending system name:

SORT ASC SITE DESC PRIORITY ASC SYSTEM

or

SORT SITE DESC PRIORITY SYSTEM

Option L: Define Managed LPARs

When you specify option L in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

--- GDDR - Define Managed LPARs ---
--- Row 1 to 7 of 7 ---

Command ===>

Press <F3> to return to the GDDR Define Host Objects Menu
<F5> Define Managed Systems <F6> Define System Recovery Attributes
Type SAVE to save, LOAD to restart from last saved values.

Line commands: A dd, D elete, R epeat

<table>
<thead>
<tr>
<th>CMD</th>
<th>Site</th>
<th>System</th>
<th>CPC Name</th>
<th>LPAR</th>
<th>Bypass</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>DC1</td>
<td>PRD1</td>
<td>IBM390PS.PRD1</td>
<td>LPARPRD1</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST1</td>
<td>IBM390PS.TST1</td>
<td>LPARTST1</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST2</td>
<td>IBM390PS.TST2</td>
<td>LPARTST2</td>
<td>YES</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST5</td>
<td>IBM390PS.TST5</td>
<td>LPARTST5</td>
<td>YES</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>TST6</td>
<td>IBM390PS.TST6</td>
<td>LPARTST6</td>
<td>YES</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>CNT1</td>
<td>IBM390PS.CP3</td>
<td>LPARCNT3</td>
<td>NO</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>PRD1</td>
<td>IBM390PS.CP3</td>
<td>LPARPRD3</td>
<td>NO</td>
</tr>
</tbody>
</table>

Figure 48 Define Managed LPARs panel

Note: Changes to this panel and Figure 46, “Define Managed Systems panel” are likely to require changes to COMM or CPC parameters in GDDMPARM. If necessary, those changes must be completed before using the GDDR Parameter Wizard.

GDDR HMC command requests are directed to HMC using the LPAR name and the processor name where the LPAR is defined. To enable HMC functions (IPL, CBU, and so forth), these variables must be defined for every GDDR-managed system and GDDR C-System. Auto-Discovery populates this panel with a list of LPARs, which has at least one LPAR for each system defined on the Define Managed systems panel, for the home site of that system.

1. If the panel is not pre-populated with existing entries, type Add in the Command field at the top of the panel to display a new line for field entry.
2. Insert additional systems by entering R (Repeat) in the command field to the left of an existing entry and overtype the existing data.

3. Confirm or change the values that are displayed using the information provided with each field described below, subject to the precedence rules that apply to data returned to the Parameter Load wizard by the Auto-Discovery feature.

- **Site**
  
  Indicates the site location of the LPAR being defined. Site values are initially populated from the Define Managed Systems panel. Valid values are DC1 or DC3. For each system, you can add one LPAR per site in the configuration, except for systems protected with LPAR_RECOVERY as described under “Option R: Define System Recovery Attributes” on page 153. LPAR_RECOVERY protected systems can have one LPAR defined at the home site, plus one at DC3 in 3-site configurations. They cannot have an alternate LPAR defined, neither at the home site, nor at the site of the Recovery LPAR location.

- **System**
  
  This field is populated from the System name supplied in the Define Managed Systems panel, one row per system, for the site which is the home site for that system.

  **Note:** If the field has not been pre-filled, this is the z/OS system name of a GDDR-managed system or a GDDR C-System which resides at the specified site. The name may be found on the SYSNAME=system-name statement in SYS1.PARMLIB(EASYS00) or the equivalent parameter file.

  Add rows for each additional site housing a CPC where a populated system can be IPL’d.

- **Central Processor Complex (CPC)**
  
  Specify the name of the central processor where the managed LPAR is defined. Auto-Discovery cannot determine the NETID portion of the CPC name, therefore you must provide this information on this panel.

- **Logical Partition (LPAR)**
  
  Specify the name of the LPAR within which this system runs at the specified site.

  You can find the CPC and LPAR names using the D M=CPU MVS console command, as shown in the following sample output:

  ![Sample output](image-url)

  **RESPONSE=SYSB**
  IEIE174I 20.59.34 DISPLAY M 781
  PROCESSOR STATUS
  ID CPU SERIAL
  0 02F94E2096
  1 02F94E2096
  2 02F94E2096

  CPC ND = 002096.S07.IBM.83.00000008F94E
  CPC SI = 2096.N03.IBM.83.000000000008F94E
  CPC ID = 00
  CPC NAME = C
  LP NAME = ZOSESYSB LP ID = 2
  CSS ID = 0
MIF ID = 2

ONLINE - OFFLINE . DOES NOT EXIST W WLM-MANAGED
N NOT AVAILABLE

CPC ND CENTRAL PROCESSING COMPLEX NODE DESCRIPTOR
CPC SI SYSTEM INFORMATION FROM STSI INSTRUCTION
CPC ID CENTRAL PROCESSING COMPLEX IDENTIFIER
CPC NAME CENTRAL PROCESSING COMPLEX NAME
LP NAME LOGICAL PARTITION NAME

- **Bypass HMC**

Identifies the LPARs where hardware management console actions should be bypassed. When YES, this bypass affects all HMC actions for the specified LPARs. This includes Load, Reset, Activate, Deactivate, manual and automatic CBU Activate and Undo, Couple DS Realignment, and CF Structure Rebuild. Set this field to NO for GDDR-managed LPARs.

If you set this field to * for any LPAR at site DCm, then on the next Save command, HMC Bypass will be in effect for ALL LPARs at site DCm. This is visible once you issue a LOAD after that Save.

4. If you made any changes, type **Save** on the command line and press **Enter**.
5. Press **F6** to proceed to the Define System Recovery Attributes panel shown in **Figure 49**.

**Option R: Define System Recovery Attributes**

When you specify option **R** in the Define Host Objects panel shown in **Figure 45 on page 146**, the Define System Recovery Attributes panel appears:

---

**Figure 49** Define System Recovery Attributes panel

1. If the panel is not pre-populated with existing entries, type **Add** in the Command field at the top of the panel to display a new line for field entry.
2. Insert additional sites by entering **R** (Repeat) in the command field to the left of an existing entry and overtype the existing data.
3. Complete the Define System Recovery Attributes panel fields as follows:
• System
  The System name is populated from the System name supplied in the Define Managed Systems panel, one row per system.
• Type
  C indicates C-System, P indicates production system.
• Home Site
  The Site ID of the site where the system normally runs.
• Contingency System
  A system at a remote site on which applications can run if a primary managed system fails. A contingency system may be specified for each GDDR-managed system running at DC1 or DC3. A contingency system for a system located at site DC3 must be located at DC1.
  The GDDR-managed system and contingency system pairing is exclusive, meaning that the home LPAR location of a contingency system cannot also be a target LPAR for CPC/LPAR Recovery purposes.
  The field may be left blank, as contingency systems are not required.
The following Recovery Site, CPC and LPAR values must either all be specified or must all be omitted. They indicate the Site, CPC, and LPAR where the failing system should be IPL’d if it fails in its normal location. If another system is running in the Recovery LPAR, it will be reset-cleared and the failing system will be IPL’d in its place. LPARs which are the home of a system with LPAR recovery defined may not be the target of recovery for a different system.
• Recovery Site
  The site at which the recovery CPC is located. If this is different from the Home site, the configuration must include AutoSwap.
• Central Processor Complex (CPC)
  The name of the central processor where the LPAR is defined.
• Logical Partition (LPAR)
  The name of the LPAR that this system runs in at the specified site. The CPC and LPAR names can be found using the D M=CPU MVS console command, as shown in the sample output on page 152.

4. If you made any changes, type **Save** on the command line and press **Enter**.
5. Press **F6** to proceed to the Define Managed CPCs panel shown in **Figure 50**.
Option P: Define Managed CPC CBU Options

When you specify option P in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

![Figure 50 Define Managed CPCs panel](image)

Auto-Discovery prepopulates this panel with a list of unique CPC names resulting from:

- CPCs for C-System LPARs
- CPCs for Managed LPARs
- CPCs for LPAR_RECOVERY LPARs

The AUTOCBU Options allow script control of capacity backup activation by cpcname and site. The presence of this parameter invokes script automation to activate licensed processing capacity on specified central processing complexes (CPCs) at the recovery site specified by site in preparation for restart of workload.

Operator control of capacity backup activation is performed using the Perform GDDR Actions menu. The option to cancel capacity backup following an activation is only available from the Perform HCM CBU Actions panel, using the Undo CBU option. “Option CBU: Perform HMC CBU actions” on page 224 provides more information.

1. If the panel is not pre-populated with existing entries, type Add in the Command field at the top of the panel to display a new line for field entry.

2. Insert additional CPCs by entering R (Repeat) in the command field to the left of an existing entry and overtype the existing data.

3. Confirm or change the values that are displayed using the information provided with each field described below, subject to the precedence rules that apply to data returned to the Parameter Load wizard by the Auto-Discovery feature.

- Site
  Indicates the ID of the site location being specified. Valid values are DC1 or DC3.

- Central Processor Complex (CPC)
  Indicates the name of the central processor where the managed LPAR is defined. There are no limits on the number of CPC name entries per site.
• **Capacity Backup (CBU)**
  
  Defines whether Capacity Backup Upgrade is enabled from the GDDR ISPF interface for the named CPC. Specify **YES** or **NO**. If YES is specified, CBU actions will be allowed using the GDDR ISPF interface.

• **AUTO CBU** options for Planned and Unplanned scripts
  
  – Planned: Valid values are **REAL**, **TEST**, or **NONE**.
    
    When **REAL**, capacity backup licenses will be activated as a part of planned script processing.
    
    When **TEST**, capacity backup capacity will be simulated as part of planned script processing.
    
    When **NONE**, no capacity backup automation actions are performed. This is the default value.
  
  – Unplanned: Valid values are **REAL**, **TEST**, or **NONE**.
    
    When **REAL**, capacity backup licenses will be activated as a part of unplanned script processing.
    
    When **TEST**, capacity backup capacity will be simulated as part of unplanned script processing.
    
    When **NONE**, no capacity backup automation actions are performed. This is the default value.

4. If you made any changes, type **Save** on the command line and press **Enter**.

5. Press **F6** to proceed to the Define IPL Parameters panel shown in **Figure 51**.

**Option I: Define IPL Parameters**

When you specify option I in the Define Host Objects panel shown in **Figure 45 on page 146**, the following panel appears:

```
--- GDDR - Define IPL Parameters ---
Command ===> Press <F3> to return to the GDDR Define Host Objects Menu
          <F5> Define Managed CPCs     <F6> Define HMC Load Activation Profiles
          Type SAVE to save, LOAD to restart from last saved values.

Line commands: A dd, D erase, R epeat

CMD  DASD Site System TYP STD IPL Parameters GOLD BCV IPL Parameters TEST BCV IPL Parameters HMC Only
---  ---   --------  ---  -------------  -------------  -------------   ---
   DC1 PHASER RES    7608,7609BTM1 7108,7109BTM1 7508,7509BTM1    NO
       ALT  7408,7409BTM1 7308,7309BTM1 7708,7709BTM1
       DRT  7208,7209BTM1

   DC2 PHASER RES    8608,8609BTM1 8108,8109BTM1 8508,8509BTM1    NO
       ALT  8408,8409BTM1 8308,8309BTM1 8708,8709BTM1
       DRT  8208,8209BTM1

   DC3 PHASER RES    5608,5609BTM1 5108,5109BTM1 5508,5509BTM1    NO
       ALT  5408,5409BTM1 5308,5309BTM1 5708,5709BTM1
       DRT  5208,5209BTM1

```

**Figure 51** Define IPL Parameters panel
The Define IPL parameters panel specifies the IPL parameters that GDDR may use to IPL an GDDR-managed system at the primary DASD site. All IPL parameters are optional.

The following statements are recommendations for production systems where GDDR is expected to perform HMC LPAR LOAD actions for the affected systems.

- Add an STD IPL parameter for each site where the listed system has an LPAR defined.
- Add a GOLD BCV IPL and/or a TEST IPL BCV parameter for systems for which GDDR is expected to perform HMC LPAR LOAD actions during GDDR Test IPL from BCV scripts.
  - For the IPL parameter-types listed previously, add ALT IPL parameters if you want GDDR to perform HMC LPAR LOAD actions using an alternate SYSRES volume.
  - Add DRT IPL parameters for systems for which you want GDDR to perform HMC LPAR LOAD actions during DR-test scripts (GDD2P03A GDDRPA27 GDD2U09A GDD2U13AGDD2U10A GDD2U12A GDDRPA05 GDDRPA06 GDDRPA07) and if you want to have a SYSRES volume for DR-test circumstances which is different from the normal or alternate SYSRES volumes.
  - Only RES STD IPL and ALT STD IPL parameters are allowed for HMC-only systems
  - This panel also allows you to specify ALT STD IPL parameters for C-systems. The only use case in GDDR for these is during system recovery for a C-system.

To use the panel:

1. If the panel is not prepopulated with existing entries, type Add in the Command field at the top of the panel to display a new line for field entry.
2. Insert additional sites by entering R (Repeat) in the command field to the left of an existing entry and overtype the existing data.
3. Confirm or change the values that are displayed using the information provided with each field described below, subject to the precedence rules that apply to data returned to the Parameter Load wizard by the Auto-Discovery feature.

- **DASD site**
  Indicates the site location of the IPL parameters being defined. Valid values are DC1 or DC3. This is the site location of the DASD devices reflected in the LOAD address portion of the IPL parameters. During GDDR scripts, GDDR will select STD or BCV IPL parameters based on the type of script and on which site is current primary DASD site, or DC3 when running at DC3 in Star configurations.

- **System**
  This field is populated from the System name supplied in the Define Managed Systems panel, one row per system, for the site which is the home site for that system.

  **Note:** If the field has not been pre-filled, this is the z/OS system name of a GDDR-managed system or a GDDR C-System which resides at the specified site. The name may be found on the SYSNAME= system-name statement in SYS1.PARMLIB(IEASYS00) or the equivalent parameter file.

  Add rows for each additional site housing a CPC where a populated system can be IPL’d.
Using GDDR Online Facilities

- **TYP**
  
  Valid values are:
  
  - ALT: Alternate STD IPL parameter
  - DRT: DR Test RES IPL parameter
  - RES: RES STD IPL parameter

  **Note**: GDDR IPL capabilities using BCVs now include support for alternate SYSRES volumes on BCV devices.

- STD IPL parameters
- GOLD BCV IPL parameters
- TEST BCV IPL parameters
  
  In the format: \texttt{ssss,iiiixxmn}
  
  Where
  
  - \texttt{ssss} is the device address
  - \texttt{iiii} is the IODF device address
  - \texttt{xx} is the LOADxx member suffix
  - \texttt{m} is the IMSI Field
  - \texttt{n} is the IEANUC0n suffix
- HMC Only
  
  Indicates if the specified system is external to the population of systems managed by GDDR in scripts. HMC Only ‘YES’ Systems may be managed using the GDDR HMC LPAR Actions panel. GDDR parameter validation is bypassed for systems defined \texttt{YES} in the HMC Only field. This field is not editable.

You can locate IPL parameters using the D IPLINFO MVS console command, as shown in the following sample output:

```
RESPONSE=SYSB
IHE254I 21.14.59 IPLINFO DISPLAY 860
SYSTEM IPLIED AT 15.59.20 ON 02/02/2009
RELEASE z/OS 01.10.00 LICENSE = z/OSe
USED LOAD02 IN SYS1.IPLPARM ON 0A26
ARCHLVL = 2 MTLSHARE = N
IEASYM LIST = B0
IEASYS LIST = B0 (OP)
IODF DEVICE 0A26
IPL DEVICE 0A1D VOLUME RESB14
```

4. If you made any changes, type **Save** on the command line and press **Enter**.

5. Press **F6** to proceed to the Define HMC Load Activation Profiles panel shown in **Figure 53**.
Option A: Define HMC Load Activation Profiles

When you specify option A in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

```
------------- GDDR - Define HMC Load Activation Profiles --------- Row 1 to 7 of 7
Command -->

Press <F3> to return to the GDDR Define Host Objects Menu
<F5> Define IPL Parameters  <F6> Define Managed HMCs
Type <SAVE> to save, <LOAD> to restart from last saved values.

Enter the required information. Press <Enter> to Validate.

<table>
<thead>
<tr>
<th>Site</th>
<th>System</th>
<th>C/P</th>
<th>Load type</th>
<th>HMC Load Profile</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC1</td>
<td>PRD1</td>
<td>P</td>
<td>STANDARD</td>
<td>PRD1STDPROF1</td>
<td>70FD,70FDQ3M1</td>
</tr>
<tr>
<td>DC1</td>
<td>TST1</td>
<td>P</td>
<td>STANDARD</td>
<td></td>
<td>70FD,70FDQ3M1</td>
</tr>
<tr>
<td>DC1</td>
<td>SYS1</td>
<td>C</td>
<td>STANDARD</td>
<td>SYS1STDPROF1</td>
<td>72FD,72FDQ3M1</td>
</tr>
<tr>
<td>DC1</td>
<td>TST2</td>
<td>P</td>
<td>STANDARD</td>
<td></td>
<td>71FD,71FDQ3M1</td>
</tr>
<tr>
<td>DC3</td>
<td>PRD3</td>
<td>P</td>
<td>STANDARD</td>
<td>PRD3STDPROF1</td>
<td>C1FD,C1FDQ3M1</td>
</tr>
<tr>
<td>DC3</td>
<td>TST3</td>
<td>P</td>
<td>STANDARD</td>
<td></td>
<td>C1FD,C1FDQ3M1</td>
</tr>
<tr>
<td>DC3</td>
<td>SYS3</td>
<td>C</td>
<td>STANDARD</td>
<td>SYS3STDPROF3</td>
<td>C1FD,C1FDQ3M1</td>
</tr>
</tbody>
</table>
```

Figure 52 Define HMC Load Activation Profiles

The Define HMC Load Activation Profiles panel is not applicable in sites where cross-site host-DASD channels are not available.

Press F6 to proceed to the Define Managed HMCs panel shown in Figure 53.

Option H: Define Managed HMCs

When you specify option H in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

```
------------- GDDR - Define Managed HMCs -------------- Row 1 of 2
Command -->

Press <F3> to return to the GDDR Define Host Objects Menu
<F5> Define IPL Parameters  <F6> Define HMC Community Names
Type <SAVE> to save, <LOAD> to restart from last saved values.

Enter the required information. Press <Enter> to Validate.

<table>
<thead>
<tr>
<th>Site</th>
<th>IP-address</th>
<th>Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC1</td>
<td>nnn.nnn.nnn.nnn</td>
<td>25000</td>
</tr>
<tr>
<td>DC3</td>
<td>nnn.nnn.nnn.nnn</td>
<td>25000</td>
</tr>
</tbody>
</table>
```

Figure 53 Define Managed HMCs panel

1. Complete the Define Managed HMCs panel fields as follows:
   - Site
     Indicates the ID of the site location being specified. Valid values are DC1 or DC3. This field is not editable.
   - IP-address
Specify the IP address of the hardware management console located at the location specified by Site. GDDR uses this IP-address to perform console operations at that site location.

**Note:** Only one HMC IP-Address parameter per Site is permitted. If a site is specified in the current configuration in the Define Managed Systems panel, then an HMC IP-Address must be present for that site.

- **Timeout**

  Specify the maximum number of milliseconds a request to an HMC console at a specified site may remain outstanding before it is considered to have timed out. The maximum value you can specify is 99999; the minimum is 1. The default value is 25000.

2. If you made any changes, type **Save** on the command line and press **Enter**.

3. Press **F6** to proceed to the Define HMC Community Names panel shown in Figure 54.

**Option C: Define HMC Community Names**

When you specify option C in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

```
---------------------- GDDR - Define HMC Community Names ----- Row 1 to 9 of 9
Command ==> 
Press <F3> to return to the GDDR Define Host Objects Menu
<F5> Define Managed HMCs                <F6> Define Managed Couple DS (1/2)
Type SAVE to save, LOAD to restart from last saved values.
Line commands: A dd, D elete

<table>
<thead>
<tr>
<th>CMD</th>
<th>Site</th>
<th>C-System</th>
<th>Community Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>DC1</td>
<td>SYS3</td>
<td>zosesys3</td>
</tr>
<tr>
<td>_</td>
<td>DC1</td>
<td>SYS1</td>
<td>zosesys1</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>SYS3</td>
<td>zosesys3</td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>SYS1</td>
<td>zosesys1</td>
</tr>
</tbody>
</table>

Figure 54  Define HMC Community Names panel
```

1. If the panel is not pre-populated with existing entries, type **Add** in the Command field at the top of the panel to display a new line for field entry.

2. Insert additional sites by entering **R** (Repeat) in the command field to the left of an existing entry and overtype the existing data.

3. Complete the panel fields as follows:

   - **HMC Site**
     Indicates the site location of the HMC being defined. Site values are populated from the Define Managed Systems panel. Valid values are DC1 or DC3.

   - **C-System**
     Indicates the system name of a C-System. C-System values are populated from the GDDMPARM CSYSSITE parameters. There will be one row per HMC per C-system in the configuration.
• **HMC Community Name**

Specify a case-sensitive community name to be used to allow the named C-system access to the HMC on the indicated site. The default community name for each C-System/site combination is the system name in lowercase. If you provide a different community name, GDDR uses the specified name in place of the default name when an LPAR initiates a command to the HMC console.

*Note:* The maximum length of the community name for the HMC is established by the HWMCA MAX COMMUNITY LEN parameter for the site. For more information, refer to the *System z Application Programming Interfaces* manual, SB10-7030-11, Data exchange APIs and commands API structures and definitions topic. The current maximum is 16 characters.

4. If you made any changes, type **Save** on the command line and press **Enter**.

5. Press **F6** to proceed to the Define Managed Couple Datasets panel shown in **Figure 55**.

### Option D: Define Managed Couple Datasets

When you specify option **D** in the Define Host Objects panel shown in **Figure 45 on page 146**, the following panel appears:

![Define Managed Couple Datasets panel 1 of 2](image)

---

**Figure 55** Define Managed Couple Datasets panel 1 of 2

The use of Couple Datasets is dependent on the site's exploitation of sysplex architecture. If no managed systems are part of a sysplex, the following message is returned upon opening this panel:

GDDI371W No sysplexes defined, please either assign systems to a sysplex or proceed to the next panel.

The fields in panel 1 of the Define Managed Couple Datasets option indicate the types of couple datasets that are used on a specified sysplex.

1. View, define, or modify the panel 1 fields as follows:

   • **Sysplex**

     Indicates the sysplex name, if applicable. Sysplex values are populated from the Define Managed Systems panel.

   • **Couple Dataset Types:** ARM BPX CFR LOGR SFM SYS WLM

     Type **YES** or **NO** under each dataset type to indicate whether or not you want GDDR to ensure correct couple dataset placement for that type during planned and unplanned site swap scripts for a specified sysplex.
Couple Facility dataset and structure management is further controlled for planned and unplanned scripts by the Rebuild CF Dataset, Rebuild CF Structure, and the CF Rebuild Timeout fields described in “Option S: Script Sysplex Options” on page 171.

2. If you made any changes, type Save on the command line and press Enter.

3. Press F6 to complete the couple dataset definitions as shown in the following panel:

![GDDR - Define Managed Couple Datasets (2/2)](image)

Figure 56 Define Managed Couple Datasets panel 2 of 2

The fields in panel 2 of the Define Managed Couple Datasets option indicate the names of primary and alternate couple datasets of the specified type to be used on the systems belonging to the specified sysplex.

**Note:** Each couple dataset must be cataloged on all systems in the sysplex.

There must be a primary and an alternate couple dataset each sysplex, for each of the possible primary DASD sites, and for each couple dataset type being used.

GDDR ensures that the primary couple datasets are located wherever the primary DASD currently resides. If the secondary DASD site is available, EMC GDDR ensures that an alternate couple dataset is used on the secondary DASD site. If the secondary DASD site is unavailable, and valid dataset names have been provided for DSN3 and DSN4 as described in Table 16 below, GDDR ensures that an alternate couple dataset is used at the primary DASD site.

Four lines display for each sysplex and managed type, one line for each dataset's role (P or A) for both sites DC1 and DC2. For each type and role within a sysplex, specify the dataset names that are cataloged at DC1 and DC2 as shown in Table 16.

### Table 16 Defining Managed Couple Datasets

<table>
<thead>
<tr>
<th>Sysplex</th>
<th>TYPE</th>
<th>SITE</th>
<th>ROLE</th>
<th>Couple Dataset Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>TESTPLX1</td>
<td>ARM</td>
<td>DC1</td>
<td>P</td>
<td>DSN1: used as Primary when DC1 is Primary and used as Alternate when DC1 is Secondary</td>
</tr>
</tbody>
</table>
Using GDDR Online Facilities

Using GDDR Online Facilities

1. View, define, or modify the fields displayed in panel 2 as follows:

   - **Sysplex**
     
     Indicates the sysplex name. Sysplex values are populated with the list of sysplexes which have at least one managed couple dataset type, as specified on panel 1.

   - **Type**
     
     Indicates the couple dataset type as specified on panel 1. Valid values are ARM, BPX, CFR, LOGR, SFM, SYS, or WLM.

   - **Site**
     
     Indicates the ID of the site location.

   - **Role**
     
     Indicates the role of the dataset. Valid values are P for primary or A for Alternate.

   - **Couple Dataset Name**
     
     Specify the dataset names that are cataloged at DC1 and DC2.

2. If you made any changes, type **Save** on the command line and press **Enter**.

3. Press **F6** to proceed to the Define Managed CF Structures panel shown in Figure 57.

### Table 16 Defining Managed Couple Datasets

<table>
<thead>
<tr>
<th>Sysplex</th>
<th>TYPE</th>
<th>SITE</th>
<th>ROLE</th>
<th>Couple Dataset Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>DC2</td>
<td>P</td>
<td></td>
<td>DSN3: used as Alternate when DC1 is Primary and DC2 is unavailable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P</td>
<td></td>
<td>DSN2: used as Primary when DC2 is Primary and used as Alternate when DC2 is Secondary</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A</td>
<td></td>
<td>DSN4: used as Alternate when DC2 is Primary and DC1 is unavailable</td>
</tr>
</tbody>
</table>
Option CF: Define Managed CF Structures

When you specify option CF in the Define Host Objects panel shown in Figure 45 on page 146, the following panels appear:

![Define Managed CF Structures panel 1 of 2](image1)

Note: If the GDDR-managed systems are not members of a sysplex, the following message is displayed in the panel:

No sysplexes defined, please either assign systems to a sysplex or proceed to the next panel.

![Define Managed CF Structures panel 2 of 2](image2)
During Planned and Unplanned Swap scripts, GDDR will issue Display XCF commands for the CF Structures identified in the Define Managed CF Structures panel and determine if they are currently located in the preferred couple facility. If a structure is found in a different couple facility than the one which is first in the list for the current primary DASD site, GDDR will issue REBUILD commands and verify the results. Verification is repeated until all structures are located in one of the acceptable couple facilities.

Couple Facility structure management is controlled for planned and unplanned scripts by the Rebuild CF Structure and the CF Rebuild Timeout fields described in “Option S: Script Sysplex Options” on page 171.

1. View, define, or modify the panel fields as follows:

- **Sysplex**
  Indicates the sysplex name, if applicable. Sysplex values are populated from the Define Managed Systems panel.

  If the panel is not pre-populated with existing Sysplex name entries, type Add in the Command field at the top of the panel to display a new line for an entry. Insert additional Sysplex names by entering R in the command field to the left of an existing Sysplex entry.

- **Structure Name**
  Provide a Couple Facility Structure name or names for each specified Sysplex using the following guidelines:
  - Length: 1-16 characters
  - Position 1: Uppercase alphabetic
  - Positions 2-16: Uppercase alphabetic, numeric, or _, @, $, #

- **CF Site**
  The ID of the site location being specified. Site values are populated from the Define Managed Systems panel. This is the site which is the primary DASD site when the listed sequence of Eligible Coupling Facility names should be used to determine correct location of CF Structures.

- **Eligible Coupling Facility Names**
  Provide up to 4 Couple Facility names, delimited by commas, using the following guidelines:
  - Length: 1-8 characters
  - Position 1: Uppercase alphabetic
  - Position 2-8: Uppercase alphabetic, numeric, or _, @, $, #

2. If you made any changes, type Save on the command line and press Enter.

3. Press F6 to proceed to the Define External Workloads panel shown in Figure 59.
Option W: Define External Workloads

External workloads run in mainframe systems which do not have their DASD in the managed VMAX systems. GDDR can coordinate Stop and Start of the workload on these “non-managed” mainframe systems with the workload Stop and Start for managed systems. When EMC GDDR takes planned or unplanned actions that impact the managed systems, it has the capability to trigger the stop or start of external workloads through a user-exit.

Note: The definition of external workloads is optional. If no external workloads are to be Stopped or Started in sequence with managed systems, press F6 to proceed to the Define EMC Mainframe Enablers Started Tasks panel.

When you specify option W in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

![Figure 59 Define External Workloads panel](image)

1. Type Add in the Command field at the top of the panel to display a new line for an entry.
2. View, define, or modify the panel fields as follows:
   - Site
     Indicates the ID of the specified site location.
   - System
     Indicates the system names which host external workload requiring coordination with system actions performed by GDDR.

Note: If you are using the user exits provided with EMC GDDR, then communication to the targeted systems requires that message interception rules GDDX191I and GDDX291I be enabled. Use the following GDDRMAIN console commands to enable these rules:
   - FGDDRMAIN,MSGS,GDDX191I,ENABLE
   - FGDDRMAIN,MSGS,GDDX291I,ENABLE

3. If you made any changes, type Save on the command line and press Enter.
4. Press F6 to proceed to the Define EMC MF Enablers STCs panel shown in Figure 60.
Option E: Define EMC MF Enablers STCs

GDDR uses information about the site’s installation parameters to communicate with EMC Mainframe Enablers started tasks that reside on GDDR-managed systems and GDDR C-Systems.

When you specify option E in the Define Host Objects panel shown in Figure 45 on page 146, the following panel appears:

--- GDDR - Define EMC Mainframe Enablers STCs Row 1 to 7 of 10 ---

<table>
<thead>
<tr>
<th>CMD</th>
<th>System</th>
<th>Type</th>
<th>Name</th>
<th>MSTR</th>
<th>Parameter Dataset and Member(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>SYS1</td>
<td>RDF</td>
<td>N/A</td>
<td>NO</td>
<td>EMC.GDDR500.CUSTOM.PARMLIB</td>
</tr>
<tr>
<td>_</td>
<td>SYS2</td>
<td>RDF</td>
<td>N/A</td>
<td>NO</td>
<td>EMC.GDDR500.CUSTOM.PARMLIB</td>
</tr>
<tr>
<td>_</td>
<td>SYS3</td>
<td>RDF</td>
<td>N/A</td>
<td>NO</td>
<td>EMC.GDDR500.CUSTOM.PARMLIB</td>
</tr>
<tr>
<td>_</td>
<td>SYS1</td>
<td>SCF</td>
<td>GDDRSCF</td>
<td>NO</td>
<td>N/A</td>
</tr>
<tr>
<td>_</td>
<td>SYS2</td>
<td>SCF</td>
<td>GDDRSCF</td>
<td>NO</td>
<td>N/A</td>
</tr>
<tr>
<td>_</td>
<td>SYS3</td>
<td>SCF</td>
<td>GDDRSCF</td>
<td>NO</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Figure 60  Define EMC Mainframe Enablers STCs panel

1. Type **Add** in the Command field at the top of the panel to display a new line for an entry.

2. The pre-populated fields are:
   - **System**
     
     Indicates the z/OS system name of a GDDR-managed system or a GDDR C-System which resides at the specified site. System Name values are populated from the Define Managed Systems panel.
   
   - **Type**
     
     Indicates the Mainframe Enablers application. Valid abbreviations are:
     
     - SCF: Symmetrix Control Facility (ResourcePak Base)
     - RDF: SRDF Host Component
     - CG: Consistency Group
   
   - **Parameter Dataset and Member(s)**
     
     Provides the name of the partitioned dataset and members containing the customization parameters for the specified started task.

**Note:** “Install GDDR started procedures” on page 65 provides more information.
3. Requirements for GDDR-managed systems:
   - SCF: STC name, SUB=MSTR YES
     Parameter Dataset and Member name are not applicable.
     \[\text{Note: SUB=MSTR is ignored for SCF entries.}\]

4. Requirements for C-Systems:
   - RDF: STC name, SUB=MSTR NO
     Parameter dataset and member pointed to by the HC STC RDFPARM DD statement.
     Required.
   - SCF: STC name, SUB=MSTR NO
     Parameter Dataset and Member name are not applicable.
     \[\text{Note: SUB=MSTR is ignored for SCF entries.}\]

5. If you made any changes, type Save on the command line and press Enter.
6. Press F6 to proceed to the Specify GDDR Options panel.

Option O: Specify GDDR Options

When you specify option O in the Parameter Management Options Menu shown in Figure 9 on page 110, the Specify GDDR Options panel appears:

\[\begin{array}{|c|c|}
\hline
\text{Option} & \text{Description} \\
\hline
O & Default Script Call Overrides \quad \text{This System: SYS1} \\
S & Script Sysplex Options \quad \text{This Site: DC1} \\
J & Script JCL Parameters \quad \text{Master-C: SYS1} \\
U & Utility Parameters \quad \text{Primary Site: DC1} \\
M & Messaging Options \quad \text{Primary DASD: DC1} \\
T & Tuning Values \\
L & GDDR user labels \quad \text{Automation: ON} \\
& \quad \text{Planned script: None} \\
& \quad \text{Unplanned script: None} \\
\hline
\end{array}\]

Select an option and press <Enter>

\[<F5> \text{Define EMC MF Enabler STCs} \quad <F6> \text{Default Script Call Overrides}\]

Press <F3> to return to the GDDR Parameter Management Options Menu

Figure 61 Specify GDDR Options panel

Complete the following options listed in this panel to define your site-specific script options:

* "Option O: Default Script Call Overrides" on page 170
* "Option S: Script Sysplex Options" on page 171
* "Option J: Script JCL Parameters" on page 173
◆ “Option U: Utility Parameters” on page 174
◆ “Option M: Messaging Options” on page 176
◆ “Option T: Tuning Values” on page 177
◆ “Option L: GDDR User Labels” on page 180
Option O: Default Script Call Overrides

When you specify option O in the Specify GDDR Options panel, the following panel appears:

```
Command ===>
Press <F3> to return to the GDDR Specify Options Menu
<F5> Specify GDDR Options Menu          <F6> Script Sysplex Options
Type <SAVE> to save, <LOAD> to restart from last saved values.

Enter Y to call the function, N to NOT call the function.

Call Program Function Description
----- -------- ----------------------------------------------------------
More:     +
Y  GDDDRDF0 Call DYNAPI Interface
Y  GDDDRDF0 DYNAPI - SRDF/S Commands
Y  GDDDRDF0 DYNAPI - DeletePair and Half-DeletePair Commands
Y  GDDDRDF0 DYNAPI - CreatePair and ResumePair Commands
Y  GDDDRDF0 DYNAPI - SRDF/A Commands
Y  GDDDRDF0 DYNAPI - Swap and Half-Swap commands
N  GDDRXDRV Manage Distributed Workload
N  GDDRKF0C Trigger Production Workload Stop
N  GDDRKF0I Trigger Production Workload Startup
N  GDDRGFO8 Use ConGroup Shutdown/Startup instead of Refresh
Y  GDDRKF0E Transfer AutoSwap Ownership
Y  GDDRKF0H Transfer Master Function Ownership
N  GDDRKF20 Simulate BCV Actions
N  GDDRKF20 Simulate BCV Actions
Y  GDDRKF20 Manage BCVs at DC1
Y  GDDRKF20 Manage BCVs at DC2
Y  GDDRKF20 Manage BCVs at DC3
Y  GDDRKF20 Manage BCVs at DC4
N  GDDRKF20 Manage External BCV devices
N  GDDRKF20 Manage TEST BCVs at DC1
N  GDDRKF20 Manage TEST BCVs at DC2
N  GDDRKF20 Manage TEST BCVs at DC3
N  GDDRKF20 Manage TEST BCVs at DC4
N  GDRCCL00 Perform SDDF Session Cleanup
Y  GDDDRDF0 Manage External SRDF devices
N  GDDRGPHS Allow Primary Site HMC Actions in Planned Scripts
N  GDDRGPHS Allow Secondary Site HMC Actions in Planned Scripts
N  GDDRGPHS Allow Primary Site HMC Actions in Unplanned Scripts
N  GDDRGPHS Allow Secondary Site HMC Actions in Unplanned Scripts
N  GDDRGPHS Allow Primary Site HMC Actions in Test Scripts
N  GDDRGPHS Allow Secondary Site HMC Actions in Test Scripts
N  GDDRGPHS Allow Primary Site HMC Actions in RDR Scripts
N  GDDRGPHS Allow Secondary Site HMC Actions in RDR Scripts
Y  GDDRGPHS Allow LOAD CLEAR when Activating LPARs in Scripts
Y  GDDRGPHS Confirm GDDR Script HMC actions by WTOR
N  GDDRGPHS Allow Secondary Site HMC Actions in Unplanned Scripts
N  GDDRGPHS Allow Primary Site HMC Actions in Test Scripts
N  GDDRGPHS Allow Secondary Site HMC Actions in Test Scripts
N  GDDRGPHS Allow Primary Site HMC Actions in RDR Scripts
N  GDDRGPHS Allow Secondary Site HMC Actions in RDR Scripts
Y  GDDRGPHS Allow LOAD CLEAR when Activating LPARs in Scripts
Y  GDDRGPHS Confirm GDDR Script HMC actions by WTOR
N  GDDDRHM2 Prompt for Alternate SYSRES usage
N  GDDDRHM2 Use DR-Test IPL parameters
N  GDDRKF20 BACKGROUNDCOPY for TF/Clone

******************************************************************************
```

Figure 62  Specify Default Script Call Overrides panel
1. Overtype the default Y or N call override values to change the default overrides used by all scripts.

You can set the ‘Confirm GDDR Script HMC actions by WTOR’ call override setting to P for Priority, meaning that WTORs for HMC actions during scripts will be reduced to 1 per priority level. The usual confirmation WTORs for each individual action will then be suppressed.

Table 21, “GDDR call overrides,” on page 333 lists the possible call overrides and their meanings.

Note: GDDR BCV management is controlled using call overrides to program GDDRKF20. If you set call overrides to allow GDDR BCV management for a given site, it is assumed that BCVs are in the proper state prior to the script being started; therefore, you should not change BCV states in between scripts using non-GDDR actions.

Ensure that Call Overrides for BCV management at a given site are set to N, if no BCVs are defined to GDDR for that site.

2. If you made any changes, type Save on the command line and press Enter.
3. Press F6 to proceed to the Script Sysplex Options panel.

Option S: Script Sysplex Options

When you specify option S in the Specify GDDR Options panel shown in Figure 61 on page 168, the following panel appears:

![Figure 63 Script Sysplex Options panel](image.png)

**Figure 63** Script Sysplex Options panel

1. Specify YES or NO for the following script management options for each site:
   - **Realign Couple Datasets**
     Specify whether Planned or Unplanned scripts should manage couple datasets when the primary DASD site is DCn. Further control is provided by the couple datasets defined in Option D, Define Managed Couple Datasets, as shown in Figure 56 on page 162 and the External field in the Define IPL Parameters panel as shown in Figure 51 on page 156.
     - When Planned is YES: Automate the management of couple datasets as a part of planned script processing.
– When Unplanned is **YES**: Automate the management of couple datasets as a part of unplanned script processing.

**Rebuild CF Structures**

Specify whether Planned or Unplanned scripts should manage Coupling Facility (CF) Structures at site DCn. Further control is provided by the PSTR.* parameters, and the System Exclude list.

– When Planned is **YES**: Automate the management of Coupling Facility Structures as a part of planned script processing.

– When Unplanned is **YES**: Automate the management of Coupling Facility Structures as a part of unplanned script processing.

**CF Rebuild Timeout**

Specify the number of seconds to allow for the processing required to rebuild Coupling Facility structures. Valid values are 1-3600. The default value is 600.

2. If you made any changes, type **Save** on the command line and press **Enter**.

3. Press **F6** to proceed to the Script JCL Parameters panel.

**System qualification for sysplex options**

The following qualifications apply both to couple dataset management and CF Structure management in scripts, as well as stand-alone actions using Option S: Manage Couple Datasets and Option CF: Manage CF Structures from the “Option A: Actions—Perform GDDR Actions” on page 215.

◆ **z/OS XCF commands** are issued for a list of eligible systems constructed from Primary and Secondary sites in AutoSwap configurations. The source of eligible systems in configurations without AutoSwap is only the Primary site.

◆ Exclude systems from the Exclude list.

◆ Exclude systems with Desired State = D.

◆ Exclude systems where GDDRMAIN is down.

◆ Limit resulting list of systems to 1 system per sysplex.

◆ The eligible system is preferably located at the site where the script runs.

Usage of GDDR couple dataset and CF Structure management require that systems are defined in GDDMPARM COMM and CPC parameters, and have GDDRMAIN running at the time of the action.

Systems not defined in GDDMPARM or where GDDRMAIN is found inactive are not considered eligible for these features, and sysplexes where no systems meet the criteria will be skipped. GDDR issues a warning message when no systems are retained.

The reduction of the list of eligible systems to 1 per sysplex requires correct sysplex names to be provided in “Option S: Define Managed Systems” on page 147.
Option J: Script JCL Parameters

GDDR Planned and Unplanned automation sequences run as MVS batch jobs. The Script JCL Parameters panel provides the customization required for those batch jobs to run successfully in your MVS environment. For GDDR scripts started internally by GDDR, this panel is the only place where this information can be provided. The jobcard information can be overridden as part of the script submission started from the Select Script to Run panel as shown in “Option S: Scripts — Run GDDR Scripts” on page 208.

When you specify option J in the Specify GDDR Options panel shown in Figure 61 on page 168, the following panel appears:

1. View, define, or modify the panel fields as follows:
   - **Site - C-System (display only)**
     
     Script JCL values are maintained for each C-System. Complete the values for the first displayed C-System, and scroll to the next C-System using F8.
   - **HC-Prefix**
     
     Specifies the prefix that will be used for SRDF Host Component API commands. The default value set by GDDR is null, and must be replaced by a valid prefix during GDDR customization.
   - **SCF-Suffix**
     
     Specifies the suffix appended to ‘SCF$’ to form the DDname on a DD statement in unplanned script jobs and GDDRMAIN and the Console and Heartbeat monitors, associating these with an SCF subsystem. The default value set by GDDR is "EMC" which is suitable only if you want EMC GDDR scripts to connect to an SCF instance which either does not specify an SCF subsystem ID, or also specifies "EMC".
   - **Work Unit**
     
     Indicates the user-specified device esoteric value that represents the storage device type to be used when GDDR dynamically allocates temporary datasets. The default value is SYSALLDA.

   **Note:** If you wish to change the default from SYSALLDA, ensure that the replacement device esoteric is not defined to z/OS using the ‘VIO’ attribute.
• **Jobname Prefix and Enforce**
  
  Specifies a 3-character prefix that is used as the first three characters of EMC GDDR jobnames. When you provide the jobname prefix and enter Y in the Enforce field, EMC GDDR internally-submitted jobs will use the jobname prefix specified in this panel. Any attempts to start a script from the Select Script to Run panel using a different jobname prefix will fail.

  If no prefix is defined, or the prefix is not enforced, GDDR uses the 3 first characters of the user-provided jobname as a prefix, or "GDD" if no jobname can be identified in the JOB cards you provided in the GDDRJCLn variables.

• **Work HLQ**
  
  Specifies a dataset high-level-qualifier with a maximum length of 17 characters that is used for work datasets. The default is GDDR. Table 6, “RACF permissions,” on page 55 provides recommended authorizations for this HLQ.

• **Surrogate User ID and Enforce**
  
  Specifies the user ID that has been authorized to all of the resources required by GDDR processes. EMC recommends specifying the user ID provided in the ADDUSER statement of the GDDCRACF C-System RACF definitions, as described in “Specify GDDR security” on page 54.

  When you provide the user ID and enter Y in the Enforce field, GDDR internally-submitted jobs will use the surrogate user ID specified in this panel. Any attempts to start a script from the Select Script to Run panel using a different user ID will fail.

• **Enter Jobcards**
  
  These fields specify the jobcards that will be used when a job is submitted internally by GDDR. Always include the REGION=0M jobcard parameter.

2. If you made any changes, type **Save** on the command line and press **Enter**.
3. Press **F6** to proceed to the Utility Parameters panel.

**Option U: Utility Parameters**

GDDR script processing sometimes requires invoking external utilities, which have their own space allocation requirements and control settings. This is done using values supplied by the GDDR Utility Parameters panel.
When you specify option U in the Specify GDDR Options panel shown in Figure 61 on page 168, the following panel appears:

```
-------------------------- GDDR - Utility Parameters --------------------------
Command ===>
Press <F3> to return to the Specify GDDR Options Menu
<F5> Script JCL Parameters  <F6> Messaging Options
Type <SAVE> to save, <LOAD> to restart from last saved values.
Enter the required information. Press <Enter> to Validate.
More: +

- EMCGROUP Allocation Parameters
  - SYSIN : ________________
  - REPORT : ________________
  - SYSPRINT : ________________

- EMCTF Parameters
  - SYSIN : ________________
  - SYSOUT : ________________
  - MAXREQ : ________________

- RPTOUT Allocation Parameters
  - SCFRDFME : ________________
  - SCFRDFM6 : ________________
  - SCFRDFM9 : ________________

- ECGUTIL Parameters
  - COMMAND allocation: ________________
  - Number of Tasks : __
  - Message Level : __

Utility Names
- IEBCOPY : ________
- IEBGENER : ________
- SDDF Cleanup : ________
- SDDF List : ________

eSTEM exec dsname for GDDR exec: (Default: Prefix.STEM.REXX(ESFRRUN))
```

Figure 65 Utility Parameters panel

1. View, define, or modify the panel fields as follows:

   - **EMCGROUP Allocation Parameters**
     - SYSIN: Recommended allocation is TRACKS,1,1
     - REPORT: Recommended allocation is CYLINDERS,1,1
     - SYSPRINT: Recommended allocation is CYLINDERS,1,1

   - **EMCTF Parameters**
     - SYSIN: Recommended allocation is TRACKS,5,2
     - SYSOUT: Recommended allocation is TRACKS,50,50
     - MAXREQ: Recommended value is 20000

   - **RPTOUT Allocation Parameters**
     - SCFRDFME: Recommended allocation is TRACKS,5,2
     - SCFRDFM6: Recommended allocation is TRACKS,5,2
     - SCFRDFM9: Recommended allocation is TRACKS,5,2

   - **ECGUTIL Parameters**
     - COMMAND: Recommended allocation is CYLINDERS,10,10
     - Number of Tasks: Recommended value is 22
     - MSGLEVEL: Recommended value is 5
• Utility Names

The Utility Names fields allow site-specific customization of these utilities. JCL generated for Parm Load validation and activation jobs use the specified names for IEBCOPY and IEBGENER.

– IEBCOPY: IEBCOPY
– IEBGENER: IEBGENER
– SDDF Cleanup: GDDFCLN1
– SDDF List: GDDFList

• HMC Optional settings

– Use Direct Initialize: Enter Y or N.

Direct Initialize is an HMC API option that directs HMC API communication to use port 3161. This option is useful for installations where a firewall exists between GDDR and the HMC console. Specifying Y allows an installation to open a particular port (port 3161) in the firewall.

– Security check on Discover: Recommended value is Y.

This is a customer preference with regards to daily usage. With regards to installing the product, suppressing the security check may be required to verify that the HMC setup steps are completed successfully.

• eSTEM exec dsname for GDDR exec

This is the full name of the exec to be executed when Option ST: eSTEM is selected on the GDDR Primary Options Menu.

2. If you made any changes, type Save on the command line and press Enter.

3. Press F6 to proceed to the Messaging and SMF Options panel.

Option M: Messaging Options

When you specify option M in the Specify GDDR Options panel shown in Figure 61 on page 168, the following panel appears:

```
Command ===>  GDDR - Messaging Options
Press <F3> to return to the Specify GDDR Options Menu
<F5> Utility Parameters  <F6> GDDR Tuning Values
Type <SAVE> to save, <LOAD> to restart from last saved values.

Enter the required information. Press <Enter> to Validate.

Include MSGID in messages issued via SAY : N
Include Time in messages issued via SAY : N

******************************** Bottom of Data ******************************
```

**Figure 66** Messaging Options panel

1. View, define, or modify the panel fields as follows:

• Include MSGID in messages issued via SAY
Using GDDR Online Facilities

Y or N indicates whether to include the 8-character message ID with messages generated from program SAY functions. SAY functions are foreground operations. The default value is N.

- Include Time in messages issued via SAY

Y or N indicates whether to include the system time in hh:mm:ss format with messages generated from program SAY functions. SAY functions are foreground operations. The default value is N.

2. If you made any changes, type Save on the command line and press Enter.

3. Press F6 to proceed to the Tuning Values panel.

Option T: Tuning Values

When you specify option T in the Specify GDDR Options panel shown in Figure 61 on page 168, the following panels appear:

```
---------------------- GDDR - Specify GDDR Tuning Values ----------------------
Command ===>
Press <F3> to return to the Specify GDDR Options Menu    <F5> Messaging Options    <F6> GDDR User Labels
Type <SAVE> to save, <LOAD> to restart from last saved values.
Enter the required information. Press <Enter> to Validate.

   Event Monitor Interval:    20  (1-999, seconds, default 20)
   EVM ConGroup Verify Interval:  1440  (0-9999, minutes, default 1440)
   EVM SRDF/A Check Interval:   600  (0-9999, seconds, default 600)
   EVM MSC Activity Monitor:    1  (0 or 1, default 0)
   EVM MSC Cycle Threshold:     2  (0 or 2-100, default 0)
   Heartbeat Monitor Interval:  30  (1-999, seconds, default 30)
   Missing Heartbeat Threshold:  10  (1-999, HBM cycles, default 10)
   WTOR Wait Time:   990  (1-3600, seconds, default 600)
   WTOR Wait Retries:     0  (0-999, number, default 0)
   HMC Reset Clear Wait Time:  15  (1-99, seconds, default 15)
   HMC LPAR Activation Wait Time:     5  (1-99, seconds, default 5)

---------------------- GDDR - Specify GDDR Tuning Values ----------------------
Command ===>
Press <F3> to return to the Specify GDDR Options Menu    <F5> Messaging Options    <F6> GDDR User Labels
Type <SAVE> to save, <LOAD> to restart from last saved values.
Enter the required information. Press <Enter> to Validate.

   TimeFinder RE-ESTABLISH Wait Time:  1440  (1-9999, minutes, default 1440)
   TimeFinder SPLIT Wait Time:     240  (1-9999, minutes, default 240)
   SCFRDFM9 Cleanup Wait Time:     60  (1-600, seconds, default 60)
   SRDF Resynchronization Interval:  10  (1-30, seconds, default 10)
   DIm command API Timeout:     600  (1-3600, seconds, default 900)
   Dlm VARY OFFLINE Timeout:    900  (1-3600, seconds, default 300)

******************************************************************************* Bottom of Data *******************************************************************************
```

Figure 67 Specify GDDR Tuning Values panels

When the Specify GDDR Tuning Values panel is initially displayed, all values are blank. When you press Enter, all fields are populated with the defaults.
Note: Regardless of the settings in the fields described below, GDDR EVM MSC Activity Monitoring is only performed after MSC is seen to be Activated for the GDDR-managed group or groups, and stops when MSC is Disabled for these groups.

1. View, define, or modify the panel fields as follows:

   • Event Monitor Interval
     Specify the time, in seconds, that the event monitor on each C-System waits between successive checks of the various GDDR event indicators. The value must be a number between 1 and 999. The default is 20 seconds.

   • EVM ConGroup Verify Interval
     Specify the time, in minutes, that the GDDR Event Monitor waits between successive EMC ConGroup verify commands. The default value is 1440 minutes. A value of 0 disables the feature. Non-zero values lower than 60 are rounded up to 60.

   • EVM SRDF/A Check Interval
     Specify the time, in seconds, that the GDDR Event Monitor waits between successive polls of the configured SRDF/A groups. The default value is 600 seconds. A value of 0 disables the feature. Non-zero values lower than 600 are rounded up to 600.

   • EVM MSC Activity Monitor
     Controls whether or not GDDR EVM will perform MSC software activity monitoring. A setting of 1 causes GDDR EVM to issue message GDDS227E when the MSC Post Timestamp is 2 or more times older than MSC_CYCLE_TARGET seconds. Message GDDS228E signals the return to normal. A setting of 0 suppresses this monitor, and also causes GDDR EVM to ignore any non-zero value for the MSC Cycle Threshold field and the SDDF Cycle and Time Threshold fields.

   • EVM MSC Cycle Threshold
     Controls if and when GDDR EVM will issue E-level messages for delays in MSC Cycle switching, with allowed values 0 and range 2-100. For example, with a value of 5, GDDR EVM will issue message GDDS229E when the most recent MSC Cycle Switch TOD value is 5 or more times older than MSC_CYCLE_TARGET seconds. Message GDDS230E signals the return to normal. A setting of 0 suppresses this monitor.

   • Heartbeat Monitor Interval
     Specify the time, in seconds, that the heartbeat monitor on each C-System waits before setting and propagating its new heartbeat value. The value must be a number between 1 and 999. The default is 30 seconds.
• Missing Heartbeat Threshold

Specify the number of times a heartbeat monitor on a C-System will need to detect no change in the heartbeat value of another C-System, upon awakening from its own wait interval, before it will declare the other C-System dead. The value must be a number from 1 to 999. The default value is 10.

• WTOR Wait Time

Specify the number of seconds that a GDDR script will wait for an operator reply to a WTOR it has issued. When the specified interval has expired, the WTOR is deleted and the script proceeds as if the operator had replied 'N' or 'CANCEL' depending upon the particular message.

This parameter is optional. If not specified, the default value of 600 is used, which is equivalent to 10 minutes.

• WTOR Wait Retries:

Specify the number of times to re-issue the same WTOR. This optional parameter allows you to prolong the time GDDR has to wait for a certain event.

The GDDR WTOR module accept the RETRIES argument, which specifies the number of times to re-issue the same WTOR when intercepting a timeout condition. You can end the wait by replying negatively to the WTOR message. By default (WTOR Wait Retries=0) the WTOR will be issued just once, resulting in script termination with RC=32 if a timeout occurs.

**Note:** This parameter is used during all MSC State changing operations.

• HMC Reset Clear Wait Time

Specify the number of seconds that the HMC LPAR Reset Clear command will wait for confirmation that the operation completed. The value must be a number between 1 and 99. The default is 15 seconds.

• HMC LPAR Activation Wait Time

Specify the number of seconds that the HMC LPAR Activation command will wait for confirmation that the operation completed. The value must be a number between 1 and 99. The default is 5 seconds.

• TimeFinder Re-Establish Wait Time

Specify the number of minutes that are used when TimeFinder Establish commands specify a WAIT parameter. The default is 1440 minutes.

• TimeFinder Split Wait Time

Specify the number of minutes that are used when TimeFinder Split commands specify a WAIT parameter. The default is 240 minutes.

• SCFRDFM9 Cleanup Wait Time

Specify the number of seconds that GDDR waits for SRDF/A cleanup to complete after the SCFRDFM9 utility completes. The value must be a number between 1 and 600. The default is 60 seconds.
• SRDF Resynchronization Wait Time
   Specify the number of minutes that GDDR waits for invalids to synchronize. The value must be a number between 1 and 30. The default is 10 minutes.

• DLm command API Timeout
   Specify the number of seconds that GDDR is to wait for a response to a command sent to a DLm Access Control Point (ACP) via the TCP/IP command API. The default is 900 seconds.

• DLm VARY OFFLINE Timeout
   Specify the number of seconds that GDDR is to wait for all DLm virtual tape drives to be varied offline. The default is 300 seconds.

2. If you made any changes, type Save on the command line and press Enter.

3. Press F6 to proceed to the Define GDDR User Labels panel.

**Option L: GDDR User Labels**

When you specify option L in the Specify GDDR Options panel shown in Figure 61 on page 168, the following panel appears:

--- GDDR - Define GDDR User Labels ---

<table>
<thead>
<tr>
<th>Command</th>
<th>Press &lt;F3&gt; to return to the GDDR Options Menu</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&lt;F5&gt; Tuning Values</td>
</tr>
<tr>
<td></td>
<td>&lt;F6&gt; Parameter Management Options Menu</td>
</tr>
<tr>
<td></td>
<td>Type SAVE to save, LOAD to restart from last saved values.</td>
</tr>
</tbody>
</table>

**GDDR Complex name ====> ________ (default: GDDRPLEX)**

<table>
<thead>
<tr>
<th>Site</th>
<th>User label (uuuuuuu)</th>
<th>Region</th>
<th>User label (uuuuuuu)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC1</td>
<td>_______ (7 chars max)</td>
<td>RG1</td>
<td>_______ (7 chars max)</td>
</tr>
<tr>
<td>DC3</td>
<td>_______</td>
<td>RG2</td>
<td>_______</td>
</tr>
</tbody>
</table>

**DISPLAY FORMAT:**

<table>
<thead>
<tr>
<th>User site display format ====&gt; 0</th>
<th>User region display format ====&gt; 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format Template: Example:</td>
<td>Format Template: Example:</td>
</tr>
<tr>
<td>0 - DC# DC1 (default)</td>
<td>0 - RG# RG1 (default)</td>
</tr>
<tr>
<td>1 - uuuuuuu# London1</td>
<td>1 - uuuuuuu# Europe1</td>
</tr>
<tr>
<td>2 - #uuuuuu# London</td>
<td>2 - #uuuuuu# Europe</td>
</tr>
<tr>
<td>3 - DC#-uuuu DC1-Lond</td>
<td>3 - RG#-uuuu RG1-Euro</td>
</tr>
<tr>
<td>4 - uuuu-DC# Lond-DC1</td>
<td>4 - uuuu-RG# Euro-RG1</td>
</tr>
</tbody>
</table>

---

**Figure 68** Define GDDR User Labels panel

This panel accepts user-specified names for the GDDR Complex, Regions, and Site list.

1. To proceed, enter the following values:

   • **GDDR Complex name**
     A name assigned by the user to the entire complex of C-Systems and GDDR managed systems, using the following guidelines:
     - Format: 8 characters, alphanumeric
     - If no name is specified, the default is 'GDDRPLEX'

   • **User site label**
     Provide site names, using the following guidelines:
• Length: 1-7 characters, alphanumeric
• If no names are specified, Format '0' is assumed; sites will be displayed as DC1 and DC3

◆ User site label format
Specify the number of the template which will determine how site user labels will be shown in panels and messages.

◆ User region label
Provide region names, using the following guidelines:
• Length: 1-7 characters, alphanumeric
• If no names are specified, Format '0' is assumed; Regions will be displayed as RG1 and RG2.

◆ User region label format
Specify the number of the template which will determine how region user labels will be shown in panels and messages.

2. If you made any changes, type **Save** on the command line and press **Enter**.
3. Press **F6** to return to the Parameter Management Options Menu.

**Option V: Validate GDDR Parameter Set**

GDDR parameter validation processing determines that your proposed changes to parameter values are consistent and relationally correct. The validation operations are similar to the activation operations described in “Option A: Activate GDDR Parameter Set” on page 183, but do not update any GDDR variables.

When you specify option **V** in the Parameter Management Options Menu shown in Figure 9 on page 110, the Validate GDDR Parameter Set panel appears:

```
--- GDDR - Validate GDDR Parameter Set ---

Options below in effect for VALIDATION

Specify Options for this Parameter Set Validation :

Validation in Foreground or Background : FORE    (FORE,BACK)
   Specify GDDR Parameter Load Type : _______   (PARTIAL,FULL)
   Specify GDDR State Variables Action : ________ (RESET, ASIS, NOUPDATE)
   Propagate to Other C-systems : ___         (YES/NO/TRY)

Issue one of the following commands:

SAVE : Save options above as the default for your userid
CANCEL : <F3> return to the GDDR Parameter Management Options Menu
VAL : Proceed with GDDR Parameter Validation using options above
```

**Figure 69** Validate GDDR Parameter Set panel

Complete the options listed in this panel to validate your parameter set.

1. View, define, or modify the panel fields as follows:
   • Validation in Foreground or Background
Using GDDR Online Facilities

- FORE: Processing occurs in TSO foreground.
- BACK: Processing occurs in background as a batch job. A panel is displayed containing the jobcard and the PROCLIB and SKELETON libraries previously specified in your personal GDDR ISPF Profile.

Modify these settings if needed and then press **Enter**.

- **Specify GDDR Parameter Load Type**
  - PARTIAL: Validate only the changed work-pds members. There is one work-pds member per parameter definition panel. Validation does not include contents of other members.
    
    **Note:** Partial validation is not currently supported to implement a change in the names or the number of managed systems or LPARs.
  - FULL: Validate the entire set of GDDR parameters. This is the recommended setting.

- **Specify GDDR State Variables Action**
  - RESET: Reset all state variables to a neutral (non-error) state. This is the recommended setting.
  - ASIS: Set all state variables to the values found in the input.
  - NOUPDATE: Do not update any state variables. This option is invalid with Parameter Load type "FULL".

- **Propagate to Other C-systems**
  - YES: Send the updates to other C-systems. If any other C-system cannot be contacted, the action will fail. This is the recommended setting.
  - NO: Do not send the updates to other C-systems.
  - TRY: Try to send the updates to other C-systems, but if they cannot be contacted, continue.

2. After specifying the requested options, type the **Save** command to save your validation options to your TSO profile.

3. Type **VAL** to validate the proposed parameter definitions, or **CANCEL** to return to the GDDR Parameter Management Options Menu.
   
   If Background mode is selected, the following confirmation message displays:
   
   **GDDI116I** Job for Parameter Validation submitted successfully

4. Following a successful validation, press **F3** to return to the Parameter Management Options Menu.
Option A: Activate GDDR Parameter Set

To complete the parameter definition and load process, you need to activate the GDDR parameter set.

**IMPORTANT**

Parameters required by the GDDR Event Monitor (EVM) and GDDR Heartbeat Monitor may be changed during parameter activate processing. For this reason, GDDRMAIN will stop the Event Monitor and Heartbeat Monitor if they are not already stopped before you perform a GDDR parameter set activation. If they were started before the parameter site activation, GDDRMAIN will restart the Event and Heartbeat Monitors after the activation completes. Refer to , “,” for information about how GDDRMAIN controls the stop/start of the Event and Heartbeat Monitors.

The parameter set activation process performs the following actions:

- If a Planned or Unplanned script is running, then only an Activate of type Partial is allowed, and only if the script job is not currently running.
- Locks variables for integrity and consistency on local system.
- Performs and implicit backup and validate before activation.
- Updates variables and propagates them to other C-systems, depending on the Activate panel setting.
- Performs an implicit backup after activation.
- Clears the command queue if requested.
- Releases the variables.
- Copies the parameter work dataset to the last activated parameter dataset.

**IMPORTANT**

If you are running the parameter set activate batch job with CONSISTENCY(ENFORCE) and you cancel it after it has started the load, then you may need to disable the dynamic exit called GDDREXG1GDDR. The same is true for a parameter backup job if you are running with any CONSISTENCY option other than IGNORE. “Dynamic exits” on page 99 describes how to manually disable the GDDREXG1GDDR dynamic exit.
Specify option A in the Parameter Management Options Menu shown in Figure 9 on page 110. The Activate GDDR Parameter Set panel appears:

--- GDDR - Activate GDDR Parameter Set ---

Option ===>

Specify Options for this Parameter Set Activation:

Option: BACK

Specify GDDR Parameter Load Type:

Option: FULL

Clear the GDDR Command Queue:

Option: YES

Specify GDDR State Variables Action:

Option: RESET

Propagate to Other C-systems:

Option: YES

Enforce consistency:

Option: RETRY=5

Ignore Backup Failure:

Option: NO

Issue one of the following commands:

SAVE: Save options above as the default for your userid
CANCEL: <F3> return to the GDDR Parameter Management Options Menu
ACT: Proceed with GDDR Parameter Activation using options above

---

Figure 70 Activate GDDR Parameter Set panel

Complete the options listed in this panel to activate your parameter set.

1. View, define, or modify the panel fields as follows:

   • Validation in Foreground or Background
     - FORE: Processing occurs in TSO foreground. Activation processing messages are displayed in the Parameter Load Activation status pop-up panels shown in Figure 71 and Figure 72 on page 187.
     - BACK: Processing occurs in background as a batch job. This is the recommended setting. A panel is displayed containing the jobcard and the PROCLIB and SKELETON libraries previously specified in your personal GDDR ISPF Profile. Modify these settings if needed and then press Enter.

   • Specify GDDR Parameter Load Type
     - PARTIAL: Activates only the changed work-pds members. There is one work-pds member per parameter definition panel. Activation does not include contents of other members.
       \[\text{Note: Partial validation is not currently supported to implement a change in the names or the number of managed systems or LPARs.}\]
     - FULL: Deletes all global variables and activates the entire GDDR parameter set. This is the recommended setting.

   • Clear the GDDR Command Queue
     - YES: The command queue is cleared if the parameter load is successful. Use this choice for a full parameter load. This is the recommended setting.
     - NO: The command queue is not cleared. This choice may be appropriate for a partial parameter load.
Option M: Maintenance—GDDR Setup and Maintenance

Using GDDR Online Facilities

Note: Clearing the command queue is attempted on all C-Systems if Propagate to Other C-Systems option is ‘Yes’ or ‘Try’ and all C-Systems are in communication with one another. If the choice is to clear the command queue, but communication between the C-Systems is temporarily interrupted, please use the "Manage GDDR Internal Command Queue" panel shown in Figure 75 on page 189 to clear it on the other C-systems, or use the GDDRCLRQ job provided in hlq.GDDRvrm.SAMPLIB.

If no script is in progress, the recommendation is to clear the command queue. If a script is in progress, please consult EMC GDDR Solution Support.

• Specify GDDR State Variables Action
  – RESET: Reset all state variables to a neutral (non-error) state. This is the recommended setting.
  – ASIS: Set all state variables to the values found in the input.
  – NOUPDATE: Do not update any state variables. This option is invalid with Parameter Load type "FULL”.

• Propagate to Other C-systems
  – YES: Send the updates to other C-systems. If any other C-system cannot be contacted, the action will fail. This is the recommended setting.
  – NO: Do not send the updates to other C-systems.
  – TRY: Try to send the updates to other C-systems, but if they cannot be contacted, continue.

• Enforce consistency
  – YES: Any updates to global variables other than by parameter activate will be blocked while the backup or parameter activate is in progress. This is the recommended setting.
  – NO: Outside updates to globals will be ignored.
  – RETRY (1-5): If an outside update occurs while parameter backup is running, the backup will be retried the indicated number of times.

• Ignore Backup Failure
  – YES: If the parameter backup which is done before the parameter load fails, the parameter load will continue.
  – NO: If the parameter backup which is done before the parameter load fails, the parameter load will not be done. This is the recommended setting.

2. After specifying the requested options, type the SAVE command to save your activation options to your TSO profile.

3. Type ACT to activate the proposed parameter definitions, or CANCEL to return to the GDDR Parameter Management Options Menu.

If you already have started an activation job that did not complete yet, or was cancelled, aborted, or never was run, you will not be able to submit another job (see the message below). In this case you must either wait for the activation job to complete, or you can issue either the ACT FORCE command to force a resubmission of
a new activation job or issue ACT RESET to reset the status (if the job was cancelled or aborted or was never run). ACT RESET does not submit another activation job, but will allow you to edit parameters again once you exit the activation panel (parameter editing is not allowed while an activation job is in progress).

<table>
<thead>
<tr>
<th>Command ====&gt; ACT</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDI387I Parm activation job started (Job GDDRPACT J0003283)</td>
</tr>
</tbody>
</table>

Options below in effect for ACTIVATION

Specify Options for this Parameter Set Activation:

| Activation in Foreground or Background: BACK (FORE,BACK) |
| Specify GDDR Parameter Load Type: FULL (PARTIAL,FULL) |
| Clear the GDDR Command Queue? NO (YES/NO) |
| Specify GDDR State Variables Action: ASIS (RESET, ASIS, NOUPDATE) |
| Propagate to Other C-systems: YES (YES/NO/TRY) |
| Enforce consistency: RETRY=5 (YES/NO/RETRY=1-5) |
| Ignore Backup Failure: NO (YES/NO) |

Issue one of the following commands:

```
+------------------------------------------------------------------------------+
| GDDI387E Parm activation job started (Job GDDRPACT J0003283) - Use ACT FORCE |
| to start another activation                                                   |
+------------------------------------------------------------------------------+
```

4. If Background activation mode is selected, the following confirmation message displays:

GDDII16I Job for Parameter Activation submitted successfully
5. If Foreground activation mode is selected, activation processing messages are displayed in the Parameter Load Activation status pop-up panels:

![Figure 71 Parameter Load Activation status panel 1 of 2](image1)

```
Command ===> ACT

Options below in effect for ACTIVATION

+--------------------- GDDR - Parameter Load Activation ----------------------
| Elapsed 0:43           JABCDE1.GDDR500.PARMS.WORK           Time 18:54:05 |

Starting GDDROVRL
GDDR721I GDDR Starting GDDR Global Variable Update
GDDP308I Parmload running on system SYS1, 4 May 2011 18:53:22
GDDP308I Restore type = FULL, State variables = ASIS
GDDP308I Update - Y, Propagate - N
GDDP308I Consistency = ENFORCE
GDDP310I GLOBAL.GDDR.JVAL.SYS3.USERID
GDDP310I =GDDR

*** Please wait ***

537 output lines Date 2011/05/04

CANCEL : <F3> return to the GDDR Parameter Management Options Menu
ACT : Proceed with GDDR Parameter Activation using options above
```

![Figure 72 Parameter Load Activation status panel 2 of 2](image2)

```
Command ===> ACT

Options below in effect for ACTIVATION

+--------------------- GDDR - Parameter Load Activation ----------------------
| Elapsed 15:53          JABCDE1.GDDR500.PARMS.WORK           Time 19:09:09 |

Removing 7 globals, added 1 globals
GDDP355C EVM/HBM found active at site DC2; parmload cancelled
GDDP355C EVM/HBM found active at site DC3; parmload cancelled
GDDR639I GDDR Completed GDDR Global Variable Update with rc 8
GDDROVRL ended - RC=8
Redoing backup global variable from backup
Backup global variable from backup redone
Foreground activation failed - GDDROVRL RC=8

*** Please press Enter or PF3 key to continue ***

2500 output lines Date 2011/05/04

CANCEL : <F3> return to the GDDR Parameter Management Options Menu
ACT : Proceed with GDDR Parameter Activation using options above
```

6. Press **Enter** or **PF3** to return to the Activate GDDR Parameter Set panel.

The following message displays:

```
GDDI378E Parm activation failed, return code = 8 - Use VIEW command to see log
```

Type the **VIEW** command at the command prompt to review the messages describing activation processing.
Option D: Message, Debug and Trace Options

When you specify option D in the Setup and Maintenance menu shown in Figure 8 on page 109, the following panel appears:

```
-------- GDDR - Set Output Message Levels By Program -------
Option ===>  Scroll ===> CSR
This panel shows the message, debug and trace output levels in effect for user shown. Levels on each line apply to program on that line only, while levels in parentheses are defaults applying to any program not found in the list.

You may change the defaults or the levels for specific programs by overtyping.
Use ADD to add a new program to the list with initial output levels.
Press <F3> to save changes and return to previous panel
Press <F1> for a complete description of available actions on this panel

Program   Msg ( 1 )  Debug ( 0 )  Trace ( 0 )  For userid:     JABCD1
-  --------  ---        -----        -----        HMC Simulation? N  ( Y or N )
-  --------  ---        -----        -----        BCPii Tracing?  N  ( Y or N )

******************************* Bottom of data ********************************
```

Figure 73  Set Output Message Levels by Program panel

This panel enables you to individually customize message, debug, and trace settings for a selected Userid and GDDR module.

◆ HMC Simulation

Performs operations up to the point of the actual submit of BCPii interface functions from the Perform HMC LPAR Actions panel or from within a script. The BCPii interface functions exploited by GDDR are LIST, QUERY, COMMAND, SET, CONNECT and DISCONNECT.

◆ BCPii Tracing

When set to 'Y', detailed BCPii diagnostic information will be written to the primary sysout dataset for the GDDRWORK address space on the appropriate C-System. Keep in mind this might not be the C-System on which the original task was initiated (either via scripting, batch work, or the ISPF interface). This is the system upon which the task is executed. This can be very useful when trying to diagnose a problem with HMC management.

The defaults for these options are the settings EMC recommends for your production environment. You may be asked to make changes to the defaults if diagnostic information is needed as a result of a question or problem.
Use the **ADD** command to make changes to the default settings. The following panel appears:

```
-------------- GDDR - Add Program to MsgLevel/Debug/Trace List -----
Command ===>
               Program ===> ________       MsgLevel ===> 1
               Debug Level ===> 0
               Trace Level ===> 0

Enter program name (required)
You may overtype default message, debug and trace levels
Press <Enter> when ready to add new program to the list and return
Press <F3> to return without adding a program to the MDT list
```

**Figure 74 Add Program to MsgLevel/Debug/Trace List panel**

This panel allows the program names provided by EMC Customer Service to be specified with the requested message level, debug, or trace flags.

If the program name is less than 8 characters, extend the name with "$" signs up to a length of 8 characters.

**Option Q: Manage GDDR Internal Command Queue**

When you specify option **Q** in the Setup and Maintenance menu shown in **Figure 8 on page 109**, the following panel appears:

```
-------------- GDDR - Manage Internal Command Queue -- Row 1 to 2 of 2
Option ===> Scroll ===> CSR
WARNING: Do not use unless instructed to do so by EMC Support
Total elements: 2    Active elements: 0    Maximum active: 35
Entries listed in order by script name (A)
Press <F3> to return to the GDDR Setup and Maintenance Menu

Sel  No  RetCode  Script                  Created                Updated
---  --  -------  ------------------------  ---------------------
  1  152      GDDRPA07  0530201106023291   0530201110255782
  SC VOL,LCL(BA14,18),HSWAP(FORCE,STAR,GDDR),ALL
  2  152      GDDRPA07  0530201106023320   0530201110255163
  SC VOL,LCL(CC12,20),HSWAP(FORCE,STAR,GDDR),ALL
```

**Figure 75 Manage GDDR Internal Command Queue panel**

**WARNING**

The GDDR Manage Internal Command Queue panel provides the capability to alter GDDR processing. EMC advises against use of this panel unless specifically directed by EMC GDDR Customer Support.

The following command line commands are available:

- **Refresh** — Display from the actual command queue elements
- **Clear Queue** — Clears the entire queue
Using GDDR Online Facilities

- **SORT NUM D[A]** — Sort by command number, Ascending [Descending]
- **SORT SCR A[D]** — Sort by script name, Ascending [Descending]
- **SORT CRE A[D]** — Sort by date/time created, Ascending [Descending]
- **SORT UPD A[D]** — Sort by last update date/time, Ascending [Descending]
- **SORT QUE A[D]** — Sort by queue order, Ascending [Descending]

The initial sort order is Command number, Descending.

The following message is issued if the command is not valid: GDDI041E Invalid sort key *sort-key*.

The display is kept in sync with actions; there is no need to refresh the display after a delete.

**Option H: Perform HMC Discovery**

To discover HMC objects accessible to GDDR, specify option **H** from the Setup and Maintenance menu shown in Figure 8 on page 109. A 'please wait' pop-up dialog such as the following appears and HMC object discovery is activated:

```
---------------------- GDDR - Setup and Maintenance Menu ----------------------
Option ===>
P arms   Manage GDDR Parameters                           This System: SYS1
D ebug   Message, Debug and Trace options                This Site: DC1
Q uue    Manage GDDR Internal Command Queue             Master-C: SYS1
H MC     Perform HMC Discovery                           Primary Site: DC1
R efresh Refresh GDDR Message Table                      Primary DASD: DC1
S ystem  Man +------- Discovering HMC Objects ---------+
                   |                                       | omation: ON
                   |   Discovering HMC objects at site DC1   |  script: None
                   |                     *** PLEASE WAIT *** |  script: None
                   |                                       |  script: None
                   +-----------------------------------------+

EMC Geographically Dispersed Disaster Restart  05.00.00
Copyright © 2007-2016 EMC Corporation. All rights reserved.

Select an option and press <Enter>
Press <F3> to return to the GDDR Primary Options Menu
```

**Figure 76** HMC object discovery panel
When the discovery operation completes, an HMC Discovery Results panel containing a scrollable display such as the following appears:

```
GDDR STARTING DISCOVER HMC OBJECTS
GDDR HMC COMMUNICATION WITH DC1 USING BCPII
GDDR MANAGING THE FOLLOWING LPAR-SYSTEM PAIRS FOR SITE DC1
> IBM390PS.Q3,ZOSEQ311:Q311
> IBM390PS.C,ZOSEQ11:T101
>
> GDDR DISCOVERY INFORMATION FOR CPC : IBM390PS.Q3
> GDDR CURRENT ACTIVATION PROFILE : DEFAULT
> GDDR LAST USED ACTIVATION PROFILE : DEFAULT
> GDDR CPU MODEL : E10
> GDDR CPU TYPE : 2098
> GDDR CPU SERIAL NUMBER : 0000200F3FA4
> GDDR # GENERAL PURPOSE PROCESSORS : 00000005
> GDDR # SERVICE ASSIST PROCESSORS : 00000002
> GDDR # INTEGRATED FACILITY FOR APPLS : 00000000
> GDDR # INTEGRATED FACILITY FOR LINUX : 00000000
> GDDR # INTERNAL COUPLING FACILITY : 00000000
> GDDR # INTEGRATED INFORMATION PROCESSORS : 00000000
```

Figure 77  HMC Discovery Results panel

When you are finished examining the results of the discovery operation, press F3 to return to the Setup and Maintenance menu shown in Figure 8 on page 109.

Option R: Refresh GDDR Message Table

When you specify option R in the Setup and Maintenance menu shown in Figure 8 on page 109, the GDDRMGT table is refreshed and the message 'GDDRMGT Table refreshed' displays in the panel:

```
-------------------------- GDDR - Setup and Maintenance Menu --------------------------
Option --->
    P arms Manage GDDR Parameters                      This System: SYS1
    D ebug Message, Debug and Trace options              This Site: DC1
    Q ueue Manage GDDR Internal Command Queue            Master-C: SYS1
    H MC Perform HMC Discovery                          Primary Site: DC1
    R efresh Refresh GDDR Message Table                   Primary DASD: DC1
    S ystem Manage GDDR System variables

Automation: ON
Planned script: None
Unplanned script: None
```

Figure 78  Message table refresh indicator
Note: After applying maintenance to the GDDR software, in many cases it will be necessary to refresh GDDRMTXT, which is managed via GDDRMAIN. GDDRMTXT is refreshed daily at midnight, or whenever GDDRMAIN is restarted.

Option S: Manage GDDR System Variables

GDDR stores parameters that describe the environment as global GDDR System Variables. With the exception of this Manage GDDR System Variables panel, updating of GDDR System Variables is under the exclusive control of GDDR.

⚠️ WARNING

The GDDR Manage GDDR System Variables panel provides the capability to alter GDDR processing. EMC advises against use of this panel unless specifically directed by EMC GDDR Solution Support. Changes done on this panel remain local and they are not propagated to other C-Systems.
To view and change the values that are used by GDDR, specify option S in the Setup and Maintenance menu shown in Figure 8 on page 109. A Manage GDDR System Variables panel similar to the following displays:

![Manage GDDR System Variables panel](image)

Figure 79  Manage GDDR System Variables panel

View, define, or modify the panel fields as follows:

- **Global variable level**
  
  Specifies the name of the global variable, including the stem name of 'GLOBAL.GDDR'.

- **Add**
  
  Type `Add` in the Command line at the top of the panel to display the Create GDDR System Variable panel. This panel allows you to create new variables under the direction of EMC GDDR Solution Support.
Using GDDR Online Facilities

- **Node**
  Global variables may have a variable number of subnodes. The Node field displays information about subnodes, such as the number of subnodes and for the lowest level nodes, their value.

- **Actn**
  For each node listed in the Node field, you can specify the following line commands in the Action field:

  - **S—Select**
    If there are subnodes to the specified node, specifying the Select command shows the next level subnode.

  - **E—Edit**
    Allows the variable at the existing level to be changed.

  - **V—View**
    Displays information about the creation, update, and last access for the global variable.

  - **Z—Delete level**
    Deletes all variables at or under the level shown.

  - **D—Delete item**
    Deletes just the variable at the existing level.

### Option T: Transfer Master C-System

Selecting the Transfer Master C-System option returns the following panel:

```
Command ===>
Press <F3> to return to the Setup and Maintenance panel
Select a row to transfer the Master C-System to that site/system.

<table>
<thead>
<tr>
<th>CMD</th>
<th>Site</th>
<th>C-System</th>
<th>Link</th>
<th>MHB</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>DC1</td>
<td>SYS1</td>
<td>0</td>
<td>0</td>
<td>Current and Recommended Master, <em>Here</em></td>
</tr>
<tr>
<td>_</td>
<td>DC3</td>
<td>SYS3</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Figure 80  Transfer Master C-System panel

Based on the current GDDR parameters, site DC1 is the current master C-System and is the recommended site for the master C-System based on the parameters. Note that *Here* indicates the current local system.

No action is recommended. Press F3 to return to the Setup and Maintenance panel.
Option G: GDDR Config—View GDDR Configuration

Specify option G in the Primary Options menu (on page 106) to view your configuration settings.

**Note:** In the View GDDR Configuration panel, the RG1 and RG2 region names and the DC1 and DC3 site names may reflect user-specified names for the GDDR Complex, Regions, and Site list as defined in the Define GDDR User Labels panel. See “Option L: GDDR User Labels” on page 180 for more details.

The following panel appears:

```plaintext
-------------- GDDR - View GDDR Configuration for GDDR_50 --------------
Command ===> 
GDDR complex name:  GDDR_50
Regions:  RG1  RG2
Sites:    DC1, DC3
Features: SRDF/A
          No DLM support, No FBA devices

Region:    Site:    C-System:
RG1        DC1      GA2LB148
RG2        DC2      GA2LB29

Press <F3> to return to the previous menu
```

*Figure 81 View GDDR Configuration panel*
Option C: Checkup—Perform Pre-script Checkup

Specify Option C in the Primary Options menu (on page 106) to invoke the GDDR on-demand Health Check. A panel similar to the following displays the items that are validated.

```
------------------- GDDR - Perform Health Check -------------------  Row 1 to 6 of 6
Command ===>       Scroll ===> CSR
Primary Cmnds : SYSTEMS, GATEK, CONFIG, DLM SET|RESET, (PF1 for more info)
SORT ASC|DESC SITE|SYSTEM|TYPE|GDDRMAIN|HMC|DATE|TIME
Active Events:  C>> CAX CGT DLM MSC NCX SRA
Degraded Mode:  YES Consistency: Inconsistent Star-HA: 0
DLM Abort: 0
Command Queue: Status Job Count Active Free Max
-------- -------- ------ ------ ------ -----
Empty      0       0      0      8
--------------------- System Communication Status ---------------------
Sel Site  Name  GDDRMAIN HMC Date and Time of
---  ------- ---- --------- ---------
  DC2    GDDRDEV1 CSYS Active n/a 12/04/15 17:08:21
  DC3    CLAYB035 PSYS Unknown **None** **None**
  DC3    DVTL149  CSYS Unknown **None** **None**
  DC1    CLAYB033 PSYS Unknown **None** **None**
  DC1    DVTL1161 CSYS Unknown **None** **None**
  DC2    CLAYB034 PSYS Unknown **None** **None**
```

**Figure 82** Perform Health Check panel

The Perform Health Check panel provides the following functions:

- **Primary Commands**
  - You can enter the following primary commands on the command line:
    - **SYSTEMS**
      Refreshes the communication status of all indicated systems.
    - **GATEK <parms>**
      Validates gatekeepers, with the following optional parameters:
      - **Auto_UnBoX** - Automatically unbox devices.
      - **PRompt** - Prompt before unboxing a device.
      - **SIMulate** - Simulate unboxing of devices without actually unboxing.
      - **FoRCe** - Force unboxing of devices when normally not allowed.
      - **DeBuG** - Enable debugging.
      - **SITE=<site>** - Limit scope to gatekeepers at the specified site.
      - **CTRL=<serial>** - Limit scope to gatekeepers on the specified controller.
    - **CONFIG <parms>**
      Validates configuration, with the following optional parameters:
      - **DeBuG** - Enable debugging.
      - **MSCG=<mscgrp>** - The MSC group name to identify the configuration.
        Note: Required if GDDRMAIN is not available.
      - **CTRL=<serial>** - The controller serial number to use as a starting point.
        Note: Required if GDDRMAIN is not available.
• DLM SET or DLM RESET
  Sets or resets the DLM Abort flag.
• SORT ASC|DESC fieldname
  Performs an ascending or descending sort, where fieldname is either SITE, SYSTEM, TYPE, GDDRMAIN, HMC, DATE, or TIME.

◆ Active Events
  Indicates any exceptions to conditions monitored by the GDDR Event monitor. The following is a list of possible exceptions:
  • SRA: SRDF/A Link Down
  • CAX: Autoswap Event
  • CGD: ConGroup Disable
  • CGT: ConGroup Trip
  • CGV: ConGroup Verify In Progress
  • RDR: Regional Disaster
  • NCX: ConGroup lost AutoSwap protection
  • ECA: Enginuity Consistency Assist - secondary
  • STR: STAR Recovery not available
  • MSF: Unable to communicate between C-Systems
  • MHB: Missing Heartbeat
  • MHB (MIM): Missing Heartbeat MIM
  • C>>: C-ahead
  • LDR: Local Disaster
  • MXS: CF Signal Connectivity lost
  • CFG: Defined vs. discovered DASD discrepancy
  • MST: Master C-system conflict

◆ Degraded Mode
  This is a YES|NO field. YES indicates that one or more of the following conditions are met:
  • A C-System has a Missing Heartbeat event (MHB). This will be visible in the “Active Events” field.
  • GDDRMAIN is not running on a C-System (will be shown as Inactive under GDDRMAIN Status).
  • A GDDRMAIN subtask or worker is not running on a C-System (will be shown as Degraded under GDDRMAIN Status).
  • The MPARM results are inconsistent.
Note: In certain circumstances, Degraded Mode may indicate NO, but the System Communication Status shows GDDRMAIN Status as Inactive or Degraded for one or more managed systems (PSYS).

- **Consistency**
  Shows if there are any discrepancies between GDDR systems in regard to GDDMPARM or GDDR subsystem.
  - Consistent: All GDDR systems are using the same GDDMPARM and the same GDDR subsystem.
  - Inconsistent: GDDMPARM is in use or the GDDR subsystem is not consistent for all GDDR systems. Inconsistency will cause Degraded Mode to be set.

- **Command Queue**
  Indicates the current command queue status and size. GDDR uses an internal command queue to perform operations.

- **System Communication Status**
  - Line Commands
    - S (select) queries the system and replaces the initial Status value of 'Unknown' with one of three values: Active, Inactive, or Degraded.
    - R (refresh) redrives the Status query. You can get updated status for all systems by entering the command SYSTEMS at the command prompt.
    - D (detail) displays the GDDRMAIN System Details panel containing information about the state of GDDRMAIN subtasks, dependent address spaces, and worker tasks. The GDDRMAIN System Details panel is shown in Figure 83 on page 200.
  - Home Site
    The Site ID of the site where the system normally runs.
  - System Name
    The System names are the C-Systems and managed systems defined in the GDDMPARM parameters.
  - Type
    - CSYS indicates a C-System
    - PSYS indicates a GDDR-managed production or test system
    - CF Indicates this is a system assigned to a Coupling Facility LPAR. The S and R line commands, as well as the SYSTEMS primary command will not attempt to verify GDDRMAIN status for such a system, but will check for HMC console communication connectivity errors.
• **GDDRMAIN Status**

GDDRMAIN communicates between C-Systems and the GDDR managed systems. This field shows the status of that communication. Initially, all systems except the C-System where you are will show a status of “Unknown”. Status values are as follows:

- **Active**
  
  Indicates that GDDRMAIN is up and fully functional.

- **Inactive**
  
  Indicates that GDDRMAIN on the system where this is running is unable to communicate with GDDRMAIN on the target system.

- **Degraded**
  
  Indicates that GDDRMAIN is up on the target system, but not all subtasks or workers are active. The conditions considered when Degraded Status is set are displayed in the System Details panel as shown in Figure 83 on page 200. You can display this panel by typing D (detail) in the Sel field associated with a system.

  Degraded status reflected by one or more inactive subtasks may be a transitory state, not a cause for concern. For example, the parameter Activate process requires that Event and Heartbeat Monitors be stopped. If the Activate option of ‘FULL’ is selected, then GDDRMAIN will stop Event and Heartbeat Monitors on all C-Systems, causing Degraded status to be reported for each C-System.

- **N/A**
  
  Indicates GDDRMAIN status is not applicable; shown for CF systems. There is no GDDRMAIN task installed on CF LPAR systems.

• **HMC Status**

HMC Status indicates the HMC communication status. HMC Status values are as follows:

- **Operating**
  
  Indicates HMC functions are supported.

- **Error**
  
  Indicates the HMC communication query failed.

- **Bypass**
  
  Indicates this system was assigned HMC Bypass status on the Define Managed LPARs panel during GDDR configuration. No communication is allowed to the HMC console for this system. The S and R line commands, as well as the SYSTEMS primary command, will not attempt communication with the HMC for HMC Bypass systems. However, GDDRMAIN status will be checked by these commands on HMC Bypass systems.

- **N/A**
  
  Displays for the C-System where you are.
GDDR uses the z/OS Hardware Management Console (HMC) to maintain and view its managed systems, issue operator commands, and to discover HMC objects.

- Date and Time of Status Check

The meaning of the date and time shown vary depending on the value of GDDRMAIN status. If status is Active or Degraded, the date and time reflect the last time that health check queried that system and received a response. If the status is Unknown or Inactive, the date and time specify the last time any communication was received.

Figure 83 shows the GDDRMAIN System Details panel. This panel provides GDDRMAIN status information which may identify conditions that cause a Degraded Mode of 'Yes' on the Perform Health Check panel.

The following fields are populated from the Perform Health Check panel.

- Site
- System
- Type
- Status Date
- GDDRMAIN Status
- HMC Status

The IP address field is populated from the GDDRMAIN parameters defined in GDDMPARM. The next three lines provide the status of GDDRMAIN subtasks and worker tasks. GDDRMAIN subtasks are either shown as active with 'Y' or inactive with 'N'. The workers are represented with the count of active worker tasks. The number of worker tasks is configured during the GDDRMAIN installation procedures as shown in “Allocate the parameter management datasets” on page 64.

GDDR-managed systems will show subtasks GVT, EVM, and HBM with 'N' and workers GDDWGV and GDDWXR with '0' if these subtasks and workers are not applicable to managed systems.

The GDDRMAIN System Details panel shows all subtasks for C-Systems as 'Y', except when a parameter Activate is performed. GDDRMAIN stops EVM and HBM when a parameter Activate function is performed. Further, when the Activate operands 'FULL' and Propagate
'Y' are selected, HBM and EVM will be stopped on all C-Systems. During this time, the Perform Health Check panel and the GDDRMAIN System Details panel show GDDRMAIN Status as Degraded. A GDDRMAIN Status of Degraded also triggers the GDDR Event monitoring mode of Degraded, as shown on the Perform Health Check panel under the Active Events field.

C-Systems must have at least one instance of all worker tasks active. If the worker manager encounters three unexpected worker task failures within a one minute period, the worker manager sets the MAX for that worker to '0', sets the GDDRMAIN Status to 'Degraded' and writes message GDDM150E (Worker worker-name has repeatedly failed, maximum reset to 0).

You must take action to resolve the cause of the failing worker task, and then change the MAX value for the task to the recommended non-zero value. Change the worker task MAX value by issuing F GDDRMAIN, WORKER[,name[,MIN=min|MAX=max]]. “GDDRMAIN console commands” on page 89 provides more information.

Note: You cannot set the worker MAX value to 0; at least one instance of each worker must be available.

Health Check monitoring

The GDDR Event monitor checks for the following storage and system environment events:

Table 17  Monitored events

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DLM</strong>: Missing DLm Heartbeat</td>
<td>One or more DLm ACP systems has failed to respond to a heartbeat request in a specified time window.</td>
</tr>
<tr>
<td><strong>ECA</strong>: Enginuity Consistency Assist</td>
<td>ConGroup has cleared ECA flags.</td>
</tr>
<tr>
<td><strong>LNK.DCmVDCn</strong>: RDF Link Down</td>
<td>The RDF link between the specified sites is unavailable.</td>
</tr>
<tr>
<td><strong>MHB</strong>: Missing Heartbeat</td>
<td>One or more C-Systems has failed to issue a heartbeat in a specific time window.</td>
</tr>
<tr>
<td><strong>MSC</strong>: SRDF/A Multi-Session Consistency</td>
<td>The EVM SRDF/A check found MSC inactive or not consistent, or found SRDF/A inactive or an MSC group is in an unexpected status as reported by GDDR818I. The recommended response is to review the status of SRDF/A groups. Activate SRDF/A if necessary, then activate MSC. The Event monitor monitors for successful MSC initialization, which then clears the MSC event. If the above action fails to resolve the issue, check GDDREVM SYSTSPRT, SCF log, and HC log for additional messages.</td>
</tr>
<tr>
<td><strong>MSF</strong>: GDDRMAIN COMM Link Down</td>
<td>Communication to the system is not available.</td>
</tr>
</tbody>
</table>
When the GDDR Event Monitor detects any of the events listed in Table 17 and a GDDR Planned script is not running, it issues the following message:

GDDS027E GDDR Error: Event full_event_id detected, value event_value

For example:

GDDS027E GDDR Error: Event GLOBAL.GDDR.DCN.Unplanned.C>> detected, value 1

Message GDDS027E is issued every EVM cycle for each unplanned event which was detected during this or any previous EVM cycle, for each previously set event which has not been cleared. Message GDDS027W indicates a transient cause, based on actions performed during Planned script processing.

### GDDR Event monitoring exception notification

Notification of Event monitoring or GDDRMAIN status exceptions can easily be implemented using third-party system management software products which perform event notification tasks.

Software state analysis messages that are candidates for user system automation triage or alert actions are described in Table 18.

#### Table 17 Monitored events

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
</table>
| RDF.DCm.DCn | RDF Directors offline A GDDR-managed RDF group defined by sitepair DCm.DCn has been found to have all directors offline. Look for the following messages:  
  - GDDR650W — Cause: The named director is offline, inoperable, or has no ports.  
  - GDDR649E — Cause: All directors for the named RDF group are either inoperable or offline or disconnected, to determine which directors or RDF groups are affected. |
| MST: Master C-System conflict | A master C-System conflict was detected. There is a discrepancy/disagreement between two or more sites regarding the name of the current master C-System. |
| RDR: Regional Disaster | DC1 outage has occurred. |
| SRA: SRDF/A Link Down | SRDF/A has been disabled. |

When the GDDR Event Monitor detects any of the events listed in Table 17 and a GDDR Planned script is not running, it issues the following message:

GDDS027E GDDR Error: Event full_event_id detected, value event_value

For example:

GDDS027E GDDR Error: Event GLOBAL.GDDR.DCN.Unplanned.C>> detected, value 1

Message GDDS027E is issued every EVM cycle for each unplanned event which was detected during this or any previous EVM cycle, for each previously set event which has not been cleared. Message GDDS027W indicates a transient cause, based on actions performed during Planned script processing.

### GDDR Event monitoring exception notification

Notification of Event monitoring or GDDRMAIN status exceptions can easily be implemented using third-party system management software products which perform event notification tasks.

Software state analysis messages that are candidates for user system automation triage or alert actions are described in Table 18.

#### Table 18 Software state analysis messages (1 of 5)

<table>
<thead>
<tr>
<th>Usage</th>
<th>Message</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event is Set</td>
<td>GDDS027E</td>
<td>The GDDR Event Monitor has found an event set during this or any previous EVM cycle which has not been cleared yet.</td>
</tr>
<tr>
<td></td>
<td>+GDDR786E</td>
<td>This message is issued by the GDDR Event Monitor, each time a GDDR event changes state. (OFF to ON, or ON to OFF). It is forwarded by GDDR to managed systems to facilitate custom automation on detection of events.</td>
</tr>
</tbody>
</table>
### Table 18  Software state analysis messages (2 of 5)

<table>
<thead>
<tr>
<th>Usage</th>
<th>Message</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRMAIN</td>
<td>GDDM014W</td>
<td>The storage used for global variables has exceeded the threshold (80%) of the capacity of the DIV.</td>
</tr>
<tr>
<td>Analysis</td>
<td>GDDM015E</td>
<td>An error has occurred writing to the DIV dataset. The message gives the return code and reason code from the DIV SAVE operation.</td>
</tr>
<tr>
<td></td>
<td>GDDM031E</td>
<td>The named Z/OS service returned an error. The message lists the return and reason codes.</td>
</tr>
<tr>
<td></td>
<td>GDDM058W</td>
<td>The named lock has been set for over 5 seconds. The lock may be &quot;stuck&quot;.</td>
</tr>
<tr>
<td></td>
<td>GDDM153I</td>
<td>During HBM initialization, a conflict in master C-System definition has been discovered. HBM stops EVM since it is also impacted by this condition.</td>
</tr>
<tr>
<td></td>
<td>GDDM161W</td>
<td>Degraded mode has been set on or off at the indicated system (which is the C-System at the named site). Previously issued messages should indicate why this action was taken.</td>
</tr>
<tr>
<td></td>
<td>GDDM162E</td>
<td>An attempt was made to set degraded mode on or off on the named system (which is the C-System for the named site). However, degraded mode could not be changed. This is generally because communication could not be accomplished with the named system because it is down or GDDRMAIN is not running.</td>
</tr>
<tr>
<td></td>
<td>GDDM195E</td>
<td>During MISC processing, a GDDMPARM CPC record was found that specified BCPII connectivity for a C-System to a CPC. An attempt to verify that connectivity failed with the specified return code information. Issued by GDDRMTIM.</td>
</tr>
<tr>
<td>SDDF Analysis</td>
<td>GDDP441I</td>
<td>The GDDR SDDF Session verification utility has completed a run of verifications with the RC indicated in the message. Investigate non-zero return codes.</td>
</tr>
<tr>
<td></td>
<td>GDDP448E</td>
<td>The GDDR SDDF Session verification utility found devices with invalid SDDF sessions. The message identifies the reason why a session is considered invalid. Alternatively, the utility found devices at risk for a Full PUSH or an unusually large differential at the next CreatePair or ResumePair differential.</td>
</tr>
<tr>
<td></td>
<td>GDDP449W</td>
<td>The GDDR SDDF Session verification utility found devices for which the SDDF Session time stamps did not change compared to the first run.</td>
</tr>
<tr>
<td></td>
<td>GDDP453E</td>
<td>The GDDR SDDF utility lists the number of devices that are considered to be at risk for a Full PUSH on the next CreatePair or ResumePair Differential operation.</td>
</tr>
</tbody>
</table>
### Table 18  Software state analysis messages (3 of 5)

<table>
<thead>
<tr>
<th>Usage</th>
<th>Message</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSC Analysis in the GDDR Event Monitor</td>
<td>GDDS202E</td>
<td>The ConGroup owner, MSC Primary Server, or MSC Secondary Server (as indicated in the message) is not the same on the C-Systems in the primary region.</td>
</tr>
<tr>
<td></td>
<td>GDDS203E</td>
<td>The ConGroup name reported by MSC does not match the ConGroup name which GDDR believes to be active.</td>
</tr>
<tr>
<td></td>
<td>GDDS204W</td>
<td>The MSC server is in an unexpected state (not PRIMARY or SECONDARY). The message shows the current role.</td>
</tr>
<tr>
<td></td>
<td>GDDS205E</td>
<td>The current SRDF state is not Star with AutoSwap.</td>
</tr>
<tr>
<td></td>
<td>GDDS207E</td>
<td>Neither $DC1 nor $DC2 are aware of the Primary or Secondary (as indicated) MSC Server.</td>
</tr>
<tr>
<td></td>
<td>GDDS221E</td>
<td>GDDS222E A “split brain” condition has been detected between sites in the primary region, based on the detection of the named event at one site but not at the other. If the condition is resolved, message GDDS222E will be issued.</td>
</tr>
<tr>
<td></td>
<td>GDDS222E</td>
<td></td>
</tr>
</tbody>
</table>
MSC Analysis in the GDDR Event Monitor

<table>
<thead>
<tr>
<th>Usage</th>
<th>Message</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDS225E</td>
<td></td>
<td>The GDDR Event Monitor set the MSC and/or STR event, because when reviewing MSC configuration data, it found an unexpected condition. GDDR and/or Mainframe Enablers software setup and configuration errors are most frequently the cause of this. Examples are: The ConGroup name defined to MSC Star does not match the one defined to GDDR The user defined a Concurrent SRDF/Star configuration to GDDR, but a Cascaded SRDF/Star configuration in Host Component parameters ConGroup is disabled in a Star configuration A Star configuration is defined to GDDR, but MSC reports MSC-mode In a pair of SQAR-associated groups, MSC has different associated group-names than those defined to GDDR In a pair of SQAR-associated groups, the SQARA and SQARB groups report a different Primary MSC Server SMFID Another common cause is that the MSC group has not been initialized sufficiently, or took an error during initialization.</td>
</tr>
<tr>
<td>GDDS227E</td>
<td>GDDS228E</td>
<td>The GDDR Event Monitor reports a delay in MSC SW activity for the named MSC group. The reported MSC Post Timestamp is older than 2 times MSC_CYCLE_TARGET seconds. Message GDDS228E reports a return to normal.</td>
</tr>
<tr>
<td>GDDS229E</td>
<td>GDDS230E</td>
<td>The GDDR Event Monitor reports a delay in MSC Cycle Switching for the named MSC group. The reported MSC Cycle TOD is older than X times MSC_CYCLE_TARGET seconds. Message GDDS230E reports a return to normal.</td>
</tr>
<tr>
<td>GDDS231E</td>
<td>GDDS232E</td>
<td>The GDDR Event Monitor reports a delay in MSC SDDF Resets for the named MSC group. The reported MSC SDDF RESET TOD is more than X MSC cycles ago. Message GDDS232E reports a return to normal.</td>
</tr>
<tr>
<td>GDDS233E</td>
<td>GDDS234E</td>
<td>The GDDR Event Monitor reports a delay in MSC SDDF Resets for the named MSC group. The reported MSC SDDF RESET TOD is more than X seconds ago. Message GDDS234E reports a return to normal.</td>
</tr>
<tr>
<td>GDDR818I</td>
<td></td>
<td>This message indicates that the state of the named MSC group has changed from an earlier recorded state. It is also issued the first time the MSC group state is recorded in the life of a GDDR address space.</td>
</tr>
</tbody>
</table>

Table 18  Software state analysis messages (4 of 5)
Table 18 Software state analysis messages (5 of 5)

<table>
<thead>
<tr>
<th>Usage</th>
<th>Message</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loss of DASD Access or Loss</td>
<td>GDDR850I</td>
<td>The event monitor has detected an unplanned swap in progress. It does not take action until the swap is complete.</td>
</tr>
<tr>
<td>of Site</td>
<td>GDDR852I</td>
<td>The event monitor has detected that the current secondary site is dead, which indicates a local disaster (LDR) occurred.</td>
</tr>
<tr>
<td></td>
<td>GDDR853I</td>
<td>The event monitor at the primary site has detected a local disaster (LDR), which indicates that recovery at the secondary site is no longer available.</td>
</tr>
<tr>
<td></td>
<td>GDDR854I</td>
<td>The event monitor at a site other than the primary site has detected a local disaster (LDR), which indicates that recovery at the secondary site is no longer available.</td>
</tr>
<tr>
<td></td>
<td>GDDR855I</td>
<td>An RDR event (regional disaster) has been detected. If this message occurs in the primary region, then the secondary region has suffered the disaster; if this message occurs in the secondary region, then the primary region has suffered the disaster and further messages and action should be expected to deal with the situation.</td>
</tr>
<tr>
<td>SRDF/A Monitoring</td>
<td>GDDR592I</td>
<td>A query (local or remote) has been done to determine if SRDF/A is active or not. The message shows the path and RDF group which were interrogated, as well as other RDF information. The status (active or not active) is shown. If the status is not active, the Cycle TOD shows the last time that SRDF/A was active on this RDF group. If MSC is not active on this RDF group, the Rcv Tag is shown as all dashes (-), and C-Ahead and Host Intervention Required are shown as N/A.</td>
</tr>
<tr>
<td></td>
<td>GDDR863W</td>
<td>SRDF/A was found in the named state (Active, Inactive or Suspended) for the named gatekeeper and RDF group.</td>
</tr>
<tr>
<td>RDF Links failure</td>
<td>GDD504E</td>
<td>An RDF Links failure event has been detected. The Event Monitor continues to check if other error conditions exist.</td>
</tr>
<tr>
<td>RDF Group Analysis</td>
<td>GDDR649E</td>
<td>The named RDF group on the indicated VMAX is either offline or has no functioning director. All directors associated with the RDF group are either dead or offline or disconnected.</td>
</tr>
<tr>
<td></td>
<td>GDDR650W</td>
<td>The named director on the indicated VMAX is not operational for the reason given. Reasons could be Dead, No ports, or Offline.</td>
</tr>
<tr>
<td></td>
<td>GDDR652E</td>
<td>The C-System at the named site is found to have no useable gatekeeper devices for the specified VMAX system, which is expected to be directly channel attached at the named site. The site-id of the VMAX system can be the same site as that of the C-System or in AutoSwap configurations, a different site in the same region.</td>
</tr>
<tr>
<td></td>
<td>GDDR653W</td>
<td>The C-System at the named site is found to have no useable connections to the specified VMAX system at a different site, which is not expected to be directly channel attached.</td>
</tr>
<tr>
<td></td>
<td>GDDR655E</td>
<td>The C-System at the named site was unable to update the SRDF Topology Map. The most likely cause for this is that the C-System at the named site has no useable gatekeepers for one or more VMAX systems at its own site. If that is the case, there will be additional symptoms:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The RDF event will be ON for each sitepair involving the named site.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GDDREVM at the Master C-System will set to Degraded Mode because there are no device workers started at the named site.</td>
</tr>
</tbody>
</table>
Additional pre-script environment checks

Before you initiate any planned swap scenarios, EMC recommends that you make the following manual checks:

- Validate that GDDR managed storage matches the RDF device population.
- Verify SRDF/A is active and cycling

The initial check is also performed automatically by EMC GDDR.

EMC recommends that you perform these checks manually before you run any GDDR swap scripts to make sure that everything is active. The checks are designed to avoid possible problems later.

The following sections discuss these checks.

Validate that GDDR managed storage matches the RDF device population

Select the Automated Configuration Check - DASD script from the Planned script section of the Select Script to run panel, as shown in “Option S: Scripts — Run GDDR Scripts” on page 208.

Review the GDDRPCCD script joblog for GDDP400E and GDDP420E error messages. Resolve discrepancies by revising the RDF.DEVICES, DLM.DEVICES and STDBCV parameters before proceeding.

Verify SRDF/A is active and cycling

Note: The GDDR Event Monitor will automatically verify that SRDF/A is running in configurations where this is applicable, and set SRA and MSC events based on its findings. This monitoring is done on a cyclic basis, under control of the “EVM SRDF/A Check Interval” GDDR tuning option as shown in “Option T: Tuning Values” on page 177.

Enter the following command from the GDDR Master C-System:

> SQ SRDFA , uuuu

where:

> Represents the SRDF/Host Component command character.

uuuu

Represents the gatekeeper UCB address for any controller at the primary DASD site.

IMPORTANT

The >SQ SRDFA, uuuu command displays data about the entire SRDF/A configuration in the VMAX unit. Use >SQ SRDFA,LCL or >SQ SRDFA,RMT to display data about the single RDF group.

Expected output

The following example shows the expected output of this operation.
What to look for
To ensure that SRDF/A is active and cycling, the following fields should all be set to Y:

◆ Secondary consistent
◆ MSC active
◆ Global consistency
◆ Star recovery available

The time since last cycle switch should generally be around the number of seconds specified in the Host Component initialization parameter, MSC_CYCLE_TARGET. A large number would indicate that SRDF/A has stopped cycling. Subsequent displays should also show the Capture and Transmit tags incrementing.

What to do if SRDF/A is not active and cycling
Take the following actions if any of the previous checks fail:

◆ Report the problem to your z/OS systems programming support group.
◆ Do not start the swap.

Option S: Scripts — Run GDDR Scripts
When you specify option S in the Primary Options Menu shown in  on page 106, the Select Script to Run panel displays:

The Select Script to Run panel is a dynamic display. It lists GDDR scripts that are eligible to run based on the current state of the configuration, while providing script generation status messages about events which affect script eligibility. The panel also specifies configuration information and scripts that are in progress, if any. Depending on the location of the primary site and primary DASD locations, the panel is customized to display valid choices. Scripts that you cannot use at your current site do not appear.

Eligible scripts are grouped in the panel by category:

◆ Planned actions
◆ Test actions
◆ Unplanned actions
◆ Resumption actions
◆ Special actions

DC1 and DC3 represent the current primary DASD site or current secondary DASD site. When these representations are shown in italic type in script titles, this indicates the values are interchangeable. The descriptions assume that DC1 is the Primary DASD site and Primary site at the beginning of the script.

In the Select Script to Run panel, the DC1 and DC3 Site names may reflect user-specified names for the sites as defined in the Define GDDR User Labels panel. See “Option L: GDDR User Labels” on page 180 for more details.
Controlling which scripts are displayed in the panel

Scripts that are not applicable for the assigned configuration are not displayed in the Scripts list. If a script is applicable for the assigned configuration, but not available due to a specific state not being met, a comment is posted next to the script in the Status section stating why the script is not available. Table 19, "Script generation status messages," on page 212 lists status messages that may appear. Figure 86 shows an example of a panel with unavailable scripts displayed:
Using GDDR Online Facilities

---

Command ==> Scroll ==> CSR
Enter S next to script to run and press <Enter>  
Current Master: PLASMA
Press <F3> to return to previous menu  
Primary Site: DC1
Press <F1> to view all available commands  
Primary DASD: DC1
Configuration: 3 SITES, SRDF/S SRDF/A STAR AUTOSWAP Automation: ON
Previous script: GDDRPA21 ran on LASER

Job GDDRPA21 (JOB04678) - 05/06/2014 10:54:26 >
Sel ----------------------- Scripts ---------------------- ----- Status ------

Automated Configuration Check - DASD........ GDDRPCCD
Reconfigure to cascaded SRDF.............. GDDRPAS2 *Degraded mode ON
Swap production from DC1 to DC2............ GDDRPAS2 *Degraded mode ON
Swap DASD from DC1 to DC2................. GDDRPAS21 *Degraded mode ON
Abandon Tertiary site (DC3).............. GDDRPAS60 *Degraded mode ON

--- Scripts for TEST Actions ---
Perform test IPL from BCVs at DC2............ GDD2P01A *Degraded mode ON
Perform test IPL from BCVs at DC3............ GDD2P01A *Degraded mode ON
Perform test IPL from R2s at DC2............. GDD2P03A *Degraded mode ON
Perform test IPL from R2s at DC3............. GDD2P03A *Degraded mode ON
Resume SRDF/A after test IPL at DC3........ GDDRPAS8 *Degraded mode ON
Recover after unplanned swap.............. GDDRPAS1 *CAX is OFF
Recover after loss of DC1 (LDR)............. GDD2P13A *Requires LDR event

--- Scripts for RESUME Actions ---
Resume SRDF/S replication after ConGroup trip GDDRPAS23
Resume SRDF/A in MSC mode to DC3........ GDDRPAS29
Resume SRDF/A (SRDF/Star AutoSwap) to DC3 ... GDDRPAS29
Reclaim Secondary site (DC2).............. GDDRPAS65 *Degraded mode ON
Reclaim Tertiary site (DC3)............... GDDRPAS65 *Degraded mode ON
--- Scripts for SPECIAL Actions ---
Restart primary MSC server at DC1........ GDDRPMASC *SRA is ON
Restart primary MSC server at DC2........ GDDRPMASC *SRA is ON
Transfer AutoSwap Owner to DC1............ GDDRPXAS *Degraded mode ON
Transfer AutoSwap Owner to DC2............ GDDRPXAS *Degraded mode ON
Global Variable Backup................... GDDRPGVB

--- End of Data ---

Figure 86  Select Script to Run panel — unavailable scripts displayed

When entering this panel, GDDR always shows all scripts applicable for the assigned configuration. You can control the scripts that appear in the panel by issuing commands at the prompt. To hide scripts that cannot be run because a required state has not been met, type Hide at the command prompt. Only scripts that can currently be run are displayed. Figure 87 shows the same panel with unavailable scripts hidden:
Running a script

To request that a script be run or rerun, type **S** next to the row describing the script you want to run and press **Enter**.

Displaying details

Use the Display Script Details command to return the program name for the script and the name of the script skeleton. Type **D** next to the row describing the script and press **Enter**.

Field descriptions

The following fields are provided on the Select Script to Run panel:

**Configuration**

This field indicates the features of your configuration that GDDR takes into account when determining which scripts are eligible to run and hence which are present in the list of scripts displayed on your panel.

**Scripts listed**

For a header row, the scripts field contains a description of the category of scripts which follow. For a script row, this field contains a description of the script.

**Status**

This field displays script generation status events and dependencies. Each time you select a script to be submitted from the Select Script to Run panel, the script prerequisites are compared to the state of the environment. In cases when conflicts are identified or event prerequisites have not been satisfied, the messages shown in **Table 19** are returned to the Status field.

If a Hide command has been issued, unavailable scripts with associated status descriptions do not appear.
Note: Refer to Table 17, “Monitored events,” on page 201 for all event descriptions and related actions.

Table 19 Script generation status messages (1 of 2)

<table>
<thead>
<tr>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only after P17A</td>
<td>Valid only after prerequisite GDD2P17A is complete</td>
</tr>
<tr>
<td>Only Sec.DASD site</td>
<td>Valid only at the secondary DASD site</td>
</tr>
<tr>
<td>Only Pri.DASD site</td>
<td>Valid only at the primary DASD site</td>
</tr>
<tr>
<td>Only on Pri. MSC</td>
<td>Valid only at the primary MSC server site</td>
</tr>
<tr>
<td>LDR is OFF</td>
<td>Requires the LDR event</td>
</tr>
<tr>
<td>Invalid for state</td>
<td>Not valid for current state</td>
</tr>
<tr>
<td>SRA is ON</td>
<td>Not allowed when there is an SRA event</td>
</tr>
<tr>
<td>RDF is ON</td>
<td>Not allowed when there is an RDF event</td>
</tr>
<tr>
<td>MSC is ON</td>
<td>Not allowed when there is an MSC event</td>
</tr>
<tr>
<td>STR event is on</td>
<td>Not allowed when there is an STR event</td>
</tr>
<tr>
<td>CFG event is on</td>
<td>Not allowed when there is a CFG event</td>
</tr>
<tr>
<td>Requires LDR @ DC1</td>
<td>Requires an LDR event at DC1</td>
</tr>
<tr>
<td>Data precedence</td>
<td>Not allowed because the site with the most recent data is not appropriate</td>
</tr>
<tr>
<td>Needs prot. pref.</td>
<td>Requires user-specified protection preference</td>
</tr>
<tr>
<td>DLM is ON</td>
<td>Not allowed when there is a DLM event</td>
</tr>
<tr>
<td>SRA is OFF</td>
<td>Only allowed if there is an SRA event</td>
</tr>
<tr>
<td>LDR is on for DC1</td>
<td>Not allowed if there is an LDR event for DC1</td>
</tr>
<tr>
<td>Sec.DASD site only</td>
<td>Valid only at the secondary DASD site</td>
</tr>
<tr>
<td>Star-HA state inv.</td>
<td>The SRDF/Star with HA configuration is in an invalid state</td>
</tr>
<tr>
<td>CGT is ON</td>
<td>Not allowed when there is a CGT event</td>
</tr>
<tr>
<td>Only on Sec. MSC</td>
<td>Valid only at the secondary MSC server site</td>
</tr>
<tr>
<td>Only at DC3</td>
<td>Valid only at DC3</td>
</tr>
<tr>
<td>Not at $DC3</td>
<td>Not valid at DC3</td>
</tr>
<tr>
<td>MHB ON @ Sec. site</td>
<td>Missing Heartbeat Event is on at the secondary site</td>
</tr>
<tr>
<td>Unpl. scr. running</td>
<td>An unplanned script is running</td>
</tr>
<tr>
<td>CG Verify running</td>
<td>The GDDR Event Monitor has started a ConGroup Verify process</td>
</tr>
<tr>
<td>Degraded mode ON</td>
<td>GDDRMAIN Degraded mode is set. Refer to the System Details panel in Option C: Checkup Perform Pre-script checkup for assistance.</td>
</tr>
<tr>
<td>MSF is ON</td>
<td>Not valid if there is an MSF event</td>
</tr>
<tr>
<td>Plan. scr. running</td>
<td>A planned script is running</td>
</tr>
</tbody>
</table>
Table 19  Script generation status messages (2 of 2)

<table>
<thead>
<tr>
<th>Passed site prim.</th>
<th>Not allowed for the primary DASD site</th>
</tr>
</thead>
<tbody>
<tr>
<td>Needs RDF+CGT ON</td>
<td>Requires both an RDF and a CGT event</td>
</tr>
<tr>
<td>ECA is ON</td>
<td>Not allowed when there is an ECA event</td>
</tr>
<tr>
<td>CGD is ON</td>
<td>Not allowed when there is a CGD event</td>
</tr>
<tr>
<td>LDR is OFF</td>
<td>Requires an LDR event</td>
</tr>
<tr>
<td>LDR is OFF @ csds</td>
<td>Requires an LDR event for the secondary DASD site</td>
</tr>
<tr>
<td>RDR event is OFF</td>
<td>Requires an RDR event</td>
</tr>
<tr>
<td>DC3 is down</td>
<td>Not allowed when DC3 is down</td>
</tr>
<tr>
<td>Requires LDR event</td>
<td>Requires the LDR event</td>
</tr>
<tr>
<td>MSC is OFF</td>
<td>Requires an MSC event</td>
</tr>
<tr>
<td>CGT is OFF</td>
<td>Requires a CGT event</td>
</tr>
<tr>
<td>Last unpl.scr. NOK</td>
<td>Can only run after a specific previous script</td>
</tr>
</tbody>
</table>

Option T: Timing—View GDDR Script Statistics

When you specify option T in the Primary Options Menu shown in on page 106, the panel that appears depends on the current status of the selected script.

If a script is currently running, a panel similar to the following appears:

--- Script Selection for Status Vi Row 1 to 4 of 4 ---
Option ===>
Enter S next to a script to select it for viewing
Press <F3> to return to Primary Option Menu
Last planned: Resume SRDF/A after link loss
Last unplanned: (none)

Script
- GDD2P17A - Abandon Site DC1 (site swap)
- GDD2P18A - Restart production at DC3 after site swap
S  GDDRPA29 - Resume SRDF/A after link loss
Last planned

Figure 88  Script Selection for Status panel

Type S next to a script to select it for viewing:
- If the selected script is currently running, a panel similar to the following appears:
If a script is completed, a panel similar to the following appears:

--- View Script Step Results Row 1 to 7 of 37
Option ===>  

<table>
<thead>
<tr>
<th>Function</th>
<th>Subfunction</th>
<th>RC</th>
<th>Step start/end</th>
<th>Cnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRGP0F</td>
<td>Verify Planned script env</td>
<td>0</td>
<td>11/20/07 at 09:06:29.96</td>
<td>2</td>
</tr>
<tr>
<td>GDDRGP0L</td>
<td>broadcast variables</td>
<td>0</td>
<td>11/20/07 at 09:06:29.96</td>
<td>2</td>
</tr>
<tr>
<td>GDDRGP0L</td>
<td>Confirm SRDF/A Resumption</td>
<td>0</td>
<td>11/20/07 at 09:06:29.96</td>
<td>2</td>
</tr>
<tr>
<td>GDDRGP0Q</td>
<td>Run ME-MSC_Cleanup</td>
<td>0</td>
<td>11/20/07 at 09:06:29.96</td>
<td>2</td>
</tr>
<tr>
<td>GDDRKF20</td>
<td>Manage BCVs - SPLIT</td>
<td>0</td>
<td>11/20/07 at 09:06:29.96</td>
<td>2</td>
</tr>
<tr>
<td>GDDRGP3J</td>
<td>Make R2s R/O</td>
<td>0</td>
<td>11/20/07 at 09:06:29.96</td>
<td>2</td>
</tr>
<tr>
<td>GDDRF0G</td>
<td>Disable Congroups</td>
<td>0</td>
<td>11/20/07 at 09:06:29.96</td>
<td>2</td>
</tr>
</tbody>
</table>

--- View Script Step Results Row 30 to 37 of 37
Option ===>  

<table>
<thead>
<tr>
<th>Function</th>
<th>Subfunction</th>
<th>RC</th>
<th>Step start/end</th>
<th>Cnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRGF3M</td>
<td>JA Sync</td>
<td>0</td>
<td>11/19/07 at 04:23:39.79</td>
<td>1</td>
</tr>
<tr>
<td>GDDRGP0H</td>
<td>Set M6Parms</td>
<td>0</td>
<td>11/19/07 at 04:23:53.09</td>
<td>1</td>
</tr>
<tr>
<td>GDDRGP0U</td>
<td>Check mirror positions</td>
<td>0</td>
<td>11/19/07 at 04:23:58.11</td>
<td>1</td>
</tr>
<tr>
<td>GDDRGP3J</td>
<td>Ready R2s</td>
<td>0</td>
<td>11/19/07 at 04:26:48.77</td>
<td>1</td>
</tr>
<tr>
<td>GDDRF0D</td>
<td>Resume SRDF/A</td>
<td>*</td>
<td>11/19/07 at 04:29:46.99</td>
<td>1</td>
</tr>
<tr>
<td>GDDRF0D</td>
<td>Manage SRDF/A</td>
<td>*</td>
<td>11/19/07 at 04:29:47.80</td>
<td>1</td>
</tr>
<tr>
<td>GDDRF0A</td>
<td>Propagate Global Variables</td>
<td>*</td>
<td>11/19/07 at 04:29:49.59</td>
<td>1</td>
</tr>
</tbody>
</table>
Option A: Actions—Perform GDDR Actions

Specify option A in the Primary Options Menu (on page 106) to access the GDDR Actions Menu:

--- GDDR - Actions Menu ---

Option ==> H

H  Perform HMC Discovery
L  Perform HMC LPAR actions
CBU Perform HMC CBU actions
S  Manage Couple Datasets
CF  Manage CF Structures

This System: SYS1
This Site: DC1
Master-C: SYS1
Primary Site: DC1
Primary DASD: DC1

Automatic: ON
Planned script: None
Unplanned script: None

--- Discovering HMC Objects ---

Discovering HMC objects at site DC1

*** PLEASE WAIT ***

---

EMC Geographically Dispersed Disaster Restart 05.00.00
Copyright © 2007-2016 EMC Corporation.

Select an option and press <Enter>
Press <F3> to return to the GDDR Primary Options Menu

---

Option H: Perform HMC Discovery

To discover HMC objects accessible to GDDR, specify option H from the GDDR Actions menu. A ‘please wait’ pop-up dialog such as the following appears and HMC object discovery is activated:

--- GDDR - Actions Menu ---

Option ==> H

H  Perform HMC Discovery
L  Perform HMC LPAR actions
CBU Perform HMC CBU actions
S  Manage Couple Datasets
CF  Manage CF Structures

This System: SYS1
This Site: DC1
Master-C: SYS1
Primary Site: DC1
Primary DASD: DC1

Automatic: ON
Planned script: None
Unplanned script: None

--- Discovering HMC Objects ---

Discovering HMC objects at site DC1

*** PLEASE WAIT ***

---

EMC Geographically Dispersed Disaster Restart 05.00.00
Copyright © 2007-2016 EMC Corporation.

Select an option and press <Enter>
Press <F3> to return to the GDDR Primary Options Menu

---

Figure 89 GDDR Actions Menu

Figure 90 HMC object discovery panel
Using GDDR Online Facilities

When the discovery operation completes, an HMC Discovery Results panel containing a scrollable display such as the following appears:

```
GDDR STARTING DISCOVER HMC OBJECTS
GDDR HMC COMMUNICATION WITH DC1 USING BCPII
GDDR MANAGING THE FOLLOWING LPAR-SYSTEM PAIRS FOR SITE DC1
> IBM390PS.Q3,ZOSEQ311:Q311
> IBM390PS.C,ZOSEC11:T101
> GDDR DISCOVERY INFORMATION FOR CPC : IBM390PS.Q3
> GDDR CURRENT ACTIVATION PROFILE : DEFAULT
> GDDR LAST USED ACTIVATION PROFILE : DEFAULT
> GDDR CPU MODEL : E10
> GDDR CPU TYPE : 2098
> GDDR CPU SERIAL NUMBER : 00020F3FA4
> GDDR # GENERAL PURPOSE PROCESSORS : 00000005
> GDDR # SERVICE ASSIST PROCESSORS : 00000002
> GDDR # INTEGRATED FACILITY FOR APPLS : 00000000
> GDDR # INTEGRATED FACILITY FOR LINUX : 00000000
> GDDR # INTERNAL COUPLING FACILITY : 00000000
> GDDR # INTEGRATED INFORMATION PROCESSORS : 00000000
```

**Figure 91** HMC Discovery Results panel

When you are finished examining the results of the discovery operation, press F3 to return to the GDDR Actions menu.

**Option L: Perform HMC LPAR Actions**

HMC actions are console operations affecting a processor, such as Load (IPL) or System Reset, or operations affecting an LPAR, such as Activate LPAR. When you select Option L - Perform HMC LPAR Actions from the GDDR Actions menu, you are presented with the following panel:

```
Actions:  A Activate LPAR   D Deactivate LPAR   Current Master:   GA2LB148
         L Load Clear    R Reset Clear    Primary Site:     DC1
         X Load Recovery  W CPC Swap      Primary DASD:     DC1
         S Start LPAR    P Stop LPAR      Automation:       ON
         Q Query LPAR    E Show events    Planned script:   None
         H Query HMC Method T Toggle Desired State Unplanned script: None
```

Type action to perform next to the desired system and site and press <Enter>

When finished, press <F3> to return to the GDDR Actions Menu

```
S
E LPAR D Load Load
L System Site CPC LPAR S Addr Parm T Message
------------------------------ ------------------ ---------- ------- --------------
  GA2LB148 DC1 IBM390PS.M2964 M38 U **** ******** M
  GA2LB34 DC1 IBM390PS.K12964 K134 D 02997 0F13KGM1 RES
  GA2LB34 DC1 IBM390PS.K12964 K134 D 02997 0F13KGM1 RBG
  GA2LB34 DC1 IBM390PS.K12964 K134 D 02997 0F13KGM1 RBT
  GA2LB34 DC3 IBM390PS.K12964 K134 U 9777 0F13KGM1 C
  GA2LB34 DC3 IBM390PS.K12964 K134 D **** RES
```

**Figure 92** Perform HMC LPAR Actions panel
The Perform HMC LPAR actions panel is only accessible on the Master C-System. It should be viewed as a list of LPARs defined to GDDR, with one or more rows per LPAR, one for each type of IPL parameter defined for the system currently associated with the LPAR.

Most of the actions on this panel are meaningless if the listed system cannot be IPLed by GDDR based on the available IPL parameters for the listed system and the current state of the configuration. In that case, the relevant row will be shown with IPL parameters masked with "*". For example, if a system only has an STD IPL parameter defined for DC2, while the current Primary DASD site is DC1, this row will be shown with IPL parameters masked with "*". The R2 devices at DC2 are considered to be 'NR' in this case, and this not usable to IPL a system from. The system listed on that row can be queried (Option Q - Query LPAR) or events may be listed (Option E - Show Events).

You may enter an action code on any row for a system-site combination. The action you enter will be taken against the system or the system's LPAR using the site named on the selected row.

**Note:** All actions on this panel can also be performed using the GDDR HMC Actions Utility (GDDRHMCA). See “GDDR HMC Actions utility (GDDRHMCA)” on page 316 for details.

### Field descriptions

The values displayed in the HMC LPAR Actions panel are defined in the Define Managed Systems panel shown in Figure 46 on page 147 and the Define IPL Parameters panel shown in Figure 51 on page 156. The following informational fields are conditionally displayed on the HMC LPAR Actions panel for GDDR users who are authorized to perform HMC actions:

**Sel**
Specify the action code for the action you want to perform for the system-site combination appearing on the row.

**System**
Indicates the name of the system associated with the LPAR shown on the table row.

**Site**
Identifies the site location of the LPAR shown on the row. This value displays as DCn-H for LPAR_RECOVERY protected systems if they are at their 'Home' location. This value displays as DCn-A for LPAR_RECOVERY protected systems if they are at their 'Away' location.

**CPC**
Central Processor Complex. This is the name of the central processor where the LPAR is defined.

**LPAR**
Logical Partition. The name of the LPAR that this system runs in at the specified site.

**DS**
Desired State. The desired state of a system, type U (up) or D (down). Toggling the Desired State for a system on this panel Up or Down, has no effect on the system. This field only informs GDDR that the indicated system is intended to be down and that no action should be taken to bring it back up when GDDR observes the system is down.
If you use GDDR to bring a system down, either during scripts or by using this panel, GDDR
sets the Desired State for the system to Down. Similarly, if you use GDDR to IPL a system,
GDDR sets the Desired State for the system to Up.

**Load address**

Contains the MVS device address of the load device that will be used if you specify an
action which performs a Load as part of its operation (for example, Load-Clear). The load
address values are masked with asterisks for the Master C-System, to prevent accidental
IPL and for managed systems STP IPL parameters that are not currently usable to perform a
LOAD action against the LPAR listed on the row. A STD IPL parameter is considered usable
if it specifies a LOAD address on the primary DASD site, and the LPAR shown on the row is
supposed to have channel connectivity to the primary DASD site.

**T**

Specifies the LPAR type, in some cases qualified with an IPL-parameter type.

- **C**: a C-System
- **M**: the Master C-System
- **OS**: an HMC only LPAR, with RES IPL parameter
- **OA**: an HMC only LPAR, with ALT IPL parameter
- **RES**: a managed system, with RES IPL parameter
- **ALT**: a managed system, with ALT IPL parameter
- **DRT**: a managed system, with DR-TEST IPL parameter
- **RBG**: a managed system, with RES BCV Gold IPL parameter
- **ABG**: a managed system, with ALT BCV Gold IPL parameter
- **RBT**: a managed system, with RES BCV Test IPL parameter
- **ABT**: a managed system, with ALT BCV Test IPL parameter
- **CF**: a Coupling Facility LPAR

*Note: HMC actions LOAD and RESET CLEAR do not apply to CF LPARs.*

**Load parms**

Contains the parameter used for the Load if you specify an action which performs a Load
as part of its operation (for example, Load-Clear). The load parameter values are masked
with asterisks if the Load address is masked with asterisks.

**Message**

Messages are active events for the listed system. Table 17, “Monitored events,” on
page 201 provides a description of all events.

**Actions you can request**

To request an HMC action, enter the action code on the command line and press Enter.

- **A**: Activate LPAR

To activate the LPAR of the selected system, type A next to the selected system and press
Enter. A confirmation pop-up displays:
Confirm the request by entering **Y** or **F** for foreground processing or **B** for background batch processing.

If you choose **B**, a second confirmation pop-up displays. You can modify the job statement as needed.

- **Cancel the request by entering **N**.**
- **Save the batch execution JCL without processing by entering **S**. After pressing **Enter**, a confirmation panel similar to the following indicates where the JCL was saved:

```plaintext
JCL to run Deactivate of LPAR 0013 saved to
GDDR.GDDR500.WORK.PARMS(H0512294).
```

**Note:** The Activate action is not allowed for the Master C-system or for an LPAR which is already activated.

### D- Deactivate LPAR

To deactivate the LPAR the selected system runs in, type **D** next to the selected system and press **Enter**. A confirmation pop-up displays.

- **Confirm the request by entering **Y** or **F** for foreground processing or **B** for background batch processing. If you choose **B**, a second confirmation pop-up displays. You can modify the job statement as needed.**
- **Cancel the request by entering **N**.**
- **Save the batch execution JCL without processing by entering **S**. After pressing **Enter**, a confirmation panel indicates where the JCL was saved.**

**Note:** The Deactivate action is not allowed for the Master C-system.

### L - Load Clear

**Note:** Before taking this action, verify that the value of the Desired State field (DS) is consistent with the Load action. Taking this action will cause the Desired State for the system to become Up.
To clear and load a selected system using the displayed load address and load parameters, type **L** next to the selected system and press **Enter**. A confirmation pop-up displays.

The Load Address and Load Parm fields are populated from current IPL parameters, you may overtype these fields in the confirmation panel. If overtyped, the panel saves the new values by creating a new or overwriting an existing alternate IPL parameter, causing the Perform HMC LPAR Actions panel to dynamically update, showing the new alternate IPL parameter entry.

**Note:** The new or updated global variables representing these IPL parameters only exist on the Master C-system. A Parameter Wizard Activate is required to broadcast these changes to the other C-systems. Refer to the GDDR IPL Parameter Swap utility for an alternative approach which includes keeping the remote C-Systems up-to-date.

- Confirm the request by entering **Y** or **F** for foreground processing or **B** for background batch processing. If you choose **B**, a second confirmation pop-up displays. You can modify the job statement as needed.
- Cancel the request by entering **N**.
- Save the batch execution JCL without processing by entering **S**. After pressing **Enter**, a confirmation panel indicates where the JCL was saved.

The following rules apply to IPLs using STD GDDR-managed volumes:

- By default, use the site selected for the IPL action.
- For test-IPL scripts, always use the selected site.
- For other scripts and IPL actions, and for C-Systems, use the home site of the system, unless the system has LPAR Recovery protection and is in its 'Away' location.

The Load Clear action is not allowed:
- for the Master C-system
- for CF LPARs
- if the Load address on the selected row is masked with ****
- if the system is currently operating
- if the LPAR is not Activated

**R - Reset Clear**

**Note:** Before taking this action, verify that the value of the Desired State field (DS) is consistent with the Reset action. Taking this action will change the Desired State for this system to Down.

To reset and clear a selected system, type **R** next to the selected system and press **Enter**. A confirmation pop-up displays. A confirmation pop-up displays.

- Confirm the request by entering **Y** or **F** for foreground processing or **B** for background batch processing. If you choose **B**, a second confirmation pop-up displays. You can modify the job statement as needed.
- Cancel the request by entering **N**.
- Save the batch execution JCL without processing by entering **S**. After pressing **Enter**, a confirmation panel indicates where the JCL was saved.
**Note:** The Activate action is not allowed for the Master C-System or if the LPAR is not activated.

**X - Load Recovery**

**Note:** Before taking this action, verify that the value of the Desired State field (DS) is consistent with the Load Recovery action. Taking this action will change the Desired State for this system to Up.

To IPL at the system’s alternate location, type X next to the selected system and press **Enter**. If this option is available, the Site value will display a fourth character after DCx; either H or A, meaning the system is currently running at DCx in its home (H) location or its alternate (A) location. A confirmation pop-up displays.

The Load Address and Load Parm fields are populated from current IPL parameters, you may overtype these fields in the confirmation panel. If overtyped, and if alternate IPL parameters do not exist for the system/site, the panel saves the new values as alternate IPL parameters, causing the Perform HMC LPAR Actions panel to dynamically update, showing the new alternate IPL parameters entry.

- Confirm the request by entering **Y** or **F** for foreground processing or **B** for background batch processing. If you choose **B**, a second confirmation pop-up displays. You can modify the job statement as needed.
- Cancel the request by entering **N**.
- Save the batch execution JCL without processing by entering **S**. After pressing **Enter**, a confirmation panel indicates where the JCL was saved.

The following rules apply to IPLs using STD GDDR-managed volumes:

- By default use the site selected for the IPL action.
- For test IPL scripts always use the selected site.

For other scripts and IPL actions and for C-Systems, use the home site of the system, unless the system has LPAR Recovery protection and is in its 'Away' location.

**Note:** The Load Recovery action is not allowed for the Master C-system or for a system that does not have LPAR Recovery protection.

**W- CPC Swap**

**Note:** The CPC Swap action applies to the entire CPC that houses the LPAR shown on the selected row. All Recovery Protected LPARs on that CPC will be moved to their alternate locations (from "Home" to "Away" or from "Away" to "Home"). This action is only allowed when the system on the selected row has LPAR Recovery Protection.
To perform a planned CPC swap, type **W** next to the selected system and press Enter. A confirmation pop-up displays:

```
+---------- Confirm CPC Swap of CPC IBM390PS.P --------+
| Command ===>                                          |
| Please confirm Swap CPC of CPC IBM390PS.P             |
| CPC: IBM390PS.P                                       |
| NOTE: This action will move EVERY system currently running on this CPS which has LPAR recovery defined to its alternate location.|
| Proceed? ===> N  Reply Y or N                        |
```

- **Reply Y or N** to either continue or discontinue the requested action.

**S - Start LPAR**

To start a selected system, type **S** next to the selected system and press Enter. A confirmation pop-up displays.

- Confirm the request by entering **Y** or **F** for foreground processing or **B** for background batch processing. If you choose **B**, a second confirmation pop-up displays. You can modify the job statement as needed.
- Cancel the request by entering **N**.
- Save the batch execution JCL without processing by entering **S**. After pressing Enter, a confirmation panel indicates where the JCL was saved.

**Note:** The Start LPAR action is not allowed for C-Systems. This action is only allowed if the selected LPAR is in an Activated, Not Operating state.

**P - Stop LPAR**

To stop a selected system, type **P** next to the selected system and press Enter. A confirmation pop-up displays.

- Confirm the request by entering **Y** or **F** for foreground processing or **B** for background batch processing. If you choose **B**, a second confirmation pop-up displays. You can modify the job statement as needed.
- Cancel the request by entering **N**.
- Save the batch execution JCL without processing by entering **S**. After pressing Enter, a confirmation panel indicates where the JCL was saved.

**Note:** The Stop LPAR action is not allowed for C-Systems. This action is only allowed if the selected LPAR is in an Operating state.
Q - Query LPAR
To query the state of the LPAR of the selected system, type Q next to the selected system and press Enter. A confirmation pop-up will be displayed. Confirm the request by typing Y and pressing Enter, or cancel the request by pressing F3. When the query is performed, the Message field is updated with the current state of the selected LPAR: Not Activated, Not Operating, Operating.

E - Show Events
To show any relevant active events for the system in the Message field, type E next to the selected system and press Enter. System related events are IPL, (planned IPL), MXS, (unplanned MXS), and MHB, (unplanned MHB). Table 17, “Monitored events,” on page 201 provides additional information about these events.

T - Toggle desired state (U to D or D to U)
The Desired State of a system (U for up or D for down) appears in the DS (Desired State) column of the Perform HMC LPAR Action panel. The default value is U.

Toggling the Desired State Up or Down for a system on this panel has no effect on the system. This field only serves the purpose of informing GDDR that the indicated system is intended to be down and that no action should be taken to bring it back up when GDDR observes the system to go down.

However, if you use GDDR to bring a system down, either during scripts or using this panel, GDDR will set the Desired State for the system to Down. Similarly, if you use GDDR to IPL a system, GDDR will set the Desired State for the system to Up.

To toggle the Desired State of a system, type T next to the selected system and press Enter. A confirmation pop-up with the new state displays:

<p>| Command ==&gt;                                          |
|                                                       |
| Please confirm Toggle Desired State request           |
| System: PRD1     New Desired State: D                 |</p>
<table>
<thead>
<tr>
<th>Proceed? ==&gt; N  Reply Y or N</th>
</tr>
</thead>
</table>

◆ Reply Y or N to either continue or discontinue the requested action.
Option CBU: Perform HMC CBU actions

When you specify option CBU in the GDDR Actions menu, the following panel displays:

```
----------------------------- GDDR - Perform CBU Actions --------------------- Row 1 of 6
Command ===>                                                  Scroll ===> CSR
Type one of the following action codes next to desired site/CPC pair:
  A  activate CBU in REAL mode
  B  activate CBU in TEST mode
  C  Undo CBU in REAL mode
  D  Undo CBU in TEST mode
When ready, press <Enter> to perform the action
You may press <F3> to return to the Primary Options Menu
```

```
Sel  Site  CPC       Message
---  ----  --------  -----------------------------------------------
  _    DC1   X0
  _    DC1   X0
  _    DC1   X1
  _    DC3   X0
  _    DC3   X0
  _    DC3   X1
............................................ Bottom of data ............................
```

**Figure 93** Perform CBU Actions panel

Each row represents a site with a processor enabled for CBU (Capacity Backup Upgrade) activation.

Type one of the following action codes next to the desired site/CPC pair:

- **A** to activate backup capacity on the specified processor (REAL mode)
- **B** to simulate the activation of backup capacity on the specified processor (TEST mode)
- **C** to undo or terminate actual backup capacity activation
- **D** to undo or terminate a simulation of backup capacity activation

Option S: Manage Couple Datasets

When you specify option S in the GDDR Actions menu, the following panel displays:

```
------------------------ GDDR - Manage Couple Datasets ------------------------
Command ==>
Current Primary DASD Site: DC1
Specify site for   Primary Couple Datasets: DC1
Specify site for Alternate Couple Datasets: DC2
Simulation Mode ? NO    (YES/NO)
Enter or change the job statement:
  //jobname  JOB account,name,CLASS=class,MSGCLASS=class,
  //        USER=GDDR,REGION=0M
  /*JOBPARM LINES=999999
  //*
Press <Enter> when ready. Press <F3> to return without submitting job.
```

**Figure 94** Manage Couple Datasets panel
You can use this panel to drive GDDR management of couple datasets, aligning couple dataset usage with the sites specified on this panel. GDDR couple dataset management is driven regardless of script management options for each site specified in “Option S: Script Sysplex Options” on page 171.

**Note:** Refer to the criteria for selection of eligible systems for sysplex operations in “System qualification for sysplex options” on page 172.

In environments with geographically dispersed sysplexes supported by cross-site host-DASD channels, the sites can be the Primary [Site A] or Secondary [Site B]. In other environments, only the Primary site [R1-site] is allowed.

Currently used couple datasets are compared against those defined to GDDR for each site. If discrepancies are found, GDDRMAIN COMM services are used to issue the required SETXCF COUPLE commands and bring couple dataset usage in line with GDDR parameter specifications. This requires GDDRMAIN to be up and running on the target Managed systems.

The Current Primary DASD site is shown for information, in case you want to use Primary Couple Datasets from the Primary DASD site, and Alternate Couple Datasets from the opposite site.

You can make the following changes using this panel:

- **Specify site for Primary Couple Datasets**
  
  Informs GDDR where you want to have Primary Couple Datasets used.

- **Specify site for Alternate Couple Datasets**
  
  Informs GDDR where you want to have Alternate Couple Datasets used.

- **Simulation mode? YES/NO**
  
  Indicates whether to use simulation mode. The default setting is **NO**. Simulation mode issues real D XCF commands to determine the initial situation. All other commands will be shown in the joblog but not issued on the target system.

  **Note:** Activating simulation mode on this panel affects this and all subsequent GDDR couple dataset management actions (stand-alone or in scripts) on this C-System until the next GDDR Parameter Wizard Activate. You can also activate couple dataset management simulation using “Option S: Manage GDDR System Variables” on page 192, by adding a new variable USEROPT.CPLSIMUL=1. This variable will be removed upon the next GDDR Parameter Wizard Activate.

- **Enter or change the job statement**
  
  Provide the jobcard and any other JCL required.

Press **Enter** when ready. Press **F3** to return to the previous panel without submitting the job.
Option CF: Manage CF Structures

When you specify option CF in the GDDR Actions menu, the following panel displays:

```
------------------------ GDDR - Manage CF Structures ------------------------
Command ===>

Current Primary DASD Site: DC1

Specify site for CF Structure location: DC1

Simulation Mode ? NO    (YES/NO)

Enter or change the job statement:
//GDDRSTR0 JOB (06610),GDDR-PARMUPDT,
   //     CLASS=A,MSGCLASS=X,NOTIFY=&SYSUID,REGION=0M
   //*
   //*
Press <Enter> when ready. Press <F3> to return without submitting job.
```

Figure 95  Manage CF Structures panel

This panel is used to control GDDR management of Couple Facility structures, causing CF structures to be rebuilt in the next CF in the Preference List for each structure, as specified in the active CFRM policy. GDDR CF structure management is driven regardless of the script management options for each site specified in “Option S: Script Sysplex Options” on page 171.

Note: Refer to the criteria for selection of eligible systems for sysplex operations in “System qualification for sysplex options” on page 172.

In environments with geographically dispersed sysplexes supported by cross-site host-DASD channels, you can request rebuild to primary site [Site A] or the secondary site [Site B]. In other configurations, only the primary site [Site A] is allowed.

Current CF Structure locations are compared against those defined to GDDR for the site specified on this panel. If discrepancies are found, GDDRMMAIN COMM services are used to issue the required SETXCF START,REBUILD commands on production systems.

The Current Primary DASD site is shown for information, in case you want to move CF Structures to the Primary DASD site.

You can make the following changes using this panel:

- Specify site for CF Structures location
  Specify where you want to move CF structures.

- Simulation mode? YES / NO
  Specify whether to use simulation mode. The default is NO. Simulation mode issues real D XCF commands to determine the initial situation. All other commands will be shown in the joblog but not issued on the target system.

Note: Activating simulation mode on this panel affects this and all subsequent GDDR CF structure management actions, both stand-alone or in scripts, on this C-System until the next GDDR Parameter Wizard Activate. You can also activate CF structure
management simulation using “Option S: Manage GDDR System Variables” on page 192, by adding a new variable USEROPT.CPLSIMUL=1. This variable will then be removed upon the next GDDR Parameter Wizard Activate.

- Enter or change the job statement
  
  Specify the jobcard and any other JCL required.

Press Enter when ready. Press F3 to return to the previous panel without submitting the job.

**Option ST: eSTEM - Run EMC Started Task Execution Manager**

The EMC Started Task Execution Manager allows you to start and stop GDDR and Mainframe Enablers related tasks from a single console.

Specify option ST on the GDDR Primary Options menu (on page 106) to access the EMC Started Task Execution Manager. The following panel appears:

![Figure 96 EMC Started Task Execution Manager Procedure Member List panel](image)

A blank list is displayed if this is the first use of the panel; otherwise, any previously-defined procedures display here.

**Adding a new procedure**

A procedure is a named collection of tasks to be issued in a specific sequence.

1. Type S next to the word ADD on the Member List panel and press Enter.

   A blank Add/Update panel displays:

   ![Add/Update panel](image)

2. Type in a name and description for the procedure and press Enter.
The procedure is added to the Member List panel. The following example shows a procedure named STRTECO:

```
---------------------- EMC Started Task Execution Manager
Member STRTECO Added
Command ===> Scroll ===> PAGE
_ ADD - "S" to add a new Procedure

MCS Console ID: JPREST2

Line Cmds: C = Clone D = Del E = Edit S = Edit Description R = RMT X = Execute

------------------------------- Procedure Member List -------------------------------
Sel Procedure Description
--- --------- ----------------------------------------------------------
_ STRTECO   Start up all my ECO prefixed tasks

---------------------- EMC Started Task Execution Manager ---------------------
Command ===> Scroll ===> PAGE

_ ADD - "S" add new Task(s), or "C" to clone existing Procedure

Line Commands...: E or S or / to Edit  C to Copy  D to Delete the entry(s)

----------------= STRTECO - Start up all my ECO prefixed tasks =================
Sel SeqNbr Issue Command String to Find Skip Wait
--- ------ -------------------------- -------------------------- ---- -----

******************************* Bottom of data ********************************
```

Adding tasks to the procedure

A task is a z/OS or 3rd party vendor command to be issued.

1. To add or update tasks associated with a procedure, type E next to the procedure you want to edit and press Enter.

If there are already tasks for the procedure, they are listed on the Task List panel, otherwise the panel displays with a blank table:

```
------------------------------- Procedure Member List -------------------------------
Sel Procedure Description
--- --------- ----------------------------------------------------------
_ STRTECO   Start up all my ECO prefixed tasks

----------------= STRTECO - Start up all my ECO prefixed tasks =================
Sel SeqNbr Issue Command String to Find Skip Wait
--- ------ -------------------------- -------------------------- ---- -----
```

There were zero (0) task records found for STRTECO, use the ADD primary command to add tasks.

---
2. To add a new task, type S next to the word ADD and press Enter. A blank Task Add/Update panel displays:

```
---------------------- EMC Started Task Execution Manager ----------------------
Command ===>                                                  Scroll ===> PAGE
ENTER: Process Add / Update            PF3: Cancel Add / Update
==============================================================================
Sequence Nbr...: 000010
Issue Command.: ____________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
String to Find:
Skip............: N (Y=Yes) Skip command if Task NOT started by this procedure.
Wait Seconds...: 3     Check for String every Wait Seconds up to 10 times.
Would you like to add another Task entry?  N ( Y or N )
```

3. Complete the panel fields as follows:

- **Sequence Nbr:** This number defines the order in which the tasks will be executed. You can accept the generated value or you can overtype this number with a number between two others to insert this command at its desired position. These sequence numbers are automatically regenerated as multiples of 10 each time the procedure is edited and saved to allow for future inserts.

- **Issue Command:** The command to be issued when the procedure is executed. Enter a valid z/OS, JES, operator, or third party vendor command. You can also use the ROUTE command to direct a command to one or more systems in a sysplex for processing. For example, if you are on X00B and X001 is an LPAR in the same sysplex, then enter the following in this field to execute a D IPLINFO command on X001: RO X001,D IPLINFO.

  **Note:** If the command string is longer than 26 characters, ISPF Column Scrolling is enabled to allow you to see the whole command on the panel.

- **String to Find:** A text string or a message number that must be found by the EMC Started Task Execution Manager before the next task will be executed. Enter a text string or a message number. After waiting the specified period (see “Wait Seconds” below), up to 10 attempts will be made to find this string. If it is not found, an informational message displays and the EMC Started Task Execution Manager exits. If the string is found, the complete string is echoed back, and execution continues.

  **Note:** If the text string is longer than 26 characters, ISPF Column Scrolling is enabled to allow you to see the whole string on the panel.

- **Skip:** Enter Y to direct the command to be skipped if the started task associated with this command is not started by this invocation of EMC Started Task Execution Manager. If the started task associated to this command is already started, then this task is bypassed. Enter N to always execute the command.
• **Wait Seconds**: Checks for the “String to Find” text string every $n$ Wait Seconds up to 10 times. For example, if you want to wait no longer than a minute for the string to be found, enter a 6 into this field.

• **Would you like to add another Task entry?**: Press Y to add the entry to the procedure and display a new blank Task Add/Update panel.

The following example shows a completed Task Add/Update panel:

```
- ---------------------- EMC Started Task Execution Manager ----------------------
Command ===>                                                  Scroll ===> PAGE

ENTER: Process Add / Update                  PF3: Cancel Add / Update
- ------------------------------- STRTECO Task Add/Update --------------------------------

Sequence Nbr...: 000010
Issue Command.: S ECOSCF

String to Find: SCF0890I
Skip..........: N (Y=Yes) Skip command if Task NOT started by this procedure.
Wait Seconds..: 10    Check for String every Wait Seconds up to 10 times.
Would you like to add another Task entry?  N ( Y or N )
```

4. **Press Enter** when you are done adding or updating tasks.

The Task List panel displays with the modified tasks. The following example shows the task has been added to the list for a procedure named STRTECO:

```
- ---------------------- EMC Started Task Execution Manager ----------------------
Command ===>                                                  Scroll ===> PAGE

ENTER: Process Add / Update                  PF3: Cancel Add / Update
- ------------------------------- STRTECO Task Add/Update --------------------------------

_ 000010 S ECOSCF SCF0890I N 10

---------------------- EMC Started Task Execution Manager ----------------------
Command ===>                                                  Scroll ===> PAGE

ENTER: Process Add / Update                  PF3: Cancel Add / Update
- ------------------------------- STRTECO Task Add/Update --------------------------------

_ 000010 S ECOSCF SCF0890I N 10

*************************************** Bottom of data ********************************************
```

You can edit, copy, or delete tasks in the list using the specified line commands.

**Executing a procedure locally**

Once you have entered all the tasks needed for a procedure, you can execute the procedure on the LPAR where you are running the EMC Started Task Execution Manager.

1. **Select the procedure by typing an X or a W line command.**
   - **X** executes all tasks without stopping between each task.
   - **W** executes a single task. You can then press **Enter** to continue with the next task in the procedure or type **QUIT** to terminate the procedure. This method allows you to view the output of each individual command before proceeding.
For example, in the following display, procedure JFPSCFD is selected for execution.

--- ADD (Select with an S to add new Procedures) ---
Line Cmds: C = Clone D = Del E = Edit S = Edit Description R = RMT X = Execute
Sel Procedure Summary
--------------------------
X JFPSCFD Startup for development SCF & SRDF
_ JFPSTOP Stop SCF & SRDF

Executed a procedure remotely

You can also execute Started Task Execution Manager (STEM) procedures on a remote host via TCP/IP.

Prerequisites

Before running a procedure on a remote host, ensure the items listed below are complete.

- The STEM listener procedure JCL must be installed in the appropriate started task PROCLIB at the remote host. The JCL for the STEM listener procedure is as follows:

```
// SETPROG APF,ADD,DSN=EMC.STEM.LINKLIB,SMS
//*
// PRODECURE ESFSTEM
//*  STEM REMOTE EXECUTION LISTENER
//*
//STEMLSTN EXEC PGM=ESFSTEM,PARM='nnnn'
//STEPLIB DD DISP=SHR,DSN=EMC.STEM.LINKLIB
//SYSPROC DD DISP=SHR,DSN=EMC.STEM.REXX
//SYSEXEC DD DISP=SHR,DSN=EMC.STEM.REXX
//STEMTASK DD DISP=SHR,DSN=EMC.STEM.JOBPARMS
//SYSUDUMP DD SYSOUT=* 
//SYSOUT DD SYSOUT=* 
//SYSPRINT DD SYSOUT=* 
```

The PARM= value on the EXEC statement specifies the TCP/IP port that will be used on this host to accept STEM procedure execution requests.

The userid associated with the ESFSTEM started task must have an OMVS segment defined that allows read/write access to TCP/IP.

You can start the ESFSTEM started task by issuing the S ESFSTEM console command. You can stop the task by issuing either the F ESFSTEM,STOP command or the P ESFSTEM command.

**Note:** The necessary LINKLIB, REXX and JOBPARMS PDS files will have been created at the time SCF was installed at the remote host.

- The STEM work task procedure JCL must be installed in the appropriate started task PROCLIB at the remote host. The JCL for the STEM work task procedure is as follows:
When a request to execute a STEM procedure is received over TCP/IP, ESFSTEM starts an ESFSTEMT address space to process the commands within the procedure. When the commands have been issued and the responses collected, the ESFSTEMT address space shuts down. The execution of STEM procedures is serialized, so there should never be more than one ESFSTEMT address space active at any point in time.

During initialization, the ESFSTEMT address space’s ACEE is set to the userid that sent the STEM procedure work request. This ensures you cannot issue any command through STEM that you would not have authority to issue via logging on to the host LPAR and entering the same command through SDSF.

To execute a Started Task Execution Manager command on a remote host you must:

- Have the same TSO userid defined for the remote host and for the Started Task Execution Manager ISPF function on the local host.
- Have COMMAND authority on the remote host.

Executing the remote procedure

1. To execute a Started Task Execution Manager procedure on a remote host via TCP/IP, type R next to the procedure. For example:

```
---------------------------- EMC Started Task Execution Manager ---- Row 1 to 5 of 5
Command ===>                  Scroll ===> PAGE
 _ ADD (Select with an S to add a new Procedure)

Line Cmds: C = Clone D = Del E = Edit S = Edit Description R = RMT X = Execute

---------------------------- Procedure Member List ----------------------------
Sel Procedure Description
 _ ADDOWN    Shut down SRDF and SCF
 R ADUP      Start up SCF and SRDF
         _ ADUP6 Start SCF, SRDF, HSMDX
         _ WTORTEST Test WTOR reply feature
************************************************** Bottom of data **************************************************
```

2. Press Enter.
A panel similar to the following appears:

```
-------------- EMC Started Task Execution Manager -- Row 1 to 15 of 15 ------------
  ADD - "S" to add new Host
  Line Cms.:  S = Select Host   E = Edit Host   D = Delete Host
  ENTER: Process   PF3: Cancel   PF7: Up   PF8: Down   PF10: Left   PF8: Right
  --------------- Host List for procedure: DIPLINFO --------------------------
  TCP/IP  Replace
  Sel  Hostname  IP address or DNS Name        STC Name  Procedure
  -------  ----------  ------------------------  --------  ---------
  -ALL-   N
  DC1CO012 10.243.11.90   TCPIP         Y
  DC1PQ317 10.243.11.118  TCPIP         N
  DC2CO017 10.243.11.103  TCPIP         N
  DC2PQ31A 10.243.11.70   TCPIP         N
  DC3CO01A 10.243.11.104  TCPIP         N
```

3. Type E next to the host to edit and press Enter.

A panel similar to the following appears:

```
-------------- EMC Started Task Execution Manager -----------------------------
Command ==>  Scroll ==> PAGE

ENTER: Submit Remote Execution Request   PF3: Cancel Remote Execution
-------------- ADUP Remote Execution ------------------------------------------
Remote Host...: 10.243.150.75       Host Name or IP Address
TCP/IP Port...: 7836
Replace Member: N ( Y or N ) Replace like named member at remote host.
```

4. Complete the panel fields as follows:
   - **Remote Host**: Enter the host name (if DNS is available) or the IP address of the LPAR where the Started Task Execution Manager procedure commands will be executed.
   - **TCP/IP Port**: Enter the TCP/IP port number that the ESFSTEM started task at the remote host uses to listen for incoming requests.
   - **Replace Member**: The name and contents of the Started Task Execution Manager procedure member will be sent via TCP/IP to the remote host. Reply Y to save this member at the remote host, replacing the content of any like-named member. Reply N to save the member using a temporary name until execution completes, at which time the temporary member is deleted.

5. Type S next to the selected host to run and press Enter.
A pop-up similar to the following appears:

```
- Enter Remote User Information + 0: Left PF8: Rl
| UserID on 10.243.11.90 | ================
| Password........... | P Replace
| Password for UserID | ame Procedure
| Initiator Class: A | -----------
| Initiator Class for Batch | N
| job submission | Y
| N
| Enter: Process PF03: Cancel | N
| UserID..........: MABDEE1 | N
| =-------------------------------------|
```

6. Enter the UserID, password, and initiator class for the selected host.

7. Press Enter.

The message “Processing, Please Wait...” displays until the Started Task Execution Manager procedure commands complete on the remote host and a response is received via TCP/IP. When remote host processing is complete, the output of the commands in the procedure displays.

### Executing a procedure within a procedure

A Started Task Execution Manager procedure may also execute another procedure or series of procedures. The command syntax for executing a procedure within a procedure is:

```
EX STEM_procedure_name [HOST(lparname)] [USER(userid)] [BATCH]
```

Where:

- `STEM_procedure_name` is the Started Task Execution Manager JOBPARMS member that contains the z/OS commands to be executed.
- `lparname` is the name of the LPAR where the `STEM_procedure_name` will be executed.
- `userid` is the TSO userid on `lparname` and is used for the FTP job and setting the security environment for started task ESFSTMT.
- `BATCH` indicates the Started Task Execution Manager procedure will be executed as a batch job.

The HOST, USER, and BATCH keywords are optional. The `STEM_procedure_name`, HOST, USER, and BATCH parameters can be separated by either blanks or commas.

If the HOST keyword is not specified, the default is the LPAR where the Started Task Execution Manager ISPF application is executing. This value can be overridden at the time the procedure is executed.

If the USER keyword is not specified, the default is the TSO userid for the current Started Task Execution Manager ISPF application session. This value can be overridden at the time the procedure is executed.
If the BATCH keyword is not specified, the procedure is run in the defined sequence and subsequent procedure statements will not be executed until this procedure has completed. If the BATCH keyword is specified, the procedure is executed as a batch job and upon job submission the next sequential procedure statement is executed immediately.

Started Task Execution Manager procedures containing EX STEM_procedure_name statements can only be executed using the W (Watch) or X (Execute) line commands.

**Note:** Due to the need to provide a password on lparname, Started Task Execution Manager procedures that contain EX STEM_procedure_name statements cannot be run using the R (Remote) line command.

When a procedure that contains EX STEM_procedure_name statements is executed, Started Task Execution Manager first checks whether the lparname is in the Started Task Execution Manager host name ISPF table. If the lparname is already defined, the IP address and TCP/IP procedure name is extracted from the table.

After you select W or X in the Started Task Execution Manager Procedure Member List and press Enter, a pop-up dialog displays to obtain the TSO (FTP) password for the userid on lparname:

```
+-- Enter Remote User Information ---+
     Enter values for X00B
     IP Address....: nn.nnn.nnn.nn
     TCP STC Name...: TCPIP
     UserID.......: NNNNNN
     Password......:
         Password for UserID
         Initiator Class: A
             Initiator Class for Batch
                job submission
     Enter: Process  PF03: Cancel
+------------------------------------+
```

You can use this panel to override the values of the IP Address, TCP STC Name, UserID and Initiator Class. Override values are only stored to the Started Task Execution Manager host name ISPF table when a table entry does not already exist.

Passwords are stored in REXX and ISPF variables that are deleted when you exit the Started Task Execution Manager ISPF application.

**Usage example 1: Cycling a set of address spaces**

This usage example illustrates a scenario where a series of commands is required to shut down an address space (ADDOWN) and another series of commands is needed to start it up (ADUP). It demonstrates how a new procedure (ADCYCL) could be created to execute the existing ADDOWN procedure and then execute the existing ADUP procedure.
The ADDOWN procedure stops the SRDF and SCF address spaces:

```
------ EMC Started Task Execution Manager ---- Row 1 to 4 of 4
Command ==>                                                   Scroll ==> PAGE
__ ADD - "S" to add new Task(s)
Line Commands...: E or S or / to edit entry or D to delete the entry
------------------------------- ADDOWN - Stop SRDF and SCF -----------------------------
Sel SegBr Issue Command String to Find Skip Wait
--- ------ -------------------------- -------------------------- ---- ----
__ 000010 P ECOMPTSK N 10
__ 000020 P EC0SRDF EMCMN10I N 10
__ 000030 P EC0SCF,MSC DISABLE SCF1321I N 10
__ 000040 P EC0SCF,INI SHUTDOWN N 10
****************************************************************************** Bottom of data ******************************************************************************
```

The ADUP procedure starts SCF and SRDF:

```
------ EMC Started Task Execution Manager ---- Row 1 to 5 of 5
Command ==>                                                   Scroll ==> PAGE
__ ADD - "S" to add new Task(s)
Line Commands...: E or S or / to edit entry or D to delete the entry
------------------------------- ADUP - Start SCF and SRDF ------------------------------
Sel SegBr Issue Command String to Find Skip Wait
--- ------ -------------------------- -------------------------- ---- ----
__ 000010 S EC0SCF Y 15
__ 000020 D IPLINFO N 10
__ 000030 d m=cpu N          
__ 000040 S EC0SRDF EMCMN03I Y 10
__ 000050 S ECOMPTSK EC00061Y Y 10
****************************************************************************** Bottom of data ******************************************************************************
```

You can combine these two procedures by creating a Started Task Execution Manager procedure named ADCYCL to execute ADDOWN and then ADUP:

```
------ EMC Started Task Execution Manager ---- Row 1 to 2 of 2
Command ==>                                                   Scroll ==> PAGE
__ ADD - "S" to add new Task(s)
Line Commands...: E or S or / to edit entry or D to delete the entry
------------------------------- ADCYCL - Stop and restart AD ------------------------
Sel SegBr Issue Command String to Find Skip Wait
--- ------ -------------------------- -------------------------- ---- ----
__ 000010 EX ADDOWN N 10
__ 000020 EX ADUP N 10
****************************************************************************** Bottom of data ******************************************************************************
```

Both ADDOWN and ADUP will be executed on this LPAR and under the userid for the Started Task Execution Manager ISPF application. Note the following:

- You will be prompted for the IP address, userid and password.
- Any overrides (for example, IP address) will apply to both EX commands since neither contains a HOST keyword and both default to the local LPAR.
• Any overrides are only saved if there is not already a host entry for the local LPAR in the Started Task Execution Manager host name ISPF table.
• To keep processing consistent and the code path short, ADDOWN and ADUP are submitted via TCP/IP and execute in the ESFSTEMT address space.
• Since the BATCH keyword is not specified, procedure ADDOWN will execute to completion before procedure ADUP is executed.
• Since the BATCH keyword is not specified, the output of the ADDOWN and ADUP procedures will be written to the ISPF screen.

Usage example 2: Executing concurrent batch procedures

This usage example illustrates a scenario where multiple procedures are executed concurrently using the BATCH keyword.

A procedure named RUNIT contains display commands to obtain the last IPL date and time and the LPAR configuration information:

```
A procedure named GETIPLI executes procedure RUNIT on three LPARs simultaneously by specifying the BATCH parameter:

```

```
A procedure named GETIPLI executes procedure RUNIT on three LPARs simultaneously by specifying the BATCH parameter:

```

The complete Issue Command statements are not visible in the panel, the full syntax is the following:

EX RUNIT,HOST(X00B),USER(KANSAS1),BATCH
EX RUNIT,HOST(X118),USER(KANSAS1),BATCH
EX RUNIT,HOST(Q311),USER(KANSAS1),BATCH

When procedure GETIPLI is executed using the W (Watch) or X (Execute) option:
You are prompted for the password on each LPAR.

Two batch jobs are submitted per EX RUNIT statement; one to start the Started Task Execution Manager remote listener via FTP and one to execute the RUNIT procedure statements on the remote LPAR.

**IMPORTANT**

It is recommended that you use a secure FTP.

The output from the commands contained in procedure RUNIT appear in the SYSTSPRT DD in the batch jobs.

To maximize throughput by minimizing jobname collisions, the batch jobs are named &useridx where x is as follows:

**FTP jobs** — replace x with 9-0 then Z-A. If a jobname that matches the generated jobname is already executing or waiting for execution, the next sequential character is substituted for x until an available jobname is obtained.

The FTP jobs usually execute very quickly, resulting in most (if not all) FTP jobs being named &userid9.

**STEM procedure remote execution jobs** — replace x with A-Z then 0-9. If a jobname that matches the generated jobname is already executing or waiting for execution the next sequential character is substituted for x until an available jobname is obtained.

The STEM procedure remote execution jobs track the jobname assigned to each EX STEM_procedure_name statement and will usually result in a series of jobnames. In the example above, the RUNIT procedures would typically be executed as jobnames &useridA through &useridC.
CHAPTER 6
Using GDDR Utilities

This chapter describes the GDDR configuration utilities.

◆ Automated Configuration Discovery for DASD (GDDRACDD) ........................................ 240
◆ GDDR RDF Director Overview utility (GDDRDIRS) .......................................................... 281
◆ GDDR MSC Configuration Validation utility (GDDMSCFX) ........................................... 286
◆ GDDR DLm Serial Number utility (GDDRDLSN) ............................................................. 299
◆ GDDR Maintenance Level Reporting utility (GDDRREG) .............................................. 300
◆ GDDR Gatekeeper Validation utility (GDDGATEK) ......................................................... 302
◆ SRDF Device Status Check Utility (GDDRSC06) ............................................................. 307
◆ TimeFinder Management utility (GDDRTF20) ............................................................... 307
◆ BCV Reporting utility (GDDRTF00) ............................................................................... 310
◆ GDDR IPL Parameter Swap utility (GDDRISWP) ............................................................ 314
◆ GDDR HMC Actions utility (GDDRHMCA) ................................................................. 316
◆ GDDR BCPii Connectivity test utility (GDDRBCPI) ......................................................... 318
◆ GDDR Load Profile Management utility (GDDRLPRF) ...................................................... 319
◆ GDDR ECGUTIL Driver utility (GDDREC0G) ................................................................. 320
◆ GDDR IPL Assist Monitor utility (GDDRIAM) ............................................................... 321
◆ GDDR Invalid Track Monitor utility (GDDRMINO) .......................................................... 325
◆ GDDRMAIN Trace Print utility (GDDRTRCP) ................................................................. 328
Automated Configuration Discovery for DASD (GDDRACDD)

The GDDR Automated Configuration Discovery for DASD utility (GDDRACDD) is designed to be run at any site in any GDDR configuration. GDDRACDD serves the following purposes:

- Discovers SRDF and TimeFinder as well as DLm backend devices in a set of defined VMAX units and RDF groups.
- Generates parameters usable to define the discovered devices to GDDR.
- Validates the discovered configuration.
- Validates existing RDF.DEVICES, DLM.DEVICES, and BCV-related parameters and other configuration global variables against the discovered DASD configuration and against GDDMPARM information.
- Produces a set of reports to describe the configuration:
  - SYMMDEVS: lists contiguous device ranges
  - RDFGROUP: lists RDF groups connecting GDDR-managed VMAX systems
  - CREPAIR: lists SC VOL CREATEPAIR commands usable to create the GDDR-managed device pairs
  - FBAMETA: lists FBA-meta devices and RDF relationships
  - TFLIST: reports GDDR-managed BCV devices

The GDDRACDD utility is supported with REXX module GDDRACDD and sample JCL in hlq.GDDRvrm.SAMPLIB (GDDRACDD). Member GDDRPACD in hlq.GDDRvrm.PARMLIB provides parameters.

Operating modes

Discovery mode

Run GDDRACDD in Discovery mode to discover an EMC VMAX DASD configuration and define it to GDDR. Discovery mode is typically run on the same system that has the GDDR Parameter wizard session active, and only there. There is no GDDR requirement to run it on any other system.

Validation mode

Run GDDRACDD in Validation mode to verify that a previously defined configuration is still correct. Validation mode is typically part of the check-list used before any GDDR script is run. Validation mode is also used at the end of the GDDRPEDD script to validate that all the changes during dynamic device add/delete result in a valid configuration.

Prerequisites

The prerequisites to run GDDRACDD are:

- GDDRSCF running with any currently supported version of Mainframe Enablers.
- Any VMAX Enginuity level currently supported with Mainframe Enablers.
- GDDRMAIN running, unless it is running in DISCOVERY mode and the COMP(Y) argument is specified.
Procedure

1. Ensure that the GDDR and prerequisite host software installation is complete on the C-System where you also have GDDRSCF running.
2. Use the GDDR Parameter wizard to navigate to the Define SRDF Devices panel.
3. When you reach the Define SRDF Devices panel, run GDDRACDD as a batch job or use the GDDRACDD panel interface. When GDDRACDD completes, return to the wizard.
4. Issue a LOAD command on the Define SRDF Device Ranges panel. This refreshes the panel with the generated RDF.Devices data.
5. Review and, where necessary, modify the RDF.DEVICES parameters created by GDDRACDD.
6. If you made any changes to the RDF.DEVICES parameters, issue a SAVE command in the Define SRDF Devices panel.
7. Repeat steps 4-6 above for TimeFinder devices, using the Define TimeFinder devices panel. Note that SRDF and TimeFinder devices can be discovered with a single run of GDDRACDD.

Implementation

GDDRACDD can be run from option G on the GDDR Maintenance and Setup Menu (see “Option G: Manage GDDRACDD” on page 123) or from a JCL provided in hlq.GDDRvrm.SAMPLIB. When using the latter option, customize the JCL provided in hlq.GDRvrm.SAMPLIB(GDDRACDD) to meet the requirements of your configuration. The following sections describe the GDDRACDD arguments, DD-cards, and parameters.

Sample JCL

GDDRACDD is invoked in a batch job.

```
//GDDRACDD JOB CSE,TF51,CLASS=A,MSGCLASS=X,NOTIFY=USER007,
//USER=GDDR,REGION=0M
/*JOBPARM  LINES=999999
//GDDRPROC JCLLIB ORDER=(EMC.GDDR.CUSTOM.LIB)
//ACTION   EXEC GDDRPROC
//GDDR.SYSTSIN DD *
EXECUTIL SEARCHDD(YES)
ISPSTART CMD(GDDRACDD 256 +
NORAID NOSIZE META +
CONFIG(DC1,CASCADDD) DLM(Y) BCV(Y) +
| DEBUG(0) TRACE(0) +
}

//GDDR.E04SRDFD DD DISP=SHR,DSN=USER007.GDDR.PARMS.WORK(E04SRDFD)
//GDDR.E05TFDEV DD DISP=SHR,DSN=USER007.GDDR.PARMS.WORK(E05TFDEV)
//GDDR.SYMDEVS DD SYSOUT=* 
//GDDR.RDFGROUP DD SYSOUT=* 
//GDDR.CREPAIR DD SYSOUT=* 
//GDDR.PBAMETA DD SYSOUT=* 
//GDDR.TFLIST DD SYSOUT=* 
://GDDR.ACCDDPARM DD *
INCLUDE DC1 GDDR I,2000,10,20
INCLUDE DC2 GDDR I,6000,10,30
INCLUDE DC1 GDDR K,2000,00,F9
INCLUDE DC2 GDDR K,6000,00,31
INCLUDE DC1 GDDR I,DLM,2000,16,26
INCLUDE DC2 GDDR I,DLM,6000,16,36
/*
//SCF$V800 DD DUMMY
```
Arguments

All GDDRACDD arguments are optional. If no arguments are specified, GDDRACDD will run with:

```
"256 ALIGN RAID SIZE META BCV (N) DLM (N) GLOBALS (NONE)
  CONFIG (DC1, CONCURRENT) SORT GROUPS (NOBREAK) NOFORCE DEBUG (0)
  TRACE (0) GDRPFX (GDDR)"
```

range_size

Default: 256

Defines the maximum number of contiguous devices to be listed as a single range in the device ranges report and in RDF.DEVICES parameters.

Supported range_size values are 0 to 4096. The recommended range_size is the default value of 256. You can use a very small value; for example, in a test environment to force GDDRACDD to generate a lot of RDF.DEVICES parameters. Larger range sizes (> 512) are intended for code testing purposes.

If you want to specify range_size, it must be the first argument. If you specify any other arguments, you must also specify a range_size. The other arguments can be specified in any order.

**Note:** For FBA-meta devices, a hard-coded maximum range_size of 128 is enforced, but you can use a smaller value.

ALIGN|NOALIGN

Default: ALIGN

This argument can be specified as ALIGN or NOALIGN. If ALIGN is specified (or defaulted), then device ranges will be aligned on boundaries defined by range_size. Alignment is supported on 256-, 512-, 1024-, 2048-, and 4096-device boundaries. If a range_size smaller than a supported alignment size is specified, range_size is rounded up to the next alignment size. GDDRACDD informs you about a range size override if this happens.

**Note:** If ALIGN is specified, and GDDRACDD discovers an asymmetrical configuration, it will issue a warning message (GDDP41SW) and change to NOALIGN automatically when the asymmetry is detected.

**ALIGN examples**

ACDD 256 ALIGN

Shows 003F-00FF then starts a new range going from 0100-013F.

ACDD 16 ALIGN

Forces the range_size to 256, and aligns as described above.

**NOALIGN example**

If NOALIGN is specified, then device ranges will not require alignment to boundaries defined by range_size, allowing device ranges to span alignment boundaries.
ACDD 256 NOALIGN

Could show a range going from 003F to 013F as a single contiguous range.

DLM(Y|N)

Default: DLM(N)
When DLM(Y) is specified, GDDRACDD performs the same discovery and validation functions for DLm Backend Devices as it does for other GDDR-managed RDF devices.
The DLm Backend Devices are identified to GDDRACDD by INCLUDE arguments with the DLM keyword. Refer to “INCLUDE” on page 251 for more information.

BCV(Y)

Default: BCV(N)
Performs discovery of TimeFinder Devices.

**Note:** The term BCV in this document refers to any VMAX device that provides a Business Continuance Volume function. This is not limited to BCV devices with the legacy TF/Mirror BCV flag. With support for TF/Clone and TF/SnapVX, GDDR considers any TimeFinder target device as a BCV in the broader sense.

When BCV(Y) is specified, GDDRACDD will discover TimeFinder devices in all or selected VMAX systems and generate GDDR BCV parameters for them. Site, VMAX systems, RDF groups, and/or BCV sets can be selected using TFINDER parameters in the ACDDPARM deck.

- **BCV(Y)** causes GDDRACDD to write PARMS.STDBCV.<t>.DCn.n=gk,bcvl-bcvh,stdl-stdh,h parameters to the E05TFDEV dataset.
- **BCV(Y,INC)** causes GDDRACDD to write legacy PARMS.BCV.DCn.ONLY.n=gk,bcvl-bcvh parameters to the E05TFDEV dataset.

**Note:** BCV(Y,INC) is only provided as a fall-back option to legacy GDDRACDD behavior. It should only be used under guidance of the EMC GDDR Solutions Support team. It is not supported when TF/Clone or TF/SnapVX support is required.

Refer to “TFINDER” on page 253 for more information about the PARMS.STDBCV GDDR parameter format.

NOSIZE, NORAID, NOMETA

These optional parameters affect the generation of RDF.DEVICES parameters. Each of them can be absent (default) or specified as NOSIZE / NORAID / NOMETA in any combination.

By default (absence of these 3 arguments), GDDRACDD generates a new device range, and consequently a new RDF.DEVICES parameter, every time any of the following attributes of the discovered SRDF devices changes:

- CKD versus FBA, meta-device versus regular device, STD device versus BCV.
Non-contiguous other-side or hopped other-side devices. (All discovered RDF groups between VMAX systems defined to GDDRACDD are considered.)

Different number of cylinders.

Different local RAID-scheme (RAID-0, RAID-1, RAID-5, RAID-6, none).

If NOSIZE, NORAILD and/or NOMETA are specified, then a difference in cylinder count or local protection scheme, or meta-type will still generate a new device range in the VMAX device ranges report, but will no longer cause an additional RDF.DEVICES parameter to be generated.

**GLOBALS(CHECK)**

When specified, runs GDDRACDD in Validation mode. The default setting is GLOBALS(NONE), which runs GDDRACDD in Discovery mode.

Validation of RDF/DLM.DEVICES parameters includes the following items, both for internal and external devices, except where noted otherwise:

- Syntactical validity of existing global variable names and values
- Existence of required RDF.DEVICES parameters for all site pairs in the configuration (internal devices only)
- Internal consistency of existing global variables against each other
- Consistency of global variables against GDDMPARM information
- Consistency of global variables against the discovered VMAX configuration:
  - Discovered devices are defined and are the correct R1/R2 type
  - Discovered devices are Link-Blocked where expected (internal groups only)
  - Discovered devices are not Link-Blocked where not expected (internal groups only)
  - Defined devices are discovered and are the correct R1/R2 type

**Note:** Errors during syntax and consistency validation of RDF/DLM.DEVICES parameters result in GDDP420E and RC=16.

For discrepancies between RDF.DEVICES parameters and devices discovered in the VMAX for internal devices, GDDRACDD issues GDDP420E and terminates with RC=8 or RC=16.

For discrepancies between RDF.DEVICES parameters and devices discovered in the VMAX for external devices, GDDRACDD issues GDDP421W, and terminates with RC=4. If this occurs during the GDDRPEDD script (GDDR Dynamic Device Add/Delete script) the RC is 0.

Validation of BCV-related parameters includes the following:

- Syntactical validity of parameter name and value
- Consistency of BCV-related parameters with GDDMPARM SYMM parameters
- Consistency of BCV-related parameters with the discovered configuration:
  - Defined BCV-devices are BCV devices (TimeFinder target devices)
  - Defined BCV devices are associated with GDDR-managed internal/external devices
  - Defined BCV devices are paired with expected STD devices
  - Sufficient BCV devices are defined and paired
  - Discovered BCV devices are defined to GDDR
- No BCV devices are defined twice in different parameters of the same set
- The defined and discovered BCV devices form a complete image of all GDDR-managed RDF devices at sites selected for GDDR BCV management
- The call override bytes for BCV management at any given site are matched by existing BCV devices at the affected site
- SnapVX target devices are linked to GDDR-managed RDF/DLM devices using the GDDR-defined snapshot names for the proper set of TF target devices

**Note:** Errors found during BCV validation result in GDDP421W and RC=0, except for GOLD/TEST set completeness validation, which results either in GDDP420E (RC=8) or GDDP421W (RC=4), depending on specifications on TFINDER parameters.

If GLOBALS(CHECK) is specified, GDDRMAIN must be running with version GDDR 5.0 or higher. COMP(Y) is mutually exclusive with GLOBALS(CHECK).

If GLOBALS(CHECK) is specified, ACDDPARM content is ignored with the exception of PATH parameters. GDDRACDD will use CSYSSITE and SYMM parameters from GDDMPARM and fabricate INCLUDE, GENERATE, TFINDER, SOFTLINK, and SNAPNAME parameters as required, using information from the existing RDF.DEVICES parameters. These fabricated parameters are echoed to the joblog prefixed with ‘SIM:’

**COMP(Y|N)**

Default: COMP(N)

COMP(Y) indicates compatibility mode. The purpose of this argument is to facilitate GDDR upgrades from earlier versions, as well as to run GDDRACDD for a configuration which is different from the one known to the running GDDRMAIN instance.

With COMP(Y), GDDRACDD will run without using GDDRMAIN services. COMP(Y) is mutually exclusive with GLOBALS(CHECK).

The requirements to run GDDRACDD without GDDRMAIN are:

- SCF is running.
- Proper SCF$nnnn DD DUMMY card is in the GDDRACDD JCL.
- IKJTSOxx must have all required GDDRDQEx entries as specified in “SYS1.PARMLIB(IKJTSOxx) customization” on page 48.
- gddrhlq.LINKLIB must be available to the job running GDDRACDD.
- CSYSSITE and SYMM parameters are specified in ACDDPARM.

**SCAN**

Indicates SCAN mode. Can be absent, or specified as SCAN.

If the SCAN argument is specified, GDDRACDD performs a validation of the arguments and parameters, as well as a limited infrastructure validation (RDF group status). It will not proceed to VMAX device discovery.
Using GDDR Utilities

FORCE|NOFORCE

Default: NOFORCE

By default GDDRACDD will stop immediately if certain configuration errors are found. See message GDDP420E in the *GDDR Message Guide* for a description of the types of errors detected. Cases where RC=8 is set can be rerun with FORCE, but there is no guarantee that the generated RDF.DEVICES parameters (if any) will be usable by GDDR.

CONFIG=<cpds>,<topology>)

Default: CONFIG(<my-site>,CONCURRENT)

The CONFIG argument affects the syntax of the CREATEPAIR commands generated by GDDRACDD. It has no effect on configuration validation or generation of RDF.DEVICES parameters.

CONFIG informs GDDRACDD about the intended Primary DASD site (CPDS) and, for 3-site configurations only, about the intended topology (Concurrent or Cascaded). Topology is ignored for 2-site and 4-site configurations.

By default, GDDRACDD will use the site where it is running as CPDS. In 3-site configurations the default GDDRACDD topology is Concurrent. In all other configurations, topology is ignored.

SORT|NOSORT

Default: SORT

NOSORT suppresses sorting of generated parameters. By default, GDDRACDD sorts the generated RDF.DEVICES parameters in turn over the available VMAX systems. Internal RDF groups are sorted separately from external RDF groups.

Sorting enhances GDDR script performance by increasing parallel processing over available resources. It provides the most benefit for users with multiple VMAX systems per site.

Note the following:

- If one site of a site pair has a lower number of VMAX systems than the other site, the sort algorithm will do the distribution using the site with the lowest number. This avoids cases where a site with low parallel processing capabilities would get overrun by parallel processing initiated from the other site.
- This sorting of generated parameters does not force a strict sequence of execution of script commands in the desired round-robin fashion, but it increases the likelihood of an optimal sequence.
- If the managed configuration has an uneven distribution of device ranges across the VMAX systems there will be a "tail-end" effect where the remaining commands are all targeting the same VMAX system.
GROUPS(BREAK|NOBREAK)

Default: GROUPS(NOBREAK)

Discovered VMAX device ranges can be paired in multiple RDF groups, currently limited to two by Enginuity. By default, GDDRACDD generates RDF.DEVICES parameters for each site pair independently of device pairing for other site pairs.

If GROUPS(BREAK) is specified, then if a range needs to be split due to an interruption in the device pairing in one RDF group, it will also cause a split in RDF.DEVICES parameters for the other RDF group. As a result there will be a higher total number of RDF.DEVICES parameters. This number will be higher than necessary for GDDR scripts to succeed. However, the generated RDF.DEVICES parameters will be symmetrical in nature when looking at parameters for different site pairs.

DEBUG(n)

TRACE([name,]n)

GDDRACDD does not use the GDDR standard method to set DEBUG and TRACE levels, as those require GDDRMAIN services for global variable access. The DEBUG and TRACE arguments close this gap.

DEBUG can be specified as 0, 1, or 2. Default: DEBUG(0)

**WARNING**

Use TRACE only under instruction from the GDDR Solution Support team. Tracing of GDDRACDD can produce enormous amounts of output in the SYSTSPRT dataset, and could consume all available spool space. The default value is TRACE(0).

The syntax for TRACE is (<sub-routine-name>,<trace-level>). Subroutine-names are provided by the GDDR Solution Support team. The default subroutine name is "ALL". TRACE level can be 0, 1, 2, or 3 and translates to REXX tracing values of "O", "R", "I", and "A" respectively.

GDDRPFX(GDDn)

Default: GDDRPFX(GDDR)

Indicates the target GDDR subsystem.

Optional DD-cards

In addition to the arguments, there are six optional DD-cards in the GDDRACDD JCL which slightly affect GDDRACDD behavior. If these DD-cards are not present, the affected report is not written.

E04SRDFD

The E04SRDFD dataset contains the generated RDF.DEVICES parameters. It is necessary for proper integration of GDDRACDD with the GDDR Parameter Wizard.
Using GDDR Utilities

E05TFDEV

The E05TFDEV dataset contains the generated STDBCV.<t>.DCn or BCV.DCn.ONLY parameters. It is necessary for proper integration of GDDRACDD with the GDDR Parameter Wizard.

SYMMDEVS

Contains the VMAX device ranges report. This used to be part of the joblog, but is now written to its own separate dataset. This provides better visibility to various warning and error messages written to the joblog. EMC recommends having this DD card in the JCL.

RDFGROUP

Contains the following sections:

- A listing of all GDDR groups found in VMAX systems defined on SYMM parameters. Indicates whether a group is included as regular DASD is DLM Backend, internal, external or not at all, whether the group is OFFLINE, and whether the group is Virtual.
- A listing of included groups by type and site pair, with associated gatekeepers.

CREPAIR

Contains the Host Component CREATEPAIR commands which can be used to create the configuration as it will be defined to GDDR, from scratch, in cases where it got deleted. The CREATEPAIR commands are generated according to the intended configuration specification in the CONFIG argument.

FBAMETA

The FBAMETA report contains an overview of FBA meta-devices in the GDDR-managed configuration. It shows head-member relationships and SRDF device pairing information.

TFLIST

The TFLIST report contains an overview of GDDR-managed BCV-devices with BCV-STD pairing information.

Parameters (ACDDPPARM DD statement)

GDDRACDD supports the following parameters to control its actions:

- SYMM
- CSYSSITE
- INCLUDE
- GENERATE
- PATH
- TFINDER
- SOFTLINK
- SNAPNAME
- **VGROUP**

  All GDDRACDD parameters are optional, except for the INCLUDE parameter. If GDDRMAIN is not running, and/or the COMP(Y) argument is specified, then CSYSSITE and SYMM parameters are required. If CSYSSITE or SYMM parameters are specified in ACDDPARM, then GDDRACDD will use those parameters, even if GDDRMAIN is running. All ACDDPARM parameters, except PATH, are ignored when running GDDRACDD in Validation mode (GLOBALS(CHECK) argument).

Parameter rules

ACDDPARM has specific syntax rules. All parameters are column-sensitive. You can specify GDDRACDD parameters in any sequence.

Parameter structure (columns 1-70 usable):

```
00000000001111111111222222222233333333334444444444555555555555555555
1234567890123456789012345678901234567890123456789012345678901234567890
PARMNAME  REF-id   GDDx value
```

- **Parameter name** goes in columns 1-8.
- **System name or a Site name or a Group number**, goes in columns 10-17.
- **GDDx** goes in column 19-22. This is the **targeted GDDR subsystem**.
- **Parameter values** go in columns 24-71.

**CSYSSITE**

The CSYSSITE parameter links a system name to a site.

For example:

```
CSYSSITE 0016   GDDR DC1
```

The CSYSSITE parameter tells GDDRACDD the site on which it is running. GDDRACDD needs to know this, to be able to properly construct createpair commands, if requested, and to select appropriate PATH parameters if any are specified.

GDDRACDD does not run on a system that is not named in a CSYSSITE parameter.

The recommendation is to code one CSYSSITE parameter for every C-System in your configuration. This makes your GDDRACDD parameter deck portable. However, GDDRACDD will work with only one CSYSSITE coded; the one for the system where you intend to run GDDRACDD next.

CSYSSITE parameters in ACDDPARM are optional if GDDRMAIN is running and you are satisfied with the CSYSSITE parameters in GDDMPARM. They are required if you specify COMP(Y) in the arguments.

If you specify any CSYSSITE parameter in ACDDPARM, then any CSYSSITE parameter in GDDMPARM is ignored.

**SYMM**

The SYMM parameter defines a VMAX system to GDDRACDD, as well as the pool of gatekeeper addresses to be distributed by GDDRACDD over the generated RDF/DLM.DEVICES parameters.
For example:

SYMM DC1(H) GDDR 000292601023,2000-200F

GDDRACDD requires one SYMM parameter with (H) Home indicator in the site-id column, for every VMAX system in the GDDR-managed configuration. You can code as many SYMM parameters as needed to define the gatekeeper pools. These can be lists of single devices or device ranges. Gatekeeper addresses must be unique at a site.

SYMM parameters in ACDDPARM are entirely optional if GDDRMAIN is running and you are satisfied with the SYMM parameters in GDDMPARM. They are required if you specify COMP(Y) in the arguments.

**Note:** SYMM parameters are only available in GDDMPARM with GDDR Version 4.0 or higher.

If you code SYMM parameters in ACDDPARM, you can copy and paste all SYMM parameters from GDDMPARM. SYMM parameters without the (H) Home indicator in the site-id column are only used by GDDRACDD if they refer to the site where GDDRACDD is running. In that case, GDDRACDD will use the first gatekeeper listed as a discovery path for the named VMAX, unless there is an explicit PATH parameter for the same VMAX system.

If you specify any SYMM parameter in ACDDPARM, then any SYMM parameter in GDDMPARM is ignored.

**Note:** When using GDDRACDD SYMM parameters, ensure they specify the same GK devices as those on the corresponding GDDRMAIN GDDMPARM SYMM parameters. This ensures that the RDF.DEVICES, DLM.DEVICES, and STDBCV ranges returned by GDDRACDD will pass validation. “SYMM parameters” on page 77 provides further details.

PATH

By default GDDRACDD will determine the discovery path for every defined VMAX system automatically. This is true whether the SYMM parameters are taken from ACDDPARM or from GDDMPARM. The algorithm selects the path with the fewest possible hops. For remote VMAX systems, the selected path is the one with the highest number of online local and remote RDF directors.

If the automatically selected path is unacceptable, a PATH parameter can be specified in ACDDPARM to force usage of the specified path for discovery.

For example:

PATH DC1 GDDR 123456789014,A000.10.30

The site specified on PATH parameters, or “from” site, determines whether or not the PATH parameter will be used. GDDRACDD determines the site where it is running based on CSYSSITE parameters. Any PATH parameter referring to a different site is ignored. This again allows for a portable ACDDPARM parameter deck.

The PATH parameters further specify a target VMAX system by serial number, and finally the discovery path. This path is made up of a gatekeeper address, which must be usable at the “from” site, followed by as many “hops” as required to reach the target VMAX system, in a dot-separated list. The number of hops that can be specified is limited to eight.
INCLUDE

The mandatory INCLUDE parameter instructs GDDRACDD whether to include discovered SRDF devices in the generation of GDDR RDF.DEVICES or DLM.DEVICES parameters. Inclusion is done at the RDF group level and can include consistency protected (internal) RDF groups as well as RDF groups outside of the consistency protection (external groups) both for “regular” DASD devices and for DLm Backend Devices.

Examples:

INCLUDE DC1 GDDR I,2000,10,20
INCLUDE DC2 GDDR I,6000,30
INCLUDE DC1 GDDR E,2000,00,F9
INCLUDE DC2 GDDR E,6000,31
INCLUDE DC1 GDDR I,DLM,2000,16,26
INCLUDE DC2 GDDR I,DLM,6000,36

Note: The INCLUDE parameter limits the creation of RDF.DEVICES and DLM.DEVICES parameters to included groups. It has no effect on the discovery process. GDDRACDD will always perform a complete discovery of all devices in all VMAX systems defined to it on SYMM parameters. The discovery is also not limited by SCF.CNTRL.EXCLUDE.LIST parameters in SCFINI. SCF.DEV.EXCLUDE.LIST parameters in SCFINI should be considered for the selection of discovery paths specified on PATH parameters.

INCLUDE parameters are required and GDDRACDD will not run if no INCLUDE parameters are found in ACDDPARM. If GDDRACDD runs in validation mode, it ignores any INCLUDE parameters in ACDDPARM and generates syntactically valid INCLUDE parameters, based on the existing RDF.DEVICES and DLM.DEVICES parameters. Every group for which RDF.DEVICES or DLM.DEVICES parameters are needed must have an INCLUDE parameter. There is only one exception to this, which is discussed in “GENERATE” on page 252.

Note: Each RDF group connects two VMAX systems, but only one INCLUDE is necessary per group, as the other end is automatically included.

Each INCLUDE parameter must specify:

- An include-type, which is either ‘I’ (Internal) or ‘E’ (External)
- A gatekeeper address. This is any MVS address that locates the intended VMAX system and was specified in the relevant gatekeeper pool-set on a SYMM parameter either in ACDDPARM or GDDMPARM.
- A comma-separated list of RDF groups contained in the VMAX system located by the specified gatekeeper. This can be any mixture of SRDF/S and SRDF/A groups.

INCLUDE parameters can optionally have a DLM keyword, which identifies them for generation of DLM.DEVICES parameters as opposed to RDF.DEVICES parameters. If specified, the DLM keyword must follow the Include-type (‘I’ or ‘E’) immediately.

Note: The RDF.DEVICES GDDR variables include both regular DASD devices and DLm backend devices. The DLM.DEVICES variables include only the DLm backend devices. This allows GDDR to manage DLm backend devices in the same script step as the regular DASD devices, as well as managing them separately for those actions that do not apply to regular DASD devices.
Note: All RDF groups listed on INCLUDE parameters must have SRDF device pairs in the VMAX system. Refer to “GENERATE” on page 252 for a discussion of the creation of RDF/DLM.DEVICES parameters for groups that do not have any SRDF device pairs.

Rules for GDDR-managed RDF groups:

1. All RDF groups known to GDDR must be “pure”: either all consistency protected devices or all external devices; no mixture is allowed within the same RDF group. Either regular DASD or DLm backend, no mixture is allowed within the same RDF group.

2. Star rule: Internal SRDF/A leg devices must be concurrent or cascaded with an SRDF/S leg RDF group which must be an internal RDF group.

3. SQAR rule: All SQAR devices (these are in internal groups by definition) must be protected in RDF groups so that they form closed “squares” at the device level.

4. External SRDF/S-leg devices can be concurrent or cascaded with an SRDF/A leg RDF group, which must also be an external RDF group.

5. External SRDF/A-leg devices can be concurrent or cascaded with an SRDF/S leg RDF group, which can be an external RDF group or a consistency protected RDF group.

GENERATE

The optional GENERATE parameter is a comma-separated list of pairs of RDF groups. It is required only in 3-site configurations and only if the devices at DC3 are only paired in one RDF group and there is a need for GDDRACDD to create RDF.DEVICES parameters for non-existing device pairs in one of the groups at DC3.

GENERATE parameters are ignored in 2-site configurations. They are allowed in 4-site configurations to enable generation of RDF.DEVICES parameters and CREATEPAIR commands for the “cross-hair” RDF groups connecting DC1 to DC4 and DC2 to DC3. GDDR support for SRDF/SQAR does not include usage of these “cross-hair” connections.

The site specified on GENERATE parameters must be DC3.

In 4-site configurations it can be DC3 or DC4.

Examples:

| GENERATE | DC3       | GDDR I, A000, (20, 30), (70, 60) |
|          | DC3       | GDDR E, A000, (F9, 31), (61, 71) |

Each GENERATE parameter must further specify:

- A type, which is either ‘I’ (Internal) or ‘E’ (External). This type must match the INCLUDE type of one the paired groups.
- A “gatekeeper” address: this is any MVS address that locates the intended VMAX system and was specified in the relevant gatekeeper pool-set on a SYMM parameter either in ACDDPARM or GDDMPARM.
- A list of pairs of RDF groups. Within each pair, at least one group must have been previously specified on an INCLUDE parameter with the same type as specified on the GENERATE parameter. The second group is automatically included, if it was not included earlier. One of the groups must have existing device pairs. Only one of the paired groups can be a virtual group (see “VGROUP” on page 258).
3-Site configurations — Generation for non-existing device pairs DC2-DC3 requires existing device pairs DC1-DC3 and DC1-DC2. Generation for non-existing device pairs DC1-DC3 requires existing device pairs DC2-DC3 and DC2-DC1. Generation for non-existing device pairs DC1-DC2 is not supported.

4-Site configurations — Generation for non-existing device pairs DC2-DC3 requires existing device pairs DC1-DC3 and DC1-DC2. Generation for non-existing device pairs DC1-DC4 requires existing device pairs DC2-DC4 and DC2-DC1. Generation for non-existing device pairs DC1-DC2, DC1-DC3, DC2-DC4 and DC3-DC4 is not supported.

TFINDER

Note: Mainframe Enabler software supports the association of multiple BCV devices to a single source device. GDDRAACDD will discover up to 2 sets of BCVs. To reflect their different GDDR use cases, these sets are named the GOLD set and the TEST set.

The BCV GOLD set is used during GDDR scripts to preserve a golden copy of the data from GDDR managed R2-devices, prior to an SRDF R1>R2 resynchronization.

The BCV TEST set is intended for TEST-IPL purposes. When running a GDDR Test-IPL from BCV script, if a TEST-set of BCVs is defined to GDDR for the selected site and you want to exploit this feature, the script will split off this TEST set and do the IPL off of those devices. This leaves the GOLD set of BCVs intact, so it can be used immediately if disaster strikes during a TEST-IPL scenario.

The optional TFINDER parameter serves the following purposes:

- For VMAX 10K, 20K, and 40K systems, TFINDER selects the Legacy TF method (TF/Mirror or TF/Clone) to be used by site.
  
  - If SnapVX with soft linking is required, GDDR uses TF/Clone on VMAX 10K, 20K, and 40K systems, ignoring the Legacy TF method selection.
  
  - If TF Clone is selected as the Legacy TF method, then GDDR will use SnapVX with soft linking on VMAX 100K, 200K or 400K systems even if soft linking support is not required by the user.

- Limits discovery of TimeFinder devices to specified sites and VMAX systems.

- Enforces selection of the gatekeeper address to be used by GDDR on TimeFinder commands for the listed VMAX system.

TFINDER parameters are ignored when running with BCV(N) or GLOBALS(CHECK). If running with GLOBALS(CHECK), GDDRAACDD will fabricate appropriate TFINDER parameters, based on:

- Call Override bytes 12 (DC4), 18/19/20 (DC1/2/3)

- The Manage EXT BCV configuration option: PARMS.BCVTYPE.EXT=Y/N and associated call-override-like variable: USEBCV.EXT

- The Manage TEST BCV configuration option: PARMS.BCVSET.TEST=Y/N and associated call-override-like variable USEBCV.TEST.DCn

- The selected Legacy TF method: MFEOPTS.TFMETHOD.<site>=MIRR|CLON
If no TFINDER parameters are coded, GDDRACDD will discover GOLD-set BCV devices only, for GDDR internal devices only, and use TF/Mirror as the Legacy TF method for all sites.

Examples

1. Manage GOLD-set internal BCV devices uniformly at all sites, but none at DC1. Issue warning-level messages if the GOLD-set is found to be incomplete. Use TF/Clone as the Legacy TF method for all sites.

   TFINDER  ALL       GDDR CLON, RA(INT), GOLD(WARN)
   TFINDER  DC1       GDDR NONE

2. Manage BCV devices at DC3 only, discover both a GOLD set and a TEST set, for both internal and external devices, issue error messages if the GOLD set is incomplete, and warning messages if the test set is incomplete. Use TF/Mirror as the Legacy TF method.

   TFINDER  DC3       GDDR RA(ALL), GOLD(ERROR), TEST(WARN)

Subparameters

Each TFINDER subparameter must specify the following:

- A Site (or ALL) as the second word.
- The following information in the fourth word:
  - Either the word NONE (excludes the site from BCV discovery)
  - Or the following sub-parameters:
    - MIRR or CLON as the selected Legacy TF method (optional)
    - RDF group specification (required)
    - BCV set specification (GOLD and/or TEST) (required)
    - GK specification (optional)

Required: Site: DCn|ALL

Limits BCV discovery to the sites listed on TFINDER parameters. If TFINDER parameters are coded where Site=ALL, then any other TFINDER parameters not specifying Site=ALL must specify the RDF group as NONE.

Required: RDF group selection: RA(INT|EXT|ALL|ra,ra,...)|NONE

Limits the BCV set selection to the GDDR-managed RDF groups specified here, either by type or by RDF group number. Can be specified as:

- NONE: excludes the site from BCV discovery
- RA(INT|EXT|ALL): discover BCVs for INT/EXT/ALL GDDR-managed RDF groups.
- RA(ra,ra,ra,...): an explicit list of RDF groups.

The RDF group selection can only be specified with an explicit list of RDF groups, if there is only 1 GK() coded, and the site is not specified as ALL.

You can use RA(INT|EXT|ALL) parameters to set defaults for a site or a specific system, along with parameters for specific groups for which you want a different BCV discovery.

Note: It is not recommended to use the RA() sub-parameter to restrict BCV discovery to certain RDF-groups in a system. This capability is provided for lab backwards compatibility with traditional GDDR behavior.
Required: BCV-Set selection and BCV-protection validation message level: {GOLD(WARN|ERROR), TEST(WARN|ERROR)}

**Note:** This subparameter is required unless the RDF group specification is NONE.

This subparameter indicates which sets of BCVs GDDRACDD should discover. GDDR-managed devices can have multiple BCV devices associated with them. GDDRACDD will discover up to two. If two are found, then one could be used in the GOLD set and the other in the TEST set. The GOLD set is SPLIT in GDDR scripts to preserve a golden copy during SRDF resynchronization operations. The TEST set will be used in GDDR scripts performing a test-IPL from BCVs.

During discovery of the configuration, GDDRACDD validates that ALL devices in selected RDF groups have an associated BCV in the sets selected here. If exceptions are found, GDDRACDD will issue either Warning or Error level messages, depending on the specification in this sub-parameter.

Either the GOLD set or the TEST set (or both) must be selected. If both GOLD(...) AND TEST(...) are specified:

- If ACDD finds 2 BCVs associated with a STD device, then the low BCV device number goes in the GOLD set and the high BCV number goes in the TEST set.
- If only 1 BCV is found for a STD device, it goes in the GOLD set and there is a W or E-level message for the Test set.

If only GOLD(...) OR TEST(...) are specified, then if more than 1 BCV is found associated with a STD device, the lowest BCV device number goes in the selected set.

**Optional: GK-selection: GK(gk,gk,gk,...)**

Default: no GK selection (select ALL boxes at DCn)

This subparameter serves two purposes:

- Limits BCV-discovery at the affected site to the VMAX units identified by the listed gatekeepers.
- Forces GDDRACDD to use the listed gatekeeper for all BCV parameters defining BCVs in the affected VMAX system. If no GK() sub-parameter is coded, GDDRACDD uses the first gatekeeper from the pool defined in GDDMPARM for the affected VMAX systems.

If more than one gatekeeper is listed, then the RA(...) subparameter must be coded as INT, EXT or ALL. You can code as many TFINDER parameters for the same site as needed to list all required gatekeepers.

**Note:** It is not recommended to use the GK() subparameter to restrict BCV discovery to certain VMAX systems at a site. This possibility is provided for backwards compatibility with traditional GDDR behavior.

**Optional: Legacy TF method selection: MIRR|CLON**

Default: MIRR

GDDR supports TF/Mirror and SnapVX on VMAX 100K, 200K or 400K systems, as well as the legacy local replication methods TF/Mirror and TF/Clone on VMAX 10K, 20K, 40K systems.
The specification of a Legacy TF method affects GDDR behavior during scripts as follows:

- If SnapVX soft linking support is required, then that method will be used on VMAX 100K, 200K or 400K systems, while TF/Clone will be used on VMAX 10K, 20K, 40K systems, ignoring any specification of a Legacy TF method.

- If SnapVX soft linking is not required, the behavior is determined by the specification for the Legacy TF method.
  - If the Legacy TF method is MIRR, GDDR issues TF/Mirror commands across all VMAX hardware generations. TimeFinder software translates these to SnapVX commands using hard linking on VMAX 100K, 200K or 400K systems.
  - If the Legacy TF method is CLON, GDDR issues TF/Clone commands on VMAX 10K, 20K, 40K systems and SNAP VX commands with soft linking on VMAX 100K, 200K or 400K systems.

**Default TFINDER parameter**

Specifying no TFINDER parameters has the same effect as:

```
TFINDER ALL GDDR MIRR,RA(INT),GOLD(WARN)
```

This statement means:

- Generate PARMS.STDBCV.<t>.DCn or PARMS.BCV.DCn.ONLY for internal RDF groups for all systems at all sites.
- Use the first GDDMPARM SYMM pool gatekeeper for each system.
- Issue warning messages if any internal devices are found not to have a BCV associated.
- Use TF/Mirror as the Legacy TF method on VMAX 10K, 20K, 40K systems at all sites (unless SnapVX with soft linking is required, in which case TF/clone is enforced on VMAX 10K, 20K, 40K systems).

**Global variables**

When GDDRACDD runs in Discovery mode and BCV(Y,<option>), it prints global variables to define the discovered BCV devices to GDDR in the E05TFDEV dataset. The type of variables printed by GDDRACDD depends on how it is invoked.

When running with BCV(Y), GDDRACDD writes a new global variable type to E05TFDEV:

```
PARMS.STDBCV.<t>.DCn.n=gk,bcvl-bcvh,stdl-stdh,<h>
```

Where <t> can have one of the following values:

- 1 = GOLD set BCV devices associated with GDDR managed internal devices
- 2 = GOLD set BCV devices associated with GDDR managed external devices
- 3 = TEST set BCV devices associated with GDDR managed internal devices
- 4 = TEST set BCV devices associated with GDDR managed external devices

stdl-stdh is the GDDR-managed STD device range associated with the listed BCV range.

<h> is an indication of the host type: C for CKD, F for FBA.
When running with BCV(Y,INC), GDDRACDD discovers and returns messages about all requested types and sets of BCV devices, but only writes the traditional global variables for GOLD set internal devices:

```
PARMS.BCV.DCn.ONLY.n=gk,bcv1-bcvh
```

**Note:** Running with BCV(Y,INC) is intended as a fallback option to legacy behavior and should only be used under guidance of the EMC GDDR Solutions Support team. It is NOT supported when TF/Clone or SnapVX soft linking support is required.

### SOFTLINK

The optional SOFTLINK parameter specifies whether SnapVX soft link support is required for a site. For example,

```
SOFTLINK DC3       GDDR YES
```

By default GDDR uses legacy TF/Mirror with Clone emulation across all generations of VMAX systems present in the configuration. By specifying a SOFTLINK parameter with value YES for a site, you can instruct GDDR to use SnapVX with soft linked targets on VMAX 100K, 200K or 400K systems for that site. GDDR will use TF/Clone Mainframe SNAP on VMAX 10K, 20K, 40K systems on that same site.

SnapVX with soft linked targets requires Mainframe Enablers V8.0 or higher, VMAX 100K, 200K or 400K systems, and the specification of at least one SNAPNAME parameter. SOFTLINK parameters are not supported when running GDDRACDD in legacy BCV discovery/validation mode using the BCV(Y,INC) argument.

### SNAPNAME

The optional SNAPNAME parameter specifies the 1-32 character name of the snapshot to be used by GDDR for the indicated set of TF target devices (GOLD vs. TEST, INT vs. EXT). For example:

```
SNAPNAME          GDDR GOLD,INT,GDDR_INT_MY_SNAP
```

At least one SNAPNAME parameter must be provided when a SOFTLINK parameter with value YES is specified. SNAPNAME parameters have the same environmental requirements as indicated for the SOFTLINK parameter. Note that SNAPNAMEs defined to GDDR are not qualified by SITE. A set of TF target devices is managed using the same snapshot name, regardless of the site to which it belongs.

GDDR managed snapshot names have the following requirements:

- For internal devices use prefix: GDDR_INT_
  
  GDDR_INT_* snapshot names must be used for devices in or paired with consistency protected RDF groups (ConGroup and/or MSC).

- For external devices use prefix: GDDR_EXT_
  
  GDDR_EXT_* snapshot names must be used for devices in or paired with ADCOPY-DISK RDF groups.

You can append up to 23 characters to form a syntactically valid snapshot name.
Using GDDR Utilities

Note: In SRDF/Star configurations, at DC1 and DC2, some devices can be ConGroup protected (SRDF/S) DC1-DC2, while being in ADCOPY-DISK mode DCn-DC3. These devices must use the GDDR_INT_* snapshot names. The GDDR_EXT_* snapshot names are reserved for devices exclusively replicated in ADCOPY-DISK mode or paired with such devices.

VGROUP

The VGROUP parameter defines a virtual RDF group to GDDRACDD. For example:

VGROUP  50       GDDR  123456789012,123456789013

These virtual groups have no meaning outside of GDDRACDD, and their use is limited to situations where GDDRACDD needs to run in an incomplete configuration. In a 3-site configuration, if there is SRDF-connectivity from DC1 to DC2 (required) and from DC1 to DC3 (required), then the VGROUP parameter allows GDDRACDD to generate correct parameters even if there is no connectivity from DC2 to DC3. (The roles of DC1 and DC2 can be inverted above.)

VGROUP parameters are ignored in 2-site configurations. They are allowed in 4-site configurations with the same caveats as noted for the GENERATE parameter.

Virtual groups can only be referenced on GENERATE parameters.

Each VGROUP parameter specifies an RDF group number, and a pair of VMAX systems to be considered connected by the specified RDF group number. The group cannot exist in either of the VMAX systems. In 3-site configurations, only one of the connected VMAX systems must be located at DC3. In 4-site configurations, the virtual group must connect either DC1-DC4 or DC2-DC3.

Parameter examples

This section provides examples illustrating the use of the GDDRACDD parameters.

Example 2-Site SRDF/A

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSYSSITE 0016</td>
<td>GDDR DC1</td>
</tr>
<tr>
<td>CSYSSITE 001E</td>
<td>GDDR DC3</td>
</tr>
<tr>
<td>SYMM DC1(H)</td>
<td>GDDR 123456789011,1000-100F</td>
</tr>
<tr>
<td>SYMM DC3(H)</td>
<td>GDDR 123456789033,5000-500F</td>
</tr>
<tr>
<td>INCLUDE DC1</td>
<td>GDDR I,1008,13</td>
</tr>
<tr>
<td>INCLUDE DC1</td>
<td>GDDR E,1007, E3</td>
</tr>
<tr>
<td>TFDINDER DC3</td>
<td>GDDR RA(13), TEST(ERROR), GK(5005)</td>
</tr>
<tr>
<td>SOFTLINK DC3</td>
<td>GDDR YES</td>
</tr>
<tr>
<td>SNAPNAME</td>
<td>GDDR TEST, INT, GDDR_INT_TEST_SNAP</td>
</tr>
</tbody>
</table>

Comments:
GDDRACDD will be allowed to run on systems named 0016, and 001E. If GDDRACDD detects it is running on 001E it knows it is running at DC3.

Two VMAX systems are defined for discovery, one at each site.

Of all the RDF groups found in these two VMAX systems, you only want to generate RDF-DEVICES parameters for the devices in group 13 in the system at DC1. External RDF-DEVICES parameters will be generated for group E3 in the system at DC1. Referring to
Figure 97, groups 13 and E3 are automatically included at DC3.

GDDRACDD will automatically select a discovery path for all VMAX systems.

Only BCV devices associated with GDDR-managed devices in RDF group 13 in system 123456789033 will be discovered. Only 1 BCV per STD device will be discovered, and it will be assigned to the TEST set. If some devices in group 13 are found not to have an associated BCV, error-level messages will be issued. The TimeFinder implementation at DC3 is SnapVX with soft-linked target devices, and GDDR will use the name "T.I_SNAPGDDR_INT_TEST_SNAP" to link target devices to the GDDR-managed RDF devices in group 13.

If GDDRMAIN is running and has the expected CSYSSITE and SYMM parameters, and COMP(Y) is not specified in the arguments, the minimal required ACDDPARM deck for this configuration would have been:

```
INCLUDE DC1      GDDR I,1008,13
INCLUDE DC1      GDDR E,1008,E3
```

The same result would be achieved with:

```
INCLUDE DC3      GDDR I,5008,13
INCLUDE DC3      GDDR E,5008,E3
```

Output

Note: Depending on the Mainframe Enablers version, GDDRACDD displays VMAX device numbers and RDF groups either as 4/2 digits (Mainframe Enablers V7.6) or 8/8 digits (Mainframe Enablers V8.0 and higher). These displays are no indication of current or future limits on the number of VMAX devices and RDF groups supported by EMC. In this guide, the 4/2 notation is used.

All messages in JESMSGLG are also visible in SYSTSPRT. In the sample message text below, RDF.DEVICES can be replaced by DLM.DEVICES in relevant cases.
Using GDDR Utilities

Information about arguments

GDDRACDD will always issue a message about the Range size and Alignment option used during each GDDRACDD run. The range size you see in the GDDRACDD output message information could be different from the one you specified in the run time arguments. If ALIGN is used, then range_size is rounded up to the next higher supported alignment boundary. If you specify ALIGN and a range size not supported with ALIGN, an additional informational message is issued:

GDDP415I GDDRACDD using Range size 256, ALIGN, NOSIZE, NORAID, and NOMETA
GDDP415I Range size 32 overridden due to ALIGN

Depending on the presence/absence of the optional NOSIZE argument:

GDDP415I GDDRACDD RDF.DEVICES parameters will span different device sizes
GDDP415I GDDRACDD RDF.DEVICES parameters will not span different device sizes

Depending on the presence/absence of the optional NORAID argument:

GDDP415I GDDRACDD RDF.DEVICES parameters will span different protection types
GDDP415I GDDRACDD RDF.DEVICES parameters will not span different protection types

Depending on the presence/absence of the optional NOMETA argument:

GDDP415I GDDRACDD RDF.DEVICES parameters will span meta and non-meta devices
GDDP415I GDDRACDD RDF.DEVICES parameters will not span meta and non-meta devices

If COMP(Y) is specified:

GDDP415I GDDRACDD running in Compatibility mode

If GLOBALS(CHECK) is specified:

GDDP415I GDDRACDD will perform validation of existing RDF.DEVICES parameters

Depending on whether GROUPS(BREAK) is specified:

GDDP415I Device range breaks in one ra-group will [not] affect device ranges in a second group

If FORCE is specified:

GDDP415I RDF.DEVICES parms will be generated, even if configuration errors are found

If NOSORT is specified:

GDDP415I RDF.DEVICES parms will NOT be sorted for optimal performance

If SCAN is specified:

GDDP415I GDDRACDD will stop after validation of parameters

Depending on the CONFIG(<site>,<topology>) argument:

GDDP415I Using Configuration description: <site>,<topology>
Recognized ACDDPARM statements

Recognized ACDDPARM statements, information retrieved from GDDMPARM, global variables used for “validation mode”, and discovery path information

All recognized ACDDPARM statements are echoed here, grouped by type. Please verify that this matches the input deck you thought you coded. This section also shows ACDDPARM lines that were ignored (except comments).

GDDP415W The following lines in ACDDPARM were ignored:
GDDP415I STORAGEx DC1 GDDR 1008 1008 5008,30

GDDP415I GDDMPARM: Sitelist: DC1 DC3
GDDP415I GDDMPARM: and C-systems: CRKB CRKD
GDDP415I
GDDP415I GDDRACDD running at DC1
GDDP415I
GDDP415I ACDDPARM: SYMM DC1(H) GDDR 000292601023,2000-200F
GDDP415I ACDDPARM: SYMM DC2(H) GDDR 000292601156,6000-600F
GDDP415I ACDDPARM: SYMM DC3(H) GDDR 000292601024,A000-A00F
GDDP415I
GDDP415I Selected Discovery paths
GDDP415I Site Symmetrix Path
GDDP415I DC1 000292601023 2000
GDDP415I DC2 000292601156 2000.10
GDDP415I DC3 000292601024 2000.F9
GDDP415I
GDDP415I ACDDPARM: INCLUDE DC1 GDDR I,2000,10,20
GDDP415I ACDDPARM: INCLUDE DC2 GDDR I,6000,10,30
GDDP415I ACDDPARM: INCLUDE DC1 GDDR E,2000,00,F9
GDDP415I ACDDPARM: INCLUDE DC2 GDDR E,6000,00,31
GDDP415I
GDDP415I ACDDPARM: GENERATE DC1 GDDR I,A000,(20,30)
GDDP415I ACDDPARM: GENERATE DC3 GDDR E,A000,(F9,31)
GDDP415I

If GENERATE parameters are present for a 2-site configuration, a warning message is issued:
GDDP415W GENERATE parameters ignored with 2-site configurations

If GDDRACDD runs with GLOBALS(CHECK) it echoes the relevant existing global variables:

GDDP415I
GDDP415I CONFIG: CONFIG.SRDFS=1
GDDP415I CONFIG: CONFIG.AUTOSWAP=1
GDDP415I CONFIG: CONFIG.SRDFA=1
GDDP415I CONFIG: CONFIG.STAR=1
GDDP415I CONFIG: CONFIG.R22=1
GDDP415I CONFIG: CONFIG.SQAR=0
GDDP415I CONFIG: Dasd Sitelist = DC1 DC2 DC3
GDDP415I CONFIG: C-Sitelist = DC1 DC2 DC3
GDDP415I CONFIG: Master C-system = VC1A
GDDP415I CONFIG: Primary DASD Site = DC1
GDDP415I CONFIG: Secondary DASD Site = DC2
GDDP415I CONFIG: Current Topology = Concurrent
GDDP415I CONFIG: SRDF/Star with AutoSwap
GDDP415I CONFIG: Expecting Link-Block at DC3 (from DC2)
GDDP415I CONFIG: Expected SRDF Device Pairing:
GDDP415I CONFIG: DC1-DC2: R1
GDDP415I CONFIG: DC1-DC3: R1
GDDP415I CONFIG: DC2-DC1: R2
GDDP415I CONFIG: DC2-DC3: R1
GDDP415I CONFIG: DC3-DC1: R2
GDDP415I CONFIG: DC3-DC2: R2
Using GDDR Utilities

GDDP415I CONFIG: Existing RDF.DEVICES parameters [can be repeated for DLM]
GDDP415I DC1-DC2
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.1 =2000,10,009E-00CF,009E-00CF,10,6000,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.2 =2001,10,00D4-015B,00D4-015B,10,6001,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.3 =2002,10,0174-017B,0174-017B,10,6002,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.4 =2003,10,0194-019B,0194-019B,10,6003,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.5 =2004,10,01B4-01BB,01B4-01BB,10,6004,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.6 =2005,10,0200-0220,0200-0220,10,6005,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.7 =2006,10,0223-0223,0224-0224,10,6006,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.8 =2007,10,0228-0230,0228-0230,10,6007,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.9 =2008,10,0233-0233,0234-0234,10,6008,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.10=2009,10,0238-024C,0238-024C,10,6009,0
GDDP415I PARMS.RDF.DEVICES.DC1.DC2.11=200A,10,024D-024D,024F-024F,10,600A,0

If invoked with BCV(Y) and GLOBALS(CHECK) and all relevant call override bytes are 0:
GDDP415W CONFIG: Not verifying BCV parameters for any sites

If ACDD runs with GLOBALS(CHECK) in a configuration with SNAP-VX support configured:
GDDP415I Supporting SNAP-VX Softlink at DC1: YES
GDDP415I SNAP-VX snapshotname for GOLD set, Internal: G_I_SNAP

If GDDRACDD runs with GLOBALS(CHECK) it fabricates INCLUDE, GENERATE, TFINDER, and
where applicable SOFTLINK and SNAPNAME parameters as necessary. Those are echoed to
the joblog:

GDDP415I (SIM) ACDDPARM: INCLUDE DC1 GDDR I,2000,10
GDDP415I (SIM) ACDDPARM: INCLUDE DC1 GDDR I,2000,20
GDDP415I (SIM) ACDDPARM: INCLUDE DC2 GDDR I,6000,10
GDDP415I (SIM) ACDDPARM: INCLUDE DC2 GDDR I,6000,30
GDDP415I (SIM) ACDDPARM: INCLUDE DC1 GDDR E,2000,00
GDDP415I (SIM) ACDDPARM: INCLUDE DC1 GDDR E,2000,F9
GDDP415I (SIM) ACDDPARM: INCLUDE DC2 GDDR E,6000,00
GDDP415I (SIM) ACDDPARM: INCLUDE DC2 GDDR E,6000,31
GDDP415I (SIM) ACDDPARM: INCLUDE DC1 GDDR E,6000,31

Several types of input errors are flagged by GDDP400E messages (RDF.DEVICES can be
replaced by DLM.DEVICES where relevant):

GDDP400E Validation error: <text>
<text> can be one of the following:
- GDD Domain not active. Please rerun with COMP(Y)
- GDDRPFX must be length 4
- Parameter not defined: <parameter>
- GDDRACDD not supported at <siteid> (not a configured site)
- No RDF.DEVICES parameters for site pair <site>-><site>
- Different number of RDF.DEVICES parameters for site pair <site1>-><site2> versus
  <site2>-><site1>
- Parameter <n> for <site1>-><site2> is not a mirror image of <parameter value>
- Parameter <n> for <site1>-><site2> is of a different type than <parameter value>
- ACDDPARM not allocated
- Failed to open ACDDPARM (<mode>) with RC=<rc>
- No parameters in ACDDPARM
- Running in compatibility mode but there are no [CSYSSITE] [SYMM] parameters in ACDDPARM
- Running while GDD Domain is down but there are no [CSYSSITE] [SYMM] parameters in ACDDPARM
- Running in compatibility mode but global variable validation is requested
- Running while GDD Domain is down but global variable validation is requested
- Parameter not defined: <parameter name>
- C-system site <site> from GDDMPARM is not known in CONFIG.SITELIST: <list>
- Duplicate definition of C-system <systemname>
- Duplicate definition of C-system site <siteid>. Sitelist: <sitelist>
- 2, 3 or 4 C-systems required. Sitelist: <sitelist>
- DC1 is a required site. Sitelist: <sitelist>
- One of sites DC2, DC3 is required. Sitelist: <sitelist>
- 3-Site config requires both DC2 and DC3. Sitelist: <sitelist>
- Site DC4 is required in a 4-Site config. Sitelist: <sitelist>
- Current system: <sysname> not defined in csyslist: <list>
- Duplicate usage of gatekeeper <gk> at site <siteid> in [GDDMPARM] [ACDDPARM]
- Invalid serialnumber [4] [<1] in <line>
- Duplicate definition for Symmetrix Unit <serial> at Site <siteid>
- Site DC1 must have storage defined. Sitelist: <sitelist>
- CSYSSITE <siteid> has no SYMM in [ACDDPARM] [GDDMPARM]
- Too many sites in SYMM-list in [ACDDPARM] [GDDMPARM]
- One of sites DC2, DC3 must have storage defined. Sitelist: <list> from [ACDDPARM] [GDDMPARM]
- Both sites DC2 and DC3 must have storage defined. Sitelist: <list> from [ACDDPARM] [GDDMPARM]
- Gatekeeper <gk> in <site1>-<site2> parameter value is not known in any <site1> SYMM
  parameter in GDDMPARM
- No GK Pool defined for Symmetrix <serial> (<gk>) at Site <siteid>
- Unknown Site <site> specified as CPDS
- DC3 not supported as Primary DASD Site
- Unknown Site referenced on <line>
- Unknown Symmetrix <serial> referenced on <line>
  Group <rag> in discovery path <path> for Symmetrix <serial> does not exist in Symmetrix <serial>
  Discovery path <path> locates Symmetrix <serial>. Expected to find <serial>
  Unable to determine a Discovery path for Symmetrix <serial>. Please specify a path in ACDDPARM
- Invalid ra-group <rag> on <line>. Must be 2 chars long
- Invalid character <char> in ra-group <rag> on <line> Expected one of 0123456789ABCDEF
- Expected two serial numbers on <line>. Found: <text>
  One of <serial1> <serial2> on <line> must be located at DC3. Found <site1> and <site2>
  Only one of <serial1> <serial2> on <line> must be located at DC3
  Virtual group must connect DC1-DC4 or DC2-DC3. Found <site1>-<site2> on <line
  Unknown Symmetrix <serial> on <line>
  Virtual group <rag> cannot be an existing group in <serial> on <line>
- No INCLUDE parameters found in ACDDPARM
- Invalid type (not E or I) on <line>
- GK <gk> on <line> does not locate a Symmetrix at <site>
- Invalid ra-group <rag> on <line>. Must be length 2 [Not found in Symmetrix <serial> at <sites>] [Connects to same-site Symmetrix <serial>] [Connects DC1 to DC4 Symmetrixes, <serial> and <serial>] [Connects to ra-group <rag> in <serial>. Asymmetrical ra-groups are not currently supported with GDDR.]
- Invalid external ra-group <rag> on <line>. [Connects DC1 to DC2 Symmetrixes, <serial> and <serial>] [Connects DC3 to DC4 Symmetrixes, <serial> and <serial>]
- Invalid character <char> in ra-group <rag> on <line>
- INCLUDE of virtual group <rag> on <line> not allowed. (GENERATE parameters only)
  Inclusion of <rag> as <type> on <line> conflicts with earlier inclusion as <type>
  Virtual site on <line> Expected one of <list>
  RA-groups on <line> must be specified as pairs
- Only one of <rag>, <rag> is allowed to be virtual on <line>
- One of <rag> and <rag> on <line> requires an INCLUDE
- GENERATE type <type> on <line> conflicts with INCLUDE type <type> for group <rag>
- Invalid ra-group <rag> on <line>. Should connect <site> to DC1 or DC2, not <site>
- Invalid ra-groups <rag>, <rag> on <line>. Both connect <site> to <site>
- No includes refer to Symmetrix <serial> at <site>
- No includes connect Symmetrixes for site-pair <site1>-<site2> on <line>
- Different number of RDP STAR ra-groups DC1-DC1 (nn) versus DC3-DC2 (nn)
- Different number of RDP SQAR ra-groups DC1-DC3 (nn) versus DC2-DC4 (nn)
- Larger number of RDP SQAR ra-groups <site1>-<site2> (nn) than <site3>-<site1> (nn) in Symmetrix <serial> at <site1>
- Duplicate specification of site <site> on <line>
- TFIINDER parameter for site <site> on <line> not allowed. ALL-site parameter exists
- TFIINDER parameter for ALL-site on <line> not allowed. Specific sites: <sitelist>
- RDF group specification required on <line>
- BVC-set selection required on <line>
- Invalid RDF group specification on <line>. Site ALL requires one of INT, EXT, ALL
- Invalid GK specification on <line>. Site ALL does not support a specific GK
- Invalid message level for validation of the GOLD|TEST set on <line>. Expected one of: WARN ERROR
- Invalid GK specification on <line>. Explicit RDF group specification requires a single gatekeeper
- Invalid ra-group <ra> on <line>. Not GDDR managed
- SOFTP LINK support requires SNAPNAME parameters
- Invalid SOFTP LINK parameter on <line>: <reason>
- Invalid SNAPNAME parameter value on <line>: <reason>
- Duplicate specification of set <set> on <line>
- Invalid SNAPSHOT name <name> for set <description>. Must begin with <prefix>
- Legacy TF Method must be specified as MIRR or CLON
- Legacy TF Method CLON for <site> requires STDCV parameters. >>> Legacy TF Method for <site> overridden to MIRR

**GDDRACDD site awareness**

Based on the detected system name and the CSYSSITE parameters in ACDDPARM, GDDRACDD is able to determine which GDDR site it runs on, and issues the following message:

```
GDDP415I GDDRACDD running at DC2
```

GDDP400E will be issued in one of two formats (depending on when the error is caught), if the current system does not match one of the system names listed in CSYSSITE parameters:

```
GDDP400E Validation error: Current system: VC1A not defined in csyslist: O016 VC1B O01E
GDDP400E Validation error: GDDRACDD not supported at <siteid> (not a configured site)
```

**Wizard integration**

If the JCL used to run GDDRACDD does not have an E04SRDFD or E05TFDEV DD statement, one or both of the following messages will be issued:

```
GDDP415I E04SRDFD not allocated, RDF.DEVICES parameters will not be available to GDDR Parameter Wizard
GDDP415I E05TFDEV not allocated, BCV.DCn.ONLY parameters will not be available to GDDR Parameter Wizard
```

If the JCL used to run GDDRACDD has some or all of the optional DD-cards, but GDDRACDD fails to open one of them, the following message will be issued:

```
GDDP415W Failed to open <ddname> (<mode>) with RC=<rc>
```

**Messages related to VMAX discovery**

VMAX device discovery is done one site at a time, one VMAX system at a time.

The GDDRACDD JESMSGLG shows progress messages as each VMAX system is being discovered:

```
GDDP402I Beginning Discovery of Symmetrix units
GDDP410I Site: DC1 - Device ranges in Symm: 000292601023 (5875) GK: 2000
GDDP402I Site: DC1 Symm: 000292601023 SP132 usage: 1786384
GDDP410I Site: DC2 - Device ranges in Symm: 000292601156 (5875) GK: 2000.10
GDDP402I Site: DC2 Symm: 000292601156 SP132 usage: 1781264
GDDP402I Site: DC3 Symm: 000292601024 SP132 usage: 1778504
```

Message GDDP410I identifies the site-location of the discovered VMAX system, its serial number and Enginuity level, as well as its discovery path from the site where GDDRACDD is running. It signals the start of discovery for the listed VMAX system.

Message GDDP402I shows the amount of virtual storage used for this VMAX system. It signals the completion of discovery for the listed VMAX system.
Generated RDF.DEVICES and DLM.DEVICES parameters

The final section of the JESMSGLG is a complete listing of the RDF.DEVICES and DLM.DEVICES parameters generated for the discovered VMAX systems. The parameters are listed by site-pair, once in each direction.

GDDP419I RDF.DEVICES Parameters for Discovered Symmetrixes

GDDP419I Site pair DC1-DC2

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.1 =2000,10,009E-00CF,009E-00CF,10,6000,0 ( 50 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.2 =2001,10,00D4-015B,00D4-015B,10,6001,0 ( 136 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.3 =2002,10,0174-017B,0174-017B,10,6002,0 ( 8 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.4 =2003,10,0194-019B,0194-019B,10,6003,0 ( 8 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.5 =2004,10,01B4-01BB,01B4-01BB,10,6004,0 ( 8 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.6 =2005,10,0200-0220,0200-0220,10,6005,0 ( 33 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.7 =2006,10,0223-0223,0224-0224,10,6006,0 ( 1 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.8 =2007,10,0228-0230,0228-0230,10,6007,0 ( 9 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.9 =2008,10,0233-0233,0234-0234,10,6008,0 ( 1 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.10=2009,10,0238-024C,0238-024C,10,6009,0 ( 21 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.11=200A,10,024D-024E,024F-024F,10,600A,0 ( 1 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.12=200B,10,024E-024E,024E-024E,10,600B,0 ( 1 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.13=200C,10,024F-024F,024D-024D,10,600C,0 ( 1 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.14=200D,10,0270-0270,0270-0270,10,600D,0 ( 1 FS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.15=200E,10,0300-030F,0300-030F,10,600E,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.16=200F,10,0340-034F,0340-034F,10,600F,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.17=2000,10,0380-038F,0380-038F,10,6000,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.18=2001,10,03C0-03CF,03C0-03CF,10,6001,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.19=2002,10,0400-040F,0400-040F,10,6002,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.20=2003,10,0440-044F,0440-044F,10,6003,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.21=2004,10,0480-048F,0480-048F,10,6004,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.22=2005,10,04C0-04CF,04C0-04CF,10,6005,0 ( 16 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.23=2006,10,0500-057F,0500-057F,10,6006,0 ( 128 CS )

GDDP419I PARMS.RDF.DEVICES.DC1.DC2.27=200A,00,01E0-01FF,01E0-01FF,00,600A,1 ( 32 FS )

...

Messages in SYSTSPRT

All messages visible in JESMSGLG are also echoed to SYSTSPRT. SYSTSPRT has some additional messages, not visible in JESMSGLG. The following sections discuss those elements in SYSTSPRT not visible in JESMSGLG.

Initial device range discovery

Messages in this section could be used to piece together RDF.DEVICES or DLM.DEVICES parameters if the actual device parameter generation by GDDRACDD fails. The same information is reported in several stages of discovery.

SRDF device pairs by site pair and by group as seen from either side

GDDR DC1-DC2 RDF device pairs in group 000292601023.10 as seen from DC1

009E-009E ==> 009E-009E
009F-00CF ==> 009F-00CF
00D4-0113 ==> 00D4-0113
0114-0153 ==> 0114-0153
0154-015B ==> 0154-015B
0174-017B ==> 0174-017B

GDDR DC1-DC2 RDF device pairs in group 000292601156.10 as seen from DC2

009E-00CF <= 009E-00CF
00D4-0113 <= 00D4-0113
0114-0153 <= 0114-0153
0154-015B <= 0154-015B
0174-017B <= 0174-017B
Using GDDR Utilities

Discovered ranges

Discovered RDF device ranges

RDF device ranges for site-pair DC1-DC2
> 000292601023 Group: 10 ---> 000292601156
  1 009E-009E  <--->  009E-009E     1 (CS) (CS) Groups: (10   ) (10   )
  2 009F-00CF  <--->  009F-00CF    49 (CS) (CS) Groups: (10 20) (10   )
  3 00D4-0113  <--->  00D4-0113    64 (CS) (CS) Groups: (10 20) (10   )
  4 0114-0153  <--->  0114-0153    64 (CS) (CS) Groups: (10 20) (10   )
  5 0154-015B  <--->  0154-015B    8 (CS) (CS) Groups: (10 20) (10   )
  6 0174-017B  <--->  0174-017B    8 (CS) (CS) Groups: (10 20) (10 30)
  7 0194-019B  <--->  0194-019B    8 (CS) (CS) Groups: (10 20) (10 30)
  8 01B4-01BB  <--->  01B4-01BB    8 (CS) (CS) Groups: (10 20) (10 30)

RDF device ranges for site-pair DC1-DC3
> 000292601023 Group: 20 ---> 000292601024
  1 009F-00CF  <--->  009F-00CF    49 (CS) (CS) Groups: (10 20) (20   )
  2 00D4-0113  <--->  00D4-0113    64 (CS) (CS) Groups: (10 20) (20   )
  3 0114-0153  <--->  0114-0153    64 (CS) (CS) Groups: (10 20) (20 30)
  4 0154-015B  <--->  0154-015B    8 (CS) (CS) Groups: (10 20) (20 30)
  5 0174-017B  <--->  0174-017B    8 (CS) (CS) Groups: (10 20) (20 30)
  6 0194-019B  <--->  0194-019B    8 (CS) (CS) Groups: (10 20) (20 30)
  7 01B4-01BB  <--->  01B4-01BB    8 (CS) (CS) Groups: (10 20) (20 30)
  8 0200-0213  <--->  0200-0213    20 (FS) (FS) Groups: (10 20) (20   )
  9 0214-021F  <--->  0214-021F    12 (FS) (FS) Groups: (10 20) (20   )
 10 0220-0220  <--->  0220-0220     1 (FS) (FS) Groups: (10 20) (20   )

RDF device ranges for site-pair DC2-DC3
> 000292601156 Group: 30 ---> 000292601024
> 000292601156 Group: 31 ---> 000292601024

Discovered and generated ranges

Discovered and generated RDF device ranges

RDF device ranges for site-pair DC1-DC2
> 000292601023 Group: 10 ---> 000292601156
  1 009E-009E  <--->  009E-009E     1 (CS) (CS) Groups: (10   ) (10   )
  2 009F-00CF  <--->  009F-00CF    49 (CS) (CS) Groups: (10 20) (10   )
  3 00D4-0113  <--->  00D4-0113    64 (CS) (CS) Groups: (10 20) (10   )
  4 0114-0153  <--->  0114-0153    64 (CS) (CS) Groups: (10 20) (10 30)
  5 0154-015B  <--->  0154-015B    8 (CS) (CS) Groups: (10 20) (10 30)
  6 0174-017B  <--->  0174-017B    8 (CS) (CS) Groups: (10 20) (10 30)
  7 0194-019B  <--->  0194-019B    8 (CS) (CS) Groups: (10 20) (10 30)
  8 01B4-01BB  <--->  01B4-01BB    8 (CS) (CS) Groups: (10 20) (10 30)
  9 0200-0213  <--->  0200-0213    20 (FS) (FS) Groups: (10 20) (10   )
 10 0214-021F  <--->  0214-021F    12 (FS) (FS) Groups: (10 20) (10   )

RDF device ranges for site-pair DC1-DC3
> 000292601023 Group: 20 ---> 000292601024
  1 009F-00CF  <--->  009F-00CF    49 (CS) (CS) Groups: (10 20) (20   )
  2 00D4-0113  <--->  00D4-0113    64 (CS) (CS) Groups: (10 20) (20   )
  3 0114-0153  <--->  0114-0153    64 (CS) (CS) Groups: (10 20) (20 30)
  4 0154-015B  <--->  0154-015B    8 (CS) (CS) Groups: (10 20) (20 30)
  5 0174-017B  <--->  0174-017B    8 (CS) (CS) Groups: (10 20) (20 30)
  6 0194-019B  <--->  0194-019B    8 (CS) (CS) Groups: (10 20) (20 30)
  7 01B4-01BB  <--->  01B4-01BB    8 (CS) (CS) Groups: (10 20) (20 30)
  8 0200-0213  <--->  0200-0213    20 (FS) (FS) Groups: (10 20) (20   )
  9 0214-021F  <--->  0214-021F    12 (FS) (FS) Groups: (10 20) (20   )
 10 0220-0220  <--->  0220-0220     1 (FS) (FS) Groups: (10 20) (20   )

RDF device ranges for site-pair DC2-DC3
> 000292601156 Group: 30 ---> 000292601024
  1 009F-00CF  <--->  009F-00CF    49 (CS) (CS) Groups: (10 30) (20   )
  2 00D4-0113  <--->  00D4-0113    64 (CS) (CS) Groups: (10 30) (20   )
  3 0114-0153  <--->  0114-0153    64 (CS) (CS) Groups: (10 30) (20 30)
  4 0154-015B  <--->  0154-015B    8 (CS) (CS) Groups: (10 30) (20 30)
  5 0174-017B  <--->  0174-017B    8 (CS) (CS) Groups: (10 30) (20 30)
  6 0194-019B  <--->  0194-019B    8 (CS) (CS) Groups: (10 30) (20 30)
  7 01B4-01BB  <--->  01B4-01BB    8 (CS) (CS) Groups: (10 30) (20 30)
### RDF.DEVICES parameters

Messages in this section can be used to piece together RDF.DEVICES parameters if the actual device parameter generation by GDDRACDD fails. The same information is reported in several stages of parameter generation. Each of these messages can be repeated for DLM.DEVICES parameters, if applicable.

#### Initial RDF.DEVICES parameters

Initial RDF.DEVICES Parameters

<table>
<thead>
<tr>
<th>Site pair</th>
<th>DC1-DC2</th>
<th>PARMS.RDF.DEVICES.DC1.DC2.1 =2000,10,009E-00CF,009E-00CF,10,6000,0 ( 50 CS )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.2 =2000,10,00D4-015B,00D4-015B,10,6000,0 ( 136 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.3 =2000,10,0174-017B,0174-017B,10,6000,0 ( 8 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.4 =2000,10,0194-019B,0194-019B,10,6000,0 ( 8 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.5 =2000,10,01B4-01BB,01B4-01BB,10,6000,0 ( 8 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.6 =2000,00,01E0-01FF,01E0-01FF,00,6000,1 ( 32 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.7 =2000,10,0200-0220,0200-0220,10,6000,0 ( 33 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.8 =2000,10,0223-0223,0224-0224,10,6000,0 ( 1 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.9 =2000,10,0228-0230,0228-0230,10,6000,0 ( 9 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.10=2000,10,0233-0233,0234-0234,10,6000,0 ( 1 FS )</td>
</tr>
</tbody>
</table>

#### Sorted RDF.DEVICES parameters

Sorted RDF.DEVICES parameters

| PARMS.RDF.DEVICES.DC1.DC2.1 =2000,10,009E-00CF,009E-00CF,10,6000,0 ( 50 CS ) |
| PARMS.RDF.DEVICES.DC1.DC2.2 =2000,10,00D4-015B,00D4-015B,10,6000,0 ( 136 CS ) |
| PARMS.RDF.DEVICES.DC1.DC2.3 =2000,10,0174-017B,0174-017B,10,6000,0 ( 8 CS ) |
| PARMS.RDF.DEVICES.DC1.DC2.4 =2000,10,0194-019B,0194-019B,10,6000,0 ( 8 CS ) |
| PARMS.RDF.DEVICES.DC1.DC2.5 =2000,10,01B4-01BB,01B4-01BB,10,6000,0 ( 8 CS ) |
| PARMS.RDF.DEVICES.DC1.DC2.6 =2000,00,01E0-01FF,01E0-01FF,00,6000,1 ( 32 FS ) |
| PARMS.RDF.DEVICES.DC1.DC2.7 =2000,10,0200-0220,0200-0220,10,6000,0 ( 33 FS ) |
| PARMS.RDF.DEVICES.DC1.DC2.8 =2000,10,0223-0223,0224-0224,10,6000,0 ( 1 FS ) |
| PARMS.RDF.DEVICES.DC1.DC2.9 =2000,10,0228-0230,0228-0230,10,6000,0 ( 9 FS ) |
| PARMS.RDF.DEVICES.DC1.DC2.10=2000,10,0233-0233,0234-0234,10,6000,0 ( 1 FS ) |

#### RDF.DEVICES parameters with gatekeepers from pools

RDF.DEVICES Parameters with GateKeepers from pools

<table>
<thead>
<tr>
<th>Site pair</th>
<th>DC1-DC2</th>
<th>PARMS.RDF.DEVICES.DC1.DC2.1 =2000,10,009E-00CF,009E-00CF,10,6000,0 ( 50 CS )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.2 =2001,10,00D4-015B,00D4-015B,10,6001,0 ( 136 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.3 =2002,10,0174-017B,0174-017B,10,6002,0 ( 8 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.4 =2003,10,0194-019B,0194-019B,10,6003,0 ( 8 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.5 =2004,10,01B4-01BB,01B4-01BB,10,6004,0 ( 8 CS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.6 =2005,10,0200-0220,0200-0220,10,6005,0 ( 33 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.7 =2006,10,0223-0223,0224-0224,10,6006,0 ( 1 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.8 =2007,10,0228-0230,0228-0230,10,6007,0 ( 9 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.9 =2008,10,0233-0233,0234-0234,10,6008,0 ( 1 FS )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARMS.RDF.DEVICES.DC1.DC2.10=2009,10,0238-024C,0238-024C,10,6009,0 ( 21 FS )</td>
</tr>
</tbody>
</table>

### Messages in SYMMDEVS

If the SYMMDEVS DD card is present in the GDDRACDD JCL, it contains the Device ranges report as created during discovery of the devices. It consists of a sequence of the following messages:

```
GDDP402I Beginning Discovery of Symmetrix units
GDDP410I Site: DC1 - Device ranges in Symm: 000192600975 (5876) GK: 0E40
GDDP402I Site: DC1 Symm: 000192600975 SP132 usage: 1769908
GDDP410I Site: DC1 - Device ranges in Symm: 000192600975 (5876) GK: 0E40
GDDP411I LDEV-HDEV-COUNT-HTM-MBR- CYLS  -MR- RDF-RDFGROUPS
GDDP412I 0000-00C7-  200-SVA-----    5120---------/-
GDDP412I 00C8-00CB-    4-SFI-----    8738--1------/-
```

---

Automated Configuration Discovery for DASD (GDDRACDD) 267
Message GDDP410I identifies the site-location of the discovered VMAX system, its serial number and Enginuity level, as well as its discovery path from the site where GDDRACDD is running.

Message GDDP402I shows the amount of virtual storage used for this VMAX system.

Message GDDP411I is a header preceding the GDDP412I messages.

Message GDDP412I has the following content:

- **LDEV-HDEV**
  This is the low and high device for the discovered device range in the VMAX system identified by message GDDP410I. These are the "local" devices, as seen from the perspective of that VMAX system.

- **COUNT**
  The number of VMAX devices in the reported range. For FBA-meta devices, this includes both the meta-heads and the meta-member devices in this range.

- **HTM**
  The local device type, reported as three flags:
  - **H Host-type**: C for CKD devices, F for FBA devices, A for AS/400 devices, S for special devices. The following special devices are recognized:
    - SVC: VCMDB devices
    - SIB: Internal BCV devices
    - SWO: WORM devices
    - SFI: Symmetrix File-system devices
    - SVA: Symmetrix Vaulting devices
    - SDU: Dummy devices
    - STH: Thin Backend devices
  - **T TimeFinder-type**: B for BCV-devices, S for standard devices
  - **M Meta-type**: H for meta-heads, M for meta-members, - for non-meta devices

  **Note:** For FBA-meta-devices, an 'FSH' or 'FBH' range typically includes meta-heads and meta-members.

- **MBR**
  For meta-devices, this is the number of VMAX devices making up each meta-device. This number includes the meta-head. Reported as '--' for non-meta devices.

- **CYLS**
  The number of cylinders per device. For meta-head devices, this is the total number of cylinders summed over all devices in the meta. For meta-members, this is the number of cylinders per member.

- **MR**
  The local protection type for the reported range. It is reported as:
- '---': device has no local protection
- '-0': RAID-0 device (striping, not a protection level really)
- '-5': RAID-5 device (can be 3+1 or 7+1)
- '-6': RAID-6 device (can be 6+2 or 14+2)
- '-1': RAID-1 device (local mirroring)

**Note:** For RAID-10, also known as CKD-Meta devices (striped by definition), if they are RAID-1, the current code shows this as '-1', not '10'.

- **RDF**
  The local (that is, in the VMAX system being discovered) SRDF personality of the devices in the reported range. It is reported as:
  - '----': not an SRDF device
  - 'R1--': R1 device
  - 'R2--': R2 device
  - 'R11-': Concurrent SRDF device, R1 in both reported RA-groups.
  - 'R12-': R21 device which is R1 in first RA-group, R2 in second RA-group
  - 'R21-': R21 device which is R2 in first RA-group, R1 in second RA-group
  - 'R22-': R22 device

- **RDFGROUPS**
  Because of the RDFGROUPS portion, the GDDP412I message can be spread over 2 lines, as shown in the following example. There will be 1 line per RDF group to which the reported device range belongs. The RDF groups are always listed in ascending order. (groups 25 and 28 below):

```
  LDEV-HDEV-COUNT-HTM-MBR-  CYLS  -MR- RDF-RDFGROUPS
  0078-007F-    8-CSH-  4-   10017--0-R11--25/-/-/-/---/--W--/--W--/192601314/0070-0077
                     28/-/-/-/T--D--/---/--W--/--------/192601313/007C-0083
```
The RDFGROUPS part of the GDDP412I message consists of 7 fields, separated by '/'.

NN/CETC/SAMDB/UNRI/tttttttttt/ssssssssss/LDEV-HDEV

NN 2-digit RDF group number

CEUTC This field reflects the CG-status, reported as 4 (future 5) flags
  C CG-flag '·' if no CG flag
  E ECA-flag '·' if no ECA flag
  U UNR-on-Timeout flag '·' if no UNR-on-Timeout flag
  T CG Trip occurred '·' if no CG trip occurred
c represents the ECA-window status
  o ECA-window is open
  c ECA-window is closed normally
  t ECA-window is closed due to a timeout

SAMDB This field reflects the SRDF status/mode, reported as 5 flags
  S SRDF replication status
  T if the devices are TNR '·' if not
  A if active in SRDF/A '·' if not
  M MSC-mode, reported as:
    '·' if not in MSC mode
    "0" "8" "C" "E" "F" = MSC-Cycle Tag
  D Adaptive Copy Disk Mode '·' if not
  B Link-Blocked '·' if not

UNRI This field reflects the host-availability status, reported as 4 flags
  U UNR '·' if not
  N NR '·' if not (applies to R2 only)
  R RNR '·' if not
  I ITA-attribute '·' if not

tttttttt This field has different meanings for SRDF/S versus SRDF/A devices
  SRDF/S CG Hash code
  SRDF/A MSC Group name

ssssssss VMAX serial number on the other side of this RA-group

LDEV-HDEV Low and high VMAX device number at the other side

For GDDR-managed devices, the RDFGROUPS field can have the following appendages:

**RDF group type indicator**
- (I) This is an internal GDDR managed group
- (E) This is an external GDDR managed group
- (ID) This is an internal GDDR managed group, for DLM backend devices
- (ED) This is an external GDDR managed group, for DLM backend devices (not currently used)

**BCV pairing and state indicator**
Format: (xG) (xT) (nn)
- (xG) : provides information about the GOLD set BCVs, blank if none requested
- (xT) : provides information about the TEST set BCVs, blank if none requested
Automated Configuration Discovery for DASD (GDDRACDD)

Using GDDR Utilities

- (nn) : number of not-requested BCVs found to be associated with the listed RDF device range. GDDR ignores these. Blank if no not-requested BCVs are found.

**BCV information for the GOLD and TEST sets**

If BCV devices are requested in these sets, they can have the following values:

- !G or !T: A BCV was requested in the indicated set, but none is found
- -G or -T: A BCV was requested, found associated, but is currently SPLIT
- >G or >T: A BCV was requested, found associated, and attached due to a (RE-)ESTABLISH
- <G or <T: A BCV was requested, found associated, and attached due to a RESTORE
- ?G or ?T: A BCV was requested, found associated, but ACDD cannot determine the pair-state

**Messages in RDFGROUP**

If the RDFGROUP DD card is present in the GDDRACDD JCL, it contains a summary report at the RDF group level. The report is presented in two parts:

- Part 1: List of RDF groups linking GDDR-managed VMAX systems
- Part 2: Unique GDDR RDF groups by site-pair and type

Part 1 lists all RDF groups between VMAX systems for which a SYMM parameter is found, organized by site-pair. For each group an indication is given of its inclusion type (Internal or External or None). If applicable, the report indicates that a group is for DLm backend devices ("- DLm"). Offline groups are marked as "<< OFFLINE <<". Virtual groups are marked "- VIRTUAL". Groups connecting same-site VMAX systems are marked "- ignored".

**Part 1 example:**

```
GDDP417I
GDDP417I List of RDFgroups linking GDDR Managed Symmetrixes
GDDP417I
GDDP417I Symmetrix units at DC1
GDDP417I
GDDP417I Site pair DC1-DC2
GDDP417I 2000 000292601023 (5875) <--00 --> 000292601156 (5875) (6000) (External)
GDDP417I 2000 000292601023 (5875) <--02 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--03 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--04 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--05 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--06 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--07 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--08 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--09 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--0A --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--0B --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--0C --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--0D --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--0E --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--0F --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--10 --> 000292601156 (5875) (6000) (Internal)
GDDP417I ...
GDDP417I 2000 000292601023 (5875) <--F5 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--F6 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--F7 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--F8 --> 000292601156 (5875) (6000) (Non-GDDR)
GDDP417I
GDDP417I Site pair DC1-DC3
GDDP417I 2000 000292601023 (5875) <--16 --> 000292601024 (5875) (A000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--17 --> 000292601024 (5875) (A000) (Non-GDDR)
GDDP417I 2000 000292601023 (5875) <--20 --> 000292601024 (5875) (A000) (Internal)
```
Part 2 lists all INCLUDE-d RDF groups, first the internal groups and then the external groups. For each type, the list is organized by site-pair.

Part 2 example:

Unique GDDR ra-groups by site-pair and type
RA-groups of type Internal

DC1 GK RA GK DC2 ( DC1 Symm - DC2 Symm )
2000 10 6000 (000292601023-000292601156)

DC1 GK RA GK DC3 ( DC1 Symm - DC3 Symm )
2000 20 A000 (000292601023-000292601024)

DC2 GK RA GK DC3 ( DC2 Symm - DC3 Symm )
6000 30 A000 (000292601156-000292601024)

RA-groups of type External
DC1 GK RA GK DC2 ( DC1 Symm - DC2 Symm )
Messages in CREPAIR

If the CREPAIR DD card is present in the GDDRACDD JCL, it contains a set of createpair commands which could be used to reconstruct the GDDR-managed environment from scratch. The content of the report is influenced by the CONFIG(...) argument. For example:

GDDP418I Createpair commands by site-pair and type
GDDP418I >>> Intended configuration: 3-site, CASCADED, Primary DASD site: DC1
GDDP418I RA-groups of type Internal
GDDP418I Sitepair DC1-DC2
GDDP418I Use these commands from a host with channel access to the Symmetrix unit at DC1
GDDP418I Caution: use LCLISR2 if you want the devices at DC1 to be R2 devices
GDDP418I SC VOL,LCL(2000,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),009B-00CF,009E
GDDP418I SC VOL,LCL(2001,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),00D4-015B,00D4
GDDP418I SC VOL,LCL(2002,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0174-017B,0174
GDDP418I SC VOL,LCL(2003,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0240-024F,0240
GDDP418I SC VOL,LCL(2004,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),02CB-02D0,02CB
GDDP418I SC VOL,LCL(2005,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),1200-13FF,1200
GDDP418I SC VOL,LCL(2006,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),1400-15FF,1400
GDDP418I SC VOL,LCL(2007,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),4240-42FF,4210
GDDP418I
GDDP418I Sitepair DC2-DC1
GDDP418I Use these commands from a host with channel access to the Symmetrix unit at DC2
GDDP418I Caution: use LCLISR1 if you want the devices at DC2 to be R1 devices
GDDP418I SC VOL,LCL(6000,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),009B-00CF,009E
GDDP418I SC VOL,LCL(6001,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),00D4-015B,00D4
GDDP418I SC VOL,LCL(6002,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),0174-017B,0174
GDDP418I SC VOL,LCL(6003,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),0240-024F,0240
GDDP418I SC VOL,LCL(6004,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),02CB-02D0,02CB
GDDP418I SC VOL,LCL(6005,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),1200-13FF,1200
GDDP418I SC VOL,LCL(6006,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),1400-15FF,1400
GDDP418I SC VOL,LCL(6007,10),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),4240-42FF,41F0
GDDP418I
GDDP418I Sitepair DC1-DC3
GDDP418I Use these commands from a host with channel access to the Symmetrix unit at DC1
to create an R22 configuration
GDDP418I Caution: use LCLISR2 if you want the devices at DC1 to be R2 devices
GDDP418I SC VOL,LCL(2000,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),009F-00CF,009F
GDDP418I SC VOL,LCL(2001,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),00D4-015B,00D4
GDDP418I SC VOL,LCL(2002,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0174-017B,0174
GDDP418I SC VOL,LCL(2003,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0240-024F,0240
GDDP418I SC VOL,LCL(2004,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),02CB-02D0,02CB
GDDP418I SC VOL,LCL(2005,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),1200-13FF,1200
GDDP418I SC VOL,LCL(2006,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),1400-15FF,1400
GDDP418I SC VOL,LCL(2007,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),4240-42FF,41F0
GDDP418I
GDDP418I Sitepair DC3-DC1
GDDP418I Use these commands from a host with channel access to the Symmetrix unit at DC3
to create an R22 configuration
GDDP418I Caution: use LCLISR1 if you want the devices at DC3 to be R1 devices
GDDP418I SC VOL,LCL(A000,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),009F-00CF,009F
GDDP418I SC VOL,LCL(A001,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),00D4-015B,00D4
GDDP418I SC VOL,LCL(A002,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),0174-017B,0174
GDDP418I SC VOL,LCL(A003,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),0240-024F,0240
GDDP418I SC VOL,LCL(A004,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),02CB-02D0,02CB
GDDP418I SC VOL,LCL(A005,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),1200-13FF,1200
GDDP418I SC VOL,LCL(A006,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),1400-15FF,1400

Automated Configuration Discovery for DASD (GDDRACDD)  273
Using GDDR Utilities

GDDP418I  SC VOL,LCL(A007,20),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),41F0-42AF,4240
GDDP418I  Sitepair DC2-DC3
GDDP418I  Use these commands from a host with channel access to the Symmetrix unit at DC2
GDDP418I  Caution: use LCLISR2 if you want the devices at DC2 to be R2 devices
GDDP418I  SC VOL,LCL(6000,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),009F-00CF,009F
GDDP418I  SC VOL,LCL(6001,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),00D4-015B,00D4
GDDP418I  SC VOL,LCL(6002,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0174-017B,0174
GDDP418I ...
GDDP418I  SC VOL,LCL(6004,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),1200-13FF,1200
GDDP418I  SC VOL,LCL(6005,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),1400-15FF,1400
GDDP418I  SC VOL,LCL(6006,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),4210-42CF,41F0
GDDP418I  Sitepair DC3-DC2
GDDP418I  Use these commands from a host with channel access to the Symmetrix unit at DC3
GDDP418I  Caution: use LCLISR2 if you want the devices at DC3 to be R2 devices
GDDP418I  SC VOL,LCL(A000,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),009F-00CF,009F
GDDP418I  SC VOL,LCL(A001,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),00D4-015B,00D4
GDDP418I  SC VOL,LCL(A002,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),0174-017B,0174
GDDP418I ...
GDDP418I  SC VOL,LCL(A004,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),1200-13FF,1200
GDDP418I  SC VOL,LCL(A005,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),1400-15FF,1400
GDDP418I  SC VOL,LCL(A006,30),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),41F0-42AF,4210
GDDP418I  RA-groups of type External
GDDP418I  Sitepair DC1-DC2
GDDP418I  Use these commands from a host with channel access to the Symmetrix unit at DC1
GDDP418I  Caution: use LCLISR2 if you want the devices at DC1 to be R2 devices
GDDP418I  SC VOL,LCL(200A,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),01E0-01FF,01E0
GDDP418I  SC VOL,LCL(200B,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0254-0254,0256
GDDP418I  SC VOL,LCL(200C,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0254-0254,0256
GDDP418I ...
GDDP418I  SC VOL,LCL(2002,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),43FD-43FD,43CA
GDDP418I  SC VOL,LCL(2003,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),43FE-43FE,43C9
GDDP418I  SC VOL,LCL(2004,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),43FF-43FF,43C8
GDDP418I  Sitepair DC2-DC1
GDDP418I  Use these commands from a host with channel access to the Symmetrix unit at DC2
GDDP418I  Caution: use LCLISR2 if you want the devices at DC2 to be R2 devices
GDDP418I  SC VOL,LCL(600A,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),01E0-01FF,01E0
GDDP418I  SC VOL,LCL(600B,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0254-0254,0256
GDDP418I  SC VOL,LCL(600C,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0254-0254,0256
GDDP418I ...
GDDP418I  SC VOL,LCL(6002,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43CA-43CA,43FD
GDDP418I  SC VOL,LCL(6003,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43C9-43C9,43FE
GDDP418I  SC VOL,LCL(6004,00),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43C8-43C8,43FF
GDDP418I  Sitepair DC1-DC3
GDDP418I  Use these commands from a host with channel access to the Symmetrix unit at DC1
GDDP418I  to create an R22 configuration
GDDP418I  Caution: use LCLISR2 if you want the devices at DC1 to be R2 devices
GDDP418I  SC VOL,LCL(2008,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),01E0-01FF,01E0
GDDP418I  SC VOL,LCL(2009,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0254-0254,0256
GDDP418I  SC VOL,LCL(200A,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0254-0254,0256
GDDP418I ...
GDDP418I  SC VOL,LCL(2003,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43FE-43FE,43C9
GDDP418I  SC VOL,LCL(2004,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43C8-43C8,43FF
GDDP418I  Sitepair DC3-DC2
GDDP418I  Use these commands from a host with channel access to the Symmetrix unit at DC3
GDDP418I  to create an R22 configuration
GDDP418I  Caution: use LCLISR1 if you want the devices at DC3 to be R1 devices
GDDP418I  SC VOL,LCL(A008,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),01E0-01FF,01E0
GDDP418I  SC VOL,LCL(A009,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),0254-0254,0256
Using GDDR Utilities

GDDP418I SC VOL,LCL(A00A,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),025A-025A,0255

GDDP418I SC VOL,LCL(A00F,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43AA-43AA,43FD

GDDP418I SC VOL,LCL(A000,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43A9-43A9,43FE

GDDP418I SC VOL,LCL(A001,F9),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),43A8-43A8,43FF

GDDP418I Sitepair DC2-DC3

GDDP418I Use these commands from a host with channel access to the Symmetrix unit at DC2

GDDP418I Caution: use LCLISR2 if you want the devices at DC2 to be R2 devices

GDDP418I SC VOL,LCL(6007,31),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),01E0-01FF,01E0

GDDP418I SC VOL,LCL(6008,31),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),0252-0252,025A

GDDP418I SC VOL,LCL(6009,31),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR1),43A9-43AA,43FD

GDDP418I Sitepair DC3-DC2

GDDP418I Use these commands from a host with channel access to the Symmetrix unit at DC3

GDDP418I Caution: use LCLISR1 if you want the devices at DC3 to be R1 devices

GDDP418I SC VOL,LCL(A007,31),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),41B0-41CF,41D0

GDDP418I SC VOL,LCL(A008,31),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),41D0-41EF,41F0

GDDP418I SC VOL,LCL(A009,31),CREATEPAIR(ADCOPY-DISK,SUSPEND,STAR,LCLISR2),42B0-43AF,42D0

Messages in FBAMETA

If the FBAMETA DD card is present in the GDDRACDD JCL, it contains an overview of GDDR-managed FBA meta devices. It shows meta-head versus meta-member relations as well as SRDF device pairing information for FBA meta devices. For example:

GDDP422I FBA Meta Devices in GDDR managed ra-groups

GDDP422I RA-groups of type Internal

GDDP422I Site: DC1

GDDP422I Symmetrix: 000292601023

GDDP422I Head: 0220 Members: 0221 0222

GDDP422I Group: 10 0220 0221 0222

GDDP422I Group: 20 0220 0221 0222

GDDP422I Head: 0223 Members: 0224 0225 0226 0227

GDDP422I Group: 10 0223 0224 0225 0226 0227

GDDP422I Group: 20 0223 0224 0225 0226 0227

GDDP422I Head: 0228 Members: 0229 022A 022B

GDDP422I Group: 10 0228 0229 022A 022B

GDDP422I Group: 20 0228 0229 022A 022B

GDDP422I Head: 022C Members: 022D 022E 022F

GDDP422I Group: 10 022C 022D 022E 022F

GDDP422I Group: 20 022C 022D 022E 022F

GDDP422I Head: 0230 Members: 0231 0232 0233

GDDP422I Group: 10 0230 0231 0232 0233

GDDP422I Group: 20 0230 0231 0232 0233

GDDP422I Head: 0233 Members: 0234 0235 0236 0237

GDDP422I Group: 10 0233 0234 0235 0236 0237

GDDP422I Group: 20 0233 0234 0235 0236 0237

GDDP422I Head: 0238 Members: 0239 023A 023B

GDDP422I Group: 10 0238 0239 023A 023B

GDDP422I Group: 20 0238 0239 023A 023B

GDDP422I Head: 023C Members: 023D 023E 023F

GDDP422I Group: 10 023C 023D 023E 023F

GDDP422I Group: 20 023C 023D 023E 023F
Messages in TFLIST

If the TFLIST dd-card is present in the GDDRACDD JCL, it contains an overview of GDDR managed BCV devices. It shows BCV to STD pairing information, as well as information pertaining to the requested BCV types and sets. For example:

GDDP460I Site: DC1 - BCV Device ranges in Symm: 000195700579 (5876) Gx: 7309
GDDP461I BCV -STD -LBCV-H- STATUS -SDDF-INT-EXT-GOLD-TEST-IMPLEMENTATION
GDDP462I 0144-1144-0144-C-RESTORE -F5FD- Y - N -WARN- -MIRR
GDDP462I 0145-1145-0145-C-ESTABLISH-F5FD- Y - N -WARN- -MIRR
GDDP462I 0190-1190-0190-C-SPLIT -F5FD- Y - N -WARN-WARN-MIRR

Message GDDP462I has the following content:

- BCV

This is the TimeFinder target VMAX device number being discovered and defined to GDDR.
◆ STD
This is the TimeFinder source VMAX device number currently paired with the listed BCV.

◆ LBCV
This is the TimeFinder target VMAX device number most recently paired with the listed STD device. When the TF implementation is SnapVX, this is replaced with the last used snapshot name for the listed STD device.

◆ H
This is the host type for the listed BCV device, C for CKD, F for FBA.

◆ STATUS
This is the STD-BCV pair status for the listed STD and BCV devices.
The possible values depend on the TimeFinder implementation method for the listed VMAX unit. (See “IMPLEMENTATION” on page 278 for details.)
- For Legacy TF/Mirror and TF/Clone MF SNAP, possible values are ESTABLISH, SPLIT, RESTORE
- For SnapVX, possible values are LINKED, LINKEDR (LINKED with RESTORE option)

◆ SDDF
This is the SDDF session ID associated with the listed BCV device. When the TF implementation is SnapVX, this is the internal snapshot ID.

◆ INT
Indicates whether or not the listed STD device is in an internal RDF group.

◆ EXT
Indicates whether or not the listed STD device is in an external RDF group.

◆ GOLD
Indicates whether or not a GOLD set BCV is requested for the listed STD device, and whether ERROR or WARNING level messages where requested when the BCV set for the RDF group of this STD device is found to be incomplete.

◆ TEST
Indicates whether or not a TEST set BCV is requested for the listed STD device, and whether ERROR or WARNING level messages where requested when the BCV set for the RDF group of this STD device is found to be incomplete.

◆ IMPLEMENTATION
This is the TimeFinder implementation method used for the specified VMAX unit.
Possible values are:
- MIRR for legacy TF/Mirror with Clone Emulation
- CLON for TF/Clone MF Snap
- SNVX:<snapname1-32> for SnapVX, where <snapname1-32> is the 1 to 32 character snapshotname used for SnapVX devices at the listed site.
Exception reporting

Asymmetrical configurations

If GDDRACDD discovers asymmetry in the VMAX device numbers and the ALIGN option is specified, an warning message is issued:

```
GDDP415W ALIGN was specified or defaulted, but an asymmetrical device configuration was discovered.
GDDP415W Generated RDF.DEVICES parameters may not be aligned.
```

This message is issued to explain why generated parameters may not be aligned, which may be unexpected if you requested them to be aligned. It is of no further consequence.

Configuration validation

The following configuration errors are flagged with message GDDP420E. Some of these cause RC=8 and others cause a higher return code. See the following list.

**Note:** Items flagged with RC=8 may lead to the generation of RDF.DEVICES parameters, if GDDRACDD is run with the FORCE argument. There is no guarantee of any kind that the generated parameters are usable or even correct.

```
GDDP420E Error: <text>
<text> can be one of the following:

- No device pairs discovered for <site>-<site> group <rag> in Symmetrix <serial> at <site>
- No device pairs discovered for <site>-<site> group <rag> in Symmetrix <serial> at <site> and no other group associated in a GENERATE parameter
- Site pair <site>-<site> has device pairs in some groups, but group <rag> in Symmetrix <serial> at <site> has no device pairs
- Device pairs discovered for groups <rag> and <rag> in Symmetrix <serial> at <site>
- Different device pairs discovered for groups <rag> and <rag> in Symmetrix <serial> at <site>
- Device range <symmdev>-<symmdev> in Symmetrix <serial> is paired <DCm>-DC3 but not <DCn>-DC3, and no GENERATE parameter for <rag>
- All devices paired <DCm>-DC3 in group <rag> in Symmetrix <serial> should be paired <DCn>-DC3 in the same group. Found <rag> and <rag>
- Device pairs discovered for <site>-<site> group <rag> in Symmetrix <serial>
- Device <symmdev> in Symmetrix <serial> at <site> is paired with device <symmdev> in Symmetrix <serial> at <site>, but the remote device is paired with <symmdev> in <rag>
- Required <site>-<site> replication not found for device <symmdev> RA(<rag>) in Symmetrix <serial> at <site>
- Replication for device <symmdev> RA(<rag>) in Symmetrix <serial> at <site> does not form a [triplet] [quadruplet]. Device <symmdev> in Symmetrix <serial> at <site> is paired with <symmdev> in Symmetrix <serial>
- Invalid SRDF/Star configuration: <site>-<site> device range <symmdev>-<symmdev> in Symmetrix <serial> has no <site>-<site> protection.
- Expected Link-Block not found for devices <dev1>-<dev2> in ra-group <group> in Symmetrix <serial> at <site>
- Unexpected Link-Block found for devices <dev1>-<dev2> in ra-group <group> in Symmetrix <serial> at <site>
- Devices <dev1>-<dev2> in ra-group <group> in Symmetrix <serial> at <site> are not defined to GDDR
- Expected rdf-type <type1> for devices <dev1>-<dev2> in ra-group <group> in Symmetrix <serial> at <site>. Found <type2>
- Device <dev#> is defined to GDDR in range <dev1>-<dev2>, but could not be found in ra-group <group> in Symmetrix <serial> at <site>
```
Using GDDR Utilities

- Device <dev#> is <type1> in ra-group <group> in Symmetrix <serial> at <site>. Expected <type2>
- Unable to generate <site> device pair for range <symmdev>-<symmdev> in Symmetrix <serial> (GK <gk>) at <site>. No <site> ra-group found for range <symmdev>-<symmdev> in Symmetrix <serial> (GK <gk>) at <site>. (Groups found: <rag-list>)
- Unable to generate <site> device pair for range <symmdev>-<symmdev> in Symmetrix <serial> (GK <gk>) at <site>. Expected ra-group <rag>
- Site: <siteid> Symm: <serial> Meta-member: <symmdev> has different ra-groups than head <symmdev> (<mbr-groups> versus <head-groups>)
- Site: <siteid> Symm: <serial> Meta-member: <symmdev> has different RAID-level than head <symmdev> (<mbr-level> versus <head-level>)
- Site: <siteid> Symm: <serial> Meta-member: <symmdev> has different RDFtype than head <symmdev> (<mbr-type> versus <head-type>)

If running with BCV(Y) and one or more TFINDER parameters specify GOLD(ERROR) or TEST(ERROR):

- Site: <siteid> Symm: <serial> Device <symmdev> requires <n> BCV devices. Found {NONE | n (bcv#)}

The following configuration issues are flagged with message GDDP421W, and cause RC=4. These messages should be reviewed as they could either result in incorrect RDF.DEVICES parameters or cause issues during GDDR scripts.

GDDP421W Warning: <text>
<text> can be one of the following:
- Site: <siteid> Symm: <serial> Meta-member: <symmdev> has different RDF-mirror flags than head <symmdev> (<mbr-flags versus <head-flags>)
- Site: <siteid> Symm: <serial> Meta-member: <symmdev> has different ADCOPY flag than head <symmdev> (<mbr-flags versus <head-flags>)
- Site: <siteid> Symm: <serial> Meta-member: <symmdev> has different CG flags than head <symmdev> (<mbr-flags versus <head-flags>)
- Site: <siteid> Symm: <serial> GK: <gk> RA: <rag>. NON-GDDR group <rag> found for GDDR-managed device range <dev1>-<dev2>

If running with BCV(Y) and one or more TFINDER parameters specify GOLD(WARN) or TEST(WARN):

- Site: <siteid> Symm: <serial> Device <symmdev> requires <n> BCV devices. Found {NONE | n (bcv#)}

The following configuration issues are flagged with message GDDP421W, and cause RC=0. These messages occur when running with BCV(Y) or BCV(Y,FULL) and should be reviewed as they could either result in incorrect BCV related parameters or cause issues during GDDR scripts. These messages are prefixed with the selected TF method for the affected VMAX (MIRR, CLON or SNVX).

GDDP421W Warning: <text>
<text> can be one of the following:
- BCV Parameter <value> defines non-existing device <dev#> in Symmetrix <serial> at <site>
- BCV Parameter <value> defines special device <dev#> in Symmetrix <serial> at <site>
- BCV Parameter <value> defines non-BCV device <dev#> in Symmetrix <serial> at <site>
- BCV Parameter <value> defines BCV <bcv#>, which has state INVALID in Symmetrix <serial> at <site>
- BCV Parameter <value> defines BCV <bcv#>, which was never attached in Symmetrix <serial> at <site>
- BCV Parameter <value> defines BCV <bcv#>, which is associated with non-existing STD device <std#> in Symmetrix <serial> at <site>
- BCV Parameter <value> defines BCV <bcv#>, which is associated with special STD device <std#> in Symmetrix <serial> at <site>
- BCV Parameter <value> defines BCV <bcv#>, which is associated with CKD Meta member STD device <std#> in Symmetrix <serial> at <site>
- BCV Parameter <value> defines BCV <bcv#>, which is associated with non-GDDR STD device <std#> in Symmetrix <serial> at <site>
Using GDDR Utilities

GDDR RDF Director Overview utility (GDDRDIRS)

The GDDR RDF Director Overview Utility is designed to be run at any site with a GDDR configuration, and provides an overview of the currently configured directors used for SRDF in VMAX systems known to GDDR.

GDDRDIRS reports are available in two formats:

- The default format provides RDF director status for all directors used by all RDF groups in VMAX systems defined to GDDR.
- Use of the optional GDDR argument limits the report output to only the directors and RDF groups defined in GDDR RDF.DEVICES parameters (this includes the consistency protected RDF groups as well as the external RDF groups).

Sample JCL for the GDDRDIRS utility is in hlq.GDDRvrm.SAMPLIB (GDDRDIRS).

This sample report has been edited for brevity. The sections of interest are:

- GDDR SRDF directors overview for ALL RDF groups Figure 98 on page 282 and Figure 99 on page 283
- RDF groups by director by VMAX by site, Site: DC1, Figure 100 on page 284
- RDF groups by director by VMAX by site, Site: DC2, Figure 101 on page 285
- RDF groups by director by VMAX by site, Site: DC3 and Site UNK, Figure 102 on page 286
Using GDDR Utilities

<table>
<thead>
<tr>
<th>GDDR SRDF Directors overview for ALL ra-groups (MFE730) running at DC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>+GDDP430I  GDDR SRDF Directors overview for ALL rdf groups, 1 of 2</td>
</tr>
<tr>
<td>+GDDP432I  (I) DC1 1072 000190103389 (5773) 81 &lt;-ONL-&gt; 81 000190101982 (5073) 5072 DC2</td>
</tr>
<tr>
<td>+GDDP432I  28 (F) ONL(Y) CONN(Y) 28 (F) ONL(Y) CONN(Y)</td>
</tr>
<tr>
<td>+GDDP432I  38 (F) ONL(Y) CONN(Y) 38 (F) ONL(Y) CONN(Y)</td>
</tr>
<tr>
<td>+GDDP432I</td>
</tr>
</tbody>
</table>

+GDDP431I  (I) DC1 1072 000192601301 (5874) 80 <-ONL-> 80 000192601314 (5074) 27A DC2 |
| +GDDP431I  48 (F) ONL(Y) CONN(Y) 48 (F) ONL(Y) CONN(Y) |
| +GDDP432I  49 (F) ONL(Y) CONN(Y) 49 (F) ONL(Y) CONN(N)* |
| +GDDP432I  58 (F) ONL(Y) CONN(Y) 58 (F) ONL(Y) CONN(Y) |
| +GDDP432I  59 (F) ONL(Y) CONN(Y) 59 (F) ONL(Y) CONN(Y) |
| +GDDP432I |

+GDDP431I  (I) DC1 1073 000190103389 (5773) 82 <-ONL-> 82 000190100868 (5773) 3072 DC3 |
| +GDDP432I  03 (E) ONL(Y) CONN(Y) 03 (E) ONL(Y) CONN(Y) |
| +GDDP432I  0E (E) ONL(Y) CONN(Y) 3E (E) ONL(Y) CONN(Y) |
| +GDDP431I  (E) DC1 1073 000190103389 (5773) 88 <-ONL-> 88 000190100868 (5773) 3073 DC3 |
| +GDDP432I  03 (E) ONL(Y) CONN(Y) 03 (E) ONL(Y) CONN(Y) |
| +GDDP432I  0E (E) ONL(Y) CONN(Y) 3E (E) ONL(Y) CONN(Y) |
| +GDDP432I |

+GDDP431I  (I) DC1 ABC2 000192601301 (5874) 82 <-ONL-> 82 000192601313 (5874) BA7A DC3 |
| +GDDP432I  78 (E) ONL(Y) CONN(Y) 78 (E) ONL(Y) CONN(Y) |
| +GDDP432I  79 (E) ONL(Y) CONN(Y) 79 (E) ONL(Y) CONN(Y) |
| +GDDP431I  (E) DC1 ABC2 000192601301 (5874) 88 <-ONL-> 88 000192601313 (5874) BA7B DC3 |
| +GDDP432I  78 (E) ONL(Y) CONN(Y) 78 (E) ONL(Y) CONN(Y) |
| +GDDP432I  79 (E) ONL(Y) CONN(Y) 79 (E) ONL(Y) CONN(Y) |
| +GDDP432I |

+GDDP431I  (I) DC2 5073 000190101982 (5073) 83 <-ONL-> 83 000190100868 (5773) 3072 DC3 |
| +GDDP432I  03 (E) ONL(Y) CONN(Y) 03 (E) ONL(Y) CONN(Y) |
| +GDDP432I  1E (E) ONL(Y) CONN(Y) 1E (E) ONL(Y) CONN(N)* |
| +GDDP431I  (E) DC2 5073 000190101982 (5073) 89 <-ONL-> 89 000190100868 (5773) 3073 DC3 |
| +GDDP432I  03 (E) ONL(Y) CONN(Y) 03 (E) ONL(Y) CONN(Y) |
| +GDDP432I  1E (E) ONL(Y) CONN(Y) 1E (E) ONL(Y) CONN(N)* |
| +GDDP432I |

+GDDP431I  (I) DC2 B27B 000192601313 (5074) 83 <-ONL-> 83 000192601313 (5874) BA7A DC3 |
| +GDDP432I  78 (E) ONL(Y) CONN(Y) 78 (E) ONL(Y) CONN(Y) |
| +GDDP432I  79 (E) ONL(Y) CONN(Y) 79 (E) ONL(Y) CONN(Y) |
| +GDDP431I  (E) DC2 B27B 000192601313 (5074) 89 <-ONL-> 89 000192601313 (5874) BA7B DC3 |
| +GDDP432I  78 (E) ONL(Y) CONN(Y) 78 (E) ONL(Y) CONN(Y) |
| +GDDP432I  79 (E) ONL(Y) CONN(Y) 79 (E) ONL(Y) CONN(Y) |
| +GDDP432I |

Figure 98  GDDR SRDF directors overview for ALL RDF groups, 1 of 2
Using GDDR Utilities

Figure 99  GDDR SRDF directors overview for ALL RDF groups, 2 of 2
Using GDDR Utilities

Figure 100  RDF groups by director by VMAX by site, Site: DC1

+GDDP431I Site: DC1
+GDDP431I Symmetrix: 000190103389 (5773) - GK: 1072 Path: 5072.01
+GDDP431I Symmetrix: 000192601301 (5874) - GK: ABC1 Path: B27A.00
+GDDP431I Director: 28 (P) ONL(Y) CONN(Y)
+GDDP431I RA-groups: (I) 81
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 01 05 10 11 15 16 18 19 1A 3C 3D 3E 3F 40 4C 5A 70 80 B1 E1
+GDDP431I Director: 38 (P) ONL(Y) CONN(Y)
+GDDP431I RA-groups: (I) 81
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 01 05 10 11 15 16 18 19 1A 40 4C 70 80 B1 E1 F1 F4
+GDDP431I Director: 03 (B) ONL(Y) CONN(Y)
+GDDP431I RA-groups: (I) 82
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 02 04 06 18 20 21 25 36 B3 E2 E8 F2 F6
+GDDP431I Director: 0E (B) ONL(Y) CONN(Y)
+GDDP431I RA-groups: (I) 82
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 02 04 06 18 20 21 25 36 B3 E2 E8 F2 F6
+GDDP431I Director: 2E (B) ONL(Y) CONN(Y)
+GDDP431I RA-groups: (I) 82
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 02 06 20 21 5B E2
+GDDP431I Director: 13 (B) ONL(Y) CONN(Y)
+GDDP431I RA-groups: (I) 82
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 20 21
+GDDP431I Director: 33 (B) ONL(Y) CONN(N)
+GDDP431I RA-groups: (I) 80
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 00 03 10 15 1A 1F 25 2A 41 60 70 A0 B1 C1 D0 E4 F0 F4 05 5A
+GDDP431I Director: 49 (F) ONL(Y) CONN(Y)
+GDDP431I RA-groups: (I) 80
+GDDP431I RA-groups: (E) 88
+GDDP431I RA-groups: (N) 00 03 10 15 1A 1F 25 2A 41 60 70 A0 B1 C1 D0 E4 F0 F4 05 5A
Figure 101 RDF groups by director by VMAX by site, Site: DC2
Using GDDR Utilities

The GDDMSCFX utility validates the VMAX system information for each RDF group against the GDDR-managed configuration (if GDDR is available), validates the system information for each RDF group in the configuration against each other, and ensures only the expected system information exists. For each RDF group, the following information is validated where applicable:

- MSC scratch area
- Multi-box list
- Star/SQAR mode (Star, Star Recovery, SQAR, SQAR Recovery, MSC)
- Multi-box mode (MSC active)
- SRDF/A active and cycle-switching

Figure 102 RDF groups by director by VMAX by site, Site: DC3 and Site UNK

GDDR MSC Configuration Validation utility (GDDMSCFX)
- Group type (SRDF/A, SRDF/S, Recovery)
- Configuration type (MSC, Star, SQAR)
- Site (A, B, C, D)
- Star/SQAR recovery available
- Global consistency
- Site C not ahead of site B
- RDF group exists
- Other side RDF group

If GDDR is not available, the site type is validated at the controller level, and the configuration mode is validated at the configuration level, since they cannot be checked against the GDDR configuration. Any discrepancies are indicated in the report with an asterisk before and after the corresponding message. This can be used to detect any stale system info that may have been left behind by a previous configuration or recovery procedure.

If available, GDDR is interrogated for the current configuration. If GDDR is not available, an MSC group name and a controller to use as a starting point (local or remote to the system where GDDMSCFX is running) must be provided to identify the configuration. GDDMSCFX looks for RDF groups on the specified controller matching the specified MSC group name. The multi-box lists are then followed from controller to controller in order to discover all of the RDF groups in the configuration. Paths to each VMAX system are obtained from SCF.

**Note:** SCF is required and must be active. SCF V7.6 and higher is supported.

GDDMSCFX can run at any site and will adaptively find paths to each VMAX in the configuration. For better reliability in a disaster situation, the utility tries all possible local and remote paths before concluding that a controller is inaccessible. Before a path is selected, it undergoes a series of validations to ensure the UCB was not swapped, the UCB is accessible, I/O can be done to the UCB, I/O can be done to the remote controller (if applicable), and the serial number read matches the controller it is attempting to access. If no valid paths are found, that controller is deemed inaccessible, and the utility moves on to the next controller in the configuration.

**Report generation**

The GDDMSCFX utility generates a report containing a configuration summary followed by a breakdown of RDF groups by site. The summary indicates the configuration type, configuration mode, and topology. It also shows all GDDR-managed RDF groups between each site, including their expected roles.

Following the configuration summary, for each site the report lists all RDF groups in the configuration in addition to any non-GDDR-managed RDF groups matching the active MSC group name for that site. For each site, RDF groups are listed by VMAX with their corresponding multi-box lists.
Using GDDR Utilities

Stale or inconsistent system information is indicated in the report with asterisks before and after the corresponding message. This allows you to detect any stale data that may have been left behind by a previous configuration or recovery procedure. Lastly, a summary message indicates the return code:

0    Successful, no stale or inconsistent data was found
4    Warning, stale or inconsistent data was found
8    Error, processing failed unexpectedly
16   Abend occurred (R0 = completion code), SVCDUMP is taken

The report is written to the SYSPRINT DD. If diagnostic information is requested, it is also written to SYSPRINT in line with the report. All GDDR global variables used by the utility are echoed to the GVAROUT DD.

Using the Perform Health Check panel

You can run the GDDMSCFX utility in batch as well as in foreground using the CONFIG command on the Perform Health Check panel. Specify Option C in the Primary Options menu (on page 106) to invoke the Perform Health Check panel.

**Note:** To use the utility with the Perform Health Check panel, give GDDMSCFX APF-authorization by adding it to the IKJTSO parameter file under AUTHPGM.

Sample JCL

You can use the following sample JCL to run the GDDMSCFX utility, with these modifications:

- The STEPLIB must specify your GDDR LINKLIB.
- The xxxx in your SCF$xxxx DD DUMMY must match your SCF subsystem name.
- The xxxx in your GDD$xxxx DD DUMMY must match your GDDR subsystem name.

**Note:** These DDs link the utility to the correct SCF and GDDR tasks. The default SCF subsystem name is 'EMC', in which case the SCF$xxxx DD DUMMY is optional. The default GDDR subsystem name is 'GDDR', in which case the GDD$xxxx DD DUMMY is optional.

- The EXEC parameters may also need to be adjusted.

In the following sample, the default parameters are shown. If GDDR is not available, the CTRL and MSCG parameters are required to identify the desired configuration.

```
//GDDMSCFX JOB (EMC),'GDDR',NOTIFY=&SYSUID,
//         CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1)
/*/JOBPARM SYSAFF=*
//*
//* GDDMSCFX - VALIDATE GDDR CONFIG AGAINST BOX INFO
//*
//GDDMSCFX EXEC PGM=GDDMSCFX,PARM='DEBUG=N'
//STEP LIB DD DISP=SHR,DSN=DS-PREFIX.LINKLIB <--- YOUR GDDR LINKLIB
//SYSPRINT DD SYSOUT=* 
//GVAROUT DD SYSOUT=*
```
Using GDDR Utilities

//SYSABEND DD SYSOUT=*  
//SCF$EMC DD DUMMY <--- YOUR SCF SUBSYS NAME ('EMC' IS DEFAULT)  
//GDD$GDDR DD DUMMY <--- YOUR GDDR SUBSYS NAME ('GDDR' IS DEFAULT)  
//*

Parameters

**MSCGroup = <msc_group_name>**
The MSC group name identifying the configuration to validate.

**Note:** This parameter is required if GDDR is not available. Since GDDR cannot be interrogated for the configuration, GDDMSCFX will look for RDF groups matching this MSC group name on the controller indicated by the ConTRoLler parameter. Then the multi-box lists are followed from controller to controller to discover all of the RDF groups in the configuration.

**ConTRoLler = <serial#>**
The serial number of any accessible controller in the configuration to be used as a starting point (similar to a gatekeeper). The controller may be local or remote to the system where GDDMSCFX is running.

**Note:** This parameter is required if GDDR is not available.

**DeBuG = YES|NO**
Indicates whether to enable debugging. The default is NO.

Example reports

This section shows output generated by various scenarios.

Example 1: GDDR available

The following is an example GDDMSCFX report for a small SRDF/Star configuration. GDDR was available and was interrogated for the configuration. The program was run with MODE=VALIDATE and ended with return code 0 to indicate no inconsistent data was found.

```
*****************************************************
* GDDMSCFX-02/09/16-20.17-GDDR500-REL *
*****************************************************
```

**Program mode** : Validate  
**SCF version** : 8.0.0  
**SCF subsystem name** : DVT  
**GDDR version** : 5.0.0  
**GDDR subsystem name** : GDDR  
**GDDMSCFX PTFs applied** : none  
**Running at site** : DC2  
**Running on system** : GDDRDEVL (master C-system)  
**Date and time** : 02/10/16 12:56:09

**Configuration Summary**

**Config type:** Star, **Config mode:** Star, **Topology:** Concurrent
Using GDDR Utilities

GDDR-managed RDF groups:

<table>
<thead>
<tr>
<th>Site A (DC1)</th>
<th>SRDF/S</th>
<th>Site B (DC2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site A (DC1)</td>
<td>SRDF/A</td>
<td>Site C (DC3)</td>
</tr>
<tr>
<td>Site B (DC2)</td>
<td>Recovery</td>
<td>Site C (DC3)</td>
</tr>
</tbody>
</table>

Controller 000197200606 (5977.785)

RDF group 19 (SRDF/A R1)

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRDF/A active</td>
<td>Y</td>
</tr>
<tr>
<td>MSC active (MSC mode)</td>
<td>Y</td>
</tr>
<tr>
<td>Global consistency</td>
<td>Y</td>
</tr>
<tr>
<td>Star recovery available</td>
<td>Y</td>
</tr>
<tr>
<td>R1 Cleanup running</td>
<td>N</td>
</tr>
<tr>
<td>R1 Transmit Cycle empty</td>
<td>Y</td>
</tr>
<tr>
<td>R2 Apply cycle empty</td>
<td>Y</td>
</tr>
<tr>
<td>Capture cycle tag</td>
<td>E0000000000002BAD</td>
</tr>
<tr>
<td>Transmit cycle tag</td>
<td>E0000000000002BAC</td>
</tr>
<tr>
<td>Time since last cycle</td>
<td>3 sec</td>
</tr>
<tr>
<td>Mode</td>
<td>Star</td>
</tr>
<tr>
<td>Other side RDF group</td>
<td>000197200652/19</td>
</tr>
</tbody>
</table>

Multi-box list:

000197200606/18 --- 000197200652/18
000197200606/19 --- 000197200652/19

RDF group 18 (SRDF/A R1)

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRDF/A active</td>
<td>Y</td>
</tr>
<tr>
<td>MSC active (MSC mode)</td>
<td>Y</td>
</tr>
<tr>
<td>Global consistency</td>
<td>Y</td>
</tr>
<tr>
<td>Star recovery available</td>
<td>Y</td>
</tr>
<tr>
<td>R1 Cleanup running</td>
<td>N</td>
</tr>
<tr>
<td>R1 Transmit Cycle empty</td>
<td>Y</td>
</tr>
<tr>
<td>R2 Apply cycle empty</td>
<td>Y</td>
</tr>
<tr>
<td>Capture cycle tag</td>
<td>E0000000000002BAD</td>
</tr>
<tr>
<td>Transmit cycle tag</td>
<td>E0000000000002BAC</td>
</tr>
<tr>
<td>Time since last cycle</td>
<td>4 sec</td>
</tr>
<tr>
<td>Mode</td>
<td>Star</td>
</tr>
<tr>
<td>Other side RDF group</td>
<td>000197200652/19</td>
</tr>
</tbody>
</table>

Multi-box list:

000197200606/18 --- 000197200652/18
000197200606/19 --- 000197200652/19

RDF group 04 (SRDF/S R1)
Mode : Star
Other side RDF group : 000197200601/04

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

RDF group 03 (SRDF/S R1)
Mode : Star
Other side RDF group : 000197200601/03

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

Site B (DC2)
============
MSC site type : B
MSC group name : MSC_DVT
Controller 000197200601 (5977.785)
-----------------------------

RDF group 04 (SRDF/S R2)
Mode : Star
Other side RDF group : 000197200606/04

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

RDF group 25 (Recovery)
Mode : Star Recovery
Other side RDF group : 000197200652/25

No scratch area or MB list for Recovery groups (referenced in RDF group 04 scratch area)

RDF group 03 (SRDF/S R2)
Mode : Star
Other side RDF group : 000197200606/03

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

RDF group 24 (Recovery)
Mode : Star Recovery
Other side RDF group : 000197200652/24

No scratch area or MB list for Recovery groups (referenced in RDF group 03 scratch area)

Site C (DC3)
============
MSC site type : C
MSC group name : MSC_DVT
Controller 000197200652 (5977.785)
-----------------------------

GDDR MSC Configuration Validation utility (GDDMSCFX)
RDF group 19 (SRDF/A R2)

SRDF/A active : Y
MSC active (MSC mode) : Y
Global consistency : Y
Star recovery available : Y
Intervention required : N
R2 Cleanup running : N
R2 Restore complete : Y
R2 Apply cycle empty : Y
R1 Transmit Cycle empty : N
Receive cycle tag : E000000000002BAD
Apply cycle tag : E000000000002BAC
Time since last cycle : 0 sec
Mode : Star
Other side RDF group : 000197200606/19

Multi-box list:
000197200606/18 ---> 000197200652/18
000197200606/19 ---> 000197200652/19

RDF group 25 (Recovery)

Mode : Star Recovery
Other side RDF group : 000197200601/25

No scratch area or MB list for Recovery groups (referenced in RDF group 19 scratch area)

RDF group 18 (SRDF/A R2)

SRDF/A active : Y
MSC active (MSC mode) : Y
Global consistency : Y
Star recovery available : Y
Intervention required : N
R2 Cleanup running : N
R2 Restore complete : Y
R2 Apply cycle empty : Y
R1 Transmit Cycle empty : N
Receive cycle tag : E000000000002BAD
Apply cycle tag : E000000000002BAC
Time since last cycle : 0 sec
Mode : Star
Other side RDF group : 000197200606/18

Multi-box list:
000197200606/18 ---> 000197200652/18
000197200606/19 ---> 000197200652/19

RDF group 24 (Recovery)

Mode : Star Recovery
Other side RDF group : 000197200601/24

No scratch area or MB list for Recovery groups (referenced in RDF group 18 scratch area)

GDDMSCFX ended with RC 0
Example 2: GDDR not available

The following is an example GDDMSCFX report for a small SRDF/Star configuration. GDDR was not available. To identify the configuration, an MSC group name and a controller to use as a starting point were provided. The program was run with MODE=VALIDATE, CTRL=000192600455, and MSCG=MSC_DVT. It ended with return code 0, as no inconsistent data was found. Because GDDR was not available, the configuration summary showing the GDDR-managed configuration is omitted.

```
*******************************************************
* GDDMSCFX-02/09/16-20.17-GDDR500-REL *
*******************************************************
Program mode          : Validate
Starting controller   : 000197200606
MSC group name        : MSC_DVT
SCF version           : 8.0.0
SCF subsystem name    : DVT
GDDR version          : 5.0.0
GDDR subsystem name   : n/a
GDDMSCFX PTFs applied : none
Config mode           : Star
Running on system     : GDDRDEVL
Date and time         : 02/10/16 12:59:48

Site A
=====

Controller 000197200606 (5977.785)
--------------------------

RDF group 19 (SRDF/A R1)

SRDF/A active         : Y
MSC active (MSC mode) : Y
Global consistency    : Y
Star recovery available : Y
R1 Cleanup running   : N
R1 Transmit Cycle empty : Y
R2 Apply cycle empty  : Y
Capture cycle tag     : E00000000000002BD9
Transmit cycle tag    : E00000000000002BD8
Time since last cycle : 4 sec
Mode                  : Star
Other side RDF group  : 000197200652/19

Multi-box list:
000197200606/18 ---> 000197200652/18
000197200606/19 ---> 000197200652/19

RDF group 18 (SRDF/A R1)

SRDF/A active         : Y
MSC active (MSC mode) : Y
Global consistency    : Y
Star recovery available : Y
R1 Cleanup running   : N
R1 Transmit Cycle empty : Y
R2 Apply cycle empty  : Y
Capture cycle tag     : E00000000000002BD9
Transmit cycle tag    : E00000000000002BD8
Time since last cycle : 4 sec
Mode                  : Star
Other side RDF group  : 000197200652/18
```
Multi-box list:
000197200606/18 ---> 000197200652/18
000197200606/19 ---> 000197200652/19

RDF group 04 (SRDF/S R1)
Mode                    : Star
Other side RDF group    : 000197200601/04

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

RDF group 03 (SRDF/S R1)
Mode                    : Star
Other side RDF group    : 000197200601/03

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

Site B
======
Controller 000197200601 (5977.785)
----------------------------------
RDF group 04 (SRDF/S R2)
Mode                    : Star
Other side RDF group    : 000197200606/04

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

RDF group 25 (Recovery)
Mode                    : Star Recovery
No scratch area or MB list for Recovery groups (referenced in RDF group 04 scratch area)

RDF group 03 (SRDF/S R2)
Mode                    : Star
Other side RDF group    : 000197200601/03

Multi-box list:
000197200606/03 ---> 000197200601/03
000197200606/04 ---> 000197200601/04

RDF group 24 (Recovery)
Mode                    : Star Recovery
No scratch area or MB list for Recovery groups (referenced in RDF group 03 scratch area)

Site C
======
Controller 000197200652 (5977.785)
----------------------------------
RDF group 19 (SRDF/A R2)
SRDF/A active : Y
MSC active (MSC mode) : Y
Global consistency : Y
Star recovery available : Y
Intervention required : N
R2 Cleanup running : N
R2 Restore complete : Y
R2 Apply cycle empty : Y
R1 Transmit Cycle empty : N
Receive cycle tag : E000000000002BD9
Apply cycle tag : E000000000002BD8
Time since last cycle : 1 sec
Mode : Star
Other side RDF group : 000197200606/19

Multi-box list:
  000197200606/18 ---> 000197200652/18
  000197200606/19 ---> 000197200652/19

RDF group 25 (Recovery)
Mode : Star Recovery

No scratch area or MB list for Recovery groups (referenced in RDF group 19 scratch area)

RDF group 18 (SRDF/A R2)
SRDF/A active : Y
MSC active (MSC mode) : Y
Global consistency : Y
Star recovery available : Y
Intervention required : N
R2 Cleanup running : N
R2 Restore complete : Y
R2 Apply cycle empty : Y
R1 Transmit Cycle empty : N
Receive cycle tag : E000000000002BD9
Apply cycle tag : E000000000002BD8
Time since last cycle : 1 sec
Mode : Star
Other side RDF group : 000197200606/18

Multi-box list:
  000197200606/18 ---> 000197200652/18
  000197200606/19 ---> 000197200652/19

RDF group 24 (Recovery)
Mode : Star Recovery

No scratch area or MB list for Recovery groups (referenced in RDF group 18 scratch area)

GDDMSCFX ended with RC 0

Example 3: SQAR configuration

The following is an example GDDMSCFX report for a small SQAR configuration. DDR was available and was interrogated for the configuration. The program was run with MODE=VALIDATE and ended with return code 0, as no inconsistent data was found.
Using GDDR Utilities

SCF version : 8.0.0
SCF subsystem name : DVT
GDDR version : 5.0.0
GDDR subsystem name : GDDR
GDDMSCFX PTFs applied : none
Running at site : DC2
Running on system : GDDRDEVL (master C-system)
Date and time : 02/10/16 12:56:09

Configuration Summary
=====================
Config type: SQAR, Config mode: SQAR

GDDR-managed RDF groups:

Site A (DC1)   SRDF/S   Site B (DC2)
--------------  ----------  ============
000195700240  2A ---> 2A  000195700255

Site A (DC1)   SRDF/A   Site C (DC3)
--------------  ----------  ============
000195700240  BB ---> BB  000195700941

Site B (DC2)   SRDF/A   Site D (DC4)
--------------  ----------  ============
000195700255  2C ---> 2C  000195700235

Site C (DC3)   Recovery   Site D (DC4)
--------------  ----------  ============
000195700941  BC ---> BC  000195700235

Site A (DC1)
--------------

MSC site type : A
MSC group name: SQAR_A_1

Controller 000195700240 (5876.269)

RDF group BB (SRDF/A R1)

SRDF/A active : Y
MSC active (MB mode) : Y
Global consistency : Y
SRDF recovery available : Y
Intervention required : N
R1 cleanup running : N
Capture cycle empty : Y
Transmit cycle empty : N
Capture cycle tag : C200000000001BF59
Transmit cycle tag : C200000000001BF58
Time since last cycle : 3 sec
Mode : SQAR
Other side RDF group : 000195700941/BB

Multi-box list:
000195700240/BB ---> 000195700941/BB

RDF group 2A (SRDF/S R1)

Mode : SQAR
Other side RDF group : 000195700255/2A

No scratch area or MB list for SRDF/S groups in SQAR mode
Site B (DC2)
==============

MSC site type: A
MSC group name: SQAR_A_2

Controller 000195700255 (5876.269)
----------------------------------

RDF group 2C (SRDF/A R1)

SRDF/A active : Y
MSC active (MB mode) : Y
Global consistency : Y
SQAR recovery available : Y
Intervention required : N
R1 cleanup running : N
Capture cycle empty : Y
Transmit cycle empty : N
Capture cycle tag : C20000000001BF59
Transmit cycle tag : C20000000001BF58
Time since last cycle : 3 sec
Mode : SQAR
Other side RDF group : 000195700235/2C

Multi-box list:
000195700255/2C ---> 000195700235/2C

RDF group 2A (SRDF/S R2)

Mode : SQAR
Other side RDF group : 000195700240/2A

No scratch area or MB list for SRDF/S groups in SQAR mode

Site C (DC3)
==============

MSC site type: C
MSC group name: SQAR_A_1

Controller 000195700941 (5876.269)
----------------------------------

RDF group BB (SRDF/A R2)

SRDF/A active : Y
MSC active (MB mode) : Y
Global consistency : Y
SQAR recovery available : Y
Intervention required : N
R2 cleanup running : N
R2 restore complete : Y
Receive cycle empty : N
Apply cycle empty : Y
Receive cycle tag : C20000000001BF58
Apply cycle tag : C20000000001BF57
Time since last cycle : 5 sec
Mode : SQAR
Other side RDF group : 000195700240/BB

Multi-box list:
000195700240/BB ---> 000195700941/BB

RDF group BC (Recovery)
Using GDDR Utilities

Mode : SQAR Recovery
Other side RDF group : 000195700235/BC

No scratch area or MB list for Recovery groups (referenced in RDF group BB scratch area)

Site D (DC4)
==============

MSC site type : C
MSC group name: SQAR_A_2

Controller 000195700235 (5876.269)
----------------------------------

RDF group 2C (SRDF/A R2)

SRDF/A active : Y
MSC active (MB mode) : Y
SQAR recovery available : Y
Intervention required : N
R2 cleanup running : N
R2 restore complete : Y
Receive cycle empty : N
Apply cycle empty : Y
Receive cycle tag : C20000000001BF58
Apply cycle tag : C20000000001BF57
Time since last cycle : 5 sec
Mode : SQAR
Other side RDF group : 000195700255/2C

Multi-box list:
000195700255/2C ---> 000195700235/2C

RDF group BC (Recovery)

Mode : SQAR Recovery
Other side RDF group : 0001957000255/BC

No scratch area or MB list for Recovery groups (referenced in RDF group 2C scratch)

Example 4: Stale system information found

This example is a snippet from a GDDMSCFX report for an SRDF/Star configuration. GDDR was available and was interrogated for the configuration. The program was run in VALIDATE mode and ended with return code 4, as stale system information was found at sites B and C. Specifically, scratch areas were found for the recovery groups at sites B and C, but recovery groups should not have scratch areas. These scratch areas are leftover from when these RDF groups were active in SRDF/A, as they were not cleaned up properly.

Note: This is an output snippet; only the stale system information from site B is shown.

RDF group D4 (SRDF/A R1)

SRDF/A active : N
MSC active (MB mode) : Y
Global consistency : Y
Star recovery available : Y
Intervention required : N
R1 cleanup running : N
Capture cycle empty : Y
Transmit cycle empty : N
Capture cycle tag       : E0000000000202B2
Transmit cycle tag      : E0000000000202B1
Time since last cycle   : 956 sec
Mode                    : Star Recovery
Other side RDF group    : 000195700600/D4

Multi-box list:
000195700578/3C ---> 000195700600/3C
000195700578/C4 ---> 000195700600/C4
000195700578/D4 ---> 000195700600/D4

* RDF group type inconsistent (scratch area indicates SRDF/A, GDDR expects Recovery) *
* Site inconsistent (scratch area indicates MSC site A, GDDR expects MSC site B) *
* SRDF/A not active *
* Last cycle switch was 956 seconds ago *

RDF group D1 (Recovery)

Mode                    : Star Recovery
Other side RDF group    : 000195700579/D1

No scratch area or MB list for Recovery groups (referenced in RDF group D4 scratch area)

* RDF group no longer exists *
* RDF group not known by GDDR (may be a stale definition) *

If GDDR is not available, validation against the GDDR configuration is not possible. However, the MSC scratch areas and multi-box lists for each RDF group in the configuration are validated against each other. The site type is validated at the controller level and the configuration mode is validated at the configuration level. The following are examples of the warning messages displayed if a site or configuration mode mismatch is detected:

* Site mismatch on controller 000195700080 (some RDF groups indicate site C, others indicate site A) *

* Config mode mismatch (some RDF groups indicate Star, others indicate MSC) *

GDDR DLm Serial Number utility (GDDRDLSN)

The DLm Serial Number utility is used in SRDF/Star configurations with DLm support to list FICON serial numbers of all DLM devices (and DLMSlaves, if any) defined to GDDR. The serial numbers are obtained from the DLm. Appropriate DLm code is required for the utility to work.

The sample GDDRDLSN utility JCL is in hlq.GDDRvrm.SAMPLIB (GDDRDLSN).

Requirements

- The STEPLIB DD statement must be pointing to the hlq.GDDRvrm.LINKLIB library.
- The SYSEXEC DD statement must be pointing to the hlq.GDDRvrm.rcxfe library
- GDDRSCF must be up and running, with the same SCFS$ connector used in the GDDRDLSN //SCFS$XXX DD DUMMY statement.
GDDR Maintenance Level Reporting utility (GDDRREG)

The GDDRREG utility reports the GDDR maintenance level. It can be run in batch and is also called from GDDRMAIN at startup.

At GDDRMAIN startup, message GDDM072I displays (after the GDDM070I herald message). This message indicates the latest date found in all ASM modules and the latest date found in all REXXC modules, as well as the highest version and PTF number found in all ASM modules.

Note: The highest version and PTF number can only be determined for ASMs. In REXXC modules, only the date and time are visible as the text is compressed/encrypted, but the date can still be used to identify the maintenance level.

GDDM070I GDDR V5.0.0 main task starting - GDDRMAIN-12/01/15-17.51-GDDR500-GD50001
GDDM072I GDDR maintenance level - ASM: 12/01/15 (V5.0.0 GD50001), REXX: 12/01/15

Also at GDDRMAIN startup, the GDDR Module Maintenance Levels report is written to SYSPRINT.

Note: For GDDR V5.0 and higher, the SYSPRINT DD has been added to the GDDRMAIN PROC. Update existing GDDRMAIN PROCs to include 'SYSPRINT SYSOUT='* to ensure the module maintenance levels are available.

Sample JCL

The following sample JCL can be used to run the GDDRREG utility, with some modifications:

- The STEPLIB must specify your GDDR LINKLIB.
- The xxxx in your SCF$xxxx DD DUMMY must match your SCF subsystem name. This DD is what links the utility to the correct SCF task. The default SCF subsystem name is 'EMC', in which case the SCF$xxxx DD DUMMY is optional.
- The EXEC parameters may also need to be adjusted.

```
//GDDRREG JOB (EMC), 'GDDR', NOTIFY=&SYSUID,
//CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1)
//*
//********************************************************************
//* GDDRREG - DISPLAY GDDR MAINTENANCE LEVEL INFORMATION              *
//*                                                                   *
//* EXEC PARAMETERS:                                                  *
//*                                                                   *
//*  [SUMMARY], - SHOW MAINTENANCE SUMMARY REPORT (DEFAULT)           *
//*  [COMPOSITE], - SHOW COMPOSITE LOAD MODULE MAINTENANCE REPORT     *
//*  [DETAIL], - SHOW LOAD MODULE CSECT DETAIL REPORT                 *
//*  [LOADMOD=LMOD], - SHOW CSECT DETAIL REPORT FOR SPECIFIED MODULE   *
//*  [DEBUG] - ENABLE DEBUGGING TO SYSPRINT DD                         *
//*                                                                   *
//* NOTE: LOADMOD NOT VALID WITH ANY OTHER PARAMETER EXCEPT DEBUG.    *
//********************************************************************
```
GDDR Utilities

GDDRREG example output

The following output example is with PARM='SUMMARY,COMPOSITE,DETAIL', therefore application, load module, and CSECT level reports are displayed.

```
* GDDR Maintenance Summary      12/03/15 18:32 *
***************************************************
Load module group         Date      Ver  PTF
------------------------  --------  ---  ---
GDDR ASM                  12/01/15  500  001
GDDR REXX                 12/01/15  500

* GDDR Module Maintenance Levels   12/03/15 18:32 *
***************************************************
Load module group         Date      Ver  PTF
------------------------  --------  ---  ---
GDDR ASM                  12/01/15  500  001
Module     Date       Ver   PTF
--------   --------   ---   ---
GDDBCPCM   12/01/15   500   001
GDDBCPCO   12/01/15   500   001
GDDBCPDI   12/01/15   500   001
GDDBCPMLS  12/01/15   500   001
GDDBCPQY   12/01/15   500   001
.  .  .
Load module group         Date
------------------------  --------
GDDR REXX                 12/01/15
Module     Date
--------   --------
GDDCHKDS   12/01/15
GDDDLMAC   12/01/15
GDDDLMMST  12/01/15
GDDDLMAK   12/01/15
GDDDLMAP   12/01/15
GDDDLMLAS  12/01/15
.  .  .

* GDDR Module CSECT Detail         12/03/15 18:32 *
***************************************************
Load module group         Date      Ver  PTF
------------------------  --------  ---  ---
GDDR ASM                  12/01/15  500  001
Module     Date       Ver   PTF
--------   --------   ---   ---
GDDBCPCM   12/01/15   500   001
```

GDDR Maintenance Level Reporting utility (GDDRREG) 301
GDDR Gatekeeper Validation utility (GDDGATEK)

Use the GDDGATEK utility to display and validate the GDDR gatekeepers that should be accessible to the site where it is run. For example, the utility is run on site B in a Star configuration, gatekeepers at sites A and B should be accessible. In addition, GDDGATEK can optionally unbox boxed devices.

Invoking the utility

GDDGATEK can be run in batch as well as via in foreground using either of the following methods:

- Issue the F GDDRMAIN, GATEK modify command.
- Issue the GATEK primary command on the Perform Health Check panel as shown in “Option C: Checkup—Perform Pre-script Checkup” on page 196.

The syntax for the GATEK modify command and Perform Health Check panel primary command is GATEK [parms], where parms supports all parameters supported in batch and the syntax is the same.

**Note:** GDDGATEK must be APF-authorized to use the AUTO_UNBOX option and should be added to the IKJTSO parameter file under AUTHPGM.

Sample JCL

Use the following sample JCL to run the GDDGATEK utility, with the following modifications:

- The STEPLIB must specify your GDDR LINKLIB.
- The xxxx in your SCF$xxxx DD DUMMY must match your SCF subsystem name.
- The xxxx in your GDD$xxxx DD DUMMY must match your GDDR subsystem name.

**Note:** These DDs link the utility to the correct SCF and GDDR tasks. The default SCF subsystem name is 'EMC', in which case the SCF$xxxx DD DUMMY is optional. The default GDDR subsystem name is 'GDDR', in which case the GDD$xxxx DD DUMMY is optional.

- The EXEC parameters may also need to be adjusted.

In the sample below, the default parameters are shown.

```
//GDDGATEK JOB (EMC), 'GDDR', NOTIFY=&SYSUID,
  CLASS=A, MSGCLASS=X, MSGLEVEL=(1, 1)
/*JOBPARM SYSAFF=* 
```
Parameter syntax

[[MODE]=VALIDate]
[,SITE={DC1|DC2|DC3|DC4}]
[,ConTroller=<serial#>]
[,Auto_UnBoX={Yes|No}]
[,PRompt={Yes|No}]
[,SIMulate={Yes|No}]
[,CONsole={Yes|No}]
[,DeBuG={Yes|No}]

Note: Parameters can be specified as keyword only or keyword=value. For example, DeBuG implies DeBuG=YES. The omission of DeBuG implies DeBuG=No.

Parameter descriptions

**MODE=VALIDate**
Validates GDDR gatekeepers that should be accessible to the site where the utility is running.

**SITE = {DC1|DC2|DC3|DC4}**
Limits the scope of the utility to gatekeepers at the specified site.

**ConTroller = <serial#>**
Limits the scope of the utility to gatekeepers on the specified controller.

**AutoUnBoX =Yes|No**
Yes specifies to automatically unbox devices. Note that this must be APF-authorized.
No specifies not to unbox devices (default).

**PRompt =Yes|No**
Yes specifies to prompt before unboxing a device. Note that this is valid with AutoUnBoX=Yes only.
No suppresses all prompts (default)

**SIMulate =Yes|No**
Yes simulates unboxing of devices without actually unboxing them. Note that this is valid with AutoUnBoX=Yes only.
No disables simulation mode (default).
FoRCe =Yes|No
Yes forces unboxing of devices when it normally would not be allowed (for example, the device is online or pending offline). There is no guarantee a device can be unboxed, but the usual validation will be bypassed. Note that this is valid with AutoUnBoX=Yes only.

No disables forcefully unboxing devices (default).

CONsole =Yes|No
Yes specifies to write the body of report to console in addition to SYSPRINT.
No specifies to write the report to SYSPRINT only (default).

DeBuG =Yes|No
Yes enables debugging to SYSPRINT DD.
No disables debugging (default).

Return codes

Table 20 lists GDDGATEK return code values.

<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful, all gatekeepers accessible and offline</td>
</tr>
<tr>
<td>4</td>
<td>Warning, one or more gatekeepers are online or allocated</td>
</tr>
<tr>
<td>8</td>
<td>Error, one or more gatekeepers are inaccessible</td>
</tr>
<tr>
<td>12</td>
<td>Severe error, processing failed unexpectedly</td>
</tr>
<tr>
<td>16</td>
<td>Abend occurred, SCVDUMP is taken</td>
</tr>
</tbody>
</table>

Gatekeeper status

For each gatekeeper device, its status consists of a combination of the following strings.

- Valid:
  - accessible
  - offline
  - (unBOXed)

- Undesirable, will result in warning message(s) and RC=4:
  - online
  - allocated

- Device is inaccessible, will result in error message(s) and RC=8:
  - BOXed
  - defer BOXed
  - no paths
  - no logical paths
  - not connected
Using GDDR Utilities

- hot IO
- permanent error
- swap in progress
- IO stopped by IOACTION STOP command
- wrong controller (serial#)
- IO timeout
- IO error
- UCB not valid
- UCB not found
- syscall error (syscall_rc)
- FC01 error (rc/rcs/rcx)

Examples

In the following examples, two gatekeepers are online (and one is allocated), a few of them are boxed, and an IOACTION STOP has been issued to one of them.

GDDGATEK report with AUTO_UNBOX=N:

GDDUG60I ******************************************************
GDDUG61I * GDDGATEK-11/24/15-19.39-GDDR500-REL *
GDDUG60I ******************************************************
GDDUG00I
GDDUG10I Program mode : Validate
GDDUG13I SCF version : 8.0.0
GDDUG14I SCF subsystem name : DVT
GDDUG15I GDDR version : 5.0.0
GDDUG16I GDDR subsystem name : GDDR
GDDUG17I GDDGATEK PTFs applied : n/a
GDDUG18I Running at site : DC2
GDDUG19I Running on system : GDDRDEVL (master C-system)
GDDUG20I Date and time : 11/24/15 21:04:37
GDDUG00I
GDDUG30I ------------------------------------------------------------------
GDDUG31I  CUU   Site  Controller     uCode  Status
GDDUG32I  ----  ----  -------------  -----  ------------------------------
GDDUG32W  7000  DC1   0123456-00579  5876   online   accessible
GDDUG32W  7001  DC1   0123456-00579  5876   online   accessible  allocated
GDDUG32I  7002  DC1   0123456-00579  5876   offline  accessible
GDDUG32I  7003  DC1   0123456-00579  5876   offline  accessible
GDDUG32I  7004  DC1   0123456-00579  5876   offline  accessible
GDDUG32E  7005  DC1   0123456-00579  5876   offline  BOXed
GDDUG32E  7006  DC1   0123456-00579  5876   offline  BOXed
GDDUG32E  7007  DC1   0123456-00579  5876   offline  BOXed
GDDUG32I  7008  DC1   0123456-00579  5876   offline  accessible
GDDUG32I  7009  DC1   0123456-00579  5876   offline  accessible
GDDUG32I  8000  DC2   0123456-00578  5876   offline  accessible
GDDUG32I  8001  DC2   0123456-00578  5876   offline  accessible
GDDUG32I  8002  DC2   0123456-00578  5876   offline  accessible
GDDUG32I  8003  DC2   0123456-00578  5876   offline  accessible
GDDUG32E  8004  DC2   0123456-00578  5876   offline  IO stopped by IOACTION STOP command
GDDUG32I  8005  DC2   0123456-00578  5876   offline  accessible
GDDUG32I  8006  DC2   0123456-00578  5876   offline  accessible
GDDUG32I  8007  DC2   0123456-00578  5876   offline  accessible
GDDUG32I  8008  DC2   0123456-00578  5876   offline  accessible
GDDUG32I  8009  DC2   0123456-00578  5876   offline  accessible
GDDUG30I ------------------------------------------------------------------
GDDUG00I
GDDUG70W  2 online gatekeeper devices found
GDDUG71W  1 allocated gatekeeper devices found
GDDUG90E GDDGATEK ended with RC 8
GDDGATEK report with AUTO_UNBOX=Y:

GDDUG60I ***************************************************************
GDDUG61I * GDDGATEK-11/24/15-19.39-GDDR500-REL *
GDDUG60I ***************************************************************
GDDUG10I Program mode          : Validate, AutoUnbox, NoPrompt
GDDUG13I SCF version           : 8.0.0
GDDUG14I SCF subsystem name    : DVT
GDDUG15I GDDR version          : 5.0.0
GDDUG16I GDDR subsystem name   : GDDR
GDDUG17I GDDR GATEK PTFs applied: n/a
GDDUG18I Running at site       : DC2
GDDUG19I Running on system     : GDDRDEVL (master C-system)
GDDUG20I Date and time         : 11/24/15 21:05:25
GDDUG00I
GDDUG30I ------------------------------------------------------------------
GDDUG31I  CUU   Site  Controller     uCode  Status
GDDUG33I  ----  ----  -------------  -----  ------------------------------
GDDUG32W  7000  DC1   0123456-00579  5876   online   accessible
GDDUG32W  7001  DC1   0123456-00579  5876   online   accessible  allocated
GDDUG32I  7002  DC1   0123456-00579  5876   offline   accessible
GDDUG32I  7003  DC1   0123456-00579  5876   offline   accessible
GDDUG32I  7004  DC1   0123456-00579  5876   offline   accessible
GDDUG32I  7005  DC1   0123456-00579  5876   offline   accessible  (unBOXed)
GDDUG32I  7006  DC1   0123456-00579  5876   offline   accessible  (unBOXed)
GDDUG32I  7007  DC1   0123456-00579  5876   offline   accessible  (unBOXed)
GDDUG32I  7008  DC1   0123456-00579  5876   offline   accessible
GDDUG32I  7009  DC1   0123456-00579  5876   offline   accessible
GDDUG32I  8000  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8001  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8002  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8003  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8004  DC2   0123456-00578  5876   offline   IO stopped by IOACTION STOP command
GDDUG32I  8005  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8006  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8007  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8008  DC2   0123456-00578  5876   offline   accessible
GDDUG32I  8009  DC2   0123456-00578  5876   offline   accessible
GDDUG30I ------------------------------------------------------------------
GDDUG00I
GDDUG30E 1 inaccessible gatekeeper devices found
GDDUG90I 3 gatekeeper devices successfully unboxed
GDDUG71W 2 online gatekeeper devices found
GDDUG71W 1 allocated gatekeeper devices found
GDDUG90E GDDGATEK ended with RC 8

The following is an example of the GATEK GDDRMAIN modify command.

GATEK
GDDUG30I --------------------------------------------------------------
GDDUG31I  CUU   Site  Controller     uCode  Status
GDDUG33I  ----  ----  -------------  -----  --------------------------
GDDUG32I  2376  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  2377  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  2378  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  2379  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  237A  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  237B  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  237C  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  237D  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  237E  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  237F  DC1   0001972-00606  5977   offline   accessible
GDDUG32I  C376  DC2   0001972-00601  5977   offline   accessible
GDDUG32I  C377  DC2   0001972-00601  5977   offline   accessible
GDDUG32I  C378  DC2   0001972-00601  5977   offline   accessible
GDDUG32I  C379  DC2   0001972-00601  5977   offline   accessible
GDDUG32I C37A DC2 0001972-00601 5977 offline accessible
GDDUG32I C37B DC2 0001972-00601 5977 offline accessible
GDDUG32I C37C DC2 0001972-00601 5977 offline accessible
GDDUG32I C37D DC2 0001972-00601 5977 offline accessible
GDDUG32I C37E DC2 0001972-00601 5977 offline accessible
GDDUG32I C37F DC2 0001972-00601 5977 offline accessible
GDDUG30I --------------------------------------------------------------

Using GDDR Utilities

The GDDRSC06 utility queries the SRDF device status for selected sites and RDF groups. This may be helpful in identifying issues in the SRDF replication groups impacting script execution. The utility can also verify that devices are in the desired replication status.

**Sample JCL**

To execute GDDRSC06, update the SAMPLIB example PROC `GDDRRXST` statement using the indicated options as desired.

```
//GDDRSC06 JOB (EMC), 'GDDRSC', CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1)
//*  RUN THE GDDRSC06 SRDF DEVICE STATUS CHECKER
//*  **************************************************************************
//*  POINT TO YOUR CUSTOMIZED GDDRPROC
//GDDRPROC JCLLIB ORDER=(............................................)
//*  **************************************************************************
//SC06 EXEC GDDRPROC
//GDDR_SYSTsin DD *
EXECUTIL SEARCHDD(YES)
GDDRRXST ........................
//SCF$XXXX DD DUMMY
//*  **************************************************************************
//*  USAGE NOTES:                                                        *
//*  GDDRRXST GDDRSC06 SITEID F_CODE J-TYPE'                             *
//*                                                                  *
//*  SITEID : SPECIFY DC1, DC2, DC3 OR DC4                              *
//*                                                                  *
//*  F_CODE : SPECIFY ONE OF: 03 04 06 07 08                             *
//*    - 03: VERIFY THAT DEVICES ARE R1 ON THE SPECIFIED RDF-LEG        *
//*    - 04: VERIFY THAT DEVICES ARE R2 ON THE SPECIFIED RDF-LEG        *
//*    - 06: VERIFY THAT R1 DEVICES ARE NOT TNR ON THE SPECIFIED         *
//*          RDF LEG                                                   *
//*    - 07: VERIFY THAT R1 DEVICES ARE TNR ON THE SPECIFIED RDF LEG    *
//*    - 08: VERIFY THAT DEVICES ARE R21                                *
//*                                                                  *
//*  J-TYPE = SRDF-TYPE. SPECIFY J0 FOR SRDF/S OR JA FOR SRDF/A         *
//*          FOR SITE DC3, IN SRDF/STAR CONFIGURATIONS,                 *
//*          JA2 INDICATES DC3-DC2 RELATIONSHIP.                         *
//*                                                                  *
//*  EXAMPLE:                                                         *
//*  GDDRRXST GDDRSC06 DC1 03 J0                                       *
//*  **************************************************************************
```

******************************** Bottom of Data *******************************

**TimeFinder Management utility (GDDRTF20)**

The TimeFinder Management utility allows you perform GDDR local replication management functions, using TimeFinder/Mirror, TimeFinder/Clone and TimeFinder/SnapVX. This includes creating a PiT copy of your data at a selected site as well as performing LINK and UNLINK operations for your SnapVX target devices.

GDDRTF20 dynamically determines which technology applies to each VMAX system at the target site based on the choices made on the Define Configuration Features panel in the GDDR parameter wizard.
Note: EMC recommends executing GDDRTF20 at a site with channel connectivity to the VMAX units being managed.

The GDDRTF20 SPLIT command creates a consistent PIT image. The SPLITINSTANT command does not guarantee consistency. GDDRKF20 supports the following EMC local replication methods:

- TimeFinder/Mirror with the RE-ESTABLISH, SPLIT and SPLITINSTANT commands
- TimeFinder/Clone with the RE-ESTABLISH, SPLIT, and SPLITINSTANT commands
- TimeFinder/SnapVX with the RE-ESTABLISH, SPLIT, SPLITINSTANT, LINK, and UNLINK commands

Sample JCL

GDDRGDDR500.SAMPLIB(GDDRTF20)

/GDDRTF20 JOB (EMC), 'GDDRHIC', CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1)
/****
/* RUN THE GDDR TF MANAGEMENT UTILITY (GDDRKF20) TO CHANGE
/* LOCAL REPLICATION STATE OF GDDR MANAGED DEVICES.
/****
/************************************************************************/
/* RECOMMENDATION IS TO RUN THIS UTILITY AT A SITE WHICH HAS
/* CHANNEL CONNECTIVITY TO THE SYMMETRIX UNITS BEING MANAGED.
/************************************************************************/
/* GDDRKF20 SUPPORTS THE FOLLOWING EMC LOCAL REPLICATION METHODS:
/* - TF/MIRROR WITH RE-ESTABLISH, SPLIT AND SPLITINSTANT COMMANDS
/* - TF/CLONE WITH RE-ESTABLISH, SPLIT AND SPLITINSTANT COMMANDS
/* - TF/SNAPVX WITH RE-ESTABLISH, SPLIT, SPLITINSTANT, LINK AND
/* UNLINK COMMANDS.
/************************************************************************/
/* THE GDDRKF20 SPLIT COMMAND CREATES A CONSISTENT PIT IMAGE.
/* SPLITINSTANT DOES NOT GUARANTEE CONSISTENCY.
/************************************************************************/
/* GDDRKF20 AUTOMATICALLY DETERMINES THE LOCAL REPLICATION METHOD
/* TO BE USED FOR EACH SYMMETRIX UNIT, BASED ON THE FOLLOWING
/* FEATURES PANEL, WHICH CAN BE SET INDIVIDUALLY FOR EACH SITE
/* IN YOUR CONFIGURATION:
/* - SNAPVX SOFTLINK SUPPORT:
/*   - IF THIS FIELD IS Y, GDDRKF20 USES
/*     - TF/SNAPVX ON EMC SYMMETRIX VMAX V3 HW
/*     - TF/CLONE ON EMC SYMMETRIX VMAX V1/V3 HW
/* - IF THIS FIELD IS N, GDDRKF20 USES THE NEXT FIELD:
/*   - LEGACY TF METHOD:
/*     - IF THIS FIELD IS M (MIRROR), GDDRKF20 USES
/*       - TF/MIRROR ACROSS V1/V2/V3 HW
/*     - IF THIS FIELD IS C (CLONE), GDDRKF20 USES
/*       - TF/SNAPVX ON EMC SYMMETRIX VMAX V3 HW
/*       - TF/CLONE ON EMC SYMMETRIX VMAX V1/V3 HW
/* ARGUMENTS:
/* REQUIRED:
/* - C(<COMMAND>)
/* RE-ESTABLISH, SPLIT, SPLITINSTANT, LINK, UNLINK
GDDRKF20 performs the operations for these commands using different commands, based on the selected local replication method for managed EMC Symmetrix VMAX controllers.

- RE-ESTABLISH
  - TF/MIRROR: RE-ESTABLISH
    (NO-OP FOR ESTABLISHED DEVICES)
  - TF/CLONE: SNAP VOLUME (TARGET(SYMNV#(<SYMSTD#>)) ... READY(YES) )
    (SOURCE(SYMNV#(<SYMSTD#>)) ...)
  - TF/SNAPSHOT: TERMINATE (NAME(<SNAPSHOTNAME>) -)
    (SOURCE(SYMNV#(<SYMSTD#>)) -)
    AUTO UNLINK(YES) -
    TERMINATE ALL(NO) )
  - TF/CLONE & TF/SNAPSHOT: ACTIVATE
  - TF/SNAPSHOT: LINK (NAME(<SNAPSHOTNAME>) -)
    (SOURCE(SYMNV#(<SYMSTD#>)) -)
    TARGET(SYMNV#(<SYMSTD#>)) -
    ... READY(YES) )
  - (LINK NOT DONE BY DEFAULT, ONLY WITH OPTIONAL LINK(Y) ARGUMENT)

Note: GDDRKF20 supports creation of consistent PIT copies across V1/V2/V3 HW. This can be done in 2 ways:
- USE TF/MIRROR ACROSS ALL HW GENERATIONS
- USE TF/CLONE ON V1/V2 HW & TF/SNAPSHOT ON V3 HW
In the latter case, GDDRKF20 generates the required SNAP volume commands and create commands, followed by a single activate command in the same QCINPUT.

- A(<DCX>) (ACTION SITE)
  SITE WHERE THE MANAGED DEVICES ARE LOCATED.
- L(<DCX>) (LINKED SITE)
  SITE WHICH HAS AN SRDF DEVICE PAIRING WITH THE MANAGED DEVICES. GDDR MANAGEMENT OF LOCAL REPLICAATION IS APPLICABLE TO GDDR MANAGED SRDF DEVICES.

OPTIONAL:

- T(<TYPE>)
  BESIDES CONSISTENCY PROTECTED SRDF DEVICES, GDDR CAN ALSO MANAGE ACOPY-DISK DEVICES, AKA "EXTERNAL" DEVICES.
  GDDR MANAGEMENT OF THESE DEVICES IS A SMALL SUBSET OF THE FEATURES PROVIDED FOR THE "INTERNAL" SRDF DEVICES.
  EXTERNAL SRDF DEVICES CAN ALSO HAVE BCV'S ATTACHED TO THEM. THE T(.) ARGUMENT CAN BE SPECIFIED AS:
  - I: LIMIT MANAGEMENT TO BCV'S FOR INTERNAL SRDF DEVICES
  - E: LIMIT MANAGEMENT TO BCV'S FOR EXTERNAL SRDF DEVICES
  - A: MANAGE BOTH INTERNAL AND EXTERNAL SRDF DEVICES
  BY DEFAULT, ONLY INTERNAL BCVS ARE MANAGED.

- S(<SET>)
  GDDR SUPPORTS UP TO 2 SETS OF BCVS PER SRDF STD DEVICE. ONE SET IS NAMED "GOLD". IT IS USED TO SPLIT OFF A GOLD COPY TO PRODUCE A CONSISTENT RESTARTABLE IMAGE DURING RESYNCHRONIZATIONS. THE SECOND SET IS NAMED "TEST".
  IT CAN BE USED DURING TEST-IPL SCRIPTS, AVOIDING THE NEED TO SPLIT OFF THE GOLD COPY DURING SUCH SCRIPTS. USING THE S(...) ARGUMENT YOU CAN SPECIFY WHETHER YOU WANT TO MANAGE THE GOLD SET (DEFAULT) OR THE TEST SET OF BCVS.
BCV Reporting utility (GDDRTF00)

The GDDRTF00 batch utility checks BCV status. GDDRTF00 dynamically determines which technology applies to each VMAX system at the queried site based on the choices made in the “Option F: Define Configuration Features” panel in the GDDR parameter wizard as described below.

- If SnapVX Softlink support is required, use SnapVX on VMAX3 systems and use TimeFinder/Clone on earlier systems (ignore the legacy TimeFinder method).
- If SnapVX Softlink support is not required:
  - If TF/MIRROR is selected as the legacy TimeFinder method, use TimeFinder/Mirror for all VMAX system types.
  - If TF/CLONE is selected as legacy TimeFinder method, use SnapVX on VMAX3 systems (ignore SnapVX support option). Use TimeFinder/Clone on earlier systems.
Supported device types

GDDRTF00-supported local replication states can apply to source devices or target devices.

- For TimeFinder/Mirror:
  
  The ESTABLISH, SPLIT, RESTORE states apply to target devices only. TimeFinder/Mirror target devices must have the legacy BCV Flag.

- For TimeFinder/Clone:
  
  The ESTABLISH, SPLIT, RESTORE states apply to source and target devices. The ESTABLISH state is rarely expected in GDDR configurations, because all GDDR SNAP volume commands receive an ACTIVATE in the same step. Therefore the most common TimeFinder/Clone state is SPLIT (the snap was activated). RESTORE means a SNAP was performed where a GDDR-defined BCV is the source and a GDDR-defined SRDF device is the target.

- For TimeFinder/SnapVX:
  
  The CREATE, ACTIVATE, RESTORE states apply to source devices. CREATE is rarely expected in GDDR configurations since all creates receive an ACTIVATE in the same step. RESTORE indicates a source is limited to its own SNAPSHOT. LINK and UNLINK apply to target devices.

TimeFinder states

GDDRTF00 allows you to specify a list of acceptable TimeFinder states. Because you can mix TimeFinder technologies at the same site (within the supported limits) multiple states may be required to perform a complete site query.

For example, when you are using SnapVX on a VMAX3 system and TimeFinder/Clone on an earlier VMAX system:

- STATE(SPLIT,ACTIVATE) queries TimeFinder/Clone source and target devices and SnapVX source devices.
- STATE(SPLIT,ACTIVATE,LINK) queries TimeFinder/Clone and SnapVX source and target devices.

For example, you are using both TimeFinder/Clone and SnapVX on a VMAX3 system:

- STATE(ACTIVATE,LINK) queries SnapVX source and target devices.
- STATE(SPLIT) queries TimeFinder/Mirror target devices. In this case only one state is required.

This example assumes you did not request SnapVX Softlink support on the Define Configuration Features panel and specified TimeFinder/Mirror as the legacy TF method.

Sample JCL

GDDR.GDDR500.SAMPLIB (GDDRTF00)

//GDDRTF00 JOB (EMC), 'GDDRHC', CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1)
//*
//* RUN THE GDDR BCV STATUS CHECKER TO BE SURE THAT ALL
//* BCV’S ARE IN ONE OF THE EXPECTED STATES.
Using GDDR Utilities

//
// *******************************************
// /* TO YOUR CUSTOMIZED GDDRPROC */
// GDDRPROC JCLLIB ORDER=(.................................)
// *******************************************
// TF00 EXEC GDDRPROC
// GDDR.SYSTSIN DD *
// EXECUTIL SEARCHDD=YES
// GDDRXXST GDDRTF00 ...........
// SCFSXXX DD DUMMY
// *******************************************

// USAGE NOTES:
//
// * GDDRTF00 B(<DCX>) Q(<STATELIST>) +
// * R(<DCY>) T(<TYPE>) S(<SET>) H(<HOST>) DLM(N) PCHECK(Y)
// *
// RECOMMENDATION IS TO RUN THIS UTILITY AT A SITE WHICH HAS
// * CHANNEL CONNECTIVITY TO THE SYMMETRIX UNITS BEINGqueried.
// *
// ARGUMENTS:
// *
// REQUIRED:
// *
// - B(<DCX>)
// * SITE WHERE THE BCV’S ARE LOCATED
// *
// - Q(<STATELIST>)
// * SPECIFIES WHICH STATES YOU CONSIDER ACCEPTABLE AT THIS TIME.
// * APPLICABLE STATES DEPEND ON THE CHOSEN EMC TF TECHNOLOGY.
// * GDDR SUPPORTS TF/MIRROR, TF/CLOCK AND TF/SNAPVX.
// * AT ANY GIVEN SITE, YOU CAN USE TF/MIRROR ON ALL VMAX SYSTEMS,
// * OR YOU CAN USE TF/CLOCK ON V1/V2 VMAX HW, WHILE USING
// * TF/SNAPVX ON V3 VMAX HW AT THE SAME SITE.
// * GDDRTF00 DYNAMICALLY DETERMINES WHICH TECHNOLOGY APPLIES TO
// * EACH VMAX SYSTEM AT THE QUERIED SITE, BASED ON CHOICES YOU
// * MADE ON THE DEFINE CONFIGURATION FEATURES PANEL IN THE GDDR
// * PARAMETER WIZARD, AS DESCRIBED BELOW:
// *
// - IF SNAPVX SOFTLINK SUPPORT IS REQUIRED:
// * - USE SNAPVX ON V3 HW
// * - USE TF/CLOCK ON V1/V2 VMAX HW (IGNORE LEGACY TF METHOD)
// *
// - IF SNAPVX SOFTLINK SUPPORT IS NOT REQUIRED:
// * - IF TF/MIRROR IS SELECTED AS LEGACY TF METHOD:
// * - USE TF/MIRROR ACROSS V1/V2/V3 HW
// * - IF TF/CLOCK IS SELECTED AS LEGACY TF METHOD:
// * - USE SNAPVX ON V3 HW (IGNORE SNAPVX SUPPORT OPTION)
// * - USE TF/CLOCK ON V1/V2 VMAX HW
// *
// GDDRTF00 SUPPORTED STATES:
// *
// LOCAL REPLICATION STATES CAN APPLY TO SOURCE DEVICES OR
// TO TARGET DEVICES. SECTION BELOW DESCRIBES SUPPORTED
// STATES BY TECHNOLOGY AND DEVICE TYPE.
// *
// FOR TF/MIRROR: ESTABLISH, SPLIT, RESTORE
// *
// TF/MIRROR STATES APPLY TO TARGET DEVICES ONLY.
// TF/MIRROR TARGET DEVICES MUST HAVE THE LEGACY BCV FLAG.
// *
// FOR TF/CLOCK: ESTABLISH, SPLIT, RESTORE
// *
// TF/CLOCK STATES APPLY TO SOURCE AND TARGET DEVICES.
// THE ESTABLISH STATE IS RARELY EXPECTED IN GDDR CONFIGS
// AS ALL GDDR SNAP VOLUME COMMANDS GET AN ACTIVATE IN THE
// SAME STEP. THEREFORE THE MOST COMMON TF/CLOCK STATE
// WILL BE: SPLIT. (SPLIT MEANS THE SNAP WAS ACTIVATED)
// RESTORE MEANS WE HAVE A SNAP WAS PERFORMED WHERE A
// GDDR DEFINED “BCV” IS THE SOURCE, AND A GDDR DEFINED
// SRDF DEVICE IS THE TARGET.
// *
// FOR TF/SNAPVX: CREATE, ACTIVATE, LINK, UNLINK, RESTORE
// *
// CREATE, ACTIVATE, RESTORE APPLY TO SOURCE DEVICES.
// CREATE IS RARELY EXPECTED IN GDDR CONFIGS SINCE
// ALL CREATES GET AN ACTIVATE IN THE SAME STEP.
// RESTORE INDICATES A SOURCE DEVICE IS LINKED TO
// ITS OWN SNAPSHOT.
// LINK, UNLINK APPLY TO TARGET DEVICES.
// *
// GDDRTF00 ALLOWS YOU TO SPECIFY A LIST OF ACCEPTABLE STATES,
// AND SINCE YOU CAN MIX TECHNOLOGIES AT THE SAME SITE (WITHIN
// THE LIMITS DESCRIBED ABOVE) MAY REQUIRE MULTIPLE STATES
/* IN ORDER TO PERFORM A COMPLETE SITE QUERY. */
/* E.G. WHEN USING TF/CLONE ON V1/V2 HW, AND SNAPVX ON V3: */
/* STATE(SPLIT,ACTIVATE) */
/* QUERIES TF/CLONE SOURCE AND TARGET DEVICES AND SNAPVX */
/* SOURCE DEVICES */
/* STATE(SPLIT,ACTIVATE,LINK) */
/* QUERIES TF/CLONE & TF/SNAPVX SOURCE AND TARGET DEVICES */
/* E.G. IN A PURE V3 HW CONFIG: */
/* STATE(activate, LINK) */
/* QUERIES TF/SNAPVX SOURCE AND TARGET DEVICES */
/* STATE(SPLIT) */
/* QUERIES TF/MIRROR TARGET DEVICES */
/* IN THIS CASE ONLY ONE STATE IS REQUIRED. */
/* THIS EXAMPLE ASSUMES YOU DID NOT REQUEST SNAPVX */
/* SOFTLINK SUPPORT ON THE DEFINE CONFIGURATION FEATURES */
/* PANEL, AND SPECIFIED TF/MIRROR AS LEGACY TF METHOD. */

/* OPTIONAL: */
/* */
/* - R(<DCY>) - MAY BE REQUIRED! */
/* GDDR-MANAGED BCV'S ARE ATTACHED TO GDDR-MANAGED SRDF DEVICES. */
/* IN SRDF/STAR CONFIGURATIONS, SRDF/A DEVICES PAIRED DC1-DC3 */
/* OR DC2-DC3 ARE POSSIBLY A SUBSET OF SRDF/S DEVICES PAIRED */
/* DC1-DC2. THE R(...) ARGUMENT THEN HELPS IDENTIFY WHICH BCV'S */
/* YOU WANT TO VERIFY. R(...) DEFAULTS TO THE SRDF/S PARTNER */
/* SITE, IF APPLICABLE. */
/* WHEN VERIFYING EXTERNAL BCV'S, R(...) IS ALWAYS REQUIRED IF */
/* THERE ARE MORE THAN 2 SITES IN THE CONFIG. */
/* */
/* - T(<TYPE>) */
/* BESIDES CONSISTENCY PROTECTED SRDF DEVICES, GDDR CAN ALSO */
/* MANAGE ADCOPY-DISK DEVICES, AKA "EXTERNAL" DEVICES. */
/* GDDR MANAGEMENT OF THESE DEVICES IS A SMALL SUBSET OF THE */
/* FEATURES PROVIDED FOR THE "INTERNAL" SRDF DEVICES. */
/* EXTERNAL SRDF DEVICES CAN ALSO HAVE BCV'S ATTACHED TO THEM. */
/* THE T(...) ARGUMENT CAN BE SPECIFIED AS: */
/* - I: LIMIT VERIFICATION TO BCV'S FOR INTERNAL SRDF DEVICES */
/* - E: LIMIT VERIFICATION TO BCV'S FOR EXTERNAL SRDF DEVICES */
/* - A: VERIFY FOR BOTH INTERNAL AND EXTERNAL SRDF DEVICES */
/* BY DEFAULT, ONLY INTERNAL BCVS ARE VERIFIED. */
/* */
/* - S(<SET>) */
/* GDDR SUPPORTS UP TO 2 SETS OF BCVS PER SRDF STD DEVICE. */
/* ONE SET IS NAMED "GOLD". IT IS USED TO SPLIT OFF A GOLD COPY */
/* TO PRESERVE A CONSISTENT RESTARTABLE IMAGE OF THE DATA */
/* DURING RESYNCHRONIZATIONS. THE SECOND SET IS NAMED "TEST". */
/* IT CAN BE USED DURING TEST-IPL SCRIPTS, AVOIDING THE NEED TO */
/* SPLIT OFF THE GOLD COPY DURING SUCH SCRIPTS. */
/* USING THE S(...) ARGUMENT YOU CAN SPECIFY WHETHER YOU WANT */
/* TO VERIFY THE GOLD SET (DEFAULT) OR THE TEST SET OF BCVS. */
/* */
/* - H(<HOST>) */
/* CAN BE SPECIFIED AS CKD, FBA OR ALL. SPECIFIES WHICH HOST */
/* TYPE OF BCV DEVICES YOU'RE INTERESTED IN. DEFAULTS TO ALL */
/* */
/* - D(<DLM>) */
/* CAN BE SPECIFIED AS YES, NO, OR ONLY. SPECIFIES WHETHER OR */
/* NOT DLM BACKEND DEVICES ARE OF INTEREST. DEFAULTS TO YES. */
/* */
/* - PCHECK(Y/N) */
/* CAN BE SPECIFIED AS YES OR NO. DEFAULT IS YES. */
/* SPECIFIES YOU WANT GDDRTF00 TO PERFORM A "PROTECTION" */
/* CHECK, I.E. VERIFY THAT ALL INCLUDED SOURCE DEVICES ARE */
/* PROTECTED BY A LOCAL REPLICA TARGET DEVICE OR SNAPSHOT, */
/* AND THAT THE DEVICE FOUND PAIRED IS ALSO DEFINED TO GDDR. */
/* */
/* EXAMPLES: */
/* GDDRXST GDDRTF00 B(DC1) Q(ESTABLISH) */
/* - VERIFIES WHETHER THE GOLD SET OF INTERNAL BCV DEVICES AT DC1, */
/* ATTACHED TO SRDF-DEVICES PAIRED WITH DC2, ARE CURRENTLY */
/* ESTABLISHED. IN A 2-SITE SRDF/A CONFIG, THE SELECTED SRDF */
/* PAIRING IS TO DC3, AS THAT IS THE ONLY AVAILABLE. */
/* IN SRDF/STAR (OR SQAR) CONFIGURATIONS, THE SELECTED SRDF */
/* PAIRING IS TO DC2, AS THAT IS THE LARGEST SET. */
/* */
/* GDDRXST GDDRTF00 B(DC3) Q(ESTABLISH) R(DC2) T(E) S(EST) */
/* - VERIFIES WHETHER THE TEST SET OF EXTERNAL BCV DEVICES AT DC3, */
/* ATTACHED TO SRDF-DEVICES PAIRED WITH DC2, ARE CURRENTLY */
/* ESTABLISHED. */
/* */
******************************************************************************
GDDR IPL Parameter Swap utility (GDDRISWP)

The GDDRISWP stand-alone utility performs swaps of normal and alternate IPL-addresses and load-parameters for managed systems by site or by system while propagating the changes to all C-Systems.

The utility executes on the Master C-System as a batch job to swap the normal and alternate IPL-address and load-parameter values for the following pairs of global variables:

<table>
<thead>
<tr>
<th>Normal</th>
<th>swapped with</th>
<th>Alternate</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARMS.IPL</td>
<td></td>
<td>PARMS.IPLALT</td>
</tr>
<tr>
<td>PARMS.IPLBCVS</td>
<td></td>
<td>PARMS.IPLBCVA</td>
</tr>
<tr>
<td>PARMS.IPLBCTS</td>
<td></td>
<td>PARMS.IPLBCTA</td>
</tr>
</tbody>
</table>

**Note:** The IPL-address and load-parameters for C-Systems are not processed by this utility. Only the global variable values for managed systems are swapped.

The following required parameters specify the system and/or site to which the updates are applied:

- **SYSTEM(name|*)**
  - System name or <*> for all systems
- **SITE(DCn|*)**
  - Site name or <*> for all sites

To run the utility

1. Edit the ds-prefix.SAMPLIB(GDDRISWP) JCL, specifying the job name, STEPLIB dataset name, and SYSTEM and SITE parameter values.

```plaintext
//JOBCARD JOB (EMC), 'GDDR', CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1)
/*
 /* THIS JOB SWAPS THE MANAGED SYSTEM IPL LOAD ADDRESS AND LOAD
 /* PARAMETER VALUES FOR THE GDDR IPL AND ALTERNATE IPL GLOBAL
 /* VARIABLES.
 /* THE UPDATED GLOBAL VARIABLES ARE BROADCAST TO ALL OTHER C-SYSTEMS.
 /*
 PARAMETERS:
 /* SYSTEM(<NAME>|*) REQUIRED: SYSTEM NAME OR <*> FOR ALL SYSTEMS
 /* SITE(DCn|*) REQUIRED: SITE NAME OR <*> FOR ALL SITES
 /*
 PROCESSING:
 /* THIS JOB MUST RUN ON THE MASTER C-SYSTEM.
 /*
 /* THIS JOB SWAPS THE LOAD ADDRESS AND LOAD PARAMETER VALUES
 /* OF THE FOLLOWING GDDR GLOBAL VARIABLES:
 /*
 /* PARMS.IPL <-> PARMS.IPLALT
 /* PARMS.IPLBCVS <-> PARMS.IPLBCVA
 /* PARMS.IPLBCTS <-> PARMS.IPLBCTA
 /*
 /* RETURN CODES:
 /* 0 = PARAMETER UPDATES COMPLETE AND BROADCAST SUCCESSFULLY
 /* 4 = NO CORRESPONDING GLOBAL VARIABLE FOR SWAP OF VALUES
 /* 8 = UPDATE OF GLOBAL VARIABLE WITH NEW VALUE FAILED, OR
 /* PARAMETER UPDATE COMPLETED BUT BROADCAST FAILED
 /* 12 = MISSING/INVALID PARAMETER, NOT THE MASTER SYSTEM, OR
 /* NO IPL-RELATED GLOBAL VARIABLES WERE FOUND FOR THE
 /* SYSTEM/SITE SPECIFIED.
```
2. Submit the job.

3. Check the job return code and messages.

The utility sets the following return codes:

- 0  Parameter updates complete and broadcast successfully to the other C-Systems.
- 4  No corresponding global variable found (for example, PARMS.IPL exists but PARMS.IPLALT does not exist or has a null value).
- 8  The update of the global value failed on the Master C-System.
  -or-
  The update of the global variable was successful, but the broadcast to one or more other C-Systems failed.
- 12 The job was not executed on the Master C-System.
  -or-
  The SYSTEM or SITE parameter value is missing or has an invalid value.
  -or-
  No IPL-related global variables were found for the SYSTEM/SITE specified.

Messages appear in the job’s SYSTSPRT DD and are issued in the following sections:

- The SYSTEM and SITE parameters for this execution of the utility
- The values of the GDDR global variables at the time of job initiation
- The C-System IPL-related global variables that are excluded from processing
- The results of swapping the managed system load-address and load-parameter values
- The results of broadcasting the updated global variables to the other C-Systems

Sample SYSTSPRT output

Swapping Managed System primary and alternate IPL Parameters for 
System = *, Site = *

Initial Global Variable values:

GLOBAL.GDDR.PARMS.IPL.BTMUL144.DC2 : 94F1,0670MGM1
GLOBAL.GDDR.PARMS.IPL.BTMUL150.DC1 : 94F1,0670MGM1
GLOBAL.GDDR.PARMS.IPL.BTMUL48.DC3 : 94FF,0F13KGM1
GLOBAL.GDDR.PARMS.IPL.BTMUL51.DC1 : 01111,1111STM1
GLOBAL.GDDR.PARMS.IPLALT.BTMUL51.DC1 : 1234,1111STM1
GLOBAL.GDDR.PARMS.IPLBCVS.BTMUL51.DC1 : 01111,1111BGM1
GLOBAL.GDDR.PARMS.IPLBCVA.BTMUL51.DC1 : 03333,333333M3
GLOBAL.GDDR.PARMS.IPLBCTS.BTMUL51.DC1 : 04444,444444M4
GLOBAL.GDDR.PARMS.IPLBCTS.BTMUL51.DC1 : 04444,444444M4

C-System Global Variables excluded from processing:

GLOBAL.GDDR.PARMS.IPL.BTMUL144.DC2 : 94F1,0670MGM1
GLOBAL.GDDR.PARMS.IPL.BTMUL150.DC1 : 94F1,0670MGM1
GLOBAL.GDDR.PARMS.IPL.BTMUL48.DC3 : 94FF,0F13KGM1

Global Variable update results:

GLOBAL.GDDR.PARMS.IPLALT.BTMUL51.DC1 set to 01111,1111STM1
Using GDDR Utilities

GLOBAL.GDDR.PARMS.IPL.BTMUL51.DC1 set to 1234,1111STM1
GLOBAL.GDDR.PARMS.IPLBCVA.BTMUL51.DC1 set to 01111,1111BGM1
GLOBAL.GDDR.PARMS.IPLBCVS.BTMUL51.DC1 set to 03333,333333M3

Variable not found: GLOBAL.GDDR.PARMS.IPLBCTA.BTMUL51.DC1
No changes made to: GLOBAL.GDDR.PARMS.IPLBCTS.BTMUL51.DC1
Return code set to: 4

Broadcasting updated Global Variables to DC1

GDDR739I GDDR Broadcasting PARMS.IPLALT.BTMUL51.DC1 = 01111,1111STM1
GDDR739I GDDR -> Set PARMS.IPLALT.BTMUL51.DC1 = 01111,1111STM1 at DC1
Send of GLOBAL.GDDR.PARMS.IPLALT.BTMUL51.DC1 Completed
GDDR739I GDDR Broadcasting PARMS.IPL.BTMUL51.DC1 = 1234,1111STM1
GDDR739I GDDR -> Set PARMS.IPL.BTMUL51.DC1 = 1234,1111STM1 at DC1
Send of GLOBAL.GDDR.PARMS.IPL.BTMUL51.DC1 Completed
GDDR739I GDDR Broadcasting PARMS.IPLBCVA.BTMUL51.DC1 = 01111,1111BGM1
GDDR739I GDDR -> Set PARMS.IPLBCVA.BTMUL51.DC1 = 01111,1111BGM1 at DC1
Send of GLOBAL.GDDR.PARMS.IPLBCVA.BTMUL51.DC1 Completed
GDDR739I GDDR Broadcasting PARMS.IPLBCVS.BTMUL51.DC1 = 03333,333333M3
GDDR739I GDDR -> Set PARMS.IPLBCVS.BTMUL51.DC1 = 03333,333333M3 at DC1
Send of GLOBAL.GDDR.PARMS.IPLBCVS.BTMUL51.DC1 Completed

Broadcasting updated Global Variables to DC3

GDDR739I GDDR Broadcasting PARMS.IPLALT.BTMUL51.DC1 = 01111,1111STM1
GDDR739I GDDR -> Set PARMS.IPLALT.BTMUL51.DC1 = 01111,1111STM1 at DC1
Send of GLOBAL.GDDR.PARMS.IPLALT.BTMUL51.DC1 Completed
GDDR739I GDDR Broadcasting PARMS.IPL.BTMUL51.DC1 = 1234,1111STM1
GDDR739I GDDR -> Set PARMS.IPL.BTMUL51.DC1 = 1234,1111STM1 at DC1
Send of GLOBAL.GDDR.PARMS.IPL.BTMUL51.DC1 Completed
GDDR739I GDDR Broadcasting PARMS.IPLBCVA.BTMUL51.DC1 = 01111,1111BGM1
GDDR739I GDDR -> Set PARMS.IPLBCVA.BTMUL51.DC1 = 01111,1111BGM1 at DC1
Send of GLOBAL.GDDR.PARMS.IPLBCVA.BTMUL51.DC1 Completed
GDDR739I GDDR Broadcasting PARMS.IPLBCVS.BTMUL51.DC1 = 03333,333333M3
GDDR739I GDDR -> Set PARMS.IPLBCVS.BTMUL51.DC1 = 03333,333333M3 at DC1
Send of GLOBAL.GDDR.PARMS.IPLBCVS.BTMUL51.DC1 Completed

Maximum return code: 4
READY
END

GDDR HMC Actions utility (GDDRHMCA)

GDDRHMCA enables GDDR to perform supported HMC actions. Use GDDRHMCA to prepare for DR testing by performing LPAR actions before executing a test script.

The utility sets the following return codes:

0 Successful operation.

12 ARGUMENT ERROR. An invalid SYSTSIN statement was provided. See message GDDR998E for a description of the error.

Sample JCL

The sample GDDRHMCA utility JCL is in hlq.GDDRvrm.SAMPLIB (GDDRHMCA). The GDDRHMCA job must be run on a C-System.

GDDR.GDDR500.SAMPLIB(GDDRHMCA)

****************************************************************************** Top of Data ********************************************************************************
//GDDRHMCA JOB (EMC), 'GDDRHMCA', CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1)
/*
  /* ISSUE HMC COMMANDS
/* ********************************************************************************

316 EMC GDDR Version 5.0 Product Guide
Using GDDR Utilities

Parameters

**ACTION(HMC_ACTION_COMMAND)**

Specifies the HMC action to perform. Supported actions include:

- **SINGLE_LOAD**
- **SINGLE_LOAD_RECOVERY**
- **SINGLE_LOAD_BCVS**
Using GDDR Utilities

- SINGLE_LPAR_ACTIVATE
- SINGLE_LPAR_DEACTIVATE
- SINGLE_LPAR_START
- SINGLE_LPAR_STOP
- SINGLE_RESET_CLEAR
- CBU_ACTIVATE
- CBU_UNDO

SYSTEM(ssssssss)
Specifies the target system name.

SITE(DCx)
Specifies the GDDR SITE ID of the target CPC.

CPC(CPC_NAME)
Specifies the name of the target CPC. This must be specified in netid.nau format.

LPAR(LPAR_NAME)
Specifies the name of the target LPAR.

LOADA(LCUU)
Specifies the load address (CUU) of the SYSRES volume.

LOADP(LOADPARM)
Specifies the load parameters.

REALCPU(REAL|TEST)
When ‘ACTION(CBU_XXXXX)’ is specified:
- REALCBU(REAL) – Executes the action
- REALCBU(TEST) - Validates but does not execute the action

GDDR BCPii Connectivity test utility (GDDRBCPI)

The GDDRBCPI utility generates a list of CPC and images that can be connected to from the system on which this job is run. Run this job on all GDDR C-Systems to establish that BCPii connectivity is set properly.

Sample JCL

The sample GDDRBCPI utility JCL is in hlq.GDDRvrm.SAMPLIB (GDDRBCPI).

GDDR.GDDR500.SAMPLIB(GDDRBCPI)

************************************************************** Top of Data **************************************************************
//jobname JOB (EMC), 'GDDRBCPI', CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1),
  NOTIFY=&SYSUID
// *-----------------------------------------------------------------------------*
//* BCPii Connectivity Test for GDDR C systems.                            *
// *-----------------------------------------------------------------------------*
//** Generate list of CPC and Images that can be connected to from *
//** the system on which this job is run. This job should be run *
//** on all GDDR C systems to establish BCPii connectivity is set *
// *-----------------------------------------------------------------------------*
//** Update &GDDRPFX to be the GDDR Dataset prefix for your site **
//** This is the same prefix established in GDDCRACF setup job **
// *-----------------------------------------------------------------------------*
//**BCPI EXEC PGM=IKJEFT01,PARM='GDDRBCPI'

318   EMC GDDR Version 5.0 Product Guide
Using GDDR Utilities

---

**GDDR Load Profile Management utility (GDDRLPRF)**

This utility runs a GDDR LOAD profile with either a VALIDATE or the UPDATE parameter.

- **Execute with the VALIDATE parameter to examine the IPL parameters for the managed systems running at the current primary DASD site.** For each system that specifies a LOAD Profile stored on the HMC, the utility verifies the IPL Load Address and Load parameters stored within GDDR against those stored in the Load Profile and, as an additional check, verifies the current load address and load parameters for the managed system(s) in question.

- **Execute with the UPDATE parameter to update the HMC Load Activation Profile specified for a managed system running on the current primary DASD site with the GDDR IPL values stored for that system.**

**Requirements**

- GDDRMMAIN must be up on the C-system from which the utility is run.

- GDDRMMAIN must be up on any system from which info is requested.

**Sample JCL**

```assembler
GDDR.GDDR500.SAMLIB(GDDRLPRF)

******************************************************************************
//GDDRLPRF JOB (EMC),’GDDR’,CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1)
//*
//* Run the GDDR Load Profile Validation/Update Process
//*
******************************************************************************
/* POINT TO YOUR CUSTOMIZED GDDRPROC
//GDDRPROC JCLLIB ORDER=(............................................)
/*
//ACCD EXEC GDDRPROC
//GDDR.SYSTSIN DD *
EXECUTIL SEARCHDD(YES)
GDDRRXST GDDRLPRF VALIDATE
******************************************************************************
/*
/* USAGE NOTES: GDDRLPRF
/*
/* Requirements:
/*
/* GDDRMMAIN must be up on the C system you are running this on
/* and GDDRMMAIN must also be up on any system from which info is
/* requested.
/*
/* Optional arguments:
/*
/* VALIDATE: Examine the IPL Parameters for the managed systems
/* running at the current primary DASD site and, for each one specifying a LOAD Profile stored on the
/* HMC, verify the IPL Load Address and LoadParms stored within GDDR against that stored in the Load
/* Profile and, as an additional check, verify the current load address and load parameters for the
/* managed system(s) in question...
/*
/* UPDATE: Update the HMC Load Activation Profile specified
/*
```
GDDR ECGUTIL Driver utility (GDDRECG0)

Use the GDDRECG0 utility to change device status. Note that this utility only affects GDDR-managed devices and that it must be run at the site with channel access to the DASD at the selected site.

Sample JCL

GDDR.GDDR500.SAMPLIB(GDDRECG0)

```plaintext
****** Top of Data ******
//GDDRECG0 JOB (EMC),'GDDRSC',CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1)
/*
/* RUN THE GDDR ECGUTIL DRIVER MODULE
/*
/****** Top of Data ********
/****
/**** POINT TO YOUR CUSTOMIZED GDDRPROC 
/****
@@ECG0 EXEC GDDRPROC
@@GDDR.SYSTSIN DD *
EXECUTIL SEARCHDD(YES)
GDDRRXST GDDRECG0 .......... 
//Scf$xxxx DD DUMMY
/****** Bottom of Data ******* *

// USAGE NOTES:
/*
/* GDDRECG0 ONLY AFFECTS GDDR MANAGED DEVICES. IT MUST BE RUN *
/* AT A SITE WITH CHANNEL ACCESS TO THE DASD AT THE SELECTED SITE. *
/*
/* INVOCATION: GDDRXST GDDRECG0 <SITE> <FUNCTION> <DEVTYPE>
/*
/* REQUIRED:
/*
/* SITE: DC1, DC2 , DC3 OR DC4
/*
/* FUNCTION:
/*
/* UNR: MAKE DEVICES USRNRDY
/*
/* RDY: MAKE DEVICES USRRDY
/*
/* CLR: CLEAR CG+ECA FLAGS FROM DEVICES
/*
/* OPTIONAL:
/*
/* DEVTYPE: ALL OR FBA (DEFAULT: ALL)
/*
/* EXAMPLES:
/* GDDRXST GDDRECG0 DC1 CLR  --- CLEAR CG+ECA FLAGS AT DC1
/* GDDRXST GDDRECG0 DC3 UNR FBA  --- MAKE FBA-DEVS AT DC3 USRNRDY
/****** Bottom of Data ******* *

Parameters

UNR
Makes devices USRNRDY.

RDY
Makes devices USRRDY.

CLR
Clears the CG and ECA device flags.
GDDR IPL Assist Monitor utility (GDDRGIAM)

The GDDRGIAM utility monitors systems being IPL’d for a hardcoded list of messages and WTORs and ensures that certain error conditions or other prompts do not cause unnecessary delay in system restart processing. This goal is achieved either by performing hardcoded replies to selected WTORs, or by echoing selected messages to create visibility.

⚠️ CAUTION ⚠️

Use of GDDRGIAM is not without risk. EMC recommends usage of this utility in RDR-type situations, only after a thorough study of the functionality. Before enabling GDDRGIAM, review the replies the utility gives to sysplex-related WTORs.

GDDRGIAM intercepts a hardcoded list of WTORs and messages occurring on GDDR-managed LPARs.

Each intercepted WTOR or message is echoed locally on the system where GDDRGIAM runs, using message GDDHMC1I. The GDDR message rule for GDDHMC1I is disabled by default. To use GDDRGIAM, set GDDHMC1I message interception to be enabled automatically when GDDRMAIN starts, using the MSG GDDRPARM parameter.

When the GDDHMC1I message rule is enabled, the message is intercepted by standard GDDR message interception and propagated to the Master C-System. The goal of this messaging is to provide visibility on issues during IPL of a managed system, on the C-system where the script runs. GDDR is allowed to interact with HMC and to perform LOAD CLEAR actions in scripts as specified in the default or script submission call overrides.

When GDDRGIAM usage is enabled, script GDDRPA27 will start GDDRGIAM, and script GDDRPA28 will stop it. Note that there could be significant time between the execution of scripts GDDRPA27 and GDDRPA28. You can also stop GDDRGIAM at any time, either by command or using hlq.GDDRvrm.SAMPLIB(GDDRGIAM).

- To determine if GDDRGIAM is currently active on the local system, issue:
  ```
  F GDDRMAIN,BCPII,STATUS
  ```

- To turn GDDRGIAM on or off for all LPARs on all CPCs, for all LPARs on a CPC, or for a specific LPAR, issue:
  ```
  F GDDRMAIN,GIAM,ON|OFF,<cpc>,<lpar>
  ```
  The action is taken on the system on which the command is issued and has no effect on GDDRGIAM running on other systems.

- To turn debug on or off for all LPARs on all CPCs, for all LPARs on a CPC, or for a specific LPAR, issue:
  ```
  F GDDRMAIN,BCPII,DBGON|DBGOFF,<cpc>,<lpar>
  ```
  The action is taken on the system on which the command is issued and has no effect on GDDRGIAM running on other systems.

Sample JCL

```jcl
GDDR.GDDR500.SAMPLIB(GDDRGIAM)

******************************** Top of Data ********************************
//GDDRGIAM JOB (EMC), 'GDDR', CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1)
```
Using GDDR Utilities

//***************************************************************
//*                                                          *
//* Stop the GDDR IPL Assist Monitor                          *
//*                                                          *
//****************************************************************
//* POINT TO YOUR CUSTOMIZED GDDRPROC                        *
//GDDRPROC JCLLIB ORDER=(............................................)
//***************************************************************
//GDDRGIAM EXEC GDDRPROC
//GDDR .SYSTIN DD *
EXECUTIL SEARCHDD(YES)
GDDRRXST GDDRGFS0 STOP
//***************************************************************

/* USAGE NOTES:                                              *
"Run this utility to STOP the GDDR IPL ASSIST Monitor (GIAM), *
formerly known as "the HMC Message Scanner", on all BCPii *
capable systems, as defined on GDDRPARM CPC parameters. *

The GDDR IPL Assist Monitor is normally started and stopped in *
GDDR scripts. This utility allows you to stop it in a case where *
a script, or sequence of scripts, was not run to completion. *

This utility can also be used to enable/disable GDDR BCPii code *
tracing.

Requirements:                                              *
- Must run on C-system.
- GDDRMAIN must be up and running on all BCPii capable systems.

The utility can run on any site in your configuration, and it *
only needs to be run once to stop GIAM everywhere.

Invocation: GDDRRXST GDDRGIAM

Arguments: [action]

Valid action arguments are STOP and NONE.
STOP is the default action.
NONE can be used to set BCPii debugging enabled/disabled

Set Output Message Levels By Program panel.

***************************************************************

Message interception

GDDRGIAM intercepts the following messages and uses message GDDHMC1I to echo them to the C-System where GDDRGIAM is running:

GDDRBMSG MSGID=IEA371I
Explanation: IEA371I data-set ON DEVICE nnnn SELECTED FOR IPL PARAMETERS

During system initialization, the system selected a dataset containing the IPL parameters. This is an informational message for visibility back to the C-System where the script runs.

GDDRBMSG MSGID=IEE311I
Explanation: IEE311I cm PARAMETER MISSING

While processing a command, the system detected that a required parameter is missing. This message is included for visibility back to the C-System where the script is running, as it might signal something is wrong with GDDRGIAM, or with the IPL process being monitored.

GDDRBMSG MSGID=IEE389I
Explanation: IEE389I MVS COMMAND PROCESSING AVAILABLE
The system issues this message during IPL processing when commands are expected to work successfully. This message is included for visibility back to the C-System where the script is running.

**WTOR interception**

GDDRGIAM intercepts the following WTORs and replies as indicated.

**GDDRBMSG MSGID=HASP420,ACTION=((REPLY,Y))**

Explanation: $HASP420 REPLY ’Y’ IF [memname IS | ALL MEMBERS ARE] DOWN (IPL REQUIRED), ’N’ IF NOT

JES2 has issued either message $HASP419 reporting the members considered dormant, or message $HASP405 if it cannot determine if any other member in the multi-access spool configuration is active. JES2 requires confirmation of the status of each of the displayed members.

Reply Y to inform JES2 that the listed JES2 member is dormant. After a reply Y to all $HASP420 WTORs, JES2 will perform a cold start or a warm start, depending on the initialization option you previously specified.

**GDDRBMSG MSGID=IGGN505A,ACTION=((REPLY,CANCEL))**

Explanation: IGGN505A SPECIFY UNIT FOR dsname ON volser OR CANCEL

During nucleus initialization program (NIP) processing, the system determined that the volume must be mounted to access dataset dsname. Select an available device of the type required and respond R xx,dev, where xx is the reply number and dev is the device number of the selected device. You may also respond by signaling EOB (pressing Enter on the console). This action indicates that the volume is not available and is not to be used for this IPL.

**GDDRBMSG MSGID=IXC247D,ACTION=((REPLY,U))**

Explanation: IXC247D REPLY U to ACCEPT USE OR D TO DENY USE OF THE COUPLE DATA SET FOR typename.

This system attempted to initialize a couple dataset for the specified type and determined that the dataset might be in use by another sysplex. Message IXC248E, which precedes this message, indicates the name of the dataset that is possibly in use by another sysplex.

Reply U to continue initialization of the couple dataset. Allowing initialization to continue on a couple dataset that is in use by another sysplex causes the other sysplex to lose access to the dataset, which might cause the systems in that sysplex to enter a wait state.

**GDDRBMSG MSGID=IXC269D,ACTION=((REPLY,U))**

Explanation: IXC269D REPLY U TO USE RESOLVED DATA SETS, C TO USE COUPLE DATA SETS SPECIFIED IN COUPLExx, OR R TO RESPECIFY COUPLExx

This system detected an inconsistency in the couple datasets specified in COUPLExx. This system has resolved the inconsistency and has found a consistent primary and alternate couple dataset. However, it does not appear that any of the systems using the datasets are active. The couple datasets specified by COUPLExx have been displayed via message IXC275I as well as those that this system has determined to be consistent.

Reply U to continue initialization with the resolved couple datasets that XCF has determined to be the better choice to use.
Using GDDR Utilities

GDDRBMSG MSGID=IXC289D,ACTION=((REPLY,U))
Explanation: IXC289D REPLY U TO USE THE DATA SETS LAST USED FOR typename OR C TO USE THE COUPLE DATA SETS SPECIFIED IN COUPLExx
XCF has detected an inconsistency in the couple datasets specified in COUPLExx. The couple datasets specified in COUPLExx are not the same couple datasets last used by the sysplex for this data type. The system lists, via message IXC288I, those datasets last used by the sysplex and also those specified in COUPLExx. This message reply indicates the correct set of couple datasets to be used for further processing.

Reply U to continue initialization with the primary and alternate sysplex datasets that were last used by the sysplex, which are not the same couple datasets specified in COUPLExx. This could have been caused by the removal of a primary or an alternate couple dataset, or as the result of the addition of a new alternate couple dataset after the sysplex was IPLed. A normal re-IPL should choose this option.

GDDRBMSG MSGID=IXC405D,ACTION=((REPLY,I))
Explanation: IXC405D REPLY I TO INITIALIZE THE SYSPLEX, J TO JOIN SYSPLEX sysplex-name, OR R TO REINITIALIZE XCF
This system is trying to initialize or join a sysplex, but XCF found one or more systems already active in sysplex sysplex-name. This message prompts the operator to indicate whether the systems displayed in message IXC404I are active in the sysplex and whether initialization should continue. See the explanation of IXC404I for additional information.

Reply I to request that sysplex initialization continue because none of the systems identified in message IXC404I are participating in an operating sysplex; that is, they are all residual systems. This system will perform cleanup of old sysplex data, initialize the couple dataset, and start a new sysplex. If any of the systems identified in message IXC404I are currently active in the sysplex, they will be placed into a disabled wait state.

GDDRBMSG MSGID=IXC501A,ACTION=((REPLY,Y))
Explanation: IXC501A REPLY Y TO USE COUPLING FACILITY NAMED cfname OR N TO NOT USE COUPLING FACILITY
This is the prompt associated with message IXC500I:

IXC500I CONFIRM REQUEST TO USE COUPLING FACILITY type.mfg.plant.sequence PARTITION: partition side CPCID: cpcid NAMED  cfname AUTHORITY DATA: plexname mm/dd/yyyy hh:mm:ss
Message IXC500I is issued when CFRM determines that another sysplex currently owns the named coupling facility.

Verify that the CFRM active policy correctly specifies the coupling facility that is to be used by this sysplex and ensure that the sysplex identified in message IXC500I is stopped from using the coupling facility before responding to this message.

If Y is specified, this system issues message IXC559I to ensure that the operator is made aware of the need to prevent another sysplex from using the coupling facility, and then message IXC560A to prompt for confirmation before allowing this system to gain ownership of the coupling facility. Messages IXC500I and IXC559I identify the coupling facility and the sysplex that currently owns it.
GDDR Invalid Track Monitor utility (GDDRMINO)

Run the GDDRMINO utility to query GDDR-managed SRDF or TimeFinder devices for invalid tracks. This is a query-only utility; it does not affect the state of the GDDR-managed configuration.

Sample JCL

This JCL provides information about GDDRMINO requirements, arguments, and use cases.

GDDR Utilities

Note: A reply of Y to this prompt and a reply of Y to message IXC560A may cause severe errors if the coupling facility is still being used by the sysplex identified in messages IXC500I and IXC559I.

If N is specified, the coupling facility will not be used by this system.

GDDRBMSG MSGID=IXC560A,ACTION=((REPLY,Y))

Explanation: IXC560A REPLY Y TO CONFIRM THAT COUPLING FACILITY NAME cfname SHOULD BE USED BY plexname1 OR N TO DENY THE USE.

This message appears in conjunction with message IXC559I when the reply to message IXC501A is Y. Message IXC559I is issued to warn the operator that severe errors may occur if the coupling facility is still being used by the sysplex that currently owns it. The operator is asked to ensure that the coupling facility is not being used by the currently owning sysplex and then asked to either confirm or deny the use of the coupling facility.

Verify that the CFRM active policy correctly identifies that the coupling facility should be used by this sysplex and ensure that the sysplex identified in messages IXC500I and IXC559I is stopped from using the coupling facility before responding to this message.

If Y is specified, this system will gain ownership of the coupling facility for the sysplex and coupling facility cleanup will occur. Messages IXC500I and IXC559I identify the coupling facility and the sysplex that currently owns it.

Note: A reply of Y may cause severe errors if the coupling facility is still being used by the sysplex identified in these messages.

If N is specified, the coupling facility will not be used by this system.
Using GDDR Utilities

/*
 * This is a query only utility, which does not affect the state
 * of the GDDR managed configuration.
 */

Requirements:
- GDDRSCF must be up and running, with the SCF$ connector used
  in this JCL.
- GDDRMAIN must be up and running, and connected with the same
  SCF instance.
- RDF.DEVICES parameters reflecting sitepairs as expected for
  the defined type of configuration.
- BCV parameters reflecting the sites in the configuration are
  expected if TimeFinder devices are of interest.
- Channel and ra-group connectivity to the Symmetrix units
  relevant to the queries.

Return code meanings are those documented for message GDDP474I.

Invocation: GDDRXST GDDRMIN0 <arguments>

Arguments:

Required: S() or B()

GDDRMIN0 needs to know which site to query.
In case SRDF devices are being queried, it also
needs to know which other site defines the SRDF
pairing, for correct device selection.
The same is true when TimeFinder devices associated
with External devices are being queried. (T(E))
GDDRMIN0 further needs to know whether Invalid
Tracks for remote mirrors or local mirrors are
of interest.

- S(<DCx>) or B(<DCx>) (only 1 allowed)
  Specify either the S() or the B() argument to
tell GDDRMIN0 which site to query, and whether
to query SRDF or TimeFinder devices.

- L(<DCx>) or R(<DCx>) (only 1 allowed)
  Required when S() is specified.
  L() is required when TimeFinder devices associated
  with External devices are being queried.
  Specifies the SRDF paired site.
  Use L() to request local invalid track queries
  Use R() to request remote invalid track queries
  When B() is used, only local invalid track
  queries are supported.

  Note: Using R() does NOT cause remote devices to be
  queried! Rather it means that the devices local
  to the site specified in S() will be queried in
  their mirror positions reflecting the partner
devices.

Optional:

- T(<type>) (I, E or A. Default: I)
  Indicates whether Internal (I), External (E)
or All (A) devices should be queried.

- SET(<set>) (GOLD or TEST. Default: GOLD)
  Used only with B().
  Indicates whether GOLD set or TEST set TimeFinder
devices should be queried.

- N(<number>): (any integer. Default: 1)
or
- W(<waitoption>) (number,objtype,stallmax)

  Use the N() or the W() arguments to indicate
  how long the Invalid Track monitor should run.
  You should only use one of these, not both.
  Use N() to indicate a fixed number of runs.
  Use W() to specify a "resynchronization goal".
  W() is ignored if N() is specified.
  Without N() or W(), N() is used with a default of 1.
using the RANGE level. 

detail than the RA-group level is wanted, EMC recommends this may produce an excessive number of messages. If more progress during any run will be called out. Any ra-group for which there is no resynchronization (Tracks/s) and ETA (time required to completion). produce a report at the RA-group level, showing: Site, As of the second run, the Invalid Track monitor will exit for one of 3 conditions: - resynchronization goal reached - stallmax exceeded - more than 99999 runs are needed for the resynchronization goal to be reached

- I(<interval>): (any integer. Default: 15)

  Specifies the number of minutes in between runs. Ignored if N() is specified with value of 1, or defaults to 1.

Recommendations:

- The utility can run on any site in your configuration, but is recommended to run at the site where the queried devices are located. As a typical GDDRM0 use case is during a massive resynchronization over SRDF links, running at the queried site avoids impact from the data traffic on the queries.

Use cases:

1. Do I have invalid tracks on my CG-protected devices at DC1, owed to DC2, Yes or No?

   GDDRXXST GDDRM0 S(DC1) R(DC2) T(I) N(1)

   The Invalid Track Monitor will do 1 query of all GDDR managed Internal devices at DC1.
   RC=0 if there are no invalid tracks owed to DC2.
   Any devices with such invalid tracks will be reported.

2. SRDF/A from DC1 to DC3 has been out for several days. I have invalid tracks all over the place at DC1. Resynchronization was just started. How long is it going to take, and please let me know when it's done or when it stops.

   GDDRXXST GDDRM0 S(DC1) R(DC3) W(0,RAG,5)

   The Invalid Track Monitor will run every 15 minutes, querying the DC1 devices for invalid tracks owed to DC3, until one of the following conditions is met:
   - ALL GDDR managed ra-groups DC1-DC3 show 0 Invalid tracks
   - During 6 consecutive runs, the number of invalid tracks owed from DC1 to DC3 does not decrease
   - More than 99999 times 15 minutes elapse

   During each run, any RA-group with invalid tracks will be reported.

   As of the second run, the Invalid Track monitor will produce a report at the RA-group level, showing: Site, Symmetrix, RA-group, Invalid tracks, Resynchronization speed (Tracks/s) and ETA (time required to completion).

   Any ra-group for which there is no resynchronization progress during any run will be called out.

   The exact same thing can be done at the Device level, but this may produce an excessive number of messages. If more detail than the RA-group level is wanted, BMC recommends using the RANGE level.
3. During my GDDRPA07 script, GDDR has just done the ResumePair from DC2 to DC3 with KEEPR2, and resumed SRDF. The operator then accidentally canceled the script. How do I monitor resynchronization progress?

```
GDDRTRCP GDDRMIN0 S(DC3) L(DC2) W(0,RAG,3) I(5)
```

The Invalid Track Monitor will run every 5 minutes, querying the DC3 devices paired with DC2 for local invalid tracks, until one of the following conditions is met:

- ALL GDDR managed ra-groups DC3-DC2 show 0 Invalid tracks
- During 4 consecutive runs, the number of local invalid tracks at DC3 does not decrease
- More than 99999 times 5 minutes elapse

Same comments as for use case 2 apply.

4. I am getting TF error messages. Could I have local invalid tracks on my BCVs at DC4?

```
GDDRTRCP GDDRMIN0 B(DC4) N(1)
```

The Invalid Track Monitor will query the BCV devices at DC4 for local invalid tracks, one time, and exit with RC=0 if there are none. Any GDDR managed BCV device at DC4 with invalid tracks will be reported.

GDDRMAIN Trace Print utility (GDDRTRCP)

Use the GDDRTRCP utility to troubleshoot suspected GDDR software issues with assistance from the GDDR Solution Support team.

Sample JCL

```
GDDR.GDDR500.SAMPLIB(GDDRTRCP)
```

********************************************************************
****************************** Bottom of Data **************************
GDDRMAIN Trace Print utility (GDDRTRCP)

Use the GDDRTRCP utility to troubleshoot suspected GDDR software issues with assistance from the GDDR Solution Support team.

Sample JCL

```
GDDR.GDDR500.SAMPLIB(GDDRTRCP)
```

********************************************************************
****************************** Bottom of Data **************************
CHAPTER 7
Performing Script Operations

This chapter describes the scripts that you can run from the GDDR Operator interface.

- Running scripts ..................................................................................................... 330
- Planned script operations ..................................................................................... 337
- Test operations ..................................................................................................... 339
- Unplanned script operations ................................................................................. 339
- Resumption operations ........................................................................................ 340
- Special operations ................................................................................................ 340
Performing Script Operations

Running scripts

GDDR allows you to run the following types of scripts:

- Planned actions
- Test actions
- Unplanned actions
- Resumption actions
- Special actions

**IMPORTANT**
Do not run GDDR scripts during bin file or configuration changes.

Take the following steps to use any of these scripts:

1. Navigate to the Select Script to Run panel as shown in “Option S: Scripts — Run GDDR Scripts” on page 208.

2. Select the script you want by typing $ next to the script and pressing **Enter**.

   **Note:** If another script is currently in progress, a message similar to the following appears on the Select Script to Run panel:

   Script GDDRPA29 is in progress. Only the script in progress may be submitted at this time.

3. If no script is in progress, the Specify Parameters for Initial Script Run panel appears:

   ![Figure 103 Specify Parameters for Initial Script Run panel]

   4. You can then confirm or clear any of the call overrides that apply to this script by entering **Y** in “Specify call override changes for this script?”
The Specify Call Overrides panel appears:

<table>
<thead>
<tr>
<th>Call?</th>
<th>Program</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>GDDDRDXRV</td>
<td>Manage Distributed Workload</td>
</tr>
<tr>
<td>Y</td>
<td>GDDDRDF0</td>
<td>Call DYNAPI Interface</td>
</tr>
<tr>
<td>Y</td>
<td>GDDDRDF0</td>
<td>DYNAPI - SRDF/S Commands</td>
</tr>
<tr>
<td>Y</td>
<td>GDDDRDF0</td>
<td>DYNAPI - DeletePair and Half-DeletePair Commands</td>
</tr>
<tr>
<td>Y</td>
<td>GDDDRDF0</td>
<td>DYNAPI - CreatePair and ResumePair Commands</td>
</tr>
<tr>
<td>Y</td>
<td>GDDDRDF0</td>
<td>DYNAPI - SRDF/A Commands</td>
</tr>
<tr>
<td>Y</td>
<td>GDDDRDF0</td>
<td>DYNAPI - Swap and Half-Swap Commands</td>
</tr>
<tr>
<td>N</td>
<td>GDDDRGF08</td>
<td>Use ConGroup Shutdown/Startup instead of Refresh</td>
</tr>
<tr>
<td>N</td>
<td>GDDRKF0C</td>
<td>Trigger Production System Shutdown</td>
</tr>
<tr>
<td>N</td>
<td>GDDRKF0I</td>
<td>Trigger Production System Startup</td>
</tr>
<tr>
<td>N</td>
<td>GDDRKF20</td>
<td>(Scan Mode) - Manage BCVs</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF0H</td>
<td>Transfer Master Function Ownership</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>(DC1) - Manage BCVs</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>(DC2) - Manage BCVs</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>(DC3) - Manage BCVs</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>(DC4) - Manage BCVs</td>
</tr>
<tr>
<td>N</td>
<td>GDDRKF20</td>
<td>Manage External BCV devices</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>Manage TEST BCV devices at DC1</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>Manage TEST BCV devices at DC2</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>Manage TEST BCV devices at DC3</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRKF20</td>
<td>Manage TEST BCV devices at DC4</td>
</tr>
<tr>
<td>N</td>
<td>GDDDRDF0</td>
<td>Manage External Devices</td>
</tr>
<tr>
<td>N</td>
<td>GDDRGF0S</td>
<td>Allow Primary Site HMC Actions in Test Scripts</td>
</tr>
<tr>
<td>N</td>
<td>GDDRGF0S</td>
<td>Allow Secondary Site HMC Actions in Test Scripts</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRGF0S</td>
<td>Allow LOAD CLEAR when Activating LPARs in Scripts</td>
</tr>
<tr>
<td>Y</td>
<td>GDDRGF0S</td>
<td>Confirm GDDR Script HMC actions by WTOR</td>
</tr>
<tr>
<td>N</td>
<td>GDDRGF0S</td>
<td>Prompt for ALT IPL Parameters</td>
</tr>
<tr>
<td>N</td>
<td>GDDRGF0S</td>
<td>Use DR-Test IPL parameters</td>
</tr>
<tr>
<td>N</td>
<td>GDDRGF0S</td>
<td>Use BACKGROUNDCOPY for TF/Clone</td>
</tr>
</tbody>
</table>

**Figure 104**  Specify Call Overrides panel

**Note:** The functions listed in this panel vary based on the current configuration and script selected.

The Specify Call Overrides panel provides help information (accessed by pressing PF1) that describes the function of each program. Use PF3 to return to the Specify Call Overrides panel.

**Note:** All call overrides do not apply to all scripts. For any script you run, you normally see a subset of the possible call overrides.

You can overtype the default "Y" or "N" call override values for this script. (Table 21 on page 333 lists the possible call overrides and their meanings.)

**Note:** You can specify call overrides only once for each execution of a script. If a script terminates before completion, you can rerun the script (as discussed in "Recovering from script errors" on page 335), but any changes you make to the call overrides will be ignored. However, when you execute the same script again after it successfully completes, you can specify call overrides for it.
Performing Script Operations

5. When you are finished, press F3 to return to the Specify Parameters for Initial Script Run panel. Then press Enter to continue.

   **Result:** A pop-up similar to the following appears and you are prompted *twice* to confirm job submission.

   ![Figure 105 Confirm Job Submission panel]

   At each prompt, reply Yes to confirm or No to terminate job submission.

   **Result:** After GDDR receives both confirmations, it submits the job to start the selected planned script. Upon starting, you are prompted (through WTOR) to confirm that you want to run the script.

6. Reply Yes to allow the script to continue or No to terminate the script.
## Call overrides

Table 21 lists the call overrides that you may see when running a script. All call overrides do not apply to all scripts. For any script you run, you normally see a subset of the possible call overrides.

### Table 21  GDDR call overrides (1 of 3)

<table>
<thead>
<tr>
<th>Program</th>
<th>Function</th>
</tr>
</thead>
</table>
| GDDRRDF0    | Call_DYNAPI_Interface  
Call Override byte 2  
Specify Y to:  
- Perform SRDF device action commands using the GDDR command queue, enabling parallel command execution.  
- Perform CreatePair, ResumePair, DeletePair, Half-DeletePair, Swap, Half-Swap, R22SWICH, MovePair, Half-MovePair, and composite (CAS-xx) commands in parallel at the VMAX system level.  
Call Overrides 3-7 further control GDDR command queue usage by command type. Specify N to suppress all GDDR command queue usage. Commands will be issued one at a time. Call Overrides 3-7 are ignored. |
| GDDRRDF0    | DYNAPI - DeletePair and Half-DeletePair commands  
Call override byte 4  
Specify Y to perform DeletePair and Half-DeletePair commands using the GDDR command queue. They will be executed in parallel at the VMAX unit level.  
Specify N to suppress GDDR command queue usage for all DeletePair and Half-DeletePair commands. They will be executed one command at a time. |
| GDDRRDF0    | DYNAPI - CreatePair and ResumePair commands  
Call Override byte 5  
Specify Y to perform CreatePair and ResumePair commands using the GDDR Command queue. They will be executed in parallel at the VMAX unit level.  
Specify N to suppress GDDR command queue usage for all CreatePair and ResumePair commands. They will be executed one command at a time. |
| GDDRRDF0    | DYNAPI - SRDF/A commands  
Call override byte 6  
Specify Y to perform SRDF device actions against SRDF/A devices using the GDDR command queue.  
Specify N to suppress usage of the GDDR command queue for SRDF/A devices. In both cases, call overrides 4 and 7 further control GDDR command queue usage. |
| GDDRKF0C    | TRIGGER_PRODUCTION_SYSTEM_SHUTDOWN  
Calls user exit GDDRXDRV to synchronously stop all production system application workloads. |
| GDDRKF0I    | TRIGGER_PRODUCTION_SYSTEM_STARTUP  
Calls user exit GDDRXDRV to asynchronously start all production system application workloads. |
| GDDRKF20    | (Scan Mode) - Manage_BCVs  
Perform actions in simulation mode. For TimeFinder/Mirror actions, the commands are echoed to the script joblog. For TimeFinder/Clone and SnapVX actions, EMCSNAP is invoked with the specification of GLOBAL TYPRUN(SCAN). |
| GDDRTNG1    | GDDRTNG1  
Generates an OPS/MVS alert and call GDDRXDRV exit with message. |
### Table 21 GDDR call overrides (2 of 3)

<table>
<thead>
<tr>
<th>Program</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>GDDRKF0H</td>
<td>TRANSFER_MASTER_FUNCTION_OWNERSHIP</td>
</tr>
<tr>
<td></td>
<td>Add a step to Site Swap scripts to transfer the Master-C role to the appropriate site.</td>
</tr>
<tr>
<td>GDDRKF20</td>
<td>(DC1) - Manage_BCVs</td>
</tr>
<tr>
<td></td>
<td>Performs TimeFinder/Mirror, TimeFinder/Clone, and/or SnapVX actions at DC1.</td>
</tr>
<tr>
<td></td>
<td>Set this to N if you do not have BCVs configured at DC1.</td>
</tr>
<tr>
<td>GDDRKF20</td>
<td>(DC3) - Manage_BCVs</td>
</tr>
<tr>
<td></td>
<td>Performs TimeFinder/Mirror, TimeFinder/Clone, and/or SnapVX actions at DC3.</td>
</tr>
<tr>
<td></td>
<td>Set this to N if you do not have BCVs configured at DC3.</td>
</tr>
<tr>
<td>GDDRKF20</td>
<td>Manage external BCVs</td>
</tr>
<tr>
<td>GDDRKF20</td>
<td>Manage Test BCVs at DC1</td>
</tr>
<tr>
<td>GDDRKF20</td>
<td>Manage Test BCVs at DC3</td>
</tr>
<tr>
<td>GDDRD6F0</td>
<td>Manage external devices</td>
</tr>
<tr>
<td></td>
<td>Support for external devices enables automated restart at a site which is an SRDF/A target by supporting page volumes and the TEMP pool volumes not replicated with SRDF/A due to bandwidth constraints.</td>
</tr>
<tr>
<td>GDDR5FHS</td>
<td>Allow Primary Site HMC Actions in Unplanned Scripts</td>
</tr>
<tr>
<td></td>
<td>Allows the listed script to perform a RESET and DEACTIVATE of LPARs currently defined at the Primary DASD site.</td>
</tr>
<tr>
<td></td>
<td>U12A Recover after loss of DC1 (LDR)</td>
</tr>
<tr>
<td>GDDR5FHS</td>
<td>Allow Secondary Site HMC Actions in Unplanned Scripts</td>
</tr>
<tr>
<td></td>
<td>Allows the listed script to perform an ACTIVATE and LOAD of LPARs currently defined at the Secondary DASD site.</td>
</tr>
<tr>
<td></td>
<td>U12A Recover after loss of DC1 (LDR)</td>
</tr>
<tr>
<td>GDDR5FHS</td>
<td>Allow Primary Site HMC Actions in Test Scripts</td>
</tr>
<tr>
<td></td>
<td>Allows the listed scripts to perform an ACTIVATE and LOAD of LPARs at the DC3 site, for systems currently defined at the Primary DASD site.</td>
</tr>
<tr>
<td></td>
<td>P01A Perform test IPL from BCVs at DC3</td>
</tr>
<tr>
<td></td>
<td>Allows the listed scripts to perform a RESET and DEACTIVATE of LPARs at the DC3 site, for systems currently defined at the Primary DASD site.</td>
</tr>
<tr>
<td></td>
<td>P02A Resume after test IPL from BCVs at DC3</td>
</tr>
<tr>
<td>GDDR5FHS</td>
<td>Allow Secondary Site HMC Actions in Test Scripts</td>
</tr>
<tr>
<td></td>
<td>Allows the listed scripts to perform an ACTIVATE and LOAD of LPARs at the DC3 site, for systems currently defined at the Secondary DASD site.</td>
</tr>
<tr>
<td></td>
<td>P01A Perform test IPL from BCVs at DC3</td>
</tr>
<tr>
<td></td>
<td>Allows the listed scripts to perform a RESET and DEACTIVATE of LPARs at the DC3 site, for systems currently defined at the Secondary DASD site.</td>
</tr>
<tr>
<td></td>
<td>P02A Resume after test IPL from BCVs at DC3</td>
</tr>
</tbody>
</table>
Performing Script Operations

Recovering from script errors

The Resiliency Expert utility automates recovery from GDDR-detected script errors. Prior to GDDR V4.2, if a Host Component command failed during a GDDR script, the script stopped and manual intervention was required. The Resiliency Expert attempts to correct devices which have failed to successfully complete a Host Component command.

The Resiliency Expert is only called from GDDRDF0. It is controlled via global variables which permit GDDR to either:

- Identify the failing commands and devices without attempting fixes.
- Identify them and attempt correction.

Table 22 lists the global variables that determine the operating mode:

**Table 22 Resiliency Expert global variables**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLOBAL.GDDR.PARMS.RDF0.FIX</td>
<td>If the value is 0, the Resiliency Expert is disabled. If the value is greater than 0, the Resiliency Expert is enabled. The default value is 1.</td>
</tr>
<tr>
<td>GLOBAL.GDDR.PARMS.RDF0.FIX.LIMIT</td>
<td>Sets the maximum number of iterations allowed to fix a device. If the value is 0, fix attempts are blocked, but failing devices are identified. If the value is greater than 0, fixes are attempted, but are limited by the value of the global. For example, if GLOBAL.GDDR.PARMS.RDF0.FIX.LIMIT = 10, the fix process terminates after 10 consecutive failures to fix devices (10 separate devices, not 10 attempts at the same device). The failures must be consecutive; a success resets the counter to 0. The default value is 10.</td>
</tr>
</tbody>
</table>
Performing Script Operations

You can enable or disable the Resiliency Expert via the FIXPERT command as follows:

- `F GDDRMAIN,FIXPERT,ON` Toggles GLOBAL.GDDR.PARMS.RDF0.FIX ON
- `F GDDRMAIN,FIXPERT,OFF` Toggles GLOBAL.GDDR.PARMS.RDF0.FIX OFF
- `F GDDRMAIN,FIXPERT` Displays the current state of the Resiliency Expert

The Resiliency Expert supports the following Host Component commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADCOPY</td>
<td>HSWAP</td>
</tr>
<tr>
<td>ADCOPY-DISK</td>
<td>ITA</td>
</tr>
<tr>
<td>CASRSUM</td>
<td>MOVEPAIR</td>
</tr>
<tr>
<td>CASSUSP</td>
<td>NADCOPY</td>
</tr>
<tr>
<td>CASSWAP</td>
<td>NDOMINO</td>
</tr>
<tr>
<td>CREATEPAIR</td>
<td>NITA</td>
</tr>
<tr>
<td>DELETEPAIR</td>
<td>NRDY</td>
</tr>
<tr>
<td>DOMINO</td>
<td>RDF-NRDY</td>
</tr>
<tr>
<td>HDELETEPAIR</td>
<td>RDF-RDY</td>
</tr>
<tr>
<td>HMOVEPAIR</td>
<td>RDF-RSUM</td>
</tr>
</tbody>
</table>

The Resiliency Expert handles only one device at a time. If it does not have a specific action to take for the failing command, it reissues the failing command. Commands are typically issued for all devices or a range of devices, so if the reissue is effective in fixing the problem, it may fix it for a number of devices. The original command is only reissued one time.

### Rerunning a script

If you are not running the Resiliency Expert or any step of a script completes with a nonzero return code, the script terminates. The reason for the failure must be investigated and resolved. You can find a description of the GDDR639I message you receive and the return codes that can accompany it in the *EMC GDDR Message Guide*.
After the cause of the failure has been resolved, submit the script again. No JCL changes are required. GDDR determines the correct step from which to resume scripting.

Depending on how the issue was resolved, it may be necessary to skip the failing step. The GDDR Solution Support team will assist you in making this decision and taking appropriate action.

WTOR messages

During execution of a script, GDDR displays WTOR messages for each of the steps required to execute a site swap.

No operator reply is required for GDDB-prefixed WTOR messages, as these will be replied to automatically by GDDR message interception rules.

An operator reply is always required for GDDO-prefixed WTOR messages.

An operator reply is optional for GDDR-prefixed WTOR messages. GDDR will automatically DOM these messages when the condition it is waiting for is reached. To stop the GDDR wait, and operator can reply causing the script to stop, or stop waiting and continue.

Planned script operations

This section describes each planned script operation that you can run from the EMC GDDR Primary Options menu, Option S, Scripts, Run GDDR Scripts. You choose the script you want to run from the Scripts for Planned Actions list in the Select Script to Run panel.

Note: DC1 and DC3 represent the current primary DASD site or current secondary DASD site. When these representations are shown in italic type in script titles, this indicates the values are interchangeable. The descriptions assume that DC1 is the Primary DASD site and Primary site at the beginning of the script.

Automated Configuration Check - DASD - GDRRPCCD

Usage

The GDRRPCCD script runs the GDDRACDD Utility in validation mode. This utility then validates that the GDDR-managed storage matches the RDF device population defined storage configuration at any time.

Review the GDRRPCCD script joblog for GDDP400E and GDDP420E error messages.

Resolve discrepancies by revising the RDF.DEVICES parameters before proceeding.

Refer to “Automated Configuration Discovery for DASD (GDDRACDD)” on page 240 for more information about the validations performed by GDDRACDD.

Note the following:

◆ The GDRRPCCD script should be run before any script from the Planned, Test, or Resumption category is run.

◆ If the GDRRPCCD script discovers any discrepancies between the discovered storage and the defined storage it will raise the CFG event. As long as this event is true, no scripts touching the storage configuration are allowed. Investigate the source of the
Performing Script Operations

discrepancy. Either correct the storage configuration, or start a GDDR Parameter wizard session and run GDDRCDD in Discovery mode to update the defined storage configuration.

Abandon Site DC1 (site swap) - GDD2P17A

Usage
This script is used to shut down the single site workload at the primary site in preparation for the restart of processing at the secondary site.

Restrictions
Before starting, the script verifies that the script is running from the Master C-System at the primary site. If these checks fail, the script terminates with GDDR926E script generation error. For details, refer to the EMC GDDR Message Guide.

Abandon Secondary site (DC2) - GDDRPA60

Usage
This script is used to shut down the single site workload at the designated site in preparation for site maintenance.

Restrictions
Before starting, the script verifies that the script is running from the Master C-System at the primary site. If these checks fail, the script terminates with a GDDR926E script generation error. For details, refer to the EMC GDDR Message Guide.

Restart production at DC2 after site swap - GDD2P18A

Usage
This script is used to restart the single site workload after the ‘Abandon Site DC1 (site swap)’ script has completed successfully.

Restrictions
The script can only start after the Master C-System has transferred from the original primary site to the secondary (new primary) site. The Master C-System function is automatically transferred when the ‘Abandon Site DC1 (site swap)’ script completes successfully. If these checks fail, the script terminates with a nonzero return code and error message GDDR639I. For details, refer to the EMC GDDR Message Guide.
Test operations

Perform test IPL from BCVs at DC3 - GDD2P01A

Usage
This script is used to IPL contingency systems at site DC3 using BCV devices.

Restrictions
The script can only be run from the current Master C-System.

Resume after test IPL from BCVs at DC3 - GDD2P02A

Usage
This script is used to reset clear contingency systems after a test at DC3.

Restrictions
The script can only be run from the current Master C-System.
The following precondition must also be met before running the script:
◆ All DC3 testing is complete.

Unplanned script operations

This section describes GDDR unplanned script processes which are invoked by one of two methods:
◆ The GDDR Event Monitor prompts the operator to request management confirmation of a trigger event or events which may indicate an outage before the script is permitted to start.
◆ The operator submits a script from the Unplanned Actions list in the Select Script to Run panel.

GDDR validates the status of the current host and storage environment against script rules before the script is allowed to begin processing. The steps you need to take to run these scripts are described in “Running scripts” on page 330.

Recover after loss of DC1 (RDR) - GDD2U12A

Usage
This script is used to restart the single site workload after the GDDR Event Monitor has detected an SRDF/A link down event and the GDDR Heartbeat Monitor has timed out. These events occurring concurrently are viewed by EMC GDDR as a regional disaster.

Restrictions
The script can only be run from the current Master C-System.
Resume replication after loss of DC1 - GDD2PA0A

Usage

This script is used to resume the SRDF/A link to the secondary site after a primary site disaster.

Restrictions

The script appears in the ‘Select Script to Run’ panel of the current Master C-System after completion of the ‘Recover after loss of DC1 (RLDR)’ script.

If the ‘Recover after loss of DC1 (RLDR)’ script failed to complete the transfer of the Master C-System to the new primary site, complete the transfer using GDDR Setup and Maintenance menu option T, Transfer Master C-System, before starting this script.

Resumption operations

This section describes each resumption script that you can run from the GDDR Primary Options menu, Option S, Scripts, Run GDDR Scripts. You choose the script you want to run from the Scripts for Resumption list on the Select Script to Run panel. The steps you need to take to run these scripts are described in “Running scripts” on page 330.

Resume SRDF/A after link loss - GDDRPA29 (internally known as GDDRPM29)

Usage

This script is used to restore the SRDF/A link after a loss of the link.

Restrictions

The script can only be run from the current Master C-System.

The following precondition must also be met before running the script:

- The link service must be restored.

Reclaim Secondary site (DC2) - GDDRPA65

Usage

This script is used to restore normal operations after a site has been abandoned for maintenance.

Special operations

Transfer Master C System to <DCx> - GDDRPXMC

- Broadcast global variables
- Transfer Master to the site specified
One GDDRPXMC script is shown for each site in the configuration which is not currently the Master C-system site.

**Global Variable Backup - GDDRPGVB**

Perform backup of global GDDR variables from the current Master C-System.

**Move systems to alternate CPC - GDDRMCPG**

**Usage**

The Move systems to alternate CPC script performs a planned CPC swap. This script is also available for use as Option W, CPC Swap on the Perform HMC LPAR Actions panel shown in Figure 92 on page 216.

**Restrictions**

A planned CPC swap is much more restrictive than unplanned CPC recovery, and if the following conditions are not met, the swap will not occur:

1. There must be at least one recoverable system on the input CPC.
2. The recoverable systems on the input CPC must all have recovery defined to the same CPC (the target CPC). See the Recovery Site, CPC, and LPAR parameter descriptions on page 154.
3. The recoverable systems on the input CPC must all be either at home or must all be away; no mixture.
4. The recoverable systems on the target CPC (if there are any), must all have recovery defined to the input CPC.
5. The recoverable systems on the target CPC must all either be at home or away; no mixture.

In addition, workload is not stopped prior to the swap and is not restarted once the swap is complete. This is a user responsibility.
Performing Script Operations
This chapter discusses how to deal with unplanned events.

- **Introduction** ................................................................. 344
- **Regional disaster operations** ......................................... 344
- **System failure operations** ............................................. 345
- **GDDR Master Function transfer** .................................... 347
Introduction

More crucial to your enterprise than operations that can be handled by scripts are those tasks you need to undertake to handle unplanned events. This chapter describes some unplanned events you may need to deal with and the steps you need to take to deal with them. The events include:

- Regional disaster operations
- System failure operations
- Master function transfer operations

Regional disaster operations

GDDR declares a regional disaster (RDR) when the SRDF/A link down condition occurs concurrently with a heartbeat timeout (HBM). A regional disaster may occur as:

- The instantaneous loss of DC1 (primary site).
- A rolling disaster that eventually leads to the loss of the DC1 site.

In either case, a regional disaster is only detected and reported by the GDDR C-System running at the DC3 site.

Confirm loss of DC1

When the GDDR C-System at DC3 detects the loss of the DC1 site, the following panel is displayed. You are requested to confirm the loss of DC1.

**IMPORTANT**

Reply only after you have management approval.

* * * R D R  D e t e c t e d  * * *
* * * R D R  D e t e c t e d  * * *
Please Confirm Loss of Site DC1[DC3]

Seek Management Approval Before Replying
Seek Management Approval Before Replying

*nn Please Confirm Management Approval (Yes/No):

**Note:** The use of DC1 [DC3] indicates that the value is supplied for the primary site, whether it is DC1 or DC3.

A reply of No terminates the dialog. A reply of Yes displays a confirmation message on the system console (and in the SYSLOG). After you have confirmed the regional disaster (RDR), EMC GDDR takes the following steps:

- Makes the C-System at DC3 the Master C-System.
- Attempts to RESET_CLEAR all z/OS systems at the DC1[DC3] site.
**System failure operations**

The following section details how GDDR handles individual system failures and how you should respond to the GDDR prompts.

GDDR distinguishes between the following types of systems:

- A GDDR C-System
- GDDR-managed z/OS systems

**EMC GDDR C-System failure**

When GDDR detects that a C-System has failed at a particular site, the following WTOR message is displayed on the system console:

```
GDDR Unplanned Event ===>
> GDDR Detects C-System SYS1 has Failed at Site DC1,
> You have the following choices...>
> Reply IPL:SYS1 - GDDR to restart SYSS at Current Location DC1
> Ignore - GDDR to Do Nothing.
> *nn Enter desired Option...IPL:SYS1 or I:
```

This condition could be because of some type of disaster at DC1, but more likely, is caused by some type of network problem that has caused a C-System heartbeat timeout which has resulted in GDDR declaring the C-System at DC1 "dead". Therefore, before replying, you must confirm the status of the GDDR C-System that has been reported as failed.

**GDDR to restart sssss at current location DCn**

To have GDDR restart the failed C-System (at its present location), reply:

R nn, IPL:sssss

where:

- **nn** is the WTOR ID number.
- **sssss** is the z/OS system name of the failed GDDR C-System.

GDDR performs a LOAD CLEAR on the failed system. You are prompted to confirm the load address and load parameters.

**GDDR to take no action**

To have GDDR take no further action for the failed C-System, reply:

R nn, I

where:

- **nn** is the WTOR ID number.
Handling Unplanned Events

Production system failure

GDDR declares a managed system failure when one of the messages listed below occurs on any GDDR-managed system and both of the following conditions are true for the managed system:

- GDDRMAIN is found not active on the GDDR managed system.
- An HMC query either is unable to determine system status or returns a status different from "Operating".
- SCF0646W CSC (ccccccc-ccccc) HOST system-name (hhhhhhhhhhhhhhhh) REMOVED, MISSING HEART BEAT FOR sssssss SECONDS
- SCF0696W CSC (ccccccc-ccccc) HOST system-name (hhhhhhhhhhhhhhhh) HAS BEEN UNREGISTERED BY HOST system-name (hhhhhhhhhhhhhhhh)

The following WTOR message is displayed on the system console:

```
GDDR Unplanned Event ===>
> GDDR Detects Production System SYS2 has Failed at Site DC1,
> You have the following choices...
> > Reply IPL:SYS2 - IPL the system in the LPAR in which it was running.
> > SYSRESET — RESET CLEAR the system and leave it down.
> > SYSCONV — Recover the system on its recovery LPAR.
> > SYSSITEn — IPL the contingency system for the failed system.
> > I — Ignore the event and take no action.
> *nn Enter desired Option...IPL:SYS2, SYSRESET,SYSCONV, SYSSITEn or I:
```

**IMPORTANT**

Before making any reply, confirm the status of the GDDR-managed system that has been reported as failed.

z/OS SFM sysplex timer failures have the potential for expressing themselves to GDDR as system failures. You should review all z/OS systems consoles including the HMC’s system consoles for STP failure WTORs.

**IPL the system in the LPAR in which it was running**

To have GDDR restart the failed system (at its present location), reply:

R nn, IPL:ssss

where:

- nn is the WTOR ID number.
- ssss is the z/OS system name of the failed system.

GDDR performs a LOAD CLEAR on the failed system. You are prompted to confirm the load address and load parameters.

**IPL the contingency system for the failed system**

To have GDDR restart the business applications at the opposite site to the failed production system, reply:
Handling Unplanned Events

GDDR Master Function transfer

If the GDDR Heartbeat Monitor detects that the C-System that owns the Master Function is no longer updating its heartbeat and has been declared dead, GDDR transfers Master Function ownership to another C-System.

GDDR issues the following message to the system console requesting confirmation of GDDR Master Function ownership transfer:

R nn, SYSSITE\$n
where:

nn is the WTOR ID number.

GDDR performs a RESET_CLEAR of the failed managed system, then triggers the restart of the business applications on the system that is the contingency partner for the failed managed system.

RESET CLEAR the system and leave it down
To have GDDR do a system reset for the failed managed system, reply:
R nn, SYSRESET
where:

nn is the WTOR ID number.
GDDR performs a RESET_CLEAR of the failed system. It performs no other actions.

Recover the system on its recovery LPAR
To have GDDR recover the failed system in its recovery LPAR, reply:
R nn, SYSRECOV
where:

nn is the WTOR ID number.

Ignore the event and take no action
To have GDDR take no further action for the failed managed system, reply:
R nn, I
where

nn is the WTOR ID number.
GDDR Confirmation ===>

Please Confirm C -System Master Transfer

From System : SYS3
to System : SYS5

Reason: CSYSDEAD

* nn Please Confirm Transfer Master Function  (Yes/No):

IMPORTANT
Before replying, check with your z/OS systems programming support group to confirm what to reply.

GDDR to transfer Master Function ownership
To have GDDR complete the transfer of Master Function ownership, reply:

R nn, yes

where:

nn is the WTOR ID number.

GDDR to take no action
To prevent GDDR from transferring the Master Function ownership, reply:

R nn, no

where:

nn is the WTOR ID number.
CHAPTER 9
Performing Maintenance Procedures

This chapter describes GDDR maintenance procedures.

- Setting up a new GDDR C-System ................................................................. 350
- Renaming an existing GDDR C-System .......................................................... 350
- Changing the GDDR C-System or GDDR managed system IP address .......... 351
- Changing the GDDR C-System or GDDR managed system IP port ............... 351
- Adding a new system or sysplex to GDDR .................................................. 352
- Adding new RDF groups to GDDR ............................................................... 354
- Adding new devices to GDDR ...................................................................... 357
- Removing an RDF group from GDDR control ............................................. 358
- Removing devices from GDDR control ....................................................... 359
- Removing a system or a sysplex from GDDR .............................................. 359
- Changing the global variable DIV dataset or WORKER parameters ......... 360
- Special cases ............................................................................................... 361
Performing Maintenance Procedures

Setting up a new GDDR C-System

Complete the following steps to set up a new GDDR C-System:

1. Ensure that correct system software is installed.
   a. Ensure that the operating system meets the requirements for running EMC GDDR.
   b. Ensure that ResourcePak Base, SRDF Host Component, and (optionally) TimeFinder are installed at required release and maintenance levels.

2. Follow the procedures relevant to C-Systems in “Integration tasks” on page 48, and particularly those described in “Install GDDMPARM” on page 66. Confirm that the new C-System is communicating with the other systems in the GDDR complex using the F GDDRMAIN,MPARM command described in Step 8 on page 67.

3. Perform the necessary updates to GDDMPARM CPC parameters. Refer to “GDDMPARM CPC parameter entries” on page 373 for detailed instructions.

4. Verify that parameters describing the new C-System are populated correctly in the Option C; Define C-Systems panel on page 134.

5. Update Symmetrix Control Facility and Consistency Group started task names for the specified C-System name using the Define EMC MF Enablers STCs panel shown in Figure 60 on page 167.

6. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

- Specify GDDR Parameter Load Type: FULL (PARTIAL,FULL)
- Specify GDDR State Variables Action: RESET (RESET, ASIS, NOUPDATE)
- Propagate to Other C-systems: YES (YES/NO/TRY)
- Clear the GDDR Command Queue?: YES (YES/NO)
- Enforce consistency: RETRY=5 (YES/NO/RETRY(1-5)
- Ignore Backup Failure: NO (YES/NO)

Renaming an existing GDDR C-System

Complete the following steps to rename an existing GDDR C-System:

1. Replace the old C System system name with the new C-System system name using the procedures described in “Install GDDMPARM” on page 66.

2. Perform the necessary updates to GDDMPARM CPC parameters. Refer to “GDDMPARM CPC parameter entries” on page 373 for detailed instructions.

3. Confirm that the new C-System is communicating with the other systems in the GDDR complex using the F GDDRMAIN,MPARM command described in Step 8 on page 66.

4. Confirm the SMFID, IPL Parameters, CPC and LPAR name are populated as needed using the Define C-Systems panel shown in Figure 34 on page 134.

5. Replace the existing C-System system name for the Symmetrix Control Facility and Consistency Group started task names using the Define EMC MF Enablers STCs panel shown in Figure 60 on page 167.
Perfoming Maintenance Procedures

6. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

Specify GDDR Parameter Load Type : FULL (PARTIAL, FULL)
Specify GDDR State Variables Action : RESET (RESET, ASIS, NOUPDATE)
Propagate to Other C-systems : YES (YES/NO/TRY)
Clear the GDDR Command Queue ? YES (YES/NO)
Enforce consistency : RETRY=5 (YES/NO/RETRY(1-5)
Ignore Backup Failure : NO (YES/NO)

Changing the GDDR C-System or GDDR managed system IP address

If it becomes necessary to change the IP address of one or more systems on which GDDRMAIN runs, you must update GDDMPARM in synchronization with the IP address change. Complete the following steps:

1. Stop the COMM subtask of GDDRMAIN on all systems which are having an IP address change.
2. Edit a GDDMPARM dataset to reflect the new IP addresses.
3. Once the new IP addresses are in use, start the COMM subtask on one of the changed systems.

Note: This will result in message GDDM103W being issued from systems on which COMM was not stopped; ignore these messages.

4. Verify that the new GDDMPARM does not cause any parameter initialization errors. If it does, stop COMM, correct the problem, and repeat step 3.
5. Propagate the new GDDMPARM to all GDDR C-Systems and managed systems. Keep in mind the requirement for consistency, as discussed in “Install GDDMPARM” on page 66.
6. Start COMM on systems where it is not running; restart the COMM subtask of GDDRMAIN on systems where it was left running. Once again, verify that there are no parameter error messages on any system. If there are, correct them and go back to step 3.
7. Verify connectivity and consistency for all systems as described in step 6 of “Install GDDMPARM” on page 66.

Changing the GDDR C-System or GDDR managed system IP port

This is similar to an IP address change, except that the change will affect all copies of GDDRMAIN on all GDDR systems because the port number must be the same for all systems. Complete the following steps:

1. Edit a GDDMPARM dataset, changing the port number on all COMM statements.
2. Propagate the dataset to all GDDR systems.
3. Issue the F GDDRMAIN,MPARM command on any GDDR system and verify that the value shown for dataset is the same for every system. If it is not, propagate the new GDDMPARM to all systems where a difference is shown.

4. Restart the COMM subtask of GDDRMAIN on all GDDR systems. When COMM restarts, GDDRMAIN waits one minute before broadcasting its in-use MPARM data.
   - If you restart the COMM subtask on all GDDRMAIN copies within a minute, the process should complete successfully.
   - If more than one minute elapses between the first COMM restart and the last, you will probably receive GDDM103W messages (which can be ignored) and GDDM141E messages with resultant setting of degraded mode. This should resolve itself when all COMM subtasks have been restarted. Alternatively, you can stop COMM on each system and then start COMM on each system after all are down. You can restart COMM on all systems by using the BC command of GDDRMAIN. “Broadcast console commands” on page 71 provides details.

5. Verify connectivity and consistency for all systems as described in step 6 of “Install GDDMPARM” on page 66.

Adding a new system or sysplex to GDDR

Use the following procedure when adding a new system or sysplex to the Enterprise Consistency Group, and thereby placing them under the management of GDDR.

1. Ensure that ResourcePak Base is set up using common parameter definitions already used by the other systems in the Enterprise Consistency Group.

2. Edit a GDDMPARM dataset accessible by the new system and either add the COMM statement for a new system or remove the COMM statement for a system being removed.

3. Perform the necessary updates to GDDMPARM CPC parameters. Refer to “GDDMPARM CPC parameter entries” on page 373 for detailed instructions.

4. Start GDDRMAIN on the new system and verify that no parameter initialization error messages occur. Note that GDDM103W messages will occur on other systems; these can be ignored.

5. Once the new GDDMPARM is satisfactory, propagate it to all GDDR systems.

6. Verify the propagation as described in step 3 of the “Changing the GDDR C-System or GDDR managed system IP port” on page 351.

7. Restart the COMM subtask on all GDDR systems except the new system.
   See the comments in step 4 of “Changing the GDDR C-System or GDDR managed system IP port” on page 351. You can accomplish this by using the BC command of GDDRMAIN. “Broadcast console commands” on page 71 provides details.

8. Verify connectivity and consistency for all systems as done in step 6 of “Install GDDMPARM” on page 66.

9. Verify that the parameters describing the new system are populated correctly in the “Option H: Define Host Objects” panel series, beginning on page 146.
10. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

- Specify GDDR Parameter Load Type: FULL (PARTIAL,FULL)
- Specify GDDR State Variables Action: RESET (RESET, ASIS, NOUPDATE)
- Propagate to Other C-systems: YES (YES/NO/TRY)
- Clear the GDDR Command Queue?: YES (YES/NO)
- Enforce consistency: RETRY=5 (YES/NO/RETRY(1-5)
- Ignore Backup Failure: NO (YES/NO)

Changing the MSC group name

To change the name of the MSC group, perform the following steps:

1. Replace the old MSC group name using the Define Configuration Basics menu, “Option R: Define Site Roles and Groups” on page 137.

2. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

- Specify GDDR Parameter Load Type: FULL (PARTIAL,FULL)
- Specify GDDR State Variables Action: RESET (RESET, ASIS, NOUPDATE)
- Propagate to Other C-systems: YES (YES/NO/TRY)
- Clear the GDDR Command Queue?: YES (YES/NO)
- Enforce consistency: RETRY=5 (YES/NO/RETRY(1-5)
- Ignore Backup Failure: NO (YES/NO)

3. Load the updated parameters, following the procedure described in “Option P: Profile—Update Personal GDDR ISPF Profile” on page 108.

4. Update MSC group parameters on all systems to reflect the new MSC group name.

5. Update RDF Manager parameter members to reflect the MSC group name. “Create parameter members for SRDF Host Component on C-Systems” on page 53 lists relevant member names for your configuration.

6. Update the RDFPARM member associated with the RDF entry for affected C-Systems within the Define EMC Mainframe Enablers STCs panel, “Option E: Define EMC MF Enablers STCs” on page 167 with the new MSC group name.

7. Bring the new RDF group(s) into the live SRDF/A environment by performing the following actions:
   a. Stop the SRDF/A with MSC environment by entering the MSC,PENDDROP command.
   b. Restart SRDF/A.
Performing Maintenance Procedures

Adding new RDF groups to GDDR

This section describes how to add new DASD to an existing SRDF/A environment by adding one or more new RDF groups to GDDR and including the new devices in the new RDF groups.

When adding new RDF groups to GDDR, it is possible that changes are required to the SYMM parameters in GDDMPARM. This will be the case if you add new VMAX units, or if the expanded configuration requires additional gatekeeper devices. These changes must be done before performing the procedure described below. Refer to “SYMM parameters” on page 77 for more information on how to change SYMM parameters in GDDMPARM.

1. Ensure the following:
   - The RDF groups have been defined.
   - The SRDF/A createpairs have been done and the devices are synchronized.

   **Note:** The procedure for defining dynamic RDF groups and creating device pairs is provided in the *EMC SRDF Host Component for z/OS Product Guide*.

2. Create new GNS groups by specifying the new RDF group with the EMCGROUP utility, or add the new RDF group to an existing enterprise GNS group.

   **Note:** The procedure for creating or modifying GNS groups is found in the Group Name Service description in the *EMC ResourcePak Base for z/OS Product Guide*.

   The GNS group(s) defined can be named according to the following convention:

   GDDRn_ddd_xxxxx_RGRP_nn_JA

   where:

   - \( n \) has the value 1 or 2, used to identify which group to use depending upon the location of the primary DASD, either DC1 or DC3.
   - \( ddd \) has the value CKD or FBA, depending upon what type of DASD is defined in the GNS group.
   - \( xxxxx \) is the last five (5) digits of the VMAX controller serial number.
   - \( nn \) is the new RDF group.

   For each new RDF group, two GNS groups must be defined: one for DC1 and one for DC3.

3. Define or modify MSC GNS groups, or add gatekeeper and RDF groups in the RDF manager parameter members (SITEUDC1 and SITEUDC3) referenced in step 5 on page 355.

   Depending on whether you are adding a new controller or just new RDF group(s) in one or more existing controllers, you will need to define new MSC GNS groups or update existing MSC GNS group definitions:

   - If adding new controllers, go to step 4 on page 354.
   - If adding new RDF groups to an existing controller, go to step 6 on page 356.

4. If adding new controller(s):
If adding a new DASD controller(s) then the following types of MSC GNS groups will need to be defined:

- Gatekeeper
- RDF group

For each type, one GNS group is required for DC1 and one for DC3. Create the new GNS groups using the EMCGROUP utility. The *EMC ResourcePak Base for z/OS Product Guide* describes how to create GNS groups.

The GNS group(s) defined may adhere to the following naming convention:

- **GDDRn_MSC_ddd_xxxxx_GKn** – for MSC Gatekeepers
- **GDDRn_MSC_ddd_xxxxx_RAGRPn** – SRDF/Star ragroup pairs
- **GDDRn_MSC_ddd_xxxxx_RAGRPn_NOSTAR** – for MSC-only RDF groups

where:

- **n** – has the value 1 or 3, used to identify which group to use depending upon the location of the primary DASD, either DC1 or DC3
- **ddd** – has the value CKD or FBA, depending upon what type of DASD is defined in the GNS group
- **xxxxx** – is the last five(5) digits of the VMAX Controller Serial Number
- **n** – is a number, starting at 1

c. Define MSC mode MSC GNS groups.

The following GNS groups must be defined for starting SRDF/A in MSC mode. These are the same as groups GDDRn_MSC_ddd_xxxxx_RAGRPn, except that they have no associated recovery RDF group in the GNS group definition:

- **GDDR1_MSC_ddd_xxxxx_RAGRPn_NOSTAR**
- **GDDR2_MSC_ddd_xxxxx_RAGRPn_NOSTAR**

5. Add GNS group(s) to MSC parameter members.

Add the appropriate GNS group(s) to the RDF manager parameter members for each site:

a. SITEDC1

To the site DC1 RDF manager parameter member SITEDC1, when adding one or more new controller(s), add a new MSC session for each pair of MSC GNS groups using control statements such as the following:

```plaintext
MSC_INCLUDE_SESSION = SCFG(GDDR1_MSC_ddd_xxxxx_GKn,
GDDR1_MSC_ddd_xxxxx_RAGRPn)
```

These must be added to the MSC_GROUP_NAME definition.

b. SITEDC3

To the site DC3 RDF manager parameter member SITEDC3, when adding one or more new controllers, add a new MSC session for each pair of MSC GNS groups using control statements such as the following:

```plaintext
MSC_INCLUDE_SESSION = SCFG(GDDR2_MSC_ddd_xxxxx_GKn,
GDDR2_MSC_ddd_xxxxx_RAGRPn)
```

These must be added to the MSC_GROUP_NAME definition.
Performing Maintenance Procedures

c. SITEUDC1
To the site DC1 RDF manager parameter member SITEUDC1, when running SRDF/A in MSC mode and adding one or more new controllers, add a new MSC session for each pair of MSC GNS groups using control statements such as the following:

```
MSC_INCLUDE_SESSION = SCFG(GDDR1_MSC_ddd_xxxxx_GKn, 
GDDR1_MSC_ddd_xxxxx_RAGRPn_NOSTAR)
```
These must be added to the MSC_GROUP_NAME definition.

d. SITEUDC3
To the site DC3 RDF manager parameter member SITEUDC3, when running SRDF/A in MSC mode and adding one or more new controllers, add a new MSC session for each pair of MSC GNS groups using control statements such as the following:

```
MSC_INCLUDE_SESSION = SCFG(GDDR2_MSC_ddd_xxxxx_GKn, 
GDDR2_MSC_ddd_xxxxx_RAGRPn_NOSTAR)
```
These must be added to the MSC_GROUP_NAME definition.

6. If adding new RDF group(s) to an existing controller(s):
   If adding one or more new RDF groups to an existing controller, existing MSC GNS groups will have to be extended to add the new RDF group(s) and to add the new gatekeeper devices.
   a. Add the new RDF group(s) to the following GNS groups:
      - GDDR1_MSC_ddd_xxxxx_RAGRPn
      - GDDR2_MSC_ddd_xxxxx_RAGRPn
   b. Add the new MSC gatekeepers to the following MSC GNS groups:
      - GDDR1_MSC_ddd_xxxxx_GKn
      - GDDR2_MSC_ddd_xxxxx_GKn
   c. Add the new RDF group(s) for starting SRDF/A in MSC mode to the following GNS groups:
      - GDDR1_MSC_ddd_xxxxx_RAGRPn_NOSTAR
      - GDDR2_MSC_ddd_xxxxx_RAGRPn_NOSTAR

7. Perform GDDR parameter updates:
   a. If any of the standard devices in the RDF groups being added are to have an associated BCV, add the necessary device ranges using the Define Data Storage Objects menu, “Option T: Define TimeFinder Device Ranges” on page 140.
   b. Run GDDRACDD to discover the updated SRDF device configuration and automatically create RDF.DEVICES parameters. Refer to , “,” for detailed instructions.

8. Distribute the changed EMC parameter members to all systems participating in the SRDF/A environment.

9. Bring the new RDF group(s) into the live SRDF/A environment by performing the following actions:
Performing Maintenance Procedures

a. Stop the SRDF/A with MSC environment by entering the **MSC,PENDDROP** command.

b. Restart SRDF/A.

10. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

- **Specify GDDR Parameter Load Type**: FULL (PARTIAL, FULL)
- **Specify GDDR State Variables Action**: RESET (RESET, ASIS, NOUPDATE)
- **Propagate to Other C-systems**: YES (YES/NO/TRY)
- **Clear the GDDR Command Queue**: YES (YES/NO)
- **Enforce consistency**: RETRY=5 (YES/NO/RETRY(1-5)
- **Ignore Backup Failure**: NO (YES/NO)

---

**Adding new devices to GDDR**

This section describes how to add new DASD to an existing GDDR environment by adding the devices to one or more existing RDF groups.

When adding new devices to GDDR, it is possible that changes are required to the SYMM parameters in GDDMPARM. This will be the case if the expanded configuration requires additional gatekeeper devices. These changes must be done before performing the procedure described below. Refer to “SYMM parameters” on page 77 for more information on how to change SYMM parameters in GDDMPARM.

1. Stop the GDDR Event Monitor and Heartbeat Monitor.
2. Stop SRDF/A with MSC by entering the **MSC,REFRESH** command.
3. Create SRDF/A device pairs.
   
   Add the new devices to one or more existing RDF groups using the **#SC VOL CREATEPAIR** command. Details on how to use this command can be found in the **SRDF Host Component for z/OS Product Guide**.

4. Make the following GDDR parameter changes:
   - If any of the standard devices being added are to have an associated BCV, create appropriate BCV device ranges for the MSC groups using the Define Data Storage Objects menu, “Option T: Define TimeFinder Device Ranges” on page 140.
   - Run GDDRCADD to discover the updated SRDF device configuration and automatically create RDF.DEVICES parameters. Refer to “Automated Configuration Discovery for DASD (GDDRCADD)” on page 240 for detailed instructions.

5. Add the devices to the your existing GDDR-protected GNS device groups.
6. Remove or update any SCF.DEV.EXCLUDE.LIST SCF initialization parameters which would exclude the devices you wish to add.
7. Issue the **SCF,GNS,REFRESH** command.
8. Restart SRDF/A.
Performing Maintenance Procedures

9. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

- Specify GDDR Parameter Load Type: FULL (PARTIAL, FULL)
- Specify GDDR State Variables Action: RESET (RESET, ASIS, NOUPDATE)
- Propagate to Other C-systems: YES (YES/NO/TRY)
- Clear the GDDR Command Queue?: YES (YES/NO)

Removing an RDF group from GDDR control

When removing RDF groups from GDDR, it is possible that changes are required to the SYMM parameters in GDDMPARM. This will be the case if you remove VMAX units, or if the reduced configuration requires less gatekeeper devices. These changes must be done before performing the procedure described below. Refer to “SYMM parameters” on page 77 for more information on how to change SYMM parameters in GDDMPARM.

Complete the following steps to remove an RDF group from GDDR control:

1. Ensure that the RDF groups to be removed from the control of GDDR have been removed from any relevant MSC GNS group.

2. Make the following GDDR parameter changes:

- Delete all BCV devices associated with the standard devices being removed using the Define Data Storage Objects menu, “Option T: Define TimeFinder Device Ranges” on page 140.

- Run GDDRACDD to discover the updated SRDF device configuration and automatically create RDF.DEVICES parameters. Refer to “Automated Configuration Discovery for DASD (GDDRACDD)” on page 240 for detailed instructions.

3. Stop the SRDF/A environment.

   - Stop the environment by entering the MSC,PENDDROP command.

4. Restart SRDF/A protection.

   Once successfully restarted, check to ensure that the RDF group(s) being deleted are no longer part of the MSC groups.

5. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

- Specify GDDR Parameter Load Type: FULL (PARTIAL, FULL)
- Specify GDDR State Variables Action: RESET (RESET, ASIS, NOUPDATE)
- Propagate to Other C-systems: YES (YES/NO/TRY)
- Clear the GDDR Command Queue?: YES (YES/NO)
- Enforce consistency: RETRY=5 (YES/NO/RETRY(1-5)
- Ignore Backup Failure: NO (YES/NO)
Removing devices from GDDR control

When removing devices from GDDR, it is possible that changes are required to the SYMM parameters in GDDMPARM. This will be the case if the reduced configuration requires less gatekeeper devices. These changes must be done before performing the procedure described below. Refer to “SYMM parameters” on page 77 for more information on how to change SYMM parameters in GDDMPARM.

Complete the following steps to remove devices from GDDR control.

**Note:** Ensure that the devices to be removed from the control of GDDR are not gatekeeper devices.

1. Stop SRDF/A by entering the `MSC,PENDDROP` command.
2. Delete the SRDF/A relationships for the devices being removed.
3. Update your GNS device group definitions so they no longer include the devices being removed from GDDR control.
4. Issue a `GNS,REFRESH` command.
5. Issue a ConGroup `REFRESH,FORCE` command.
6. Restart SRDF/A.
7. Update GDDR parameters:
   - If any of the standard devices being removed have an associated BCV, remove the associated BCV devices or device ranges using the Define Data Storage Objects menu, “Option T: Define TimeFinder Device Ranges” on page 140.
   - Run GDDRACDD to discover the updated SRDF device configuration and automatically create RDF.DEVICES parameters. Refer to “Automated Configuration Discovery for DASD (GDDRACDD)” on page 240 for detailed instructions.
8. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

Specify the following Validate and Activate parameters:

- Specify GDDR Parameter Load Type : FULL (PARTIAL,FULL)
- Specify GDDR State Variables Action : RESET (RESET, ASIS, NOUPDATE)
- Propagate to Other C-systems : YES (YES/NO/TRY)
- Clear the GDDR Command Queue ? YES (YES/NO)
- Enforce consistency : RETRY-5 (YES/NO/RETRY(1-5)
- Ignore Backup Failure : NO (YES/NO)

Removing a system or a sysplex from GDDR

**Note:** On the system or systems being removed, ensure that the ResourcePak Basestarted procedures have been stopped and will no longer be used.

1. Edit a GDDMPARM dataset accessible by any remaining system and remove the COMM statement for a system being removed.
Performing Maintenance Procedures

2. Perform the necessary updates to GDDMPARM CPC parameters. Refer to “GDDMPARM CPC parameter entries” on page 373 for detailed instructions.

3. Start GDDRMAIN on any remaining system and verify that no parameter initialization error messages occur. Note that GDDM103W messages will occur on other systems; these can be ignored.

4. Once the new GDDMPARM is satisfactory, propagate it to all C-Systems and GDDR-managed systems.

5. Verify the propagation as described in step 3 of “Changing the GDDR C-System or GDDR managed system IP port” on page 351.

6. Restart the COMM subtask on all GDDR managed systems.

   See the comments in step 4 of “Changing the GDDR C-System or GDDR managed system IP port” on page 351. You can accomplish this by using the BC command of GDDRMAIN. “Broadcast console commands” on page 71 provides details.

7. Verify connectivity and consistency for all systems as described in step 6 of “Install GDDMPARM” on page 66.

8. Delete remaining parameters describing the system(s) being removed using the Parameter Management Options Menu, “Option H: Define Host Objects” on page 146.

9. During an appropriate maintenance window, Validate and Activate the parameter changes using the procedures described in “Option V: Validate GDDR Parameter Set” on page 181 and “Option A: Activate GDDR Parameter Set” on page 183.

   Specify the following Validate and Activate parameters:
   
   Specify GDDR Parameter Load Type : FULL (PARTIAL,FULL)
   Specify GDDR State Variables Action : RESET (RESET, ASIS, NOUPDATE)
   Propagate to Other C-systems : YES (YES/NO/TRY)
   Clear the GDDR Command Queue ? YES (YES/NO)
   Enforce consistency : RETRY=5 (YES/NO/RETRY(1-5)
   Ignore Backup Failure : NO (YES/NO)

Changing the global variable DIV dataset or WORKER parameters

To change the global variable DIV dataset definition or WORKER parameters, complete the following steps:

1. Edit a GDDMPARM dataset and make the needed changes on one C-System.

2. Propagate the modified GDDMPARM to all GDDR C-Systems and managed systems. Keep in mind the requirement for consistency, discussed in “Install GDDMPARM” on page 66.

3. Verify the propagation as described in step 3 of “Changing the GDDR C-System or GDDR managed system IP port” on page 351.

4. Restart the COMM subtask of GDDRMAIN on all GDDR systems, even though the COMM parameters have not changed. If you do not do this, you will receive message GDDM144W informing you that the dataset and the in-use values are possibly different. You can accomplish this by using the BC command of GDDRMAIN if you wish. “Broadcast console commands” on page 71 provides details.
5. Restart the other GDDRMAIN subtasks related to the changed parameters on the systems affected by the changes (GVT for DIV dataset definitions, WORKMGR for WORKER parameters).

You do not need to restart these tasks on systems unaffected by the changes. Note that you can use the BC command to do this.

Special cases

In an SRDF/A environment, special consideration must be given to devices that are not SRDF/A linked to DC3, specifically volumes containing the following system data:

- Page datasets
- Non-LOGR couple datasets

These datasets are not managed by the GDDR restart process. EMC GDDR management of these volumes is limited to couple dataset realignment during site swap scripts, ensuring the primary couple dataset is on the primary DASD site, with an alternate on the secondary DASD Site. These devices can be defined to GDDR as external devices. “Configure GDDR support for external devices” on page 84 provides more information.

Use of the “Perform test IPL from BCVs at DC3” script requires no manual actions. GDDR will control the BCV SPLIT operations for volumes outside of EMC GDDR control.

To avoid this problem, the following procedures needed to be adopted.

Page datasets

All volumes containing page datasets must be paired up (using an adaptive copy RDF group defined specifically for this purpose) to appropriate DC3 volumes and allowed to synch up. Once in synch, the pairs must be suspended. This action must be carried out any time paging volumes are added or moved in the configuration.

Volumes used for page datasets must be dedicated volumes, that is, they must contain no other system or user data.

Non-LOGR couple datasets

All volumes containing non-LOGR couple datasets must be paired up (using an adaptive copy RDF group defined specifically for this purpose) to appropriate R2 volumes and allowed to synch up. Once in synch, the pairs must be suspended.

This action must be carried out any time non-LOGR couple dataset volumes are added or moved in the configuration. Additionally, after any policy management the relevant couple datasets must be resynched with the corresponding volume(s) at R2 to ensure the latest changes will be available in the event of a regional disaster.

Volumes used for couple datasets must be dedicated volumes, that is, they must contain no other system or user data.
Performing Maintenance Procedures

Standalone dump considerations

Special considerations for SADMP IPL volumes and DASD DUMP volumes should be made for each site in a GDDR infrastructure. Many SADMP generation options and the definitions of the DASD DUMP volumes embed device addresses for use by SADMP operations after the IPL of SADMP. It is very likely that these addresses have no meaning in a mirrored environment (SRDF/S or SRDF/A), resulting in unpredictable SADMP operations at the recovery sites.

In addition, the contents of the SADMP DASD DUMP volumes is of very little consequence at DC3 and the mirroring only slows down SADMP’s operation. Consequently, you should review your SADMP generation procedures and business continuance operations and consider maintaining separate SADMP IPL and DASD DUMP volumes at DC1 and DC3.
CHAPTER 10
Troubleshooting

This chapter documents how to resolve problems you may encounter while running EMC GDDR scripts.

◆ Detecting and resolving problems ................................................................. 364
◆ Using the GDDRXCMD batch utility ............................................................... 364
◆ Troubleshooting ............................................................................................. 364
Troubleshooting

Detecting and resolving problems

Keep the following points in mind:

◆ GDDR scripts should always end with return code=0. This indicates that the desired action has been completed successfully.

◆ If a GDDR script does not complete with return code=0, you need to identify the problem. First, examine the JOBLOG of the job that ran the script. Then, examine the SYSTSPRT and SYSPRINT DDs for any error messages and nonzero return codes.

◆ Look up GDDR error messages in the EMC GDDR Message Guide. Follow the actions recommended in that document.

Using the GDDRXCMD batch utility

As part of troubleshooting or correcting GDDR issues, you may need to print or clear the GDDR command queue. To do so, use GDDRXCMD, a batch utility that can perform these tasks.

Customize examples below to your site's requirements.

To print the current queue

```
//QPRINT EXEC PGM=GDDRXCMD,PARM=PRINT
//SYSPRINT DD SYSOUT=*  
```

The result will be a hexadecimal + character printout of the current records in the queue.

To clear the current queue

```
//QCLEAR EXEC PGM=GDDRXCMD,PARM=BOTH
//SYSPRINT DD SYSOUT=*  
```

The job must end with RC=0. Run another print job to verify that the queue has indeed been cleared.

**IMPORTANT**

Only clear the queue if you have been advised to do so by EMC Customer Support.

Troubleshooting

After maintenance has been applied, use the GDDRREG utility to verify the GDDR maintenance level.
APPENDIX A
User Exits

This appendix describes GDDR user exit routines.

- User exit programming considerations .................................................. 366
- Exit specifications .................................................................................... 367
User exits must be written in TSO/E REXX. Consult the *z/OS V1R11.0 TSO/E REXX Reference*, SA22-7790-08, for programming assistance. A special procedure is provided below for using compiled CA-OPS/MVS REXX.

In the exit descriptions that follow, all parameters are positional within a single REXX argument. That is, the arguments are accessed by a REXX instruction such as:

```
parse arg parm1 parm2 ...
```

**Sample procedure to use interpreted REXX user exits**

1. Create a user-partitioned dataset patterned after the `hlq.GDDRvrm.RCXFE` library to contain user-customized exit modules.
2. Concatenate this new library behind the `hlq.GDDRvrm.RCXFE` library referenced by the SYSEXEC DD statements in the GDDRPROC JCL procedure.
3. Rename the existing exit module(s) in `hlq.GDDRvrm.LINKLIB` to allow the system to use your customized exit modules in your user partitioned dataset. This will allow you to rename the LINKLIB member back to its original name should you decide to no longer use the exit override.

**Sample procedure to use CA-OPS/MVS compiled REXX user exits**

1. Code a TSO REXX module named `GDDRUXnn`.
2. In that module perform a `CALL OI <modname>`, where `<modname>` is the name of your actual user exit code in CA-OPS/MVS compiled REXX.
3. Modify your GDDRPROC JCL, adding the required OPSEXEC / OPSCOMP DD cards as well as your CA-OPS linklib in STEPLIB.

**Built-in routines available to exits**

Exit routines may save and retain values across separate invocations using built-in routines as follows:

- **GDDRUXSV**
  
  GDDRUXSV allows you to save a value in a specified durable variable, creating the variable if necessary.
  
  Invocation format is:
  
  ```
call gddruxsv variable-name variable-value
  ```
  
  The return code is found in REXX variable ‘result’. Any return code other than 0 indicates an error.

- **GDDRUXGV**
  
  GDDRUXGV allows you to retrieve a value previously saved by GDDRUXSV.
Invocation format is:

```
call gddrxgv variable-name
```

The value is returned in REXX variable ‘result’. If no value is available, the ‘result’ variable is dropped and consequently becomes ‘RESULT’ when evaluated.

**GDDRXDV**

GDDRXDV allows you to delete a durable variable previously created by GDDRXSV.

Invocation format is:

```
call gddrxsv variable-name variable-value
```

The return code is found in REXX variable ‘result’. Any return code other than 0 indicates an error.

**Exit specifications**

**GDDRX01**

This exit is called from Planned or Unplanned scripts at a point appropriate for starting production mainframe workloads. The exit must be named GDDRX01.

**Parameters**

1. **Mode:**
   - SYCH — Caller will wait for result
   - ASYNC — Caller will not wait for result

2. **System ID on which to start workload**

3. **Number of this system in the list of systems for which user exit 1 will be called**

4. **Number of systems in the list of systems for which user exit 1 will be called**

**Return code**

If zero, the exit will not be called on a script rerun. If non-zero, the exit will be called on a script rerun.

**Example**

An example of how user exit 1 could be programmed is provided in the GDDR SAMPLIB distribution library member GDDRX01.

**GDDRX02**

This exit is called from Planned or Unplanned scripts at a point appropriate for stopping production mainframe workloads. The exit must be named GDDRX02.
User Exits

Parameters

1. Mode:
   - SYNCH — Caller will wait for result
   - ASYNC — Caller will not wait for result
2. System ID on which to stop workload
3. Number of this system in the list of systems for which user exit 2 will be called
4. Number of systems in the list of systems for which user exit 2 will be called

Return code

If zero, the exit will not be called on a script rerun. If non-zero, the exit will be called on a script rerun.

Example

An example of how user exit 2 could be programmed is provided in the GDDR SAMPLIB distribution library member GDDRUX02.

GDDRUX03

This exit is called from Planned or Unplanned scripts at a point appropriate for starting production open systems workloads. The exit must be named GDDRUX03.

Parameters

1. Mode:
   - SYNCH — Caller will wait for result
   - ASYNC — Caller will not wait for result
2. Source (site moving from)
3. Target (site moving to)
4. Context (reason for call - values are swap, rdr, tdc3, ldr)

Return code

If zero, the exit will not be called on a script rerun. If non-zero, the exit will be called on a script rerun.

Example

An example of how user exit 3 could be programmed is provided in the GDDR SAMPLIB distribution library member GDDRUX03.

GDDRUX04

This exit is called from Planned or Unplanned scripts at a point appropriate for stopping production open systems workloads. The exit must be named GDDRUX04.
User Exits

Parameters

1. Mode:
   - SYNCH — Caller will wait for result
   - ASYNC — Caller will not wait for result

2. Source (site moving from)

3. Target (site moving to)

4. Context (reason for call—values are swap, rdr, tdc3, ldr)

Return code

If zero, the exit will not be called on a script rerun. If non-zero, the exit will be called on a script rerun.

Example

An example of how user exit 4 could be programmed is provided in the GDDR SAMPLIB distribution library member GDDRUX04.
This appendix describes the IBM BCPii interface.

- Introduction to the BCPii interface ................................................................. 372
- BCPii HMC Networking capabilities and requirements .............................. 372
- GDDMPARM CPC parameter entries ............................................................. 373
- GDDRBPCI BCPii test job for B-Systems ....................................................... 376
Introduction to the BCPii interface

The Base Control Program internal interface (BCPii) is the interface to the IBM Mainframe Service Element (SE) and the Hardware Management Console (HMC) used to communicate with the Central Processing Complex (CPC) SE. BCPii provides for a direct link via the operating system to the SE associated with the local CPC or to a remote CPC via the existing HMC LAN Network as described in “BCPii HMC Networking capabilities and requirements” on page 372.

The BCPii interface provides a Connect and Disconnect function to objects associated with the mainframe CPC. Various functions can then be performed against those objects across the established connections. The relevant object types are CPC, Image, Capacity Record, and the major HMC profile types of Image, Load, and Reset. The connection type determines the functions available to the caller. GDDR supports CPC, Image, and Load Activation Profile connections.

BCPii also provides a Command function allowing z/OS HMC command functions to be entered against the connected-to object. For example, for a CPC connection, command options such as Activate and Deactivate can be requested. For an Image connection, command options such as Load, Reset Clear, z/OS Operating System Commands, and others can be issued.

The BCPii interface also provides List and Query functionality against the object it is connected to, allowing retrieved information to be used in determining the logic path followed in GDDR. The List function lists all objects of a given type associated with the HMC network. A list of CPCs can be obtained that are within the HMC Grouping associated with the local CPC on whose image/LPAR the BCPii List functions are executed. “BCPii HMC Networking capabilities and requirements” on page 372 provides more information on this topic. A list of Images associated with a connected-to CPC can also be requested. Similar lists are available for the other connection types. The Query function returns setting and information specific to the connected-to object allowing the caller to query setup and operational settings for the CPC, the Image, the Profiles, and all other object types. Complementing this Query function is the Set function allowing modification of certain settings within the object. Load parameters for an Image can be changed. LPAR weightings within a CPC can be changed. There are many options available for modification.

BCPii HMC Networking capabilities and requirements

The Service Element (SE) is a specialized laptop in the CPC used to setup, maintain, and control system objects on that CPC. The HMC is an external PC running specialized code that communicates with the SE for all affiliated CPCs allowing, for the most part, the same level of control against the objects associated with any listed CPC but without requiring physical proximity to the CPC. An HMC can control more than a single CPC, thus introducing the concept of CPC grouping. A CPC group is a defined list of CPCs that an HMC can address.

BCPii works with the local SE. All requests that may be resolved locally do not involve any HMC with LIC Enabled status. Each SE affiliates with an HMC which introduces that SE to the CPC group. If more than one HMC has the SE in the CPC group, and all such HMCs have LIC Enabled status, then, for a remote HMC request, the first HMC to get the tap from the SE is the one with which BCPii communicates.
BCPII can only talk to those remote SE/CPCs that are defined in the CPC group on the HMC with which the local SE is affiliated. If multiple HMCs are defined to the Data Center LAN, it is up to that Data Center to ensure all SE/CPCs are defined to the CPC group for each HMC. If all CPCs are defined in each CPC group, that is, each HMC knows about all CPCs in the Network, then BCPII on one system may reach any SE in the network. Contrary to this, the Data Center may wish to fence certain SE/CPCs from BCPII interaction using separate HMCs with unique CPC groupings. The routing for a remote BCPII request involves sending the request to the local SE, then on to an affiliated HMC and then on to the SE of the remote CPC for which the original BCPII request was created. Routing for a local request is directly to the SE for that local CPC and no further.

As the CPC is affiliated with an HMC network, its fully qualified name is comprised of two network related elements: the NETID and the Network Addressable Unit (NAU). These two elements, separated by a period (.) are required when establishing the communications link between the BCPII address space on the GDDR B-System and the SE affiliated with the target of the BCPII request.

Consequently, this NETID component of the CPC name has been added to all GDDR interfaces (ISPF screen, control blocks, and so forth). When the time comes to perform an HMC function against the LPAR/CPC, the fully qualified CPC name (NETID.NAU) is used.

GDDMPARM CPC parameter entries

In addition to the SE/HMC network routing capabilities described above, GDDR provides additional routing capabilities.

If a GDDR code element requires a BCPII action against a particular CPC and does not have SE/HMC network access to the target CPC, it can route the request to another system.

The GDDRMAPPING CPC parameters allow you to inform GDDR which systems have SE/HMC access to the CPCs GDDR is expected to manage.

Note: There is no requirement for GDDR BCPII work to be done on a C-System. It can be done on a C-System or production system that meets certain requirements—consider this to be a “B-System” for BCPII work. A B-System is any GDDR system (C-System or managed system) which is named on a CPC statement in GDDMPARM. The GDDWXH worker and the MISC subtasks of MISCBSnn are applicable only to B-Systems.

CPC parameter requirements

The general syntax of the CPC parameters is explained in “CPC parameters” on page 76. There are two types of CPC parameters: NETID parameters and METHOD parameters.

CPC NETID parameters

A CPC NETID parameter entry is required in GDDMPARM for each GDDR B-System specifying the LAN network ID associated with the HMC network that is affiliated to the SE within the GDDR B-System’s CPC. The purpose is to identify the netid portion of the B-system CPC names, as this cannot be discovered.
One of the tasks within the GDDR Parameter Wizard process involves setting up the GDDR B-Systems with values including the CPC and LPAR names along with LOAD PARM data associated with the image.

GDDR requires the NETID in the B-Systems Parameter Wizard setup as this fully qualified CPC name is stored in a global variable used throughout GDDR. This requires that each B-System have a self-defining NETID entry in a CPC NETID parameter in GDDMPARM. The format for the CPC NETID parameter is identical to that of the CPC HMC method parameter, with the exception that in place of the HMC method at the end of the parameters, the literal "NETID" is substituted. As an example the CPC NETID record for the B-System at DC1 running on CPCA would be as follows:

```
CPC      CSYS1    GDDR DC1,IBM390PS,CPCA,NETID
```

IBM390PS in the above CPC parameter represents the NETID, and CPCA the NAU. The example Network ID (NETID) IBM390PS used here may be in use at your facility; however, this value may not be IBM390PS. Check with your systems programmer.

### CPC METHOD parameters

At least one CPC method parameter entry per managed CPC is required. A managed CPC is any CPC where either a GDDR C-system or a GDDR managed system can run. Multiple entries for the same CPC are allowed, one per B-System in the configuration. The METHOD records:

- Inform GDDR about the method to use for HMC actions against the listed CPC. Currently, only the BCPii method is supported.
- Define B-systems which are expected to have SE/HMC network access to the listed CPC.

Routing of BCPii actions between GDDR B-Systems is allowed and is explained in the following section.

### CPC parameter SE/HMC control requirements

A key point when discussing BCPii revolves around where BCPii is actually required. The system on which the BCPii activity is initiated is where the BCPii address space is required in order to communicate with the SE of the local CPC and, if necessary, the HMC for remote SE requests. The target systems do not need to be running the BCPii address space in order for the functions to work as BCPii is communicating with the SE (remote SE via the affiliated HMC if necessary), not the images running on those systems. As long as the GDDR B-System is running a new enough version of z/OS on a new enough CPC, it can manage z/OS images back to the earliest level of z/OS supported on the required CPC level (z9 and higher). BCPii is supported with GDDR if at least one of the CPCs on which a B-System is running meets the requirements and that system has network connectivity to the HMC/SE associated with all other CPCs.

During GDDRMMAIN startup, the CPC entries within GDDMPARM are read and validated. Updates can be made to the CPC entries within GDDMPARM if changes are desired. Those changes are applied to the running GDDRMMAIN by issuing a restart of the MISC task (F GDDRMMAIN,RESTART MISC). As the GDDMPARM members are required to be in sync across the GDDRMMAIN tasks running, the Broadcast command option (BC) can be used to restart the MISC task on all GDDR B-Systems.
All HMC tasks are now done as external work within the REXX worker (GDDRWORK tasks GDDWXH and/or GDDWXR). All information related to the HMC task including, if requested, tracing information, can be found in the SYSTSPRT sysout stream associated with the GDDRWORK address space on the appropriate C system. Again, keep in mind this might not be the C-System on which the original task was initiated (either via scripting, batch work, or the ISPF interface). This is the system upon which the task is executed.

As an example, see the following configuration:

- CSYS1 — C-System 1 running on CPC IBM390PS.CPCA at DC1
- CSYS2 — Master C-System 2 running on CPC IBM390PS.CPCB at DC2
- CSYS3 — C-System 3 running on CPC IBM390PS.CPCC at DC3
- PSYS1 — Managed system PSYS1 running on IBM390PS.CPCD at DC1
- PSYS2 — Managed system PSYS2 running on IBM390PS.CPCD at DC1
- PSYS3 — Managed system PSYS3 running on IBM390PS.CPCE at DC2
- PSYS4 — Managed system PSYS4 running on IBM390PS.CPCF at DC2
- PSYS5 — Managed system PSYS5 running on IBM390PS.CPCG at DC3

HMC functions originating on C-System CSYS1 targeting system PSYS1 at DC1 run on CSYS1.
- There is a CPC record connecting CSYS1 to CPC IBM390PS.CPCD.

HMC functions originating on C-System CSYS1 targeting system PSYS3 at DC2 run on CSYS1.
- There is a CPC record connecting CSYS1 to CPC IBM390PS.CPCE.

HMC functions originating on C-System CSYS1 targeting system PSYS4 at DC2 run on CSYS2.
- There is no CPC record connecting CSYS1 to CPC IBM390PS.CPCF so the work is run remotely at CSYS2.

HMC functions originating on z/OS system CSYS2 targeting system PSYS3 at DC2 run on CSYS2.
- There is a CPC record connecting CSYS2 to CPC IBM390PS.CPCE. The CPC parameter statements below are used for the description of the validation that occurs when GDDRMAIN is started.

CPC CSYS1 GDDR DC1,IBM390PS.CPCA,BCPII
CPC CSYS1 GDDR DC2,IBM390PS.CPCB,BCPII
CPC CSYS1 GDDR DC3,IBM390PS.CPCC,BCPII
CPC CSYS1 GDDR DC1,IBM390PS.CPCD,BCPII
CPC CSYS1 GDDR DC2,IBM390PS.CPCF,BCPII
CPC CSYS1 GDDR DC2,IBM390PS.CPCE,BCPII
CPC CSYS1 GDDR DC1,IBM390PS.CPCD,BCPII
CPC CSYS1 GDDR DC2,IBM390PS.CPCE,BCPII
CPC  CSYS1  GDDR DC2,IBM390PS.CPCF,BCPII
CPC  CSYS1  GDDR DC3,IBM390PS.CPCG,BCPII

* NETID records for GDDI123B C system processing
*
CPC  CSYS1  GDDR DC1,IBM390PS.CPCA,NETID
CPC  CSYS2  GDDR DC2,IBM390PS.CPCB,NETID
CPC  CSYS3  GDDR DC3,IBM390PS.CPCC,NETID

The CPC entries for CSYS1, CSYS2, and CSYS3 which specify NETID are populated on the Define C-Systems panel by Auto-Discovery, using the Parameter Wizard. GDDRMAIN startup or refresh of the MISC task will check and ensure a valid NETID CPC record exists for each C-System specified in the GDDMPARM member. If not found, GDDR error message GDDM194E is displayed in the GDDRMAIN output to alert the user. All HMC functions against the managed systems PSYS1, PSYS2, and PSYS3 is routed to C-System CSYS1, work for PSYS4 to CSYS2, and work for PSYS5 to CSYS3.

When GDDRMAIN is started on each B-System, an attempt is made to verify BCPII connectivity to the indicated CPC (NETID.NAU). Failure during this check will result in the display of GDDR error message GDDM195E to alert the user to a networking problem.

GDDRBPCI BCPII test job for B-Systems

The batch job GDDRBPCI is distributed with GDDR. The run JCL for GDDRBPCI is provided in SAMPLIB. This program, from each of the GDDR B-Systems, verifies connectivity to all CPCs visible through the HMC grouping. Note that this does not mean each of the GDDR B-Systems will be able to connect to all CPCs visible via CPC grouping. The purpose of this job is to ensure those CPCs defined to GDDR can be interrogated for information. Further, for each CPC that BCPII can connect to, a list of Images on each is obtained and a BCPII connection attempted to each of those images to ensure all GDDR systems under BCPII management are ready for GDDR.

For those systems not eligible for BCPII connectivity from a GDDR B-System, error messages will be generated. If connection problems are encountered, the error messages will help diagnose why. This GDDRBPCI job should be run after all hardware requirements in the MVS Programming Callable Services for High Level Languages have been met and all RACF updates creating the specific Facility Classes and setting the required Permits are made.

Sample output from job GDDRBPCI follows. This job was run from a test GDDR setup at EMC. As this output shows, the local B-System running on the Q3 CPC has visibility to three other CPCs (C, O, and X1) but has authority to only one other CPC (X1). The error message section at the start is used to diagnose any problems or to highlight issues that might be ignored (for instance, non-authorized CPCs). The BCPII Connection Report shows a list of the CPCs that were visible during processing and also a list of all Images on the CPCs to which the job found authorization. BCPII authority to a CPC is separate from its authority to an image on that CPC (that is, not all images on a CPC may be available for BCPII communication). The JES job log also contains further diagnostic information that EMC support personnel can use to diagnose at a deeper level.

BCPII Connection Test beginning...        4 Feb 2011  08:00:56

These messages may be ignored:

Error 746: EVALBLOCK too small, retrying for larger
Default area for data returned from BCPII call is not
large enough. A larger area is automatically obtained

Error 539: BCPII Connection Error - rc 000000F02 rsn FFFFFFFF
F02 indicates a lack of SAF Authority to the object of
call to BCPII. This may require investigation

rC XXXXXXXXXXX can be found in the error section in the BCPII section
of MVS Programming Callable Services for High Level
Languages for each of the BCPII Call types (CONNECT,
DISCONNECT, LIST, QUERY, etc). Each call type has a
set of specific errors as shown in the above error msg
showing F02 as a SAF AUTH error during CONNECT attempt.

BCPII Checking CPC/LPAR Connectivity... 4 Feb 2011 08:00:56

BCPII Processing messages follow. See BCPII Connection Report
below for results of BCPII Connectivity testing

GDDR992E Error 539 in GDDBCPCO: BCPII Connection Error - rc 00000101 rsn FFFFFFFF
GDDR992E Error 539 in GDDBCPCO: BCPII Connection Error - rc 00000101 rsn FFFFFFFF
GDDR992E Error 746 in GDDBCPLS: EVALBLOCK too small, retrying for larger

BCPII Connection Report for 4 identified CPCs 4 Feb 2011 08:01:32

CPC 001 is IBM390PS.C Connection Not Made - Check RACF Facility Classes
CPC 002 is IBM390PS.O Connection Not Made - Check RACF Facility Classes
CPC 003 is IBM390PS.Q3 Connection Complete - Will Check Image Connection <-- Local CPC
CPC 004 is IBM390PS.X1 Connection Complete - Will Check Image Connection

BCPII LPAR Connectivity Report 4 Feb 2011 08:01:32

<table>
<thead>
<tr>
<th>CPC</th>
<th>Image</th>
<th>Connection Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPC</td>
<td>Q11D</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>Q19</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ31A</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ31B</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ31C</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ31E</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ31F</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ311</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ312</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ313</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ314</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ315</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ316</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ317</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ318</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ84</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ85</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ86</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>ZOSEQ87</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X10A</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X10B</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X10C</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X10D</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X10E</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X10F</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X101</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X102</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X103</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X104</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X105</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X106</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X107</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X108</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X109</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC</td>
<td>X11A</td>
<td>Connection Established</td>
</tr>
</tbody>
</table>
BCPi Interface

<table>
<thead>
<tr>
<th>CPC</th>
<th>Image</th>
<th>Connection State</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X11B</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X11C</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X11D</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X11E</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X11F</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X111</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X112</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X113</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X118</td>
<td>Connection Established</td>
</tr>
<tr>
<td>CPC IBM390PS.X1</td>
<td>X119</td>
<td>Connection Established</td>
</tr>
</tbody>
</table>

BCPi Connection Report Completed

READY
END