

# **ECS COTS Test Cases**

## Table of Contents

1	FIREFOX REGRESSION TEST.....	1
1.1	AIM.....	1
1.1.1	Install ESDTs with a variety of spatial search types (ECS-ECSTC-448).....	1
1.1.2	Update multiple ESDTs, success case (ECS-ECSTC-450).....	2
1.1.3	Delete multiple ESDTs, success case (ECS-ECSTC-451).....	4
1.2	BMGT.....	5
1.2.1	Start and navigate through the BMGT GUI (ECS-ECSTC-659).....	5
1.3	CLS.....	5
1.3.1	Order Status GUI: Display Order Status, OD_S6_05. Criterion 130 (ECS-ECSTC-454).....	5
1.3.1.1	Order Status GUI: Display Order Status, OD_S6_05. Criterion 130 (ECS-ECSTC-628).....	6
1.3.2	Order Status GUI: Display Order Status based on History Range, OD_S6_05, Criterion 180 (ECS-ECSTC-455).....	7
1.3.2.1	Order Status GUI: Display Order Status, OD_S6_05. Criterion 130 (ECS-ECSTC-629).....	8
1.3.2.1.1	Order Status GUI: Display Order Status, OD_S6_05. Criterion 130 (ECS-ECSTC-630).....	9
1.4	DPL.....	10
1.4.1	Data Pool Maintenance GUI - Check Batch Insert Status (ECS-ECSTC-456).....	10
1.4.2	GUI Security (ECS-ECSTC-457).....	10
1.4.3	DPL Maintenance GUI (ECS-ECSTC-458).....	11
1.5	DPL Ingest.....	12
1.5.1	Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-459).....	12
1.5.2	Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-460).....	13
1.5.3	Use the DPL Ingest GUI to configure FTP Hosts (ECS-ECSTC-462).....	15
1.5.4	Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-463).....	16
1.5.5	View and modify the DPL Ingest GUI configuration (ECS-ECSTC-464).....	17
1.5.6	Add & Modifying Volume Groups (ECS-ECSTC-465).....	18
1.6	Data Access.....	19
1.6.1	Use DataAccess GUI to configure service for datatypes (ECS-ECSTC-631).....	19
1.6.2	Start and navigate through the Data Access Configuration Interface (ECS-ECSTC-632).....	20
1.7	OMS.....	20
1.7.1	OMS GUI: Non Privileged User Test (ECS-ECSTC-466).....	20
1.7.2	OMS GUI Config Parameters (ECS-ECSTC-467).....	21
1.8	SSS.....	21
1.8.1	Enter Subscriptions via SSS GUI (ECS-ECSTC-468).....	21
1.8.2	Unqualified Subscription with Email (ECS-ECSTC-469).....	22
1.8.3	Placing Bundling Order Subscription : OD_S3_03, criterion 10 (ECS-ECSTC-470).....	22
2	JAVA REGRESSION TEST.....	23
2.1	AIM.....	23

- 2.1.1 Install ESDTs with a variety of spatial search types (ECS-ECSTC-475)..... 23
- 2.1.2 QAUU: Update by ESDT and temporal range for specific measured parameters (ECS-ECSTC-476)..... 45
- 2.1.3 XML Replacement Utility:Successfully replace a granule metadata file (ECS-ECSTC-477)..... 49
- 2.1.4 On-line Archive Repair from non-resident tape: DP\_7F\_01 Criterion 320 (ECS-ECSTC-478)..... 51
- 2.2 BMGT..... 52
  - 2.2.1 Nominal Collection Export[S-1]: Manual Export (ECS-ECSTC-485)..... 52
  - 2.2.2 Nominal Granule Export[S-2a]: Automatic Export: Ingest Granule (ECS-ECSTC-487)..... 55
  - 2.2.3 Nominal Collection Export[S-2a]: Automatic Export: Insert Collection (ECS-ECSTC-490)..... 59
  - 2.2.4 Configuring Collections For Export (ECS-ECSTC-491)..... 63
- 2.3 CLS..... 69
  - 2.3.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-493)..... 69
  - 2.3.2 Order Status GUI: Display Order Status based on History Range, 0D\_S6\_05, Criterion 180 (ECS-ECSTC-494)..... 70
- 2.4 DPL..... 71
  - 2.4.1 Publish with Theme (ECS-ECSTC-479)..... 71
  - 2.4.2 Run DPCV against DPL, Provide a List of Granule IDs : CK\_7F\_01, Criterion 540 (ECS-ECSTC-480)..... 72
- 2.5 DPL Ingest..... 73
  - 2.5.1 Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-481)..... 73
  - 2.5.2 Ingest SIPS granules without publishing to the Data Pool (ECS-ECSTC-482)..... 75
- 2.6 Data Access..... 78
  - 2.6.1 Use DataAccess drill down to submit HEG request (ECS-ECSTC-492) ..... 78
  - 2.6.2 Use DataAccess drill down to submit GDAL request (ECS-ECSTC-599)..... 79
  - 2.6.3 Use DataAccess GUI to configure service for datatypes (ECS-ECSTC-600) ..... 79
  - 2.6.4 Submit eg/esi requests for HEG Service (ECS-ECSTC-3866)..... 80
  - 2.6.5 Submit egi/esi requests for ICESat-2 data (ECS-ECSTC-3867)..... 80
  - 2.6.6 Submit egi/esi requests for MOFIS Service (ECS-ECSTC-3868)..... 81
  - 2.6.7 Submit order request with no processing (ECS-ECSTC-3869) ..... 82
  - 2.6.8 Verify Certificates and trustStore location exist on dpl, oml, and eil. (ECS-ECSTC-3873)..... 82
  - 2.6.9 Submit egi/esi requests for FRI Service (ECS-ECSTC-3881)..... 83
  - 2.6.10 Submit egi/esi requests for SMAPL1L2 Service (ECS-ECSTC-3882) ..... 84
- 3 GDAL REGRESSION TEST ..... 84
  - 3.1 DataAccess GDAL End To End Testing (ECS-ECSTC-520) ..... 84
  - 3.2 DP\_81\_02\_TP050 GDAL processing requests and errors (ECS-ECSTC-856) ..... 86
  - 3.3 SMAP L1/L2 spatial subsetting with GeoTIFF reformatting (ECS-ECSTC-857)..... 87
- 4 APACHE REGRESSION TEST ..... 88
  - 4.1 AIM..... 88
    - 4.1.1 Install ESDTs with a variety of spatial search types (ECS-ECSTC-607)..... 88

- 4.1.2 Update multiple ESDTs, success case (ECS-ECSTC-608)..... 110
- 4.1.3 Delete multiple ESDTs, success case (ECS-ECSTC-609)..... 115
- 4.2 DMS..... 118
  - 4.2.1 Single Granule External Processing Order (ECS-ECSTC-610)..... 118
  - 4.2.2 Cancelled and Failed Granules (ECS-ECSTC-611)..... 119
- 4.3 DPL Ingest..... 119
  - 4.3.1 Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-612)..... 119
  - 4.3.2 Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-613)..... 121
  - 4.3.3 Use the DPL Ingest GUI to configure Remote Transfer Hosts (ECS-ECSTC-614)..... 122
  - 4.3.4 Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-615)..... 123
  - 4.3.5 View and modify the DPL Ingest GUI configuration (ECS-ECSTC-616)..... 124
  - 4.3.6 Add & Modifying Volume Groups (ECS-ECSTC-617)..... 126
- 4.4 DPL..... 127
  - 4.4.1 Data Pool Maintenance GUI - Check Batch Insert Status (ECS-ECSTC-618)..... 127
- 4.5 CLS..... 128
  - 4.5.1 OBSOLETE - Order Status GUI: Display Order Status, OD\_S6\_05. Criterion (ECS-ECSTC-619)..... 128
  - 4.5.2 OBSOLETE - Order Status GUI: Display Order Status based on History Range, OD\_S6\_05, Criterion 180 (ECS-ECSTC-620)..... 128
- 4.6 Data Access..... 130
  - 4.6.1 Use DataAccess GUI to configure service for datatypes (ECS-ECSTC-623)..... 130
  - 4.6.2 Start and navigate through the Data Access Configuration Interface (ECS-ECSTC-627)..... 130
- 4.7 BMGT..... 131
  - 4.7.1 Connect to CMR (ECS-ECSTC-678)..... 131
  - 4.7.2 GUI (ECS-ECSTC-679)..... 133
  - 4.7.3 Configuring Collections for Export (ECS-ECSTC-680)..... 137
  - 4.7.4 Incremental Verification – Nominal (ECS-ECSTC-681)..... 142
- 4.8 URS Authentication..... 152
  - 4.8.1 URS Authentication - HTTP download (ECS-ECSTC-682)..... 152
  - 4.8.2 [OBSOLETE] - NCR8052187 test instruction - MISR LANCE for ASDC requires URS authentication for users accessing via FTP (ECS-ECSTC-683)..... 154
  - 4.8.3 [OBSOLETE] - NCR8052188 test instruction - MISR LANCE for ASDC requires URS authentication for users accessing via HTTPS (ECS-ECSTC-684)..... 155
- 4.9 Tomcat Conf Checkout..... 157
  - 4.9.1 Apache Tomcat Load Balance and Failover Test (ECS-ECSTC-685)..... 157
  - 4.9.2 Long URL Test (ECS-ECSTC-686)..... 158
  - 4.9.3 Package/Zip file checkout (ECS-ECSTC-687)..... 161
- 4.10 Apache 2.2.23 - SSL..... 163
  - 4.10.1 C12500-1 Verify Apache 2.2.23-SSL/tomcat Web Services (ECS-ECSTC-835)..... 163
  - 4.10.2 C12500-2 Verifying the HTTP port numbers for Apache 2.2.23 with SSL can be used to access the specific modes (ECS-ECSTC-836)..... 164
  - 4.10.3 C12500-3 Testing the mode-specific SSL port number can only be used to access one specific single mode (ECS-ECSTC-837)..... 164
  - 4.10.4 Apache 2.2.23-SSL: AIM Regression Test 1 (ECS-ECSTC-838)..... 165
  - 4.10.5 Apache 2.2.23-SSL: AIM Regression Test 2 (ECS-ECSTC-839)..... 169

- 4.10.6 Apache 2.2.23-SSL: AIM Regression Test 3 (ECS-ECSTC-840)..... 173
- 4.10.7 Apache 2.2.23-SSL: DPLIngest Regression Test 1 (ECS-ECSTC-841) ..... 175
- 4.10.8 Apache 2.2.23-SSL: DPLIngest Regression Test 2 (ECS-ECSTC-842) ..... 177
- 4.10.9 Apache 2.2.23-SSL: DPLIngest Regression Test 3 (ECS-ECSTC-843) ..... 179
- 4.10.10 Apache 2.2.23-SSL: DPLIngest Regression Test 4 (ECS-ECSTC-844)..... 180
- 4.10.11 Apache 2.2.23-SSL: DPLIngest Regression Test 2 (ECS-ECSTC-845)..... 181
- 4.10.12 Apache 2.2.23-SSL: DPLIngest Regression Test 24 (ECS-ECSTC-846)..... 183
- 4.10.13 Apache 2.2.23-SSL: BMGT Regression Test 1 (ECS-ECSTC-847) ..... 184
- 4.10.14 Apache 2.2.23-SSL: BMGT Regression Test 2 (ECS-ECSTC-848) ..... 185
- 4.10.15 Apache 2.2.23-SSL: BMGT Regression Test 3 (ECS-ECSTC-849) ..... 186
- 4.10.16 Apache 2.2.23-SSL: Data Pool Regression Test 2 (ECS-ECSTC-850) ..... 188
- 4.10.17 Apache 2.2.23-SSL: DMS Regression Test 1 (ECS-ECSTC-851) ..... 190
- 4.10.18 Apache 2.2.23-SSL: DMS Regression Test 3 (ECS-ECSTC-852) ..... 190
- 4.10.19 Apache 2.2.23-SSL: Web Order Status Regression Test 1 (ECS-ECSTC-853) ... 192
- 4.10.20 Apache 2.2.23-SSL: Web Order Status Regression Test 2 (ECS-ECSTC-854) ... 193
- 5 PVC LITTLE LANCE REGRESSION TESTS ..... 195
  - 5.1 Apache ..... 195
    - 5.1.1 BMGT ..... 195
      - 5.1.1.1 GUI (ECS-ECSTC-700) ..... 195
      - 5.1.1.2 Configuring Collections For Export (ECS-ECSTC-701) ..... 201
      - 5.1.1.3 Connecting to ECHO (ECS-ECSTC-702)..... 207
      - 5.1.1.4 Configuring Collections For Export for Lance (ECS-ECSTC-703)..... 210
    - 5.1.2 AIM ..... 210
      - 5.1.2.1 Install ESDTs with a variety of spatial search types (ECS-ECSTC-704) ..... 210
      - 5.1.2.2 Update multiple ESDTs, success case (ECS-ECSTC-705)..... 212
      - 5.1.2.3 Delete multiple ESDTs, success case (ECS-ECSTC-706) ..... 213
    - 5.1.3 CLS..... 214
      - 5.1.3.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-720) 214
      - 5.1.3.2 Order Status GUI: Display Order Status based on History Range, OD\_S6\_05, Criterion 180 (ECS-ECSTC-721)..... 215
      - 5.1.3.3 Order Status GUI: Display Order Status based on History Range, OD\_S6\_05, Criterion 180 - Specific to PVC Little Lance (ECS-ECSTC-722) ..... 216
    - 5.1.4 DMS ..... 217
      - 5.1.4.1 Single Granule FtpPull and FtpPush Order (ECS-ECSTC-719) ..... 217
    - 5.1.5 DPLIngest..... 218
      - 5.1.5.1 Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-713) ..... 218
      - 5.1.5.2 Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-714).. 219
      - 5.1.5.3 Use the DPL Ingest GUI to configure Remote Transfer Hosts (ECS-ECSTC-715) 221
      - 5.1.5.4 Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-716)..... 222
      - 5.1.5.5 View and modify the DPL Ingest GUI configuration (ECS-ECSTC-717) ..... 223
      - 5.1.5.6 Add & Modifying Volume Groups (ECS-ECSTC-718) ..... 224
  - 5.2 WuFtp ..... 225
    - 5.2.1 WuFTP Regression Test (PVC Little Lance) (ECS-ECSTC-712)..... 225

- 6 HYRAX REGRESSION TESTS..... 227
  - 6.1 EDSC test - Reformatting L1/L2 collections for NetCDF-3 output format (ECS-ECSTC-723)..... 227
  - 6.2 EDSC test - Reformatting SMAP L1/L2 collections for ASCII output format (ECS-ECSTC-724) ..... 228
  - 6.3 Manual Export - Opendap URL (ECS-ECSTC-725)..... 229
  - 6.4 ESI tests - SMAPL1L2 collections - formatting (ECS-ECSTC-726)..... 234
  - 6.5 EDSC test - Reformatting L3/L4 collections for NetCDF-3 output format (ECS-ECSTC-727)..... 234
  - 6.6 EDSC test - Reformatting SMAP L3/L4 collections for ASCII output format (ECS-ECSTC-728) ..... 236
  - 6.7 EGI tests for SMAPL1L2 collections - formatting and exclusions (ECS-ECSTC-729) 237
  - 6.8 EGI tests for SMAPL3 collections - NetCDF3 and ASCII format (ECS-ECSTC-730). 239
  - 6.9 EGI tests for SMAPL1L2 collections parameter subsetting, formatting and exclusions (ECS-ECSTC-731) ..... 242
  - 6.10 EGI tests for SMAPL4 collections - NetCDF-3 and ASCII formatting and exclusions (ECS-ECSTC-732) ..... 243
  - 6.11 ESI tests - SMAPL3L4 collections - reformatting (ECS-ECSTC-733)..... 246
  - 6.12 Configure OPENDAP collection - configure single collection with multiple formats (ECS-ECSTC-734) ..... 246
  - 6.13 NCR 8053130:Support parameter subsetting for the Opendap ESI service (ECS-ECSTC-735)..... 249
- 7 WUFTP REGRESSION TEST..... 254
  - 7.1 WuFTP Regression Test (ECS-ECSTC-711)..... 254
- 8 IMAGEMAGICK REGRESSION TESTS..... 256
  - 8.1 Make sure imagemagick can annotate a PNG file with text (ECS-ECSTC-855)..... 256
- 9 PYTHON REGRESSION TESTS..... 257
  - 9.1 SMAP L1/L2 reformatting to HDF-EOS5 format (ECS-ECSTC-707)..... 257
  - 9.2 Run test script to test the OGC Web Service python library. (ECS-ECSTC-708)..... 258
  - 9.3 Run test script to exercise the python numerical analysis library (numPy) and hdf 5 library (h5py) (ECS-ECSTC-709)..... 259
  - 9.4 Run test script to exercise the python shape file library (ECS-ECSTC-710)..... 260
- 10 POSTGRES REGRESSION TEST ..... 261
  - 10.1 AIM..... 261
    - 10.1.1 Test Case 1 - Install ESDTs with a variety of spatial search types (ECS-ECSTC-862) 261
    - 10.1.2 Test Case 4 - Search Granules for Deletion (ECS-ECSTC-863) ..... 282
    - 10.1.3 Test Case 5 - Granule Logical Deletion -- physical (ECS-ECSTC-865) ..... 283
    - 10.1.4 Test Case 9 - Granule Deletion Phase II - DeletionCleanup (ECS-ECSTC-866)... 285
    - 10.1.5 Test Case 16 - DPL - AIM Inventory Validation (ECS-ECSTC-867)..... 287
    - 10.1.6 Test Case 18 - On-line Archive Repair from non-resident tape: DP\_7F\_01 Criterion 320 (ECS-ECSTC-868)..... 295

- 10.1.7 Test Case 26 - Run Archive Checksum Validation Utility (ACVU) providing a file with a list of dbIds without the -calculate option : CK\_7F\_01, Criterion 670 (ECS-ECSTC-869)..... 296
- 10.1.8 Test Case 27 - AIM XML Check Utility (ECS-ECSTC-870)..... 297
- 10.1.9 Test Case 3 - Delete multiple ESDTs, success case (ECS-ECSTC-904)..... 298
- 10.1.10 Test Case 2 - Update multiple ESDTs, success case (ECS-ECSTC-864)..... 301
- 10.2 DPLIngest..... 306
  - 10.2.1 Test Case 1 - Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-871)..... 306
  - 10.2.2 Test Case 2 - Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-872)..... 308
  - 10.2.3 Test Case 3 - Use the DPL Ingest GUI to configure Remote Transfer Hosts (ECS-ECSTC-873)..... 309
  - 10.2.4 Test Case 4 - Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-874) 310
  - 10.2.5 Test Case 5 - View and modify the DPL Ingest GUI configuration (ECS-ECSTC-875) 311
  - 10.2.6 Test Case 8 - Cross-DAAC Ingest (ECS-ECSTC-876) ..... 313
  - 10.2.7 Test Case 13 - Verifies MISBR Ingest and linkages are handled correctly (ECS-ECSTC-877)..... 314
  - 10.2.8 Test Case 18 - Browse granule(s) in separate PDR from science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-878)..... 315
  - 10.2.9 Test Case 19 - QA granule(s) in same PDR as science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-879)..... 317
  - 10.2.10 Test Case 20 - PH granule(s) in same PDR as science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-880)..... 318
  - 10.2.11 Test Case 21 - PH granule(s) in different PDR from science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-881)..... 319
- 10.3 DPL..... 320
  - 10.3.1 Test Case 3 - Publish with Theme (ECS-ECSTC-882)..... 320
  - 10.3.2 OBSOLETE - Test Case 8 - Update Granule Utility (ECS-ECSTC-883) ..... 321
  - 10.3.3 Test Case 20 - Hidden Scrambler Utility (ECS-ECSTC-884) ..... 321
  - 10.3.4 Test Case 23 - DPL Maintenance GUI (ECS-ECSTC-885)..... 322
  - 10.3.5 Test Case 34 - Un-publish Granules that were Reported by Data Pool Inventory Validation, DP\_7F\_01, Criterion 90 (ECS-ECSTC-886) ..... 323
  - 10.3.6 Test Case 1 - Qualified Subscription Insert (ECS-ECSTC-916) ..... 324
  - 10.3.7 OBSOLETE - Test Case 6 - Download Granules via Web (ECS-ECSTC-917)..... 324
  - 10.3.8 OBSOLETE - Test Case 19 - HEG Order from Web Access (ECS-ECSTC-919). 325
- 10.4 BMGT..... 326
  - 10.4.1 Test Case 3 - Nominal Granule Export[S-2a]: Automatic Export: Ingest Granule (ECS-ECSTC-907)..... 326
  - 10.4.2 Test Case 4 - Nominal Collection Export[S-1]: Manual Export (ECS-ECSTC-908) 330
  - 10.4.3 Nominal Granule Export[S-2b]: Automatic Export: Logically Delete Granule (ECS-ECSTC-909)..... 334

- 10.4.4 Nominal Granule Export[S-2c]: Automatic Export: Physically Delete Granule (ECS-ECSTC-910)..... 338
- 10.4.5 Nominal Granule Export[S-2i]: Automatic Export: Publish Granule (ECS-ECSTC-911)..... 342
- 10.4.6 Nominal Granule Export[S-2j]: Automatic Export: Unpublish Granule (ECS-ECSTC-912)..... 347
- 10.4.7 Test Case 23 - Incremental Verification - Nominal (ECS-ECSTC-913) ..... 352
- 10.5 EMS ..... 364
  - 10.5.1 Test Case 1 - EMS Extract Utility: Verify that flat files are created and exported for all datasets when Extract utility is run using default configuration (ECS-ECSTC-894) 364
  - 10.5.2 Test Case 1a - EMS Extract Utility: The EMS Dataset Extraction Utility shall maintain a record of the status and starting/ending reporting period of each execution. (ECS-ECSTC-895)..... 365
  - 10.5.3 Test Case 1b - EMS Extract Utility: The EMS Dataset Extraction Utility shall by default generate and export EMS flat file information for a 24-hour period (ECS-ECSTC-896)..... 366
  - 10.5.4 Test Case 1c - EMS Extract Utility: The EMS Dataset Extraction Utility shall by default use the ending date of the previous reporting period to determine the starting and ending date for the current execution. (ECS-ECSTC-897) ..... 366
  - 10.5.5 Test Case 1d - EMS Extract Utility: The EMS Dataset Extraction Utility shall by default generate and export each of the following types of flat files per execution (ECS-ECSTC-898)..... 367
  - 10.5.6 Test Case 1e-g - EMS Extract Utility: The EMS Dataset Extraction Utility shall provide the operator the capability to specify an output directory for the flat files. (ECS-ECSTC-899)..... 368
  - 10.5.7 Test Case 1h - EMS Extract Utility: Verify that in extracted dataset files, there is one record per line and each record is divided into fields using '|&|' as a delimiter. (ECS-ECSTC-901)..... 369
- 10.6 SSS..... 370
  - 10.6.1 Place, Edit, Trigger Spatial Subscriptions (ECS-ECSTC-861)..... 370
- 10.7 OMS..... 371
  - 10.7.1 Ftp Pull Acquire (ECS-ECSTC-859) ..... 371
  - 10.7.2 Ftp Push Acquire (ECS-ECSTC-858)..... 371
  - 10.7.3 SCP Acquire (ECS-ECSTC-860)..... 372
  - 10.7.4 OMS Checkout (Track, Edit, Cancel, Resubmit Orders) (ECS-ECSTC-914) ..... 372
- 10.8 DataAccess..... 374
  - 10.8.1 EDSC Spatial Query (ECS-ECSTC-887)..... 374
  - 10.8.2 EDSC Acquire (http) (ECS-ECSTC-888)..... 374

## List of Figures

**No table of figures entries found.**

## **List of Tables**

**No table of figures entries found.**

# 1 FIREFOX REGRESSION TEST

## 1.1 AIM

1.1.1 Install ESDTs with a variety of spatial search types (ECS-ECSTC-448)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Install ESDTs with a variety of spatial search types]</i>		#comment
2	Verify that the descriptor ODL files for the ESDT being installed are moved from the installation source directory into the configured descriptor directory.		
3	Prepare descriptor files for several (at least 5) ESDTs covering a variety of ESDT types described below, such that the descriptors conform to the ECS data model and the XML schema validation rules, guaranteeing successful ESDT installation. Examples of such can be found in /sotestdata/DROP_721/DS_7E_01/Criteria/2030, and are listed as follows:   1. ESDT descriptor with a spatial search attribute of Rectangle  MOD09CMG.005  MYD09CMG.005   2. ESDT descriptor with a spatial search attribute of GPolygon  MOD14.005  MYD14.005   3. ESDT descriptor with a spatial search attribute of NotSupported  AEPOE7W.001  PM1GBAD1.001   4. ESDT descriptor with a spatial search attribute of Orbit  AE_Land.002  AE_Rain.002   5. ESDT descriptor with a spatial search attribute of Point  g3bssp.007  g3bt.007 		
4	Ensure collections are not already installed.		
5	Copy the descriptor files to the source directory (specified in the GUI).		
6	Verify that the descriptors are listed in the GUI.		
7	From the ESDT Maintenance GUI, select the ESDTs in the source directory and initiate installation.		
8	Upon completion of the ESDT installation, verify that the GUI displays a message indicating number of ESDTs successfully installed.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

1.1.2 Update multiple ESDTs, success case (ECS-ECSTC-450)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Preconditions</i>		#comment
2	<i>Modify the ESDT descriptor ODL files for several (at least 6) previously installed ESDTs, such that the modifications include the following variety of ESDT updates:  1. removing one or more existing collection level attributes that are not restricted  2. removing DLLName parameter from the Collection metadata group  3. changing the ShortName in the CollectionAssociation group  4. changing a mandatory inventory attribute to optional  5 adding one or more optional inventory metadata attributes  6. adding one or more new qualifiers on existing events</i>		#comment
3	<i>Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2060/Replacement as the initial descriptors. Use only GLA01.013 ... GLA06.013; do not use GLA07.013.</i>		#comment
4	<i>Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2120 as the replacement descriptors.</i>		#comment
5	Compare each replacement descriptor with its initial counterpart, and note the differences.		
6	Ensure none of the test collections are installed, according to the ESDT Maintenance GUI.		
7	Note the values associated with the following names in EcDsAmESDTMaint.properties:  descriptor.target.dir descriptor.source.dir mcf.target.dir  E.g.,  descriptor.target.dir=/stornext/smallfiles/OPS/descriptor<br		

#	Action	Expected Result	Notes
	<pre> /&gt;descriptor.source.dir=/usr/ecs/OPS/CUSTOM/data/ESS&lt;br /&gt;mcf.target.dir=/stornext/smallfiles/OPS/mcf </pre>		
8	Install the initial test collections:  Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}.  Log in to the ESDT Maintenance GUI. Click the &quot;Install New ESDTs/Update Existing ESDTs&quot; button. Select all the test descriptors. Click the &quot;Proceed with installation/update&quot; button. Ensure that all descriptors install successfully. Ensure the descriptor files are removed from the source directory.		
9	<i>Setup: Testing ESDT Updates</i>		#comment
10	Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}.		
11	Log in to the ESDT Maintenance GUI.		
12	Click the &quot;Install New ESDTs/Update Existing ESDTs&quot; button.		
13	Select all the test descriptors.		
14	Click the &quot;Proceed with installation/update&quot; button.		
15	Upon completion of the ESDT update, verify that the GUI displays a message indicating the number of ESDTs successfully updated.		
16	Verify that, upon completion of updates, the ESDT Maintenance GUI displays a message saying that the ESDT changes will take effect only after the Ingest service is re-started.		

**TEST DATA:**

Any current data types.

Initial ESDTs: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2060/Replacement

Updated ESDTs: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2120

The 'update' ESDT descriptor files have been modified as follows:

1. removing one or more existing collection level attributes that are not restricted

DsESDTGIGLA01.013.desc

2. removing DLLName parameter from the Collection metadata group

DsESDTGIGLA02.013.desc

3. changing the ShortName in the CollectionAssociation group

DsESDTGIGLA03.013.desc

4. changing a mandatory inventory attribute to optional

DsESDTGIGLA04.013.desc

5 adding one or more optional inventory metadata attributes

DsESDTGIGLA05.013.desc  
6. adding one or more new qualifiers on existing events  
DsESDTGIGLA06.013.desc  
QAPercentInterpolatedData

**EXPECTED RESULTS:**

1.1.3 Delete multiple ESDTs, success case (ECS-ECSTC-451)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>DS_7E_01 ESDT Maintenance GUI ITP:  <a href="http://dmsserver.gsfc.nasa.gov/release721/ESDT_Maint_GUI/ITP_DS_7E_01_ESDTGUI.doc">http://dmsserver.gsfc.nasa.gov/release721/ESDT_Maint_GUI/ITP_DS_7E_01_ESDTGUI.doc</a> &para;</i>		#comment
2	<i>Preconditions</i>		#comment
3	Ensure a database client is connected to the ecs database.		
4	Identify 3 ESDTs for deletion (C1, C2, C3).		
5	On the ESDT Maintenance GUI &quot;ESDT List&quot; page, select C1, C2, and C3.		
6	Click the “Delete Selected ESDTs” button at the bottom of the page to begin the ESDT deletion.		
7	Select OK at the confirmation prompt.		
8	Verify the ESDT Maintenance GUI displays a message indicating 3 ESDTs were deleted.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

---

## 1.2 BMGT

### 1.2.1 Start and navigate through the BMGT GUI (ECS-ECSTC-659)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Start the BMGT GUI.		
2	Go through each tab and verify it displays correctly.		

TEST DATA:

EXPECTED RESULTS:

## 1.3 CLS

### 1.3.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-454)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request the status of a single, completed order that was handled by the OMS and by specifying a valid Order Id and user contact email address.  		
2	<i>Verify the following:</i>		#comment
3	a. Order Id is correctly displayed.		
4	b. Submission date/time is correctly displayed.		
5	c. Order state is correctly displayed.		
6	d. Order completion date/time is correctly displayed.		

---

---

---

#	Action	Expected Result	Notes
7	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed		
8	f. Order state and request states are presented in terms that an end user can understand.		
9	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
10	h. There is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

1.3.1.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-628)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request the status of a single, completed order that was handled by the OMS and that contains at least two requests by specifying a valid Order Id and user contact email address.  Each request in the order should contain at least 100 granules that have NOT been processed by HEG or the external subsetter.		
2	<i>Verify the following:</i>		#comment
3	a. Order Id is correctly displayed.		
4	b. Submission date/time is correctly displayed.		
5	c. Order state is correctly displayed.		
6	d. Order completion date/time is correctly displayed.		
7	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed		
8	f. Order state and request states are presented in terms that an end user can understand.		

---

---

#	Action	Expected Result	Notes
9	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
10	h. There is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

1.3.2 Order Status GUI: Display Order Status based on History Range, 0D\_S6\_05, Criterion 180 (ECS-ECSTC-455)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request order history by specifying a starting and ending date, user contact email address, and one valid Order Id for the user.  Ensure that at least 100 orders are displayed that represent a mix of orders for which request details are available for some orders and not available for others.  Ensure that the date range requires the Order Status Interface to retrieve information from both the Order Manager operational tables and archive tables.  Ensure that at least one of the orders contains granules that were processed by the external subsetter and one of the orders contains granules that were processed by HEG.		
2	Verify that the correct orders are returned and sorted by submission date/time.		
3	<i>For each order verify the following:</i>		#comment
4	a. Order Id is correctly displayed.		
5	b. Submission date/time is correctly displayed.		
6	c. Order state is correctly displayed.		
7	d. Order completion date/time is correctly displayed.		
8	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed, including the		

#	Action	Expected Result	Notes
	processing description for the granule to be processed by the external subsetter, in a manner which should be generally understandable by the user.		
9	f. Order state and request states are presented in terms that an end user can understand.		
10	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
11	h. An indication is provided when request details are not available for an order.		

TEST DATA:

EXPECTED RESULTS:

1.3.2.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-629)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request the status of a single, completed order that was handled by the OMS and that contains at least two requests by specifying a valid Order Id and user contact email address.  Each request in the order should contain at least 100 granules that have NOT been processed by HEG or the external subsetter.		
2	<i>Verify the following:</i>		#comment
3	a. Order Id is correctly displayed.		
4	b. Submission date/time is correctly displayed.		
5	c. Order state is correctly displayed.		
6	d. Order completion date/time is correctly displayed.		
7	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed		

#	Action	Expected Result	Notes
8	f. Order state and request states are presented in terms that an end user can understand.		
9	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
10	h. There is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

1.3.2.1.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-630)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request the status of a single, completed order that was handled by the OMS and that contains at least two requests by specifying a valid Order Id and user contact email address.  Each request in the order should contain at least 100 granules that have NOT been processed by HEG or the external subsetter.		
2	<i>Verify the following:</i>		#comment
3	a. Order Id is correctly displayed.		
4	b. Submission date/time is correctly displayed.		
5	c. Order state is correctly displayed.		
6	d. Order completion date/time is correctly displayed.		
7	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed		
8	f. Order state and request states are presented in terms that an end user can understand.		
9	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		

#	Action	Expected Result	Notes
10	h. There is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

### 1.4 DPL

#### 1.4.1 Data Pool Maintenance GUI - Check Batch Insert Status (ECS-ECSTC-456)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Data Pool Maintenance GUI – Check Batch Insert Status]</i>		#comment
2	Use the Data Pool Maintenance GUI to check the status of the batch insert (by Publish Utility) using the Batch Summary tab and also using the List Insert Queue tab (filter by batch label).		
3	Verify that the GUI correctly reports the status in the DIInsertActionQueue.		

TEST DATA:

Any current data type

EXPECTED RESULTS:

#### 1.4.2 GUI Security (ECS-ECSTC-457)

DESCRIPTION:

PRECONDITIONS:

---

STEPS:

#	Action	Expected Result	Notes
1	<i>[GUI Security]</i>		#comment
2	For the DPM GUI, verify that on startup the operator is asked to login and is allowed the use of “full access” features only after successful login.		

TEST DATA:

n/a

EXPECTED RESULTS:

1.4.3 DPL Maintenance GUI (ECS-ECSTC-458)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[DPL Maintenance GUI]</i>		#comment
2	Open the DPL Maintenance GUI in a web browser.		
3	Verify that  g. A collection can be added  h. A collection can be deleted  i. A collection can be updated  j. A collection group can be added  k. A collection group can be updated  l. A theme can be added  m. A theme can be deleted  n. A theme can be updated		

TEST DATA:

Any current data types

EXPECTED RESULTS:

## 1.5 DPL Ingest

### 1.5.1 Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-459)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Data Providers can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[View Data Providers]</i>		#comment
3	As the 'ingest admin' operator, navigate to the Data Provider page on the Data Pool Ingest GUI.		
4	Verify that all provider information in the INGST CI database appears correctly on the Data Provider page.		
5	<i>[Configure Data Providers]</i>		#comment
6	Edit existing provider information (if any) and define new providers to fulfill the following requirements:  At least one data provider should have an FTP notification method,  at least one data provider should have an scp notification method,  at least one data provider should have an email notification method,  at least one data provider should have a combination FTP/email notification method,  and at least one data provider should have a combination scp/email notification method. Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des. At least one provider must use active mode. At least one should use passive mode.		
7	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit data provider information.		
8	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing data providers before saving this information.		
9	Verify that the ingest admin operator has the authorization to define new data providers (i.e., has access to the data provider pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
10	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16110 to be entered.		

#	Action	Expected Result	Notes
11	Verify that all of the notification methods in S-DPL-16150 can be entered or selected on the Data Pool Ingest GUI, as appropriate for the selected transfer method as per S-DPL-16110.		
12	For one provider with an FTP Notification method, verify that the Data Pool Ingest GUI allows the related FTP information to be entered (FTP host, destination directory, login id, password)		
13	Verify that the FTP password entered is not shown or stored in the clear.		
14	Verify that the FTP password entered is not shown in the Data Pool Ingest GUI log.		
15	For one provider with an email notification method, verify that the Data Pool Ingest GUI allows the related email address to be entered.		
16	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new data providers before saving this information.		
17	For one provider with an scp notification method, verify that the Data Pool Ingest GUI allows the related scp information to be entered (scp host, destination directory, login id, password, scp type, and cipher).		
18	Verify that the scp password entered is not shown or stored in the clear.		
19	Verify that the scp password entered is not shown in the Data Pool Ingest GUI log.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**1.5.2 Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-460)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Polling Locations can be configured using the DPL Ingest GUI</i>		#comment

---

---

#	Action	Expected Result	Notes
2	<i>[Configure Polling Locations]</i>		#comment
3	As the Ingest Admin operator use the Data Pool Ingest GUI to define all polling locations that will be used for testing this ticket. (Table of data provider to polling location mappings, and related S-DPL-16230 information for each polling location TBS as part of ITP).  At least one polling location should have an FTP polling method,  at least one polling location should have a local polling method,  and at least one polling location should have an scp polling method. At least one polling location using FTP shall be for a provider using local transfers.		
4	Verify that the ingest admin operator has the authorization to define the polling locations (i.e., has access to the polling location pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
5	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16230 to be entered.		
6	Verify that all of the polling methods in S-DPL-16250 can be entered or selected on the Data Pool Ingest GUI.		
7	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new polling locations before saving this information.		
8	<i>[Edit Polling Locations]</i>		#comment
9	For one polling location, as the 'ingest admin' operator, edit all of its existing configuration parameters. (NOTE: After this criterion is complete, values of these configuration parameters should be reset to appropriate values for processing all PDRs in criterion 300).		
10	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit polling location information.		
11	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing polling locations before saving this information.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

### 1.5.3 Use the DPL Ingest GUI to configure FTP Hosts (ECS-ECSTC-462)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies FTP Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure FTP hosts]</i>		#comment
3	As the 'ingest admin' operator, navigate to the FTP Host page on the Data Pool Ingest GUI.		
4	Edit existing ftp host information (if any, from the INGST CI), per S-DPL-16260, and define new ftp hosts such that all ftp hosts which will be used for testing this ticket are defined. (Table of ftp hosts and related S-DPL-16260 information TBS as part of ITP). At least two ftp hosts should be defined that are not configured for the INGST subsystem.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit ftp host information.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing ftp hosts and definitions of new ftp hosts before saving this information.		
7	Verify that the ingest admin operator has the authorization to define the ftp hosts (i.e., has access to the ftp host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
8	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16260 to be entered.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

### 1.5.4 Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-463)

#### DESCRIPTION:

#### PRECONDITIONS:

#### STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies SCP Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure hosts for scp access]</i>		#comment
3	As the 'ingest admin' operator, navigate to the host page on the Data Pool Ingest GUI.		
4	Edit existing information for hosts accessed via scp (if any, from the INGST CI), per S-DPL-16290, and define new scp hosts such that all scp hosts which will be used for testing this ticket are defined. (Table of hosts accessed via scp, and related S-DPL-16290 information TBS as part of ITP.) Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des, i.e., the test needs to involve several different providers.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to enter and edit the scp host information in S-DPL-16290.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing scp host parameters and definitions of new scp host parameters before saving this information.		
7	Verify that the ingest admin operator has the authorization to define the scp host parameters in S-DPL-16290 (i.e., has access to the host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		

#### TEST DATA:

Any current data types

#### EXPECTED RESULTS:

### 1.5.5 View and modify the DPL Ingest GUI configuration (ECS-ECSTC-464)

#### DESCRIPTION:

#### PRECONDITIONS:

#### STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies DPL Ingest GUI configuration can be viewed and modified correctly.</i>		#comment
2	<i>[View collection configuration]</i>		#comment
3	As an 'ingest admin' operator, use the Data Pool Ingest GUI to list all data types for which configuration parameters were entered in criterion 120.		
4	Verify that "Ignore Validation Warnings" and "Public in Data Pool" are displayed for each ESDT.		
5	Verify that the Data Pool Ingest GUI provides the operator a method to quickly select or scan for a subset of existing Data Pool collections without having to enter the full ESDT name and version.		
6	<i>[Edit collection configuration]</i>		#comment
7	As an 'ingest admin' operator, use the Data Pool Ingest GUI to edit publication policy configuration parameters for a Data Pool collection.		
8	Verify that the edited configuration parameters are correctly updated in the database.		
9	<i>[Reconfigure data type parameters.]</i>		#comment
10	Log into the DPL Ingest GUI as an 'ingest admin' operator, and configure a data type (ESDT and Version) to be published in the public Data Pool upon insert.		
11	Submit several PDRs for the data type configured above. There needs to be a sufficient number of PDRs and granules such that some granules will complete ingest before the configuration change made during the test (see next step), and some granules will not start ingest until after the configuration change has been applied.		
12	After the first few granules completed ingest, re-configure the data type via the DPL Ingest GUI such that the data type no longer will be published in the Data Pool. Also change the minimum retention period sufficiently long so that the ingested granules are not cleaned up immediately after archiving, to allow time for the verification steps below.		

#	Action	Expected Result	Notes
13	Verify that the first few granules that were ingested before the re-configuration are queued with the Data Pool Insert Service for insertion into the public Data Pool area.		
14	Verify that all granules that started ingest one minute or more after the re-configuration are inserted into the non-public Data Pool area, and are NOT queued with the Data Pool Insert Service for insertion into the public Data Pool area. NOTE: if the Science granule has an associated Browse the Browse will be published even if the Science granule is &quot;hidden.&quot;		
15	Verify that ALL browse granules ingested as part of the test are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**1.5.6 Add & Modifying Volume Groups (ECS-ECSTC-465)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>&lt;i&gt;Description and Objectives: Verifies it is possible to add and modify volume groups using the DPL Ingest GUI and that the server correctly uses the new volume groups.&lt;/i&gt;</i>		#comment
2	<i>&lt;i&gt;[Add &amp; Modifying Volume Groups]&lt;/i&gt;</i>		#comment
3	Using the DPL Ingest GUI add a primary and backup Volume Group for a collection.		
4	Ingest a granule and verify that it goes to the primary and backup volume groups.		
5	Modify Volume group and give it a new path		
6	Ingest a granule and verify that it goes to the new path.		

#	Action	Expected Result	Notes
7	Click on reports and verified that the addition and modification from above are displayed in the report.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**1.6 Data Access**

1.6.1 Use DataAccess GUI to configure service for datatypes (ECS-ECSTC-631)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Go to the DataAccess Configuration Interface GUI		
2	Under the Collection Configuration tab, right click on an ESDT. Select &quot;Configure New Service for Collection&quot; Add the GDAL, HEG, or GLAS service.		
3	Verify the service is added successfully.		
4	Under the Service Configuration tab, right click on the same ESDT. Select &quot;Remove Collection from Service&quot; to remove the collection from the service.		
5	Go to the Collection Configuration tab to verify that the collection is no longer configured for the service.		

**TEST DATA:**

**EXPECTED RESULTS:**

---

### 1.6.2 Start and navigate through the Data Access Configuration Interface (ECS-ECSTC-632)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Start the Data Access Configuration Interface GUI.		
2	Go through each tab and verify that it displays correctly.		

TEST DATA:

EXPECTED RESULTS:

## 1.7 OMS

### 1.7.1 OMS GUI: Non Privileged User Test (ECS-ECSTC-466)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>OMS GUI: Non Privileged User Test</i>		#comment
2	Verify that the OM GUI on startup asks for a login from the operator.		
3	Verify the OM GUI sets appropriate access privileges. I.e. It won't allow "full access" features to be used until a valid password is used.		

TEST DATA:

Any current data type

EXPECTED RESULTS:

### 1.7.2 OMS GUI Config Parameters (ECS-ECSTC-467)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Change a configuration parameter in the OMS GUI		
2	Verify that the order of the configuration parameters don't change.		

TEST DATA:

EXPECTED RESULTS:

## 1.8 SSS

### 1.8.1 Enter Subscriptions via SSS GUI (ECS-ECSTC-468)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Enter Subscriptions via SSS GUI]</i>		#comment
2	Use the Spatial Subscription Server GUI to enter unqualified subscriptions for the following data types:  Email Only: MOD03 (latest public version)  Email/FtpPush: MI1B2E, MOD02HKM (latest public version)  Email/FtpPull: MIL2ASAE, AST08 (latest public version)		
3	Enter qualified subscriptions for Data Pool Insert:  Email/Data Pool Insert: NISE, AST_L1B (latest public version)		
4	For at least one of the above subscriptions, specify that the email notification text should include only the qualifying metadata.		

#	Action	Expected Result	Notes
5	For at least one other of the above subscriptions, specify that the email notification text should include all metadata.		

**TEST DATA:**

see above

**EXPECTED RESULTS:**

1.8.2 Unqualified Subscription with Email (ECS-ECSTC-469)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Unqualified Subscription with Email]</i>		#comment
2	Ingest a MOD03 granule.		
3	Verify that the subscription notification email is sent to the correct address for MOD03.		

**TEST DATA:**

current version of MOD03

**EXPECTED RESULTS:**

1.8.3 Placing Bundling Order Subscription : OD\_S3\_03, criterion 10 (ECS-ECSTC-470)

**DESCRIPTION:**

**PRECONDITIONS:**

---

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Placing Bundling Order Subscription : OD_S3_03, criterion 10]</i>		#comment
2	Use the NSBRV GUI to create a bundling order for each type of media.		
3	Verify all optional information can be entered or omitted.		
4	Verify if optional information is omitted, the correct defaults are provided where requirements specify such defaults.		
5	Verify all required information must be entered.		
6	Verify the bundling orders are stored in the database.		
7	Verify each bundling order receives a unique identification that is displayed to the operator		
8	Verify MSS order tracking information is created for the bundling order and the order source and status are set correctly		
9	Verify it is possible to enter subscriptions and pick a bundling order as their distribution action.		

**TEST DATA:**

Any current data type

**EXPECTED RESULTS:**

**2 JAVA REGRESSION TEST**

**2.1 AIM**

2.1.1 Install ESDTs with a variety of spatial search types (ECS-ECSTC-475)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Install ESDTs with a variety of spatial search types]</i>		#comment
2	Verify that the descriptor ODL files for the ESDT being installed are moved from the installation source directory into the configured descriptor directory.		



#	Action	Expected Result	Notes
			/> &quot; For Step 5, Yes, it is asking that you compare what is in the ODL ESDT descriptor file against the XML schema files. For this, I suggest doing some random sampling. Just take a few attributes from the ESDT descriptor file and find how those attributes are defined in the XML schema files.&quot; Need an automated way to compare ODL descriptor files to XML schemas. 
13	Verify that an MCF file is generated for each ESDT being installed, and stored into the configured MCF directory location ( locations pulled from /custom/ecs/TS3/CUSTOM/cfg/EcDsAmESDTMaint.properties )  descriptor.target.dir = /stornext/smallfiles/TS3/descriptor descriptor.source.dir = /usr/ecs/TS3/CUSTOM/data/ESS mcf.target.dir = /stornext/smallfiles/TS3/mcf archive.metadata.dir = /stornext/smallfiles/TS3/metadata 		20130731t124819::f5dpl01v ::cmshared:/stornext/smallfiles/TS3/mcf \$ find . -type f -mtime -1   sed 's/./([^\#]*)#([^\.]*).*\1.\2/' MOD09CMG.005 MOD14.005 MYD14.005 MYD09CMG.005 AE_Land.002 AE_Rain.002 g3bssp.007 PM1GBAD1.001 g3bt.007 AEPOE7W.001 
14	Verify that the generated MCF files are correct. 		( Define &quot;correct&quot;: The MCFs should be a pretty close match with the INVENTORYMETADAT A section and the ARCHIVEDMETADATA section in the ESDT 

#	Action	Expected Result	Notes
			<p data-bbox="1604 233 1913 1416">/&gt;descriptor file. The place where there should be a difference in in the&lt;br /&gt;AdditionalAttributes group. The ProductSpecificMetadata group in the ESDT&lt;br /&gt;descriptor file is replaced with a generic AdditionalAttributes group in the&lt;br /&gt;MCF. )&lt;br /&gt;Within the INVENTORYMETADATA group, the ProductSpecificMetadata group from the descriptor file should be replaced with this AdditionalAttributes block in the MCF file:&lt;br /&gt;&lt;br /&gt; GROUP = AdditionalAttributes&lt;br /&gt; OBJECT = AdditionalAttributesContain er&lt;br /&gt; Data_Location = "NONE"&lt;br /&gt; Mandatory = "FALSE"&lt;br /&gt; CLASS = "M"&lt;br /&gt; OBJECT = AdditionalAttributeName&lt;br /&gt; Mandatory = "FALSE"&lt;br /&gt; CLASS = "M"&lt;br /&gt; Data_Location = "PGE"&lt;br /&gt; NUM_VAL = 1&lt;br /&gt; TYPE =</p>

#	Action	Expected Result	Notes
			&quot;STRING&quot;                      END_OBJECT = AdditionalAttributeName                      GROUP = InformationContent                      CLASS = &quot;M&quot;                      OBJECT = ParameterValue                      Mandatory = &quot;FALSE&quot;                      Data_Location = &quot;PGE&quot;                      NUM_VAL = 1                      TYPE = &quot;STRING&quot;                      END_OBJECT = ParameterValue                      END_GROUP = InformationContent                      END_OBJECT = AdditionalAttributesContainer                      END_GROUP = AdditionalAttributes                      MODIS MCF files should include the descriptor file's ARCHIVEDMETADATA group.                      AE_Land#002.MCF, AE_Rain#002.MCF, and PM1GBAD1#001.MCF do not have this group 
15	Verify that an XML schema file (*.xsd) is generated for each ESDT being installed, and stored into the descriptors directory.		
16	Verify that the generated XML schema files contain validation rules based upon the contents of the Inventory section of the Descriptor file and the default element rules. 		The &quot;Inventory section&quot; probably starts here:                      GROUP = INVENTORYMETADATA                      Jon Pals: 

#	Action	Expected Result	Notes
			For Step 9, take a look at the 'Mandatory =' lines in the INVENTORYMETADATA section of the ESDT descriptor file and compare that with the ESDT's .xsd file in the /stornext/smallfiles/&lt;MO DE&gt;/descriptor directory. The INVENTORYMETADATA attributes with 'Mandatory = &quot;FALSE&quot;' lines should be listed in the .xsd file as having 'minOccurs=&quot;0&quot;' . The INVENTORYMETADATA attributes with 'Mandatory = &quot;TRUE&quot;' lines should be listed in the .xsd file as having no 'minOccurs' or having a 'minOccurs=&quot;1&quot;' . 
17	Verify that basic collection identification information and proper spatial search type have been added into the AIM Inventory database. 		
18	Verify that the PSA associations, if any, have been added into the AIM Inventory database. 		select aa.additionalattributename from amcollection c join dsmdcollectionaddnlattribx ref ax on c.collectionid = ax.collectionid join dsmdadditionalattributes aa on ax.attributeid = aa.attributeid where c.shortname = '\${SHORTNAME}' 

#	Action	Expected Result	Notes
19	Verify that an insert event, and event qualifiers are inserted into the Spatial Subscription Server database for each ESDT being installed. 		<pre data-bbox="1602 232 1913 1414">/&gt;and c.versionid = \${VERSIONID};&lt;br /&gt; Event qualifiers appear restricted to those defined in ecnbeventmetadataattrdef.att ributename.&lt;br &lt;/EVENTPARMS: Extracted from descriptor file&lt;br /&gt;&lt;br /&gt;Online EVENTPARMS known by SSS will be added.&lt;br /&gt;&lt;br &lt;/-- What qualifiers does SSS know about?&lt;br &lt;/select *&lt;br /&gt;from ecnbeventmetadataattrdef&lt;br &lt;/where attributename in (\${EVENTPARMS});&lt;br &lt;/&lt;br /&gt;&lt;/-- What qualifiers were added?&lt;br /&gt;select attributename&lt;br /&gt;from ecnbeventattrxref&lt;br &lt;/where esdt_id = '\${SHORTNAME}'&lt;br &lt;/and versionid = \$VERSIONID;&lt;br &lt;/'RangeBeginningTime' &lt;/and 'RangeEndingTime' &lt;/appear to get combined into 'GranuleTimeRange'.&lt;br &lt;/&lt;br &lt;/'RangeBeginningDate' and 'RangeEndingDate' appear to get combined into 'GranuleDateRange'.&lt;br &lt;/g3bssp.007 and g3bt.007 both lack date and time ranges, but both have GranuleTimeRange and GranuleDateRange in</pre>

#	Action	Expected Result	Notes
			<pre>ecnbeventattrref.&lt;br /&gt;example:&lt;br /&gt;select *&lt;br /&gt;from&lt;br /&gt;ecnbeventdefinition&lt;br /&gt;where (esdt_id = 'MOD14' and versionid = 5)&lt;br /&gt;or (esdt_id = 'MYD14' and versionid = 5)&lt;br /&gt;or (esdt_id = 'AEPOE7W' and versionid = 1)&lt;br /&gt;or (esdt_id = 'PMIGBAD1' and versionid = 1)&lt;br /&gt;or (esdt_id = 'AE_Land' and versionid = 2)&lt;br /&gt;or (esdt_id = 'AE_Rain' and versionid = 2)&lt;br /&gt;or (esdt_id = 'MOD09CMG' and versionid = 5)&lt;br /&gt;or (esdt_id = 'MYD09CMG' and versionid = 5)&lt;br /&gt;or (esdt_id = 'g3bssp' and versionid = 7)&lt;br /&gt;or (esdt_id = 'g3bt' and versionid = 7);&lt;br /&gt;&lt;br /&gt;eventtype   esdt_id   versionid   eventid&lt;br /&gt;-----+-----+-----&lt;br /&gt;----+-----&lt;br /&gt;INSERT&lt;br /&gt;  AE_Land   2&lt;br /&gt;&lt;br /&gt; &lt;br /&gt;INSERT  &lt;br /&gt;AE_Rain   2  &lt;br /&gt; &lt;br /&gt;INSERT   AEPOE7W&lt;br /&gt;  1  &lt;br /&gt; &lt;br /&gt;INSERT&lt;br /&gt;  MOD09CMG  5&lt;br /&gt;&lt;br /&gt; &lt;br /&gt;INSERT  &lt;br /&gt;MOD14   5  &lt;br /&gt; &lt;br /&gt;INSERT  &lt;br /&gt;MYD09CMG   5&lt;br /&gt;&lt;br /&gt; &lt;br /&gt;INSERT  </pre>

#	Action	Expected Result	Notes
			MYD14   5    INSERT    PM1GBAD1   1    INSERT  g3bssp    7    INSERT    g3bt   7   /(10 rows) 
20	Verify that the ESDT Maintenance GUI logs the processing activities including time of request, action requested, ESDT ShortName, Version ID, descriptor file name, and the result of the operation, in a configured log file.		The ESDTMaintenanceGUI.* logs include a timestamp in each log entry, but it's not clear when a request actually began.  These lines from the debug log may be the beginning of a request:  07.31.2013 12:02:09.563 : Thread ID [41] : VERBOSE : /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTAmAE_Land.002.desc dataModelType is: ECS 07.31.2013 12:02:09.563 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTAmAE_Land.002.desc The action is not stated explicitly. In the debug log, one entry includes the word "installing";  07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoM OD09CMG.005.desc The ShortName and

#	Action	Expected Result	Notes
			<p>Version ID are part of the descriptor filename. They appear in several other log entries, but it's not clear why:&lt;br /&gt;&lt;br /&gt;07.31.2013 12:02:09.651 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:10 String Value:AE_Land.002 Type:12&lt;br /&gt;&lt;br /&gt;07.31.2013 12:02:09.702 : Thread ID [41] : VERBOSE :</p> <p>AbstractServiceImpl.setObject4CallableStmt() Param Index:1 String Value:AE_Land Type:1&lt;br /&gt;07.31.2013 12:02:09.702 : Thread ID [41] : VERBOSE :</p> <p>AbstractServiceImpl.setObject4CallableStmt() Param Index:2 String Value:2 Type:4&lt;br /&gt;The debug log reports the successful installation like this:&lt;br /&gt;&lt;br /&gt;07.31.2013 12:02:10.358 : Thread ID [41] : INFORMATION : InstallESDTPage:Successfully installed ESDT DsESDTAmAE_Land.002.desc&lt;br /&gt;The ESDTMaintenanceGUI.ops0.log contains very little information.&lt;br /&gt;Example:&lt;br /&gt;From</p>

#	Action	Expected Result	Notes
			<pre>/usr/ecs/TS3/CUSTOM/logs /ESDTMaintenanceGUI.de bug0.log:&lt;br /&gt;&lt;br /&gt;time of the installation request:&lt;br /&gt;07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br /&gt;&lt;br /&gt;action requested: (this shows the action taken)&lt;br /&gt;07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br /&gt;&lt;br /&gt;ShortName:&lt;br /&gt;07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObje ct4CallableStmt() Param Index:1 String Value:MOD09CMG Type:1&lt;br /&gt;&lt;br /&gt;Version ID:&lt;br /&gt;07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObje ct4CallableStmt() Param Index:4 String Value:5 Type:4&lt;br /&gt;&lt;br /&gt;descriptor file name:&lt;br /&gt;07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO</pre>

#	Action	Expected Result	Notes
			M/data/ESS/DsESDTMoM OD09CMG.005.desc  result of the operation (successful): 07.31.2013 12:02:12.215 : Thread ID [41] : INFORMATION : InstallESDTPage:Successfully installed ESDT DsESDTMoMOD09CMG.0 05.desc
21	<i>Verification Instructions</i>		#comment
22	Verify that there are at least 5 descriptors that match the setup requirements in directory /sotestdata/DROP_721/DS_7E_01/Criteria/2030/. Verify that the value of the OBJECT = SpatialSearchType group is “rectangle”, “GPolygon”, “not supported”, “point”, or “orbit” accordingly:  ESDT   SpatialSearchType   Archiving DAAC  ACR3L2SC.001   NotSupported   ASDC  AEPOE7W.001   NotSupported   NSIDC  AE_Land.002   Orbit   NSIDC  AE_Rain.002   Orbit   NSIDC  MB2LME.198   Orbit   ASDC  MIANRCCH.198   NotSupported   ASDC  MIL2ASOS.198   Orbit   ASDC  MIL3DAE.198   Rectangle   ASDC  MOD09CMG.005   Rectangle   LP DAAC  MOD14.005   GPolygon   LP DAAC  MOP02.003   Rectangle   ASDC  MYD09CMG.005   Rectangle   LP DAAC  MYD14.005   GPolygon   LP DAAC  PM1GBAD1.001   NotSupported   NSIDC  TL3ATD.002   Rectangle   ASDC  g3assp.004   Point   ASDC  g3atb.004   Point   ASDC		
23	Return to the ESDT List Page. Select the check box next to the descriptor names set up in setup step 1. Select the “Install new ESDTs/Update existing ESDTs” button. Note the wall clock time of the start of the installation for later verification in the logs. Note the number of descriptors selected.		Login to the ESDT Maintenance GUI. Ensure no test ESDT is currently installed. If it is, delete its granules, clean up orphans, remove from the

#	Action	Expected Result	Notes
			data pool (DPL GUI), and delete it (ESDT GUI). Click the "Install new ESDTs/Update existing ESDTs" button. Note the ESDT source directory. Copy the test ESDTs to the ESDT source directory.  Login to the ESDT Maintenance GUI. Click the "Install new ESDTs/Update existing ESDTs" button. Select the test ESDTs. Click the "Proceed with installation/update" button. Wait for the screen to refresh.
24	Verify that the GUI displays the number of descriptors installed is equal to the number of descriptors selected.		
25	Verify that the metadata elements provided in the descriptor files were validated by the XML validation utility by viewing its application log in the mode. Verify for each descriptor that no failures occurred. 		
26	Select one descriptor from each part of setup step 1. 		
27	Go to the directory where the generic descriptor schema is installed and view it.		
28	Verify by inspection that rules for elements in the generic schema are executed correctly for corresponding elements in the descriptor. A similar comparison should be performed between the elements of the generic descriptor schema and the ECS data model. 		What is the XML validation utility used, and where is its log? Where is "the directory where the generic descriptor schema is installed"? 
29	Verify by inspection that rules for elements in the generic schema are executed correctly for corresponding elements in the descriptor. 		

#	Action	Expected Result	Notes
30	Verify each element in the descriptor file that has a matching element in the generic schema gets added to the descriptor's .xsd file. 		&quot;A similar comparison should be performed between the elements of the generic descriptor schema and the ECS data model.&quot;  What does this mean? /usr/ecs/TS3/CUSTOM/logs/ESDTMaintenanceGUI.ddebug0.log shows the descriptor .xml and .xsd files are generated but not whether they are validated:  07.31.2013 12:02:12.021 : Thread ID [41] : xmlsvcs.schemagen.EcAmDescSchemaGenImpl : VERBOSE : START generateDescriptorSchemaFromOdlDescriptor odlDescriptor: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc outputSchema: /stornext/smallfiles/TS3/descriptor/DsESDTMoMOD09CMG.005.xsd 07.31.2013 12:02:12.021 : Thread ID [41] : xmlsvcs.schemagen.EcAmDescSchemaGenImpl : VERBOSE : START generateDescriptorXmlFile inputODLFile: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc

#	Action	Expected Result	Notes
			<pre>outputXMLFile: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml&lt;br /&gt;07.31.2013 12:02:12.050 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : END generateDescriptorXmlFile inputODLFile: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc outputXMLFile: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml&lt;br /&gt;07.31.2013 12:02:12.051 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : START generateDescriptorSchemaF romXmlDescriptor xmlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br /&gt;07.31.2013 12:02:12.051 : Thread ID [41] : xmlsvcs.schemagen.EcAmS chemaGenerator : INFORMATION : createESDTSchema</pre>

#	Action	Expected Result	Notes
			<p>START: XML descriptor - /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml, Schema - /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br &gt;07.31.2013 12:02:12.154 : Thread ID [41] : xmlsvcs.schemagen.EcAmS chemaGenerator : INFORMATION : createESDTSchema DONE: XML descriptor - /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml, Schema - /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br &gt;07.31.2013 12:02:12.155 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : END generateDescriptorSchemaF romXmlDescriptor xmlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br &gt;07.31.2013 12:02:12.155 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl :</p>

#	Action	Expected Result	Notes
			VERBOSE : END generateDescriptorSchemaFromOdlDescriptor odlDescriptor: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc outputSchema: /stornext/smallfiles/TS3/descriptor/DsESDTMoMOD09CMG.005.xsd
31	Find the mcf target directory (parameter mcf.target.dir) in the EcAmMaintenanceGui.properties file.  Change the directory to the ESDT specific subdirectory (probably /stornext/smallfiles/&lt;mode&gt;/mcf/).  Perform an ls to verify that the mcf file is found. Repeat for each MCF file generated from the ESDT installation.		/custom/ecs/\${MODE}/CUSTOM/cfg/EcDsAmESDTMaint.properties                      />mcf.target.dir=/stornext/smallfiles/\${MODE}/mcf
32	Compare the descriptor file to the mcf file to verify that the MCF is correct. 		
33	Change the directory to this directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/).		
34	Perform an ls to verify that the schema file is found. 		/custom/ecs/\${MODE}/CUSTOM/cfg/EcDsAmESDTMaint.properties                      />descriptor.target.dir=/stornext/smallfiles/\${MODE}/descriptor
35	Make a one to one comparison of the groups in the Inventory Section of the Descriptor file with the corresponding groups in the schema.		
36	Verify that the values of the descriptor are valid according to the rules in the schema.		
37	Find the descriptor source and target directory (parameters descriptor.source.dir and descriptor.target.dir) in the EcAmMaintenanceGui.properties file. 		
38	Change to target directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/).		

#	Action	Expected Result	Notes
39	Perform an ls to verify that the descriptor file DsESDT<ShortName>.<VersionId>.desc is found. Repeat for each part descriptor installed in the setup step 1.		
40	Verify that the descriptor file was deleted in the source directory in the mode.		
41	Change to directory descriptor.source.dir.		
42	Verify that the descriptor was deleted by using the ls command and finding no files matching the descriptor's name. Also, repeat for each descriptor installed.		/custom/ecs/\${MODE}/CUSTOM/cfg/EcDsAmESDT Maint.properties descriptor.source.dir=/usr/ecs/\${MODE}/CUSTOM/data/ESS descriptor.target.dir=/stornext/smallfiles/\${MODE}/descriptor This should be at least 2 steps.
43	In the AIM Inventory database, verify that the ShortName, VersionID, and insertTime are populated in the DsMdCollections table for each ESDT installed.   select ShortName, VersionID, insertTime  from DsMdCollections  where ShortName = "<ShortName>"  and VersionID= "<VersionID>" 		
44	Verify that the spatial searchSearchType matches the one specified in the descriptor.   select configuredName, VersionID, spatialSearchType  from DsMdESDTConfiguredType  where configuredName = "<shortName>"  and VersionID = "VersionID" 		
45	Repeat for each installed ESDT.		select c.inserttime, esdt(c.shortname,c.versionid), c.collectionid, c.spatialsearchtype, e.spatialsearchtype from amcollection c join dsgeesdtconfiguredtype e on c.shortname = e.configuredname and c.versionid = e.versionid where (c.shortname = '\${SHORTNAME}' and

#	Action	Expected Result	Notes
			<pre>c.versionid = \${VERSIONID})&lt;br /&gt;or (c.shortname = '\${SHORTNAME}' and c.versionid = \${VERSIONID})&lt;br /&gt;...&lt;br /&gt;order by c.spatialsearchtype, c.shortname, c.versionid;&lt;br /&gt;dsgeesdtconfiguredtype replaces DsMdESDTConfiguredTyp e.&lt;br /&gt;select c.inserttime, esdt(c.shortname,c.versionid ), c.collectionid,&lt;br /&gt; c.spatialsearchtype, e.spatialsearchtype&lt;br /&gt;from amcollection c&lt;br /&gt;join dsgeesdtconfiguredtype e&lt;br /&gt;on c.shortname = e.configuredname and c.versionid = e.versionid&lt;br /&gt;where (c.shortname = 'MOD14' and c.versionid = 5)&lt;br /&gt;or (c.shortname = 'MYD14' and c.versionid = 5)&lt;br /&gt;or (c.shortname = 'AEPOE7W' and c.versionid = 1)&lt;br /&gt;or (c.shortname = 'PM1GBAD1' and c.versionid = 1)&lt;br /&gt;or (c.shortname = 'AE_Land' and c.versionid = 2)&lt;br /&gt;or (c.shortname = 'AE_Rain' and c.versionid = 2)&lt;br /&gt;or (c.shortname = 'MOD09CMG' and c.versionid = 5)&lt;br /&gt;or</pre>

#	Action	Expected Result	Notes
			<pre>(c.shortname = 'MYD09CMG' and c.versionid = 5)&lt;br /&gt;or (c.shortname = 'g3bssp' and c.versionid = 7)&lt;br /&gt;or (c.shortname = 'g3bt' and c.versionid = 7)&lt;br /&gt;order by c.spatialsearchtype, c.shortname, c.versionid;&lt;br /&gt;&lt;br /&gt;      inserttime        esdt        collectionid   spatialsearchtype   spatialsearchtype&lt;br /&gt;----- -----+----- -----+----- +-----+----- -----&lt;br /&gt; 2013-07-31 12:02:12.560421   MOD14.005        198330   GPolygon        GPolygon&lt;br /&gt; 2013-07-31 12:02:13.716981   MYD14.005        198332   GPolygon        GPolygon&lt;br /&gt; 2013-07-31 12:02:11.403874   AEPOE7W.001      198328   NotSupported      NotSupported&lt;br /&gt; 2013- 07-31 12:02:14.194661   PM1GBAD1.001     198333   NotSupported      NotSupported&lt;br /&gt; 2013- 07-31 12:02:09.656776   AE_Land.002      198326   Orbit             Orbit&lt;br /&gt; 2013-07-31 12:02:10.765544  </pre>

#	Action	Expected Result	Notes
			AE_Rain.002   198327   Orbit   Orbit  2013-07-31 12:02:14.865666   g3bssp.007   198334   Point   Point<br > 2013-07-31 12:02:15.367958   g3bt.007   198335   Point   Point  2013-07-31 12:02:11.866531   MOD09CMG.005   198329   Rectangle   Rectangle  2013-07-31 12:02:13.12891   MYD09CMG.005   198331   Rectangle   Rectangle (10 rows)
46	In another terminal log into the Spatial Subscription Server (SSS) database. View the &lt;ShortName&gt;.&lt;VersionId&gt; descriptor in the mode and find the INSERT object under the EVENT group in the descriptor. Under the EVENTPARMS object are a list of parameters which should be populated in the SSS database.		
47	Verify that the event was inserted into the EcNbEventDefinition table.   select * from EcNbEventDefinition  where ESDT_Id in ( &lt;a quoted list of ShortNames of installed ESDTs&gt;); 		
48	Verify that each parameter EVENTPARM parameter is populated in the SSS database .   select * from EcNbEventAttrXref  where ESDT_Id = “ShortName” 		
49	Repeat this step for each descriptor installed		
50	Change directory to the log.dir directory of the mode.		
51	View the ESDT_Maint.log and search for the first ESDT installed.		
52	Verify the time of the installation request, the ShortName, Version ID, descriptor file name and the result of the operation (successful) is listed in the log file.		
53	Repeat for each ESDT installed.		time of the installation request: 07.31.2013 12:02:11.821 : Thread ID

#	Action	Expected Result	Notes
			<pre>[41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br /&gt;&lt;br /&gt;ShortName:&lt;br /&gt;07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObj ect4CallableStmt() Param Index:1 String Value:MOD09CMG Type:1&lt;br /&gt;&lt;br /&gt;Version ID:&lt;br /&gt;07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObj ect4CallableStmt() Param Index:4 String Value:5 Type:4&lt;br /&gt;&lt;br /&gt;descriptor file name:&lt;br /&gt;07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br /&gt;&lt;br /&gt;result of the operation (successful):&lt;br /&gt;07.31.2013 12:02:12.215 : Thread ID [41] : INFORMATION : InstallESDTPage:Successful ly installed ESDT DsESDTMoMOD09CMG.0 05.desc</pre>

TEST DATA:

Any current data types

EXPECTED RESULTS:

2.1.2 QAUU: Update by ESDT and temporal range for specific measured parameters (ECS-ECSTC-476)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[QAUU: Update by ESDT and temporal range for specific measured parameters] (See Criterion 3000 in DS_7E_01)</i>		#comment
2	Select an ESDT that has at least two measured parameters per granule. The ESDT has granules in AIM inventory, as well as in Data Pool, some of which are located in public Data Pool area, and others are in hidden Data Pool area. The ESDT MOD10A1 is a good choice.		
3	Set up an input file for the QA Update Utility to update the science quality flags of the ESDT for a temporal range, such that there are at least 10 granules within the time range. Set the temporal range so that it matches the boundary of at least one of the granules acquisition times. Specify a valid email address for the originating SCF site as "Requester ID" in the input request file. Specify science QA flag values and explanation for at least 2 measurement parameters.		
4	Name the input file such that it contains, as part of its file name, the following: (a) the correct ECS mode (b) a valid site name that is allowed to perform the QA update for the ESDT (c) the request time tag in the form YYYYMMDDHHMMSS (Note: Current QAMUT uses the following file naming convention: &lt;Mode&gt;_&lt;SiteName&gt;_QAUPDATE.&lt;YYYYMMDDHHMMSS&gt;)		
5	<i>VERIFICATION</i>		#comment
6	Verify that the QA Update Utility displays the total number of granules that will be affected by this update and prompts for operator confirmation before performing the update.		

---

---

#	Action	Expected Result	Notes
7	Verify that upon completion of the QA update run, the operator can view the following information: a. A list of affected granules identified by ESDT Short Name, Version ID, LGID, dbID, along with the measured parameters and the updated QA flag values b. Total number of granules updated		
8	Verify that the granules listed are in the correct acquisition time range (inclusive of the boundary points) and the granule count is correct.		
9	Verify that the XML metadata files for each of the granules are updated correctly in the small file archive system.		
10	Specifically, verify that the QA flags and the corresponding explanation fields are updated correctly for the affected measured parameters.		
11	Verify the XML metadata files in the public Data Pool file systems are replaced by the corresponding updated XML metadata files in the small file archive system.		
12	Verify the XML metadata files in the hidden Data Pool area are NOT updated.		
13	Verify that the QA Update Utility appended the date and time of update accurate to the minute, to the explanation fields, and stored them in the corresponding metadata files in the small file archive system, as well as in the Data Pool database.		
14	Verify that the updated QA attributes for the affected granules are correctly updated in the Data Pool database.		
15	Verify that lastUpdatetime for each of the granules affected is updated correctly in the AIM Inventory database as well as in the Data Pool database using the system time.		
16	Verify that the QA update utility recorded the list of granules updated along with the updated QA values for the affected measured parameters in the AIM Inventory database (for use by BMGT)		
17	Verify that the QA update request file is moved into the configured completed request directory location.		
18	Verify that a history file is generated and stored in the QA_HISTORY_DIR directory, which contains the granule IDs (db IDs) along with the measured parameters, the QA values, and the corresponding explanation fields before and after the update for each granule updated.		
19	Verify that the QA update utility logged the processing activities including the following: a. time stamp and PID are included in each log entry b. the name of the request file is logged at the start of		

---

---

#	Action	Expected Result	Notes
	processing the update request  c. Total number of granules requested.  d. Total number of granules updated.  e. Run completion status		
20	Verify that the total number of granules requested is the same as the total number of granules updated.		
21	Verify that an email notification is sent to the configured “Reply-to” e-mail address for the originating site, as well as to the configured internal DAAC email address, and that it contains the following information:  a. The name of the request file processed.  b. Total number of granules specified in the request  c. Total number of granules updated.		
22	Verify that the total number of granules requested is the same as the total number of granules updated.		
23	Verify that a log file is created with a standard log file name in a configured log directory.		
24	<i>Verification Instructions</i>		#comment
25	<i>Note: Scripts to assist with this verification were created as part of the Integration Testing for QAUU. They should be stored in a safe location in the PVC for future use.</i>		#comment
26	Use DPL Ingest GUI to control whether the granules for an ESDT go to the data pool and whether they go to the public or hidden Data Pool area.		
27	From DPL Ingest GUI, Configuration, Data Types, configure this ESDT to be inserted into Data Pool and whether it’s public or not.		
28	Configure the ESDT as specified in the Test Data Requirement to be inserted into Data Pool public area, and ingest 5 granules.		
29	Configure the ESDT to be inserted into Data Pool but not public area, and ingest another 5 granules.		
30	Prepare a QA update request which updates the science quality flags of the specific measured parameters of the ESDT for a temporal range.  See the QAUU 609 section “4.8.9.3 Input File Format” for the format of Request Range:  <a href="http://edhs1.gsfc.nasa.gov/waisdata/eed/pdf/609eed001_File_6.pdf">http://edhs1.gsfc.nasa.gov/waisdata/eed/pdf/609eed001_File_6.pdf</a>  The measured parameters should be specified in the request file for this test.		
31	Specify some granules to have acquisition times at the exact boundary points of the temporal range.		
32	Specify the email address in the input request file.		

#	Action	Expected Result	Notes
33	Put "From hliu@raytheon.com" as the first line of the request file for the request from LDOPE.		
34	Ensure that LDOPE_EMAIL_REPLY_ADDRESSES=hliu@eos.hitc.com is in the config file		
35	Specify science QA flag and explanations for at least 2 measured parameters.		
36	Name the QA Update Request file name with the following format:  &lt;MODE&gt;&lt;SITE&gt;_QAUPDATE.&lt;timestamp&gt; (example OPS_LDOPE_QAUPDATE.20010228122030)		
37	Copy the request file to QA_REQUEST_DIR directory.		
38	Run query SearchWithTemporalRange.ksh for each entry in the request file. Save the query results.		
39	Locate the granule XML files for the affected granules in the small file archive under /stornext/smallfiles/&lt;mode&gt;/metadata/&lt;ShortName.VersionID&gt;, and copy the XML files to a temporary location for later comparison.		
40	Check if there is a pre-created QAUU log file under /usr/ecs/&lt;mode&gt;/CUSTOM/logs, and remove it if any.		
41	Run EcAmQAUUStart &lt;MODE&gt; -file &lt;QAUpdate Request File&gt;		
42	Verify the total number of affected granules is displayed and correct by comparing with the number from the query in step 3, and confirm the updating after the prompt.		
43	Verify that the QAUU history file under /usr/ecs/&lt;mode&gt;/CUSTOM/data/DSS/AIM/QAUU/history contains the information about the affected granules and the log file contains total number of granules updated.		
44	Verify that the granules listed in QAUU history file are consistent with the granules returned from the query in step 3, and the granule count is correct		
45	Locate the XML files for the affected granules in /stornext/smallfiles/&lt;mode&gt;/metadata/&lt;ShortName.VersionID&gt;, compare with the original XML files using diff, and verify that the QA flags and explanations are updated correctly for the affected measured parameters.		
46	Locate the XML files for the affected granules in public data pool /datapool/&lt;MODE&gt;/user/&lt;FILE_SYSTEM&gt;/, and verify they are the same as the ones in the small file archive using diff.		
47	Locate the XML files for the affected granules in hidden data pool area /datapool/&lt;MODE&gt;/user/&lt;FILE_SYSTEM&gt;/orderdata, and		

---

---

#	Action	Expected Result	Notes
	verify they are NOT updated by comparing with original XML files using diff and time stamps.		
48	View the updated XML files, and verify the ScienceFlagExplanation field has time tag accurate to the minute appended for the updated QAFlags.		
49	Query the Data Pool database GetDplMeasuredParameter.ksh &lt;dbID&gt; to verify the ScienceFlagExplanation field has time tag appended for the updated QAFlags		
50	Query the Data Pool database GetDplMeasuredParameter.ksh &lt;dbID&gt; to verify the QA flags and Explanations are correctly updated.		
51	Query the Inventory Database GetInDbLastUpdate.ksh &lt;dbID&gt; and the Data Pool database GetDplLastUpdate.ksh &lt;dbID&gt;, and verify that the lastUpdate is updated using the system time.		
52	Query the Inventory Database GetInDbParamUpdHistory.ksh &lt;dbID&gt; and verify that the table recorded the information about the QA updates of the granules.		
53	Verify the QA Update Request file is moved from directory QA_REQUEST_DIR to QA_COMPLETED_REQUEST_DIR.		
54	Verify the history file is created under QA_HISTORY_DIR, and it contains the dbID, ParameterName, FlagName, OldQualityFlag, OldQualityFlagExplan, NewQualityFlag and NewQualityFlagExplan		
55	Verify the QA update utility log recorded the processing activities.		
56	Verify the email notification is sent to email addresses specified in DAAC_EMAIL_ADDRESSES and LDOPE_ReplyAddress.		
57	Verify the email contains the correct information.		
58	Verify the QA Update Utility log file is created under /usr/ecs/&lt;MODE&gt;/CUSTOM/logs.		

TEST DATA:

EXPECTED RESULTS:

2.1.3 XML Replacement Utility:Successfully replace a granule metadata file (ECS-ECSTC-477)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[XML Replacement Utility:Successfully replace a granule metadata file]</i>		#comment
2	Identify a granule metadata file to replace.  Query DsMdXMLFile and DsMdXMLPath in the inventory database for a metadata file.  This should ensure us that the ESDT is installed and the xml schema is in the correct location.		
3	Go to the destination location and verify that the queried file exists and the permission allows cmshared to read and write. Also, open up the file to ensure the file is good.		
4	Copy the metadata file out and put it in a temporary place.  IMPORTANT: Do not rename the copied out file, it must have the same name as the one in the destination directory		
5	Modify the copied out metadata file. Use the xml schema as a guide to help you select the attribute to modify.		
6	Go to the mode's utilities directory and execute EcDsAmXruStart - xmlfile &lt;absolute path of the modified metadata file>;		
7	<i>Verification:</i>		#comment
8	Verify that the modified file is removed from the source location		
9	Verify that the destination location contains the modified metadata file.		
10	Verify that the lastUpdate attribute is updated to a time close to when the replacement was executed.		
11	Verify that the granule's lastUpdate time in DsMdGranules is updated.		
12	Verify that a GRUPDATE event is inserted for the granule into the DsMdGrEventHistory table.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

### 2.1.4 On-line Archive Repair from non-resident tape: DP\_7F\_01 Criterion 320 (ECS-ECSTC-478)

#### DESCRIPTION:

#### PRECONDITIONS:

#### STEPS:

#	Action	Expected Result	Notes
1	<i>[On-line Archive Repair from non-resident tape: DP_7F_01 Criterion 320] Create phantoms in the public and hidden Data Pool that satisfy the following conditions:  a. Turn at least ten (10) public science granules into phantoms including MISR, MODIS and ASTER granules by removing their science and XML files. At least one of these granules shall be in use for order purposes; remove its hidden links, as well.  b. At least one but not all of the science granules shall have a browse that is present in the public Data Pool.  c. At least two of the phantom science granules with browse shall also be missing their browse links from the Data Pool directories, including a MISR and a non-MISR granule.  d. One of the science granules shall have experienced a file name collision in the hidden Data Pool, that is, its file names shall have been suffixed during Data Pool insert.</i>		#comment
2	Also turn at least five (5) public Browse granules, including MISR, MODIS and ASTER browse into phantoms by removing their browse files from the public Data Pool.		
3	Save the removed XML and browse files for comparison during the test.		
4	Submit an on-line archive repair for these granules via the command line using the RestoreOlaFromTape utility, specifying the granules to be repaired in an input file; and request restoration of browse links that may be missing for granules whose files were restored (using the –restorelinks command line parameter).		
5	Ensure that the test requires access to granules on a tape that is resident in the silo as well as a tape that is not resident in the silo.		
6	Verify that all science granules that resided on tapes resident in the silo were repaired, i.e., their files are now in the correct public Data Pool locations and have the original file names		
7	Verify that the science granules that resided on tapes that were not resident in the silo were not repaired.		
8	Verify that the utility logs each of the granules that were not repaired, the reason for not repairing it, and the tape label.		

#	Action	Expected Result	Notes
9	Verify that the utility exits with an exit code indicating the occurrence of errors.		
10	Verify that the utility sends a notice to the specified e-mail address indicating that a repair failed due to error, the nature of the repair, and the name and location of the log file.		
11	Make the missing tape resident in the silo and re-run the repair for the skipped granules. Verify that the remaining granules are now repaired		
12	Verify that the contents of all the restored XML files (including those referenced in V-6) match the contents of the original XML files.		
13	Verify that all browse granules were repaired and are present as jpeg files in their original disk location and with their original file names.		
14	Verify that the repaired jpeg files are identical to the original jpeg files.		
15	Verify that the missing browse links are restored.		
16	Verify that the hidden links required by the public granules in condition (a) that are also referenced by orders were restored.		
17	Verify that any files that were copied from the ECS tape archive into the Data Pool during the repair were checksummed and that the checksum is recorded in the Data Pool inventory.		
18	Verify that any files that were copied from the ECS browse archive into the Data Pool during the repair had their checksum verified.		

TEST DATA:

EXPECTED RESULTS:

## 2.2 BMGT

### 2.2.1 Nominal Collection Export[S-1]: Manual Export (ECS-ECSTC-485)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	#S-1 [Nominal Collection Export] Find two collections with ECS metadata and which share the same short name, but have different version IDs. Request the manual export of collection metadata for one of these collections.		
2	Ensure collections C1 and C2 enabled for collection and granule export: update bg_collection_configuration set collectionexportflag = 'Y', granuleexportflag = 'Y' where shortname = and versionid in ( , ); *Ensure collections C1 and C2 enabled for collection and granule export: update bg_collection_configuration set collectionexportflag = 'Y', granuleexportflag = 'Y' where shortname = and versionid in ( , );		
3	Request manual export of collection metadata for collection C1: EcBmBMGTManualStart --metc --collections .		
4	#V-1 Verify that the manual export in S-1 results in a single HTTP PUT request containing the full collection metadata for the requested collection (but no other collections sharing a short name but with different version IDs).		
5	Verify the bg_export_error table shows no errors referring to collection C1 on or after the time of the manual export.		
6	Verify that the TCP proxy log reports a single HTTP PUT request for collection C1.		
7	Verify that the HTTP PUT request contains collection C1's full collection metadata.		
8	Verify the TCP proxy log reports no PUTs for collection C2		
9	#V-4 Verify that the metadata exported in S-1 and S-2, except S-2 subclause b, validates against the ECHO collection metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Collection.xsd">https://api.echo.nasa.gov/ingest/schema/Collection.xsd</a> ).		
10	Verify collection C1's exported metadata validates against the ECHO 10 collection schema. ( <a href="https://api.echo.nasa.gov/ingest/schema/Collection.xsd">https://api.echo.nasa.gov/ingest/schema/Collection.xsd</a> ). xmllint --noout --schema Collection.xsd collection.xml		
11	#V-5 Verify that the metadata exported in S-1 and S-2, except S-2 subclause b has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the collection recorded in the AIM database. d) LastUpdate = The last update time of the collection recorded in the AIM database		
12	Use an xpath utility to verify the exported metadata has the following elements.		
13	a) Visible = true xpath /Collection/Visible C1.xml	true	
14	b) Orderable = false xpath /Collection/Orderable C1.xml	false	

#	Action	Expected Result	Notes
15	c) InsertTime = The insert time of the collection recorded in the AIM database. xpath /Collection/InsertTime C1.xml	where the date time string returned is equal to what is returned from the query to the aim_ schema in the ecs database: select inserttime from amcollection where shortname = and versionid = ;	
16	d) LastUpdate = The last update time of the collection recorded in the AIM database. xpath /Collection/LastUpdate C1.xml	where the date time string returned is equal to what is returned from the query: select lastupdate from amcollection where shortname = and versionid = ;	
17	#V-6 Verify that the metadata exported in S-1 and S-2 contains version numbers with no leading zeroes.		
18	Verify that collection C1's exported metadata contains version numbers with no leading zeroes: xpath /Collection/VersionId C1.xml	#{VERSIONID}	

**TEST DATA:**

Crit id	Crit ccr no	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location	Readiness Status
10-S1			2 Collections with same shortname, different versionids (C1, C2)	MOD44W.005 MOD44W.006		2 collections	/sotestdata/DROP_802/BE_82_01/Criteria/010/010_1	

**EXPECTED RESULTS:**

You should get one scenario passed.

2.2.2 Nominal Granule Export[S-2a]: Automatic Export: Ingest Granule (ECS-ECSTC-487)

DESCRIPTION:

S	40	1	<b>[Nominal Granule Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections.		
S	40	2	For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.		

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure granule g1's collection is installed. E.g., the DPL Ingest GUI shows the collection as a configured datatype.		
3	Ensure the DPL Ingest GUI shows granule g1's collection is configured to be public on ingest.		
4	Ensure granule g1's collection is enabled for collection and granule export:  update bg_collection_configuration set		

#	Action	Expected Result	Notes
	granuleexportflag = 'Y', collectionexportflag = 'Y' &lt;SHORTNAME&gt;&lt;VERSIONID&gt;;		
5	Ensure ECHO has granule g1's collection metadata: curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/\${URL_ENCODED_DATASETID} If ECHO is missing the collection, export it: EcBmBMGTManualStart \${MODE} --metc --collections \${SHORTNAME}.\${VERSIONID}		
6	Ensure a local copy of the ECHO 10 schema files is available for schema validation (e.g., under /tools/common/test/BE_82_01/echo10/schema). If needed, download *.xsd files from https://api.echo.nasa.gov/ingest/schema/.		
7	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
8	Ensure the BMGT Dispatcher is running. On the BMGT host: ps auxww   grep \$MODE   grep EcBmDispatcher If needed, start the dispatcher: ./EcBmBMGTDispatcherStart \$MODE		
9	Ensure the BMGT Auto driver is running. On the BMGT host: ps auxww   grep \$MODE   grep EcBmAuto If needed, start the Auto driver: ./EcBmBMGTAutoStart \$MODE		
10	<i>Setup</i>		#comment
11	<i>S-2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory.</i>		#comment
12	Ingest a new granule g1 into the ECS inventory.		
13	<i>Verification</i>		#comment
14	<i>V-2 Verify that the operations in S-2, except subclauses b – d, each result in the export of one or more HTTP PUTs containing the full granule metadata.</i>		#comment
15	<i>NOTE: Each action may result in multiple distinct events being triggered. The BMGT Auto driver will consolidate these events if they are picked up within the same polling interval. If they span multiple intervals, then multiple, redundant exports could result.</i>		#comment
16	Verify the TCP proxy log one or more PUTs for granule g1. There could be 1 PUT after the granule is archived and another when the granule is published.		
17	Verify the TCP proxy log shows that each of granule g1's PUT requests contains the full granule metadata.		

#	Action	Expected Result	Notes
18	<i>V-4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).</i>		#comment
19	Verify granule g1's exported metadata validates against the ECHO granule metadata schema: xmllint --noout --schema /path/to/Granule.xsd g1.xml or /tools/common/test/BE_82_01/bin/echo10_validate_metadata g g1.xml		
20	<i>V-5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements:  a) Visible = true  b) Orderable = true  c) InsertTime = The insert time of the granule recorded in the AIM database.  d) LastUpdate = The last update time of the granule recorded in the AIM database</i>		#comment
21	<i>Verify that the exported metadata has the following elements, using the Ruby script /tools/common/test/BE_82_01/bin/xpath:</i>		#comment
22	a) Visible = true  xpath /Granule/Visible g1.xml	&lt;Visible&gt;true&lt;/Visible&gt;	
23	b) Orderable = true  xpath /Granule/Orderable g1.xml	&lt;Orderable&gt;true&lt;/Orderable&gt;	
24	c) InsertTime = The insert time of the granule recorded in the AIM database  xpath /Granule/InsertTime g1.xml	&lt;InsertTime&gt;YYYY-MM-DDTHH:mm:SSZ&lt;/InsertTime&gt;  where the date time string return is equal to what is returned from the query to the database  select archivetime from amgranule where granuleid = &lt;GRANULEID&gt;;	
25	d) LastUpdate = The last update time of the granule recorded in the AIM database.  xpath /Granule/LastUpdate g1.xml	&lt;LastUpdate&gt;YYYY-MM-DDTHH:mm:SSZ&lt;/LastUpdate&gt;  where the date time string returned is equal to what is returned from the query  select lastupdate from amgranule where granuleid = &lt;GRANULEID&gt;;	

TEST DATA:  
See Test Case 603.

EXPECTED RESULTS:

V	40	1	Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID).		
V	40	2	Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.		
V	40	3	Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.		
V	40	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).		
V	40	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database		
V	40	6	Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255.		
V	40	7	Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule.		
V	40	8	Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent.		
V	40	9	Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed.		
V	40	10	Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule.		
V	40	11	Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs.		

	V	40	12	Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs.		
	V	40	13	Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs.		
	V	40	14	Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection.		
	V	40	15	Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected.		
	V	40	16	Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.		

### 2.2.3 Nominal Collection Export[S-2a]: Automatic Export: Insert Collection (ECS-ECSTC-490)

#### DESCRIPTION:

	S	10	1	<b>[Nominal Collection Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Request the manual export of collection metadata for one of these collections.		
	S	10	2	Find collections which have ECS Metadata: a) Insert a new collection into the ECS inventory (and enable for collection export). b) Delete a collection from the ECS inventory. c) Update an existing collection. d) For a collection which is currently disabled for collection export, enable it for collection (but not granule) export.		

#### PRECONDITIONS:

#### STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure BMGT configuration is current and correct (config files, properties files, database settings, etc.).		

#	Action	Expected Result	Notes
3	Ensure a PostgreSQL prompt is available and connected to the ecs database. <code>/tools/postgres/current32/bin/psql -U \$USERNAME -h f4dbl03 -d ecs</code> <code>select public.set_search_path('AIM', MODE);</code> <code>show search_path;</code>	Should list the schemas installed in the mode. Should include 'aim_&lt;MODE&gt;'	
4	Ensure the test collection descriptor file is under <code>/sotestdata/DROP_802/BE_82_01/Criteria/010/010_2_A</code> .		
5	Ensure the ESDT Maintenance GUI shows collection C2 is not installed.		
6	Ensure the BMGT dispatcher and auto driver are up and running: <code>./EcBmBMGTAppStart &amp;lt;MODE&amp;gt;</code> <code>select propertyvalue</code> <code>from bg_configuration_property</code> <code>where propertyname = 'BMGT.Dispatcher.Running';</code> propertyvalue should be true; On the BMGT host <code>ps -ef   grep &amp;lt;MODE&amp;gt;   grep EcBmDispatcher</code> should show a single dispatcher process running. On the BMGT host <code>ps -ef   grep &amp;lt;MODE&amp;gt;   grep EcBmAuto</code> should show a single auto driver process running		
7	Ensure a local copy of the ECHO 10 schema files is available for validation (from <a href="https://api.echo.nasa.gov/ingest/schema">https://api.echo.nasa.gov/ingest/schema</a> ).		
8	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
9	<i>Setup</i>		#comment
10	<i>S-2 Find collections which have ECS Metadata:</i> a) Insert a new collection into the ECS inventory (and enable for collection export). <i></i>		#comment
11	Install collection C2 into ECS. (Use the ESDT Maintenance GUI, the DPL Maintenance GUI, a DB patch from the command line, and the DPL Ingest GUI.)		
12	Use populate script EcBgPopulateCollections.ksh or a modified populate script to check if the collection does not exist and insert a row in the bg_collection_configuration table. IF NOT EXISTS ( SELECT collectionid from bg_collection_configuration where shortname = &lt;C2_ShortName&gt; and versionid = &lt;C2_VersionId&gt;) THEN insert into bg_collection_configuration (collectionid, shortname, versionid, longname, granuleexportflag, collectionexportflag) select collectionid, shortname, versionid, longname, 'Y' as granuleexportflag, 'Y' as collectionexportflag from amcollection where shortname = &lt;C2_ShortName&gt; and VersionId = &lt;C2_VersionId&gt;; -- update the other fields based on esdt type ELSE -- Enable Collection C2 for Collection and Granule Export update bg_collection_configuration set granuleexportflag = 'Y' ,		

#	Action	Expected Result	Notes
	collectionexportflag = 'Y' where shortname = &lt;C2_ShortName&gt; and versionid = &lt;C2_VersionId&gt;;		
13	<i>V-2 Verify that the operations in S-2 subclauses a, c, and d each result in the export of a single HTTP PUT containing the full collection metadata.</i>		#comment
14	Verify that the TCP proxy log shows a single HTTP PUT request for collection C2.		
15	Verify that the HTTP PUT request contains collection C2's full metadata.		
16	<i>V-4 Verify that the metadata exported in S-1 and S-2, except S-2 subclause b, validates against the ECHO collection metadata schema (https://api.echo.nasa.gov/ingest/schema/Collection.xsd).</i>		#comment
17	Verify that collection C2's exported metadata validates against the ECHO 10 collection schema: xmllint --noout --schema Collection.xsd C2.xml		
18	<i>V-5 Verify that the metadata exported in S-1 and S-2, except S-2 subclause b has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the collection recorded in the AIM database. d) LastUpdate = The last update time of the collection recorded in the AIM database.</i>		#comment
19	<i>Use an xpath utility to verify the exported metadata has the following elements.</i>		#comment
20	a) Visible = true xpath /Collection/Visible C2.xml	&lt;Visible&gt;true&lt;/Visible&gt;	
21	b) Orderable = false xpath /Collection/Orderable C2.xml	&lt;Orderable&gt>false&lt;/Orderable&gt;	
22	c) InsertTime = The insert time of the collection recorded in the AIM database. xpath /Collection/InsertTime C2.xml	&lt;InsertTime&gt;&lt;YYYY-MM-DDTHH:mm:SSZ&lt;/InsertTime&gt; where the date time string returned is equal to what is returned from the query to the aim_&lt;MODE&gt; schema in the ecs database: select inserttime from amcollection where shortname = &lt;C2_ShortName&gt; and versionid = &lt;C2_VersionId&gt;;	
23	d) LastUpdate = The last update time of the collection recorded in the AIM database. xpath /Collection/LastUpdate C2.xml	&lt;LastUpdate&gt;&lt;YYYY-MM-DDTHH:mm:SSZ&lt;/LastUpdate&gt;	

#	Action	Expected Result	Notes
		; where the date time string returned is equal to what is returned from the query:  select lastupdate from amcollection where shortname = &lt;C2_ShortName&gt; and versionid = &lt;C2_VersionId&gt;;	
24	<i>V-6 Verify that the metadata exported in S-1 and S-2 contains version numbers with no leading zeroes.</i>		#comment
25	Verify that collection C2's exported metadata contains version numbers with no leading zeroes:  xpath /Collection/VersionId C2.xml	&lt;VersionId&gt;\${VERSIONID}&lt;/VersionId&gt;	

TEST DATA:

Crit id	Crit ccr no	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location	Readiness Status
10	S-2	Install and enable new collection for automatic export			1 Collection to be installed (C2)		/sotestdata/DROP_802/BE_82_01/Criteria/010/010_2_A	

EXPECTED RESULTS:

	V	10	1	Verify that the manual export in S-1 results in a single HTTP PUT request containing the full collection metadata for the requested collection (but no other collections sharing a short name but with different version IDs).		
	V	10	2	Verify that the operations in S-2 subclauses a, c, and d each result in the export of a single HTTP PUT containing the full collection metadata.		
	V	10	3	Verify that the operation in S-2 subclause b results in the export of a single HTTP DELETE, with the ID of the collection in the URL, but containing no collection metadata in the request body.		

---

---

V	10	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclause b, validates against the ECHO collection metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Collection.xsd">https://api.echo.nasa.gov/ingest/schema/Collection.xsd</a> ).		
V	10	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclause b has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the collection recorded in the AIM database. d) LastUpdate = The last update time of the collection recorded in the AIM database.		
V	10	6	Verify that the metadata exported in S-1 and S-2 contains version numbers with no leading zeroes.		

#### 2.2.4 Configuring Collections For Export (ECS-ECSTC-491)

DESCRIPTION:

S	280	1	<b>[Configuring Collections For Export]</b> Add a new ESDT to the ECS system and ingest some granules in that collection.		
S	280	2	Go to the GUI collection configuration page.		
S	280	3	Enable the collection for collection export.		
S	280	4	Allow the export of the collection metadata to complete.		
S	280	5	Enable the collection for granule export.		
S	280	6	Find a collection which is not enabled for collection or granule export. Request the manual export of collection and granule metadata for this collection.		
S	280	7	Find a collection which is enabled for collection, but not granule export. Request the manual export of collection and granule metadata for this collection.		
S	280	8	Find a collection which is enabled for collection and granule export. Request the manual export of collection and granule metadata for this collection.		

	S	280	9	Find a collection which is not enabled for collection or granule export, but for which there is another ESDT with the same short name, but different version, which is. Request the manual export of collection and granule metadata for this collection.		
--	---	-----	---	---	--	--

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure collection C1 is not installed.		
3	Ensure bg_collection_configuration has no row for collection C1.		
4	Ensure granules g1, g2, g3 are not in AIM.		
5	Ensure collections C2, C3, C4, C5, C6 are installed.		
6	Ensure granules g4 ... g18 are in AIM		
7	Ensure collection C2 is disabled for both collection and granule export.		
8	Ensure collection C3 is enabled for collection but not for granule export.		
9	Ensure collection C4 is enabled for both collection and granule export.		
10	Ensure collection C5 is disabled for collection or and granule export.		
11	Ensure collection C6 is enabled for both collection and granule export.		
12	Ensure a TCP proxy or mock ECHO is capturing BMGT HTTP traffic.		
13	<i>Setup and Verification</i>		#comment
14	<i>S-1 Add a new ESDT to the ECS system and ingest some granules in that collection.</i>		#comment
15	Install collection C1.		
16	Add a row for collection C1 to the bg_collection_configuration table, setting both export flags to 'N'.  See EcBgPopulateCollections.ksh for examples.		
17	Ingest granules g1, g2, g3.		
18	<i>S-2 Go to the GUI collection configuration page.</i>		#comment
19	Visit the BMGT GUI Collection Configuration page.		
20	<i>V-1 On the collection configuration page in S-2, verify that all currently installed collections are listed with their current enabled/disabled status.</i>		#comment
21	Verify the BMGT GUI's collection configuration page lists collections C1 ... C6.		

#	Action	Expected Result	Notes
22	Verify the BMGT GUI indicates C1 is disabled for both collection and granule export.		
23	Verify the BMGT GUI indicates C2 is disabled for both collection and granule export.		
24	Verify the BMGT GUI indicates C3 is enabled collection export but disabled for granule export.		
25	Verify the BMGT GUI indicates C4 is enabled for both collection and granule export.		
26	Verify the BMGT GUI indicates C5 is disabled for both collection and granule export.		
27	Verify the BMGT GUI indicates C6 is enabled for both collection and granule export.		
28	<i>V-2 On the collection configuration page in S-2, verify that the new collection is listed but not enabled for collection or granule export.</i>		#comment
29	Verify collection C1 appears on the BMGT GUI's collection configuration page.		
30	Verify the BMGT GUI indicates C1 disabled for both collection and granule export.		
31	<i>S-3 Enable the collection for collection export.</i>		#comment
32	Enable C1 for collection export by clicking the collection export check box next to the collection.		
33	<i>V-3 After enabling the collection for collection export in S-3, verify that the collection metadata for the collection is automatically exported.</i>		#comment
34	Verify that after C1 is enabled for collection export, the BMGT GUI indicates C1's metadata is successfully exported.		
35	Verify that after C1 is enabled for collection export, a BMGT log records the export of C1's metadata.		
36	Verify that after C1 is enabled for collection export, the TCP proxy logs an HTTP PUT whose body contains C1's metadata.	The request should begin with a line such as  PUT /catalog-rest/providers/\${PROVIDERID}/datasets/\${DATASETID} HTTP/1.1  where \${DATASETID} is the URL-encoded longname + ' V' + versionid, e.g.,  />MODIS%2FAqua%20Gross%20Primary%20Productivity%208-	

#	Action	Expected Result	Notes
		Day%20L4%20Global%201km%20SI N%20Grid%20V005	
37	<i>S-4 Allow the export of the collection metadata to complete.</i>		#comment
38	Wait for collection C1's export request to be marked complete:  select r.completiontime, r.status from bg_export_request r join amcollection c on c.colelectionid = r.collectionid where c.ShortName = '&lt;SHORT_NAME&gt;'; and c.VersionId = &lt;VERSION_ID&gt;;   Or, wait for the TCP proxy to log C1's HTTP PUT request.		
39	<i>S-5 Enable the collection for granule export.</i>		#comment
40	In the BMGT GUI, enable C1 for granule export by checking the granule export check box next to the collection in the collection configuration tab.		
41	<i>V-4 After enabling the collection for granule export in S-5, verify that the granule metadata for every granule in the collection is automatically exported.</i>		#comment
42	Verify that after C1 is enabled for granule export, the BMGT GUI indicates all metadata for all granules belonging to C1 is successfully exported (except logically deleted granules).		
43	Verify that after C1 is enabled for granule export, a BMGT log records the export of C1 granules' metadata (except logically deleted granules).		
44	Verify that after C1 is enabled for granule export, the TCP proxy logs an HTTP PUT for each of C1's granules, containing granule metadata (except logically deleted granules).	Each granule request should begin with a line such as  PUT /catalog- rest/providers/\${PROVIDERID}/gran ules/\${GRANULEUR} HTTP/1.1<br  where \${GRANULEUR} is a URL-encoded geoid, such as   SC%3AMYD17A2.005%3A20062	
45	<i>S-6 Find a collection which is not enabled for collection or granule export. Request the manual export of collection and granule metadata for this collection.</i>		#comment
46	Request manual export of C2:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C2_SHORT_NAME&gt;.&lt;C2_VERSION_ID&gt;		
47	<i>V-5 For the export attempt in S-6, verify that nothing is exported.</i>		#comment
48	Verify the TCP proxy logs no request for C2's collection or granule metadata.		

#	Action	Expected Result	Notes
49	Verify a BMGT log indicates no request is attempted for C2's collection or granule metadata.		
50	<i>S-7 Find a collection which is enabled for collection, but not granule export. Request the manual export of collection and granule metadata for this collection.</i>		#comment
51	Request manual export of C3:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C3_SHORT_NAME&gt;.&lt;C3_VERSION_ID&gt;		
52	<i>V-6 For the export attempt in S-7, verify that only collection metadata is exported.</i>		#comment
53	Verify the TCP proxy logs a single HTTP PUT request, containing C3's collection metadata.  Get the request ID for the collection export from the GUI. Look in the BMGT manual log for pattern like  &quot;requestId&quot;:18629,&quot;batchId&quot;;76		
54	Verify the TCP proxy logs no HTTP PUT requests for C3's granules.		
55	Verify a BMGT log records a single export request for C3's collection metadata.		
56	Verify a BMGT log records no export attempts for any C3 granule metadata.		
57	<i>S-8 Find a collection which is enabled for collection and granule export. Request the manual export of collection and granule metadata for this collection.</i>		#comment
58	Request manual export of C4:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C4_SHORT_NAME&gt;.&lt;C4_VERSION_ID&gt;		
59	<i>V-7 For the export attempt in S-8, verify that both collection and granule metadata is exported.</i>		#comment
60	Verify the TCP proxy logs a single HTTP PUT request, containing C4's collection metadata.		
61	Verify the TCP proxy logs a single HTTP PUT request for each of C4's granules, containing granule metadata (excepting any granules which are logically deleted).		
62	Verify a BMGT log records a single export request for C4's collection metadata.		
63	Verify a BMGT log records a single export request for each of C4's granules (excepting those which are logically deleted).		
64	<i>S-9 Find a collection which is not enabled for collection or granule export, but for which there is another ESDT with the same short name, but		#comment

#	Action	Expected Result	Notes
	different version, which is. Request the manual export of collection and granule metadata for this collection.</i>		
65	Request manual export of C5:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C5_SHORT_NAME&gt;,&lt;C5_VERSION_ID&gt;		
66	<i>V-8 For the export attempt in S-9, verify that nothing is exported.</i>		#comment
67	Verify the TCP proxy logs no request for C5's collection or granule metadata.		
68	Verify a BMGT log indicates no request is attempted for C5's collection or granule metadata.		

**TEST DATA:**

Crit id	Crit ccr no	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location	Readiness Status
280				6 collections (C1 ... C6)			/sotestdata/DROP_802/BE_82_01/Criteria/280	
280				3 granules for each collection (g1 ... g18)			/sotestdata/DROP_802/BE_82_01/Criteria/280	

**EXPECTED RESULTS:**

	V	280	1	On the collection configuration page in S-2, verify that all currently installed collections are listed with their current enabled/disabled status.		
	V	280	2	On the collection configuration page in S-2, verify that the new collection is listed but not enabled for collection or granule export.		
	V	280	3	After enabling the collection for collection export in S-3, verify that the collection metadata for the collection is automatically exported.		
	V	280	4	After enabling the collection for granule export in S-5, verify that the granule metadata for every granule in the collection is automatically exported.		

	V	280	5	For the export attempt in S-6, verify that nothing is exported.		
	V	280	6	For the export attempt in S-7, verify that only collection metadata is exported.		
	V	280	7	For the export attempt in S-8, verify that both collection and granule metadata is exported.		
	V	280	8	For the export attempt in S-9, verify that nothing is exported.		

### 2.3 CLS

#### 2.3.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-493)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request the status of a single, completed order that was handled by the OMS and that contains at least two requests by specifying a valid Order Id and user contact email address.  Each request in the order should contain at least 100 granules that have NOT been processed by HEG or the external subsetter.		
2	<i>Verify the following:</i>		#comment
3	a. Order Id is correctly displayed.		
4	b. Submission date/time is correctly displayed.		
5	c. Order state is correctly displayed.		
6	d. Order completion date/time is correctly displayed.		
7	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed		
8	f. Order state and request states are presented in terms that an end user can understand.		
9	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
10	h. There is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

### 2.3.2 Order Status GUI: Display Order Status based on History Range, 0D\_S6\_05, Criterion 180 (ECS-ECSTC-494)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request order history by specifying a starting and ending date, user contact email address, and one valid Order Id for the user.  Ensure that at least 100 orders are displayed that represent a mix of orders for which request details are available for some orders and not available for others.  Ensure that the date range requires the Order Status Interface to retrieve information from both the Order Manager operational tables and archive tables.  Ensure that at least one of the orders contains granules that were processed by the external subsetter and one of the orders contains granules that were processed by HEG.		
2	Verify that the correct orders are returned and sorted by submission date/time.		
3	<i>For each order verify the following:</i>		#comment
4	a. Order Id is correctly displayed.		
5	b. Submission date/time is correctly displayed.		
6	c. Order state is correctly displayed.		
7	d. Order completion date/time is correctly displayed.		
8	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed, including the processing description for the granule to be processed by the external subsetter, in a manner which should be generally understandable by the user.		
9	f. Order state and request states are presented in terms that an end user can understand.		

#	Action	Expected Result	Notes
10	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
11	h. An indication is provided when request details are not available for an order.		

TEST DATA:

EXPECTED RESULTS:

## 2.4 DPL

### 2.4.1 Publish with Theme (ECS-ECSTC-479)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Publish with Theme]</i>		#comment
2	Ingest ICEBRIDGE data. First verify that the ThemeID value is not in the themeID column of AmCollection table for the collection so that the theme association would not happen upon ingest.		For example IRPAR2.001
3	Use the -theme option to associate these granules with an existing theme. (Use the Data Pool Maintenance GUI Manage Themes tab to see the list of existing themes.)		
4	Verify that the granules were successfully inserted into the Data Pool by checking the status of the insert actions in the Data Pool database (DIInsertActionQueue).		
5	Verify that the granules were inserted into the AmGranule table.		
6	Verify, using Unix cd and ls commands, that the files for the granules were inserted into the appropriate Data Pool directories.  select fs.absoluteFileSystemPath + c.GroupId + df.DirectoryPath + df.OnlineFileName from DIFileSystems fs join AmCollection c on fs.fileSystemLabel = c.FileSystemLabel join AmGranule		

#	Action	Expected Result	Notes
	g on c.CollectionId = g.CollectionId join AmDataFile df on g.GranuleId = df.GranuleId where g.GranuleId in (&lt;GranuleIds&gt;)		
7	Verify that the granules are associated with the specified theme, i.e., that appropriate rows for the granules have been inserted in the DIGranuleThemeXref table.		

**TEST DATA:**

Any nonECS data type

**EXPECTED RESULTS:**

2.4.2 Run DPCV against DPL, Provide a List of Granule IDs : CK\_7F\_01, Criterion 540 (ECS-ECSTC-480)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Run DPCV against DPL, Provide a List of Granule IDs : CK_7F_01, Criterion 540]</i>		#comment
2	Select several granules (at least 10) from Data Pool that belong to at least two ESDTs.		
3	Manually alter the checksum values in the Data Pool database for at least 2 granule files.		
4	Set the Last Verification time to null for 2 other granules.		
5	Set the Checksum Verification Status to “failed” for 2 granules.		
6	Run DPCV against Data Pool providing the list of granule IDs as input.		
7	Verify that the DPCV run completed successfully.		
8	Verify that DPCV performed checksum verification for all data files that belong to the list of granules in S-540-1.		
9	Verify that DPCV checksum verification was successful for those files whose checksum values were not altered in S-540-1. Verify the following in DPL database: Checksum time was updated correctly for each affected file Checksum origin was updated as “DPCV” for each affected file		The “Checksum origin was updated as “DPCV” for each affected file that had a null least checksum

#	Action	Expected Result	Notes
	that had a null last checksum verification time.  Checksum verification status was set to a success status.		verification time.&quot;; should be removed
10	Verify that DPCV failed checksum verification for those files whose checksums have been altered as described in S-540-2. Verify the following in DPL database:  Checksum verification status was set to a failure status.		
11	Verify that DPCV logs an error message for each file that failed checksum verification, and that the error message includes information specified in the L4 requirement S-DPL-49180.		
12	Verify that the Last Verification time and status for the granules modified in S-540-3 was populated.		
13	Verify that the verification status was set to “success” for the granules modified in S-540-4.		
14	Verify that DPCV logs an error message for each file that failed checksum verification, and that the error message includes information specified in the L4 requirement S-DPL-49180 and S-DPL-49190.		

**TEST DATA:**

any current collection group

**EXPECTED RESULTS:**

**2.5 DPL Ingest**

**2.5.1 Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-481)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Data Providers can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[View Data Providers]</i>		#comment

#	Action	Expected Result	Notes
3	As the 'ingest admin' operator, navigate to the Data Provider page on the Data Pool Ingest GUI.		
4	Verify that all provider information in the INGST CI database appears correctly on the Data Provider page.		
5	<i>[Configure Data Providers]</i>		#comment
6	Edit existing provider information (if any) and define new providers to fulfill the following requirements:  At least one data provider should have an FTP notification method,  at least one data provider should have an scp notification method,  at least one data provider should have an email notification method,  at least one data provider should have a combination FTP/email notification method,  and at least one data provider should have a combination scp/email notification method. Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des. At least one provider must use active mode. At least one should use passive mode.		
7	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit data provider information.		
8	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing data providers before saving this information.		
9	Verify that the ingest admin operator has the authorization to define new data providers (i.e., has access to the data provider pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
10	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16110 to be entered.		
11	Verify that all of the notification methods in S-DPL-16150 can be entered or selected on the Data Pool Ingest GUI, as appropriate for the selected transfer method as per S-DPL-16110.		
12	For one provider with an FTP Notification method, verify that the Data Pool Ingest GUI allows the related FTP information to be entered (FTP host, destination directory, login id, password)		
13	Verify that the FTP password entered is not shown or stored in the clear.		
14	Verify that the FTP password entered is not shown in the Data Pool Ingest GUI log.		
15	For one provider with an email notification method, verify that the Data Pool Ingest GUI allows the related email address to be entered.		

---

---

#	Action	Expected Result	Notes
16	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new data providers before saving this information.		
17	For one provider with an scp notification method, verify that the Data Pool Ingest GUI allows the related scp information to be entered (scp host, destination directory, login id, password, scp type, and cipher).		
18	Verify that the scp password entered is not shown or stored in the clear.		
19	Verify that the scp password entered is not shown in the Data Pool Ingest GUI log.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**2.5.2 Ingest SIPS granules without publishing to the Data Pool (ECS-ECSTC-482)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies SIPS ingest without publishing to the Data Pool. This is the main regression test of DPL Ingest where all logging and ECS Service Host functionality should be tested.</i>		#comment
2	<i>[SIPS Ingest, but not to public Data Pool]</i>		#comment
3	Filter the list of insert requests by request status SUCCESSFUL.		
4	Select one successful request from the filtered list where the collection in the request is configured not to be inserted into the public Data Pool.		
5	Verify that the request details are displayed on the Data Pool Ingest GUI, and include all information in S-DPL-16670.		
6	Verify that the request details include a list of all granules associated with the request, in the default order described in S-DPL-16690, and including all granule information listed in S-DPL-16690.		
7	Verify that all granules in the request are in a SUCCESSFUL state.		

---

---

---

---

#	Action	Expected Result	Notes
8	Verify that the start and completion of PDR Validation, and the PDR Validation information in S-DPL-18345, are included in the application log for this request.		
9	Verify that the files for each granule in the request are transferred to a temporary directory on the file system configured for the collection.		
10	Verify that the files for each granule in the request are transferred using the configured transfer method.		
11	Verify that file transfers for all granules in the request were performed on one of the ECS platforms configured for that purpose.		
12	Verify that the start and completion of all file transfers, and the file transfer information in S-DPL-18350, are included in the application log for this request.		
13	Verify that the start and completion of ingest operations for the request, and all request information in S-DPL-18385, are included in the application log for this request.		
14	Verify that the start and completion of ingest operations for each granule in the request, and all granule information in S-DPL-18380, are included in the application log for this request.		
15	Verify that the Science granules in the request were inserted into a non-public directory on the Data Pool, and that all Data Pool inventory database information for non-public granules has been populated. Also verify all Browse granules ingested in this test are PUBLIC (AmGranule(IsOrderOnly) is NULL and the AmBrowseOnlineFile contains at least one entry for each Browse. Also verify the Browse JPEG files are stored in the appropriate public DataPool directory (as recorded in AmBrowseOnlineFile).		
16	Verify that queuing the Data Pool insert and all information in S-DPL-18375 are included in the application log for this request.		
17	Verify that all granules in the request are copied to the ECS archive, to the archive locations and volume groups that are configured for the collection in the AIM database. This includes all primary, backup, forward processing and reprocessing volume groups.		
18	Verify that the start and completion of archiving operations for all granules in the request, and all granule archiving information in S-DPL-18366, appears in the Data Pool Ingest Service application log.		
19	Verify that the start and completion of archiving operations for all files in the request, and all file archiving information in S-DPL-18367, appears in the Data Pool Ingest Service application log.		

---

---

#	Action	Expected Result	Notes
20	Verify that the ECS archive file names of all granules in the request are consistent with the internal file naming convention. (Verify that the file names contain the ECS granule id which is the same as the ECS granule id in the inventory database and identify the physical file format correctly.		
21	Verify that the archive write operations for the granule were executed on one of ECS service hosts configured for that purpose.		
22	Verify that the XML metadata file is copied into the appropriate directory in the small file archive.		
23	Verify that the metadata for the granule is correctly inserted into the AIM database, and that the metadata was inserted in the AIM database AFTER the granule(s) in the request were copied to the ECS archive.		Verify the AmGranule.ArchiveTime for this granule is after the timestamp of the granule data file in the stornext archive. We check the timestamp of the data file instead of xml metadata file here, because  1). the xml file could be updated after archive, e.g. archive time update, browse linkage update.  2). the xml file is copied to the archive from data pool at the same time when the data files are copied to archive.
24	Verify that the internal file(s) for the granule(s) are in the AIM metadata for the granule(s), and that this information is correct.		
25	Verify that the start and completion of metadata insert, and all metadata insert information in S-DPL-18370, appears in the Data Pool Ingest Service application log.		
26	Verify that an xml metadata file is stored in the non-public directory on the Data Pool for each granule in the request.		
27	Verify that the xml metadata file contents are correct for at least one granule of each ESDT used by the test.		
28	Verify that the ECSid of the granules in the request are stored in the "aim" schema, and in the Data Pool and AIM XML files.		

#	Action	Expected Result	Notes
29	Verify that ALL browse granules ingested as part of the test are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

**TEST DATA:**

Any SIPS data type

**EXPECTED RESULTS:**

**2.6 Data Access**

**2.6.1 Use DataAccess drill down to submit HEG request (ECS-ECSTC-492)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Go to the Data Access Testbed GUI		
2	Drill down on a collection that is enabled for ESI processing.		
3	Select a granule for HEG processing by clicking on the Order Form link.		
4	On the processing page, build a HEG request for the granule. Select the synchronous request type. Fill in an email address. Submit the HEG request.		
5	Verify the requests completes successfully. Click on the link for the External EGI Request URL.	Verify that the order complete page is displayed.	
6	Verify an email is sent to the email address entered on the processing tool page.		

**TEST DATA:**

**EXPECTED RESULTS:**

---

### 2.6.2 Use DataAccess drill down to submit GDAL request (ECS-ECSTC-599)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Go to the Data Access Testbed GUI		
2	Drill down on a collection that is enabled for ESI processing.		
3	Select a granule for GDAL processing by clicking on the Order Form link.		
4	On the processing page, build a GDAL request for the granule. Select the synchronous request type. Fill in an email address. Submit the GDAL request.		
5	Verify the requests completes successfully. Click on the link for the External EGI Request URL.	Verify that the order complete page is displayed.	
6	Verify an email is sent to the email address entered on the processing tool page.		

TEST DATA:

EXPECTED RESULTS:

### 2.6.3 Use DataAccess GUI to configure service for datatypes (ECS-ECSTC-600)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Go to the DataAccess Configuration Interface GUI		
2	Under the Collection Configuration tab, right click on an ESDT. Select &quot;Configure New Service for Collection&quot; Add the GDAL, HEG, or GLAS service.		
3	Verify the service is added successfully.		

---

TEST DATA:

EXPECTED RESULTS:

2.6.4 Submit eg/esi requests for HEG Service (ECS-ECSTC-3866)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Using EDSC/Data Access Testbed search for a collection that is configured for HEG service.		
2	Select granule(s) and perform a service request by selecting &quot;Customize&quot; on the edit option page.		
3	Perform parameter subsetting. Select the different output format options. Verify the request completes successfully.		
4	Monitor the status of the requests on the Data Access GUI.		
5	Verify email notifications were sent for the requests.		
6	Verify the status complete email has the links to the files.		
7	Verify the files can be downloaded and inspected.		

TEST DATA:

EXPECTED RESULTS:

2.6.5 Submit egi/esi requests for ICESat-2 data (ECS-ECSTC-3867)

DESCRIPTION:

PRECONDITIONS:

---

**STEPS:**

#	Action	Expected Result	Notes
1	Using EDSC/Data Access Testbed search for ICESat2 granule(s).		
2	Perform a service request on the granule(s) by selecting &quot;Customize&quot;.		
3	Perform parameter subsetting. Select the different output format options. Verify the requests complete successfully.		
4	Monitor the status of the requests on the Data Access GUI.		
5	Verify email notifications were sent for the requests.		
6	Verify the status complete email has the links to the files.		
7	Verify the files can be downloaded and inspected.		

**TEST DATA:**

**EXPECTED RESULTS:**

**2.6.6 Submit egi/esi requests for MOFIS Service (ECS-ECSTC-3868)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Using EDSC/Data Access Testbed search for MCD43GF.006 granules.		
2	Perform a service request on the granule(s) by selecting &quot;Customize&quot;.		
3	Perform parameter subsetting. Select the different output format option. Verify the request completes successfully.		
4	Monitor the status of the requests on the Data Access GUI.		
5	Verify email notifications were sent for the requests.		
6	Verify the status complete email has the links to the files.		
7	Verify the files can be downloaded and inspected.		

TEST DATA:

EXPECTED RESULTS:

2.6.7 Submit order request with no processing (ECS-ECSTC-3869)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Using EDSC/Data Access Testbed search for a collection that is configured in Data Access.		
2	Select granule(s) and perform a order request by selecting &quot;Stage For Delivery&quot; on the edit options page.		
3	Set the Distribution Type to &quot;HTTPS&quot; and the Select Processor to &quot;No Processing&quot;.		
4	Select the &quot;Download Data&quot; button. Verify that the request completes successfully.		
5	Monitor the request on the Distribution Request page of the Order Manager GUI. Verify the request shipped.		
6	Verify that a email notification containing the https download links is sent.		
7	Verify that the files can be downloaded using the links.		

TEST DATA:

EXPECTED RESULTS:

2.6.8 Verify Certificates and trustStore location exist on dpl, oml, and eil. (ECS-ECSTC-3873)

DESCRIPTION:

PRECONDITIONS:

---

**STEPS:**

#	Action	Expected Result	Notes
1	Go the tomcatEnv.sh file in /usr/ecs/OPS/COTS/tomcat/bin on f5dpl01.		
2	Verify the trustStore path exist under /usr/java/latest/jre/lib/security/cacerts		
3	Verify the path exists (ls /usr/java/latest/jre/lib/security/cacerts)		
4	Repeat steps 1 - 3 on f5oml01		
5	Repeat steps 1 - 3 on f5eil01		

**TEST DATA:**

**EXPECTED RESULTS:**

2.6.9 Submit egi/esi requests for FRI Service (ECS-ECSTC-3881)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Using EDSC/Data Access Testbed search for AST_L1T.003 granules.		
2	Perform a service request on the granule(s) by selecting &quot;Customize&quot;.		
3	Perform parameter subsetting. Select the different output format options: TIR, VNIR, and BOTH. Verify the requests complete successfully.		
4	Monitor the status of the requests on the Data Access GUI.		
5	Verify email notifications were sent for the requests.		
6	Verify the status complete email has the links to the files.		
7	Verify the files can be downloaded and inspected.		

**TEST DATA:**

---

EXPECTED RESULTS:

2.6.10 Submit egi/esi requests for SMAPL1L2 Service (ECS-ECSTC-3882)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Using EDSC/Data Access Testbed search for a collection that is configured for SMAPL1L2 service.		
2	Select granule(s) and perform a service request by selecting &quot;Customize&quot; on the edit option page.		
3	Perform parameter subsetting. Select the different output format options. Verify the request completes successfully.		
4	Monitor the status of the requests on the Data Access GUI.		
5	Verify email notifications were sent for the requests		
6	Verify the status complete email has the links to the files.		
7	Verify the files can be downloaded and inspected.		

TEST DATA:

EXPECTED RESULTS:

**3 GDAL REGRESSION TEST**

**3.1 DataAccess GDAL End To End Testing (ECS-ECSTC-520)**

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Find a collection which is enabled for both HEG and GDAL processing in Data Access. Ensure that the service options forms have been uploaded to ECHO.		
2	Go to ECHO and add 3 granules from this collection to the cart.		
3	In the cart select "perform service";		NOTE: If it is grayed out, you will need to go back to the Collection Configuration tab of the DataAccess Configuration GUI and upload the echo form for the collection. Then, log into to the PUMP to update the virtual tags. This is done on the "Service Entries" page under "Service Management" in the "Provider Context" tab. To complete the process of updating the virtual tags, under the "Virtual" tab in the "Tags" section, select a DATASET (e.g. PVC_TS2). From the list of datatypes, verify that the datatype used for the test is checked, then upload the tag.
4	Set different order options for each of the three granules:		
5	4a. select HEG processing with GeoTiff as the output format		
6	4b. select GDAL processing with NetCDF as the output format		
7	4c. select GDAL processing with png as the output format		
8	Submit the requests and verify that they succeed.		
9	Verify that the output files were processed correctly by opening and viewing them.		NOTE: To view the netCDF output, go to

#	Action	Expected Result	Notes
			<a href="http://www.giss.nasa.gov/tools/panoply/download_gen.html">http://www.giss.nasa.gov/tools/panoply/download_gen.html</a> to download and install the netCDF viewer called Panoply.

TEST DATA:

EXPECTED RESULTS:

**3.2 DP\_81\_02\_TP050 GDAL processing requests and errors (ECS-ECSTC-856)**

DESCRIPTION:

Submit requests that will result in a variety of GDALService statuses. Examples: a) Conversion of a granule to a supported format. b) Conversion of a granule with band subsetting. c) Conversion of a granule with spatial subsetting. d) Conversion of a granule to an unsupported format. e) Conversion takes too long. Time out case f) Conversion granule/file does not exist.

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>comment</i>		#comment
2	Note, requests in this test can be submitted via EGI, ESI or directly to the GDALService. The easiest way to build requests is to use the ESI inventory drilldown and then use the links provided.		
3	Submit a request for conversion of a granule to a supported format like Gtiff.	The request should succeed and the converted files should appear in the results. Verify the converted files have names that identify the data-object component processed (and not sequential number or other non-identifying naming convention).	
4	Submit a request for conversion of a granule to Gtiff format with band subsetting. Select one or multiple bands.	The request should succeed and the converted files should appear in the results. The number of files should	

---

---

#	Action	Expected Result	Notes
		correspond to the number of bands you selected. Verify the converted files have names that identify the data-object component processed (and not sequential number or other non-identifying naming convention).	
5	Submit a request for conversion of a granule to Gtiff format with spatial subsetting.	The request should succeed and the converted files should appear in the results.	
6	Submit a request for conversion of a granule to an unsupported format.	The request should fail/error out with InvalidParameterValue and a message stating that the format used was not valid.	
7	Submit a request for conversion for a granule whose files are missing from the datapool.	The request should fail/error out with InternalError and a message similar to InputHDFEOSFileNotExistErr	
8	Submit a request for conversion that will take too long to process so that a timeout occurs. You can modify the GdalService.application.timeout parameter in the GdalServiceConfig.properties file to 1 to make timeouts happen very fast. Make sure to bounce GdalService if you modify the properties file.	The request should fail/error out with InternalError and message similar to Cancelled (Timeout). Also there should not be a stray gdal_translate process for this request on the GdalService host.	

TEST DATA:

EXPECTED RESULTS:

Verify that the Internal API returns: a) The list of converted files in the case of success or b) A correct classification of the error and/or an error description

**3.3 SMAP L1/L2 spatial subsetting with GeoTIFF reformatting (ECS-ECSTC-857)**

DESCRIPTION:

Geospatial Data Abstraction Library (GDAL) is a COTS tool used by ESI to provide various types of geospatial data processing, such as reformatting. HDF5 and NetCDF are used by GDAL. These products are automounted under /tools/gdal, /tools/hdf5, and /tools/netcdf, respectively, and are accessible via symlinks under /usr/ecs/OPS/COTS. They can be tested using the steps in the Steps section:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Find a collection (such as MOD10CM.005) which is enabled for both HEG and GDAL processing in Data Access. Ensure that the service options forms have been uploaded to ECHO.		
2	Go to ECHO and add 5 granules from this collection to the cart.		
3	In the cart select &quot;perform service&quot;.		
4	Set different order options for each of the five granules:		
5	a. select GDAL processing with GeoTiff as the output format		
6	b. select GDAL processing with KML as the output format		
7	c. select GDAL processing with ASCII (Arc/Info ASCII Grid) as the output format		
8	d. select GDAL processing with NetCDF as the output format		
9	e. select GDAL processing with PNG as the output format		
10	Submit the requests and verify that they succeed.		
11	Verify that the output files were processed correctly by opening and viewing them.		

TEST DATA:

EXPECTED RESULTS:

## 4 APACHE REGRESSION TEST

### 4.1 AIM

4.1.1 Install ESDTs with a variety of spatial search types (ECS-ECSTC-607)

DESCRIPTION:

PRECONDITIONS:

---

STEPS:

#	Action	Expected Result	Notes
1	<i>[Install ESDTs with a variety of spatial search types]</i>		#comment
2	Verify that the descriptor ODL files for the ESDT being installed are moved from the installation source directory into the configured descriptor directory.		
3	Prepare descriptor files for several (at least 5) ESDTs covering a variety of ESDT types described below, such that the descriptors conform to the ECS data model and the XML schema validation rules, guaranteeing successful ESDT installation. Examples of such can be found in /sotestdata/DROP_721/DS_7E_01/Criteria/2030, and are listed as follows:   1. ESDT descriptor with a spatial search attribute of Rectangle  MOD09CMG.005  MYD09CMG.005   2. ESDT descriptor with a spatial search attribute of GPolygon  MOD14.005  MYD14.005   3. ESDT descriptor with a spatial search attribute of NotSupported  AEPOE7W.001  PM1GBAD1.001   4. ESDT descriptor with a spatial search attribute of Orbit  AE_Land.002  AE_Rain.002   5. ESDT descriptor with a spatial search attribute of Point  g3bssp.007  g3bt.007 		
4	Ensure collections are not already installed.		
5	Ensure EcDsAmESDTMaint.properties has log.debug.level=XVERBOSE (for verification).		
6	Copy the descriptor files to the source directory (specified in the GUI).		
7	Verify that the descriptors are listed in the GUI.		
8	Get the current time to help search the logs.		
9	From the ESDT Maintenance GUI, select the ESDTs in the source directory and initiate installation.		
10	Upon completion of the ESDT installation, verify that the GUI displays a message indicating number of ESDTs successfully installed.		
11	Verify that the metadata elements provided in the descriptor files were validated by the XML Services library against the ESDR common schema, which conforms to the ECS data model. If you have the ESDT Maintenance GUI debug level set to XVERBOSE, you should get a message something like the following in the ESDT Maintenance GUI log:   >EcsFileProcessor.validateXmlFile DsESDTMoMOD09A1.005.xml   >Other than that, it might require looking at the actual code.		
12	Verify by inspection that the element types and lengths in the descriptors match those stated in the ESDT common schema which in turn matches the		Are the descriptors supposed to be compared to XML

#	Action	Expected Result	Notes
	ECS data model. The ESDT common schema is under: /usr/ecs/MODE/CUSTOM/WWW/DSS/ESDTMaint/WEB-INF/lib/EcDsAmEsdtCommonSchemas.jar. 		schema? Should verify that the descriptors in the configured descriptors directory diff with the originals. Jon Pals:  &quot; For Step 5, Yes, it is asking that you compare what is in the ODL ESDT descriptor file against the XML schema files. For this, I suggest doing some random sampling. Just take a few attributes from the ESDT descriptor file and find how those attributes are defined in the XML schema files.&quot; Need an automated way to compare ODL descriptor files to XML schemas. 
13	Verify that an MCF file is generated for each ESDT being installed, and stored into the configured MCF directory location ( locations pulled from /custom/ecs/TS3/CUSTOM/cfg/EcDsAmESDTMaint.properties )  descriptor.target.dir = /stornext/smallfiles/TS3/descriptor descriptor.source.dir = /usr/ecs/TS3/CUSTOM/data/ESS mcf.target.dir = /stornext/smallfiles/TS3/mcf archive.metadata.dir = /stornext/smallfiles/TS3/metadata 		20130731t124819::f5dpl01v ::cmshared::/stornext/smallfiles/TS3/mcf \$ find . -type f -mtime -1   sed 's/..([^\#]*)#[([^\.]*)].*\^1.\2/' MOD09CMG.005 MOD14.005 MYD14.005 MYD09CMG.005 AE_Land.002 AE_Rain.002 g3bssp.007 PM1GBAD1.001 g3bt.007 AEPOE7W.001 
14	Verify that the generated MCF files are correct. 		( Define &quot;correct&quot;: The MCFs should be a pretty

#	Action	Expected Result	Notes
			<p>close match with the /&gt;INVENTORYMETADAT A section and the ARCHIVEDMETADATA section in the ESDT /&gt;descriptor file. The place where there should be a difference in in the /&gt;AdditionalAttributes group. The ProductSpecificMetadata group in the ESDT /&gt;descriptor file is replaced with a generic AdditionalAttributes group in the &gt;MCF. )&lt;br /&gt;&lt;br /&gt; /&gt;Within the INVENTORYMETADATA group, the ProductSpecificMetadata group from the descriptor file should be replaced with this AdditionalAttributes block in the MCF file:&lt;br /&gt;&lt;br /&gt; GROUP = AdditionalAttributes&lt;br /&gt; OBJECT = AdditionalAttributesContain er&lt;br /&gt; Data_Location = &amp;quot;NONE&amp;quot;&lt;br /&gt; Mandatory = &amp;quot;FALSE&amp;quot;&lt;br /&gt; CLASS = &amp;quot;M&amp;quot;&lt;br /&gt; OBJECT = AdditionalAttributeName&lt;br /&gt; Mandatory = &amp;quot;FALSE&amp;quot;&lt;br /&gt; CLASS =</p>

#	Action	Expected Result	Notes
			&quot;M&quot;                      Data_Location = &quot;PGE&quot;                      NUM_VAL = 1                      TYPE = &quot;STRING&quot;                      /> END_OBJECT = AdditionalAttributeName                      GROUP = InformationContent                      CLASS = &quot;M&quot;                      OBJECT = ParameterValue                      Mandatory = &quot;FALSE&quot;                      Data_Location = &quot;PGE&quot;                      NUM_VAL = 1                      TYPE = &quot;STRING&quot;                      /> END_OBJECT = ParameterValue                      END_GROUP = InformationContent                      END_OBJECT = AdditionalAttributesContain er                      END_GROUP = AdditionalAttributes                      />MODIS MCF files should include the descriptor file's ARCHIVEDMETADATA group.                      />AE_Land#002.MCF, AE_Rain#002.MCF, and PM1GBAD1#001.MCF do not have this group 
15	Verify that an XML schema file (*.xsd) is generated for each ESDT being installed, and stored into the descriptors directory.		

#	Action	Expected Result	Notes
16	Verify that the generated XML schema files contain validation rules based upon the contents of the Inventory section of the Descriptor file and the default element rules. 		The &quot;Inventory section&quot; probably starts here: GROUP = INVENTORYMETADATA Jon Pals:  For Step 9, take a look at the 'Mandatory =' lines in the INVENTORYMETADATA section of the ESDT descriptor file and compare that with the ESDT's .xsd file in the /stornext/smallfiles/&lt;MODE&gt;/descriptor directory. The INVENTORYMETADATA attributes with 'Mandatory = &quot;FALSE&quot;' lines should be listed in the .xsd file as having 'minOccurs=&quot;0&quot;' . The INVENTORYMETADATA attributes with 'Mandatory = &quot;TRUE&quot;' lines should be listed in the .xsd file as having no 'minOccurs' or having a 'minOccurs=&quot;1&quot;'. 
17	Verify that basic collection identification information and proper spatial search type have been added into the AIM Inventory database. 		
18	Verify that the PSA associations, if any, have been added into the AIM Inventory database. 		select aa.additionalattributename from amcollection c >join dsmdcollectionaddnlattribx ref ax on c.collectionid = ax.collectionid join

#	Action	Expected Result	Notes
			dsmdadditionalattributes aa on ax.attributeid = aa.attributeid where c.shortname = '\${SHORTNAME}'<br \${VERSIONID}; 
19	Verify that an insert event, and event qualifiers are inserted into the Spatial Subscription Server database for each ESDT being installed. 		Event qualifiers appear restricted to those defined in ecnbeventmetadataattrdef.attributename. EVENTPARMS: Extracted from descriptor file  Online EVENTPARMS known by SSS will be added.  -- What qualifiers does SSS know about? select * from ecnbeventmetadataattrdef where attributename in (\${EVENTPARMS});  -- What qualifiers were added? select attributename from ecnbeventattrxref where esdt_id = '\${SHORTNAME}' and versionid = \${VERSIONID}; 'RangeBeginningTime' and 'RangeEndingTime' appear to get combined into 'GranuleTimeRange'.  'RangeBeginningDate' and 'RangeEndingDate' appear to get combined into 'GranuleDateRange'. 

#	Action	Expected Result	Notes
			<pre>&lt;/&gt;g3bssp.007 and g3bt.007 both lack date and time ranges, but both have GranuleTimeRange and GranuleDateRange in ecnbeventattrref.&lt;br /&gt;example:&lt;br /&gt;select *&lt;br /&gt;from ecnbeventdefinition&lt;br /&gt;where (esdt_id = 'MOD14' and versionid = 5)&lt;br /&gt;or (esdt_id = 'MYD14' and versionid = 5)&lt;br /&gt;or (esdt_id = 'AEPOE7W' and versionid = 1)&lt;br /&gt;or (esdt_id = 'PM1GBAD1' and versionid = 1)&lt;br /&gt;or (esdt_id = 'AE_Land' and versionid = 2)&lt;br /&gt;or (esdt_id = 'AE_Rain' and versionid = 2)&lt;br /&gt;or (esdt_id = 'MOD09CMG' and versionid = 5)&lt;br /&gt;or (esdt_id = 'MYD09CMG' and versionid = 5)&lt;br /&gt;or (esdt_id = 'g3bssp' and versionid = 7)&lt;br /&gt;or (esdt_id = 'g3bt' and versionid = 7);&lt;br /&gt;&lt;br /&gt; eventtype   esdt_id   versionid   eventid&lt;br /&gt;----- -----+-----+----- ----+-----&lt;br /&gt; INSERT   AE_Land   2  &lt;br /&gt; INSERT   AE_Rain   2  &lt;br /&gt; INSERT   AEPOE7W   1  &lt;br /&gt; INSERT   MOD09CMG  5</pre>

#	Action	Expected Result	Notes
			<pre>  &lt;br /&gt; INSERT   MOD14   5  &lt;br /&gt; INSERT   MYD09CMG   5  &lt;br /&gt; INSERT   MYD14   5  &lt;br /&gt; INSERT   PM1GBAD1   1  &lt;br /&gt; INSERT  g3bssp   7  &lt;br /&gt; INSERT   g3bt   7  &lt;br /&gt;(10 rows)&lt;br /&gt;                     </pre>
20	Verify that the ESDT Maintenance GUI logs the processing activities including time of request, action requested, ESDT ShortName, Version ID, descriptor file name, and the result of the operation, in a configured log file.		The ESDTMaintenanceGUI.* logs include a timestamp in each log entry, but it's not clear when a request actually began.  These lines from the debug log may be the beginning of a request:  07.31.2013 12:02:09.563 : Thread ID [41] : VERBOSE : /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTAmAE_Land.002.desc dataModelType is: ECS 07.31.2013 12:02:09.563 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTAmAE_Land.002.desc The action is not stated explicitly. In the debug log, one entry includes the word "installing";  07.31.2013 12:02:11.821 : Thread ID

#	Action	Expected Result	Notes
			<p>[41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br &gt;/&gt;The ShortName and Version ID are part of the descriptor filename. They appear in several other log entries, but it's not clear why:&lt;br /&gt;&lt;br /&gt;07.31.2013 12:02:09.651 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObj ct4CallableStmt() Param Index:10 String Value:AE_Land.002 Type:12&lt;br /&gt;&lt;br &gt;/&gt;07.31.2013 12:02:09.702 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObj ct4CallableStmt() Param Index:1 String Value:AE_Land Type:1&lt;br &gt;/&gt;07.31.2013 12:02:09.702 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObj ct4CallableStmt() Param Index:2 String Value:2 Type:4&lt;br /&gt;The debug log reports the successful installation like this:&lt;br &gt;&lt;br /&gt;07.31.2013 12:02:10.358 : Thread ID [41] : INFORMATION : InstallESDTPage:Successful ly installed ESDT DsESDTAmAE_Land.002.d</p>

#	Action	Expected Result	Notes
			esc The ESDTMaintenanceGUI.ops0 .log contains very little information.<br >Example: From >/usr/ecs/TS3/CUSTOM/logs >/ESDTMaintenanceGUI.de >bug0.log:  time of >the installation request:<br >07.31.2013 12:02:11.821 : >Thread ID [41] : VERBOSE > : installing >ESDT:/usr/ecs/TS3/CUSTO >M/data/ESS/DsESDTMoM >OD09CMG.005.desc<br > action requested: >(this shows the action >taken) 07.31.2013 >12:02:11.821 : Thread ID >[41] : VERBOSE : installing >ESDT:/usr/ecs/TS3/CUSTO >M/data/ESS/DsESDTMoM >OD09CMG.005.desc<br > ShortName:<br >07.31.2013 12:02:11.862 : >Thread ID [41] : VERBOSE > : >AbstractServiceImpl.setObj >ct4CallableStmt() Param >Index:1 String >Value:MOD09CMG >Type:1  Version >ID: 07.31.2013 >12:02:11.862 : Thread ID >[41] : VERBOSE : >AbstractServiceImpl.setObj >ct4CallableStmt() Param >Index:4 String Value:5 >Type:4 <br

#	Action	Expected Result	Notes
			/>descriptor file name: 07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTMoM OD09CMG.005.desc  result of the operation (successful): 07.31.2013 12:02:12.215 : Thread ID [41] : INFORMATION : InstallESDTPage:Successful ly installed ESDT DsESDTMoMOD09CMG.0 05.desc
21	<i>Verification Instructions</i>		#comment
22	Verify that there are at least 5 descriptors that match the setup requirements in directory /sotestdata/DROP_721/DS_7E_01/Criteria/2030/ Verify that the value of the OBJECT = SpatialSearchType group is “rectangle”, “GPolygon”, “not supported”, “point”, or “orbit” accordingly:    ESDT   SpatialSearchType   Archiving DAAC     ACR3L2SC.001   NotSupported   ASDC     AEPOE7W.001   NotSupported   NSIDC     AE_Land.002   Orbit   NSIDC     AE_Rain.002   Orbit   NSIDC     MB2LME.198   Orbit   ASDC     MIANRCCH.198   NotSupported   ASDC     MIL2ASOS.198   Orbit   ASDC     MIL3DAE.198   Rectangle   ASDC     MOD09CMG.005   Rectangle   LP DAAC     MOD14.005   GPolygon   LP DAAC     MOP02.003   Rectangle   ASDC     MYD09CMG.005   Rectangle   LP_DAAC     MYD14.005   GPolygon   LP DAAC     PM1GBAD1.001   NotSupported   NSIDC     TL3ATD.002   Rectangle   ASDC     g3assp.004   Point   ASDC     g3atb.004   Point   ASDC		

#	Action	Expected Result	Notes
23	Return to the ESDT List Page. Select the check box next to the descriptor names set up in setup step 1. Select the “Install new ESDTs/Update existing ESDTs” button. Note the wall clock time of the start of the installation for later verification in the logs. Note the number of descriptors selected.		Login to the ESDT Maintenance GUI. Ensure no test ESDT is currently installed. If it is, delete its granules, clean up orphans, remove from the data pool (DPL GUI), and delete it (ESDT GUI). Click the “Install new ESDTs/Update existing ESDTs” button. Note the ESDT source directory. Copy the test ESDTs to the ESDT source directory.  Login to the ESDT Maintenance GUI. Click the “Install new ESDTs/Update existing ESDTs” button. Select the test ESDTs. Click the “Proceed with installation/update” button. Wait for the screen to refresh.
24	Verify that the GUI displays the number of descriptors installed is equal to the number of descriptors selected.		
25	Verify that the metadata elements provided in the descriptor files were validated by the XML validation utility by viewing its application log in the mode. Verify for each descriptor that no failures occurred. 		
26	Select one descriptor from each part of setup step 1. 		
27	Go to the directory where the generic descriptor schema is installed and view it.		
28	Verify by inspection that rules for elements in the generic schema are executed correctly for corresponding elements in the descriptor. A similar comparison should be performed between the elements of the generic descriptor schema and the ECS data model. 		What is the XML validation utility used, and where is its log? Where is “the directory where

#	Action	Expected Result	Notes
			the generic descriptor schema is installed" ; 
29	Verify by inspection that rules for elements in the generic schema are executed correctly for corresponding elements in the descriptor. 		
30	Verify each element in the descriptor file that has a matching element in the generic schema gets added to the descriptor's .xsd file. 		&quot;A similar comparison should be performed between the elements of the generic descriptor schema and the ECS data model.&quot;  What does this mean? /usr/ecs/TS3/CUSTOM/logs/ESDTMaintenanceGUI.ddebug0.log shows the descriptor .xml and .xsd files are generated but not whether they are validated:  07.31.2013 12:02:12.021 : Thread ID [41] : xmlsvcs.schemagen.EcAmDescSchemaGenImpl : VERBOSE : START generateDescriptorSchemaFromOdlDescriptor: odlDescriptor: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc outputSchema: /stornext/smallfiles/TS3/descriptor/DsESDTMoMOD09CMG.005.xsd 07.31.2013 12:02:12.021 : Thread ID [41] : xmlsvcs.schemagen.EcAmDescSchemaGenImpl : VERBOSE : START

#	Action	Expected Result	Notes
			<pre>generateDescriptorXmlFile inputODLFile: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc outputXMLFile: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml&lt;br /&gt;07.31.2013 12:02:12.050 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : END generateDescriptorXmlFile inputODLFile: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc outputXMLFile: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml&lt;br /&gt;07.31.2013 12:02:12.051 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : START generateDescriptorSchemaF romXmlDescriptor xmlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br /&gt;07.31.2013 12:02:12.051 :</pre>

#	Action	Expected Result	Notes
			<p>Thread ID [41] : xmlsvcs.schemagen.EcAmS chemaGenerator : INFORMATION : createESDTSchema START: XML descriptor - /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml, Schema - /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br &gt;07.31.2013 12:02:12.154 : Thread ID [41] : xmlsvcs.schemagen.EcAmS chemaGenerator : INFORMATION : createESDTSchema DONE: XML descriptor - /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml, Schema - /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br &gt;07.31.2013 12:02:12.155 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : END generateDescriptorSchemaF romXmlDescriptor xmlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09</p>

#	Action	Expected Result	Notes
			CMG.005.xsd                      />07.31.2013 12:02:12.155 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : END generateDescriptorSchemaF romOdlDescriptor odlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd
31	Find the mcf target directory (parameter mcf.target.dir) in the EcAmMaintenanceGui.properties file.  Change the directory to the ESDT specific subdirectory (probably /stornext/smallfiles/&lt;mode&gt;/mcf/).  Perform an ls to verify that the mcf file is found. Repeat for each MCF file generated from the ESDT installation.		/custom/ecs/\${MODE}/CU STOM/cfg/EcDsAmESDT Maint.properties<br />mcf.target.dir=/stornext/s mallfiles/\${MODE}/mcf
32	Compare the descriptor file to the mcf file to verify that the MCF is correct. 		
33	Change the directory to this directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/).		
34	Perform an ls to verify that the schema file is found. 		/custom/ecs/\${MODE}/CU STOM/cfg/EcDsAmESDT Maint.properties<br />descriptor.target.dir=/storn ext/smallfiles/\${MODE}/de scriptor
35	Make a one to one comparison of the groups in the Inventory Section of the Descriptor file with the corresponding groups in the schema.		
36	Verify that the values of the descriptor are valid according to the rules in the schema.		

#	Action	Expected Result	Notes
37	Find the descriptor source and target directory (parameters descriptor.source.dir and descriptor.target.dir) in the EcAmMaintenanceGui.properties file. 		
38	Change to target directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/).		
39	Perform an ls to verify that the descriptor file DsESDT&lt;ShortName&gt;.&lt;VersionId&gt;.desc is found. Repeat for each part descriptor installed in the setup step 1.		
40	Verify that the descriptor file was deleted in the source directory in the mode.		
41	Change to directory descriptor.source.dir.		
42	Verify that the descriptor was deleted by using the ls command and finding no files matching the descriptor's name. Also, repeat for each descriptor installed.		/custom/ecs/\${MODE}/CUSTOM/cfg/EcDsAmESDT Maint.properties descriptor.source.dir=/usr/ecs/\${MODE}/CUSTOM/data/ESS descriptor.target.dir=/stornext/smallfiles/\${MODE}/descriptor This should be at least 2 steps.
43	In the AIM Inventory database, verify that the ShortName, VersionID, and insertTime are populated in the DsMdCollections table for each ESDT installed.   select ShortName, VersionID, insertTime  from DsMdCollections  where ShortName = "&lt;ShortName&gt;"  and VersionID= "&lt;VersionID&gt;" 		
44	Verify that the spatial searchSearchType matches the one specified in the descriptor.   select configuredName, VersionID, spatialSearchType  from DsMdESDTConfiguredType  where configuredName = "&lt;shortName&gt;"  and VersionID = "VersionID" 		
45	Repeat for each installed ESDT.		select c.inserttime, esdt(c.shortname,c.versionid), c.collectionid, c.spatialsearchtype, e.spatialsearchtype />from amcollection c />join dsgeesdtconfiguredtype e 

#	Action	Expected Result	Notes
			<pre> /&gt;on c.shortname = e.configuredname and c.versionid = e.versionid&lt;br /&gt;where (c.shortname = '\${SHORTNAME}' and c.versionid = \${VERSIONID})&lt;br /&gt;or (c.shortname = '\${SHORTNAME}' and c.versionid = \${VERSIONID})&lt;br /&gt;...&lt;br /&gt;order by c.spatialsearchtype, c.shortname, c.versionid;&lt;br /&gt;dsgeesdtconfiguredtype replaces DsMdESDTConfiguredTyp e.&lt;br /&gt;select c.inserttime, esdt(c.shortname,c.versionid ), c.collectionid,&lt;br /&gt; c.spatialsearchtype, e.spatialsearchtype&lt;br /&gt;from amcollection c&lt;br /&gt;join dsgeesdtconfiguredtype e&lt;br /&gt;on c.shortname = e.configuredname and c.versionid = e.versionid&lt;br /&gt;where (c.shortname = 'MOD14' and c.versionid = 5)&lt;br /&gt;or (c.shortname = 'MYD14' and c.versionid = 5)&lt;br /&gt;or (c.shortname = 'AEPOE7W' and c.versionid = 1)&lt;br /&gt;or (c.shortname = 'PM1GBAD1' and c.versionid = 1)&lt;br /&gt;or (c.shortname = 'AE_Land' and c.versionid = 2)&lt;br /&gt;or </pre>

#	Action	Expected Result	Notes
			<pre>(c.shortname = 'AE_Rain' and c.versionid = 2)&lt;br /&gt;or (c.shortname = 'MOD09CMG' and c.versionid = 5)&lt;br /&gt;or (c.shortname = 'MYD09CMG' and c.versionid = 5)&lt;br /&gt;or (c.shortname = 'g3bssp' and c.versionid = 7)&lt;br /&gt;or (c.shortname = 'g3bt' and c.versionid = 7)&lt;br /&gt;order by c.spatialsearchtype, c.shortname, c.versionid;&lt;br /&gt;&lt;br /&gt;      inserttime        esdt        collectionid   spatialsearchtype   spatialsearchtype&lt;br /&gt;----- -----+----- -----+----- -----&lt;br /&gt; 2013-07-31 12:02:12.560421   MOD14.005        198330   GPolygon        GPolygon&lt;br /&gt; 2013-07-31 12:02:13.716981   MYD14.005        198332   GPolygon        GPolygon&lt;br /&gt; 2013-07-31 12:02:11.403874   AEPOE7W.001      198328   NotSupported      NotSupported&lt;br /&gt; 2013- 07-31 12:02:14.194661   PM1GBAD1.001    198333   NotSupported      NotSupported&lt;br /&gt; 2013-</pre>

#	Action	Expected Result	Notes
			07-31 12:02:09.656776   AE_Land.002   198326   Orbit   Orbit  2013-07-31 12:02:10.765544   AE_Rain.002   198327   Orbit   Orbit  2013-07-31 12:02:14.865666   g3bssp.007   198334   Point   Point<br /> 2013-07-31 12:02:15.367958   g3bt.007   198335   Point   Point  2013-07-31 12:02:11.866531   MOD09CMG.005   198329   Rectangle   Rectangle  2013-07-31 12:02:13.12891   MYD09CMG.005   198331   Rectangle   Rectangle (10 rows)
46	In another terminal log into the Spatial Subscription Server (SSS) database. View the &lt;ShortName&gt;.&lt;VersionId&gt; descriptor in the mode and find the INSERT object under the EVENT group in the descriptor. Under the EVENTPARMS object are a list of parameters which should be populated in the SSS database.		
47	Verify that the event was inserted into the EcNbEventDefinition table.   select * from EcNbEventDefinition  where ESDT_Id in ( &lt;a quoted list of ShortNames of installed ESDTs&gt;); 		
48	Verify that each parameter EVENTPARM parameter is populated in the SSS database .   select * from EcNbEventAttrXref  where ESDT_Id = "ShortName" 		
49	Repeat this step for each descriptor installed		
50	Change directory to the log.dir directory of the mode.		
51	View the ESDT_Maint.log and search for the first ESDT installed.		

#	Action	Expected Result	Notes
52	Verify the time of the installation request, the ShortName, Version ID, descriptor file name and the result of the operation (successful) is listed in the log file.		
53	Repeat for each ESDT installed.		time of the installation request: 07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoM/OD09CMG.005.desc  ShortName: 07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:1 String Value:MOD09CMG Type:1  Version ID: 07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:4 String Value:5 Type:4  descriptor file name: 07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoM/OD09CMG.005.desc  result of the operation (successful): 07.31.2013 12:02:12.215 : Thread ID [41] : INFORMATION : InstallESDTPage:Successful

#	Action	Expected Result	Notes
			ly installed ESDT DsESDTMoMOD09CMG.0 05.desc

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

4.1.2 Update multiple ESDTs, success case (ECS-ECSTC-608)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Preconditions</i>		#comment
2	<i>Modify the ESDT descriptor ODL files for several (at least 6) previously installed ESDTs, such that the modifications include the following variety of ESDT updates:  1. removing one or more existing collection level attributes that are not restricted  2. removing DLLName parameter from the Collection metadata group  3. changing the ShortName in the CollectionAssociation group  4. changing a mandatory inventory attribute to optional  5 adding one or more optional inventory metadata attributes  6. adding one or more new qualifiers on existing events</i>		#comment
3	<i>Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2060/Replacement as the initial descriptors. Use only GLA01.013 ... GLA06.013; do not use GLA07.013.</i>		#comment
4	<i>Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2120 as the replacement descriptors.</i>		#comment
5	Compare each replacement descriptor with its initial counterpart, and note the differences.		

#	Action	Expected Result	Notes
6	Ensure none of the test collections are installed, according to the ESDT Maintenance GUI.		
7	Note the values associated with the following names in EcDsAmESDTMaint.properties:  descriptor.target.dir descriptor.source.dir mcf.target.dir  E.g.,  descriptor.target.dir=/stornext/smallfiles/OPS/descriptor descriptor.source.dir=/usr/ecs/OPS/CUSTOM/data/ESS mcf.target.dir=/stornext/smallfiles/OPS/mcf		
8	Install the initial test collections:  Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}.  Log in to the ESDT Maintenance GUI. Click the &quot;Install New ESDTs/Update Existing ESDTs&quot; button. Select all the test descriptors. Click the &quot;Proceed with installation/update&quot; button. Ensure that all descriptors install successfully. Ensure the descriptor files are removed from the source directory.		
9	Verify an MCF file is created in \${mcf.target.dir} for each descriptor. 		
10	Copy the MCF files to a separate directory for later verification. 		
11	Verify each descriptor was moved to \${descriptor.target.dir}. 		
12	Copy the descriptor files to a separate directory for later verification. 		
13	Verify an XML schema file (*.xsd) was created for each descriptor in \${descriptor.target.dir}. 		
14	Copy the schema files to a separate directory for later verification. 		
15	Extract /usr/ecs/TS3/CUSTOM/lib/DSS/EcDsAmEsdtCommonSchemas.jar for later verification.		
16	<i>Setup</i>		#comment
17	Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}.		
18	Log in to the ESDT Maintenance GUI.		
19	Click the &quot;Install New ESDTs/Update Existing ESDTs&quot; button.		
20	Select all the test descriptors.		
21	Note the current time as t0.		
22	Click the &quot;Proceed with installation/update&quot; button.		
23	Upon completion of the ESDT update, verify that the GUI displays a message indicating the number of ESDTs successfully updated.		

#	Action	Expected Result	Notes
24	<i>Verify that an MCF file is created for each ESDT being updated, and stored into the configured MCF directory location replacing the previous version.</i>		#comment
25	Verify for each updated descriptor an MCF file exists in \${mcf.target.dir} with a timestamp more recent than time t0.		
26	<i>Verify that the MCF files are correct.</i>		#comment
27	For each descriptor file, note the contents of the INVENTORYMETADATA section and the ARCHIVEMETADATA section.		
28	Verify the MCF file has the same INVENTORYMETADATA group as the descriptor, with the exception that ProductSpecificMetadata group from the descriptor file is replaced with this static AdditionalAttributes block in the MCF file:   GROUP = AdditionalAttributes  OBJECT = AdditionalAttributesContainer  Data_Location = &quot;NONE&quot;  Mandatory = &quot;FALSE&quot;  CLASS = &quot;M&quot;  OBJECT = AdditionalAttributeName  Mandatory = &quot;FALSE&quot;  CLASS = &quot;M&quot;  Data_Location = &quot;PGE&quot;  NUM_VAL = 1  TYPE = &quot;STRING&quot;  END_OBJECT = AdditionalAttributeName  GROUP = InformationContent  CLASS = &quot;M&quot;  OBJECT = ParameterValue  Mandatory = &quot;FALSE&quot;  Data_Location = &quot;PGE&quot;  NUM_VAL = 1  TYPE = &quot;STRING&quot;  END_OBJECT = ParameterValue  END_GROUP = InformationContent  END_OBJECT = AdditionalAttributesContainer  END_GROUP = AdditionalAttributes		
29	Verify in the MCF file INVENTORYMETADATA section is followed by the descriptor's ARCHIVEMETADATA section.		
30	<i>Verify the descriptor file for each ESDT is replaced by the updated descriptor file, and that the updated descriptor files are consistent with the ECS data model.</i>		#comment
31	Verify each ESDT's updated descriptor file replaced the initial descriptor file in the \${descriptor.source.dir} directory, using diff.		
32	Verify each ESDT validates against the extracted schema. For each ESDT,  Log in to the ESDT Maintenance GUI. Click on the descriptor file in the list. Copy the XML version of the descriptor. Save the XML to a file. Validate the file against the extracted		

#	Action	Expected Result	Notes
	schema:  xmllint --noout --schema /path/to/schema/Descriptor.xsd ../descriptor_file.xml		
33	<i>Verify that an XML schema file is generated for each ESDT and stored into the configured location replacing the previous version</i>		#comment
34	Verify that each ESDT has an XML schema file (*.xsd) in \${descriptor.source.dir} with a timestamp on or after time t0.		
35	<i>Verify that the generated XML schema files contain validation rules based upon the contents of the Inventory section of the Descriptor file and the default element rules.</i>		#comment
36	Diff the updated XML schema files against the initial XML schema files saved earlier.		
37	Verify the differences between updated and initial XML schema files correspond to the differences between updated and initial descriptor files.		
38	<i>Verify that collection-based tables in the AIM Inventory database as identified in the Operations Concept are populated with correct information.</i>		#comment
39	For each updated ESDT, in the ecs database, in the amcollection table, verify lastupdate is after time t0 and each of LongName, CollectionDescription, SpatialSearchType, and RevisionDate match those in the descriptor file:  select lastupdate, esdt(shortname,versionid), LongName, CollectionDescription, SpatialSearchType, RevisionDate from amcollection where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID};		
40	For each updated ESDT, in the ecs database, in the DsGeESDTCfguredType table, verify descriptorFileNameBase matches the descriptor file name and spatialSearchType matches that in the descriptor file:  select descriptorFileNameBase, spatialSearchType from DsGeESDTCfguredType where configuredName = '\${SHORTNAME}' and versionid = \${VERSIONID};		
41	For each updated ESDT, in the ecs database, verify the AdditionalAttributes are correctly cross-referenced and that each value matches that in the descriptor file in the appropriate AdditionalAttributesContainer section:  select aa.attributeid, aa.additionalattributename, aa.additionalattributedescription, aa.additionalattributedatatype, aa.parameterunitofmeasure, aa.parameterrangebegin, aa.parameterrangeend, aa.parametervalueaccuracy, aa.valueaccuracyexplanation, aa.measurementresolution from amcollection c join dsmdcollectionadnlattrbsxref ax on		

#	Action	Expected Result	Notes
	<p>c.collectionid = ax.collectionid&lt;br /&gt;join dsmdadditionalattributes aa&lt;br /&gt;on  ax.attributeid = aa.attributeid&lt;br /&gt;where c.shortname =  '\${SHORTNAME}'&lt;br /&gt;and c.versionid = \${VERSIONID};&lt;br /&gt;&lt;br /&gt; /&gt;Note that if an additional attribute existed in the database prior to installing  an ESDT, it will not be updated to reflect the new definition.</p>		
42	<p>&lt;i&gt;Verify that the insert event and event qualifiers are updated correctly in  the Spatial Subscription Server database for each ESDT being updated.&lt;/i&gt;</p>		#comment
43	<p>Verify each ESDT has an INSERT event definition:&lt;br /&gt;&lt;br /&gt;select *&lt;br /&gt; /&gt;from EcNbEventDefinition&lt;br /&gt;where ESDT_Id =  '\${SHORTNAME}'&lt;br /&gt;and VersionID = \${VERSIONID}</p>		
44	<p>From each descriptor file, get a list of EVENTPARM names:&lt;br /&gt;&lt;br /&gt;sed  -n /^\s*GROUP\s*=\s*EVENT\s*\$/,/^END_GROUP\s*=\s*EVENT\s*\$/  {&lt;br /&gt;  /\s*OBJECT\s*=\s*INSERT\s*\$/,/\s*END_OBJECT\s*=\s*INSERT\s*\$/  {&lt;br /&gt;  /\s*OBJECT\s*=\s*EVENTPARMS\s*\$/,/\s*END_OBJECT\s*=\s*EVENT  TPARMS\s*\$/ {&lt;br /&gt;   /\s*OBJECT\s*=\s*EVENTPARMS\s*\$/ d&lt;br /&gt;  s/\s*OBJECT\s*=\s*//p&lt;br /&gt;   }&lt;br /&gt; }&lt;br /&gt;}' \${DESCRIPTOR_FILE}</p>		
45	<p>For each ESDT, find the EVENTPARMS that are eligible to be added to the  Spatial Subscription Server database:&lt;br /&gt;&lt;br /&gt;select attributename&lt;br /&gt; /&gt;from ecnbeventmetadataattrdef&lt;br /&gt;where attributename in  (\${EVENTPARMS})&lt;br /&gt;order by attributename;</p>		
46	<p>For each ESDT, find the EVENTPARMS that were associated with that  ESDT as event qualifiers:&lt;br /&gt;&lt;br /&gt;select attributename&lt;br /&gt;from  ecnbeventattrxref&lt;br /&gt;where esdt_id = '\${SHORTNAME}'&lt;br /&gt;and  versionid = \$VERSIONID&lt;br /&gt;order by attributename;</p>		
47	<p>Verify the associated event qualifiers match the eligible list with the  exception that&lt;br /&gt;&lt;br /&gt;'RangeBeginningTime' and 'RangeEndingTime'  from the descriptor get combined into 'GranuleTimeRange' in the  database;&lt;br /&gt;&lt;br /&gt;'RangeBeginningDate' and 'RangeEndingDate' from the  descriptor get combined into 'GranuleDateRange' in the database.</p>		
48	<p>Verify that, upon completion of updates, the ESDT Maintenance GUI  displays a message saying that the ESDT changes will take effect only after  the Ingest service is re-started.</p>		

**TEST DATA:**

Any current data types.

Initial ESDTS: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2060/Replacement

Updated ESDTS: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2120

The 'update' ESDT descriptor files have been modified as follows:

1. removing one or more existing collection level attributes that are not restricted  
DsESDTGIGLA01.013.desc
2. removing DLLName parameter from the Collection metadata group  
DsESDTGIGLA02.013.desc
3. changing the ShortName in the CollectionAssociation group  
DsESDTGIGLA03.013.desc
4. changing a mandatory inventory attribute to optional  
DsESDTGIGLA04.013.desc
- 5 adding one or more optional inventory metadata attributes  
DsESDTGIGLA05.013.desc
6. adding one or more new qualifiers on existing events  
DsESDTGIGLA06.013.desc  
QAPercentInterpolatedData

**EXPECTED RESULTS:**

4.1.3 Delete multiple ESDTs, success case (ECS-ECSTC-609)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>DS_7E_01 ESDT Maintenance GUI ITP: <a href="http://dmsserver.gsfc.nasa.gov/release721/ESDT_Maint_GUI/ITP_DS_7E_01_ESDTGUI.doc"></a></i>		#comment
2	<i>Preconditions</i>		#comment
3	Ensure a database client is connected to the ecs database.		
4	Identify 3 ESDTs for deletion (C1, C2, C3).		
5	Save the collection IDs of C1, C2, C3. For each,  select collectionid from amcollection where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID}		

#	Action	Expected Result	Notes
6	Save the attribute IDs associated with each of C1, C2, C3. For each, <pre>select attributeid from amcollection c join dsmdcollectionaddnlattrbsxref x on c.collectionid = x.collectionid where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID}</pre>		
7	Ensure C1, C2, C3 have no granules. For each, <pre>select esdt(shortname, versionid), granuleid from amgranule where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID}</pre>		
8	Ensure C1, C2, C3 are absent from the data pool. For each, <pre>select * from amcollection where groupid is null and shortname =  '\${SHORTNAME}' and versionid = \${VERSIONID}</pre>		
9	Ensure C1, C2, C3 are absent from the Spatial Subscription Server database. For each, <pre>select * from EcNbSubscription where esdt_id = '\${C1_SHORTNAME}' and versionid = \${C1_VERSIONID}</pre>		
10	<i>S-1</i> From the ESDT Maintenance GUI select a group of ESDTs for deletion. Each of the ESDTs selected must satisfy the following conditions to for deletion to be possible: 1. the ESDT contain no granules in the AIM inventory database 2. the ESDT is not defined within the Data Pool 3. the Spatial Subscription service contains no active or inactive subscriptions referencing the ESDT		#comment
11	Note the current time as t0.		
12	On the ESDT Maintenance GUI "ESDT List" page, select C1, C2, and C3.		
13	Click the "Delete Selected ESDTs" button at the bottom of the page to begin the ESDT deletion.		
14	Select OK at the confirmation prompt.		
15	<i>V-1</i> Verify that each ESDT was deleted successfully. 1. The MCF file was deleted 2. The ESDT XML schema file was deleted 3. The ESDT descriptor file was deleted. 4. The ESDT basic collections and PSA associations were deleted from the Inventory database 5. The insert events and event qualifiers for the ESDT were deleted from the Subscription Server database 6. The XML metadata file directories associated with the ESDT were removed from the XML archive 7. PSA definitions are removed if there are no other ESDTs associated with them		#comment
16	1. Verify no MCF files exist for C1, C2, or C3. For each ESDT, <pre>ls -l /stornext/smallfiles/\${MODE}/mcf/\${SHORTNAME}#\${VERSIONID}.M</pre>		

#	Action	Expected Result	Notes
	CF  For example,  ls -l /stornext/smallfiles/OPS/mcf/MOD29P1D#005.MCF		
17	2. Verify no ESDT XML schema files exist for C1, C2, or C3. For each ESDT,   /stornext/smallfiles/\${MODE}/descriptor/*\${SHORTNAME}.\${VERSIONID}.xsd  For example,  ls -l /stornext/smallfiles/OPS/descriptor/*MOD29.003.xsd		
18	3. Verify no ESDT descriptor files exist for C1, C2, or C3. For each ESDT,   /stornext/smallfiles/\${MODE}/descriptor/*\${SHORTNAME}.\${VERSIONID}.desc  For example,  ls -l /stornext/smallfiles/OPS/descriptor/*MOD29.003.desc		
19	4. Verify the collections table has no rows for C1, C3, or C3. For each ESDT,  select * from amcollection where shortname = '\${C1_SHORTNAME}' and versionid = \${VERSIONID}		
20	4. Verify additional attribute associations no longer exist for C1, C2, C3.  select * from dsmdcollectionaddnlattrbsxref where collectionid in (\${COLLECTIONIDS})		
21	5. Verify Subscription Server database has no event definitions for C1, C2, or C3. For each,  select * from ecnbeventdefinition where esdt_id = '\${SHORTNAME}' and versionid = \${VERSIONID}		
22	5. Verify Subscription Server database has no event qualifiers for C1, C2, or C3. For each,  select * from ecnbeventattrxref where esdt_id = '\${SHORTNAME}' and versionid = \${VERSIONID}		
23	6. Verify no XML metadata directory exists in the small file archive for C1, C2, or C3. For each,  ls -ld /stornext/smallfiles/\${MODE}/metadata/\${SHORTNAME}.\${VERSIONID}   For example,  ls -ld /stornext/smallfiles/OPS/metadata/MOD29.003		
24	7. Verify the additional attribute definitions for C1, C2, C3 no longer exist:  select attributeid from dsmdadditionalattributes where attributeid in (\${ATTRIBUTEIDS})  If any previously associated attributes remain, verify they are associated with existing collections:  select distinct(attributeid) from dsmdcollectionaddnlattrbsxref where attributeid in (\${REMAINING_ATTRIBUTEIDS})  should return the same list of \${REMAINING_ATTRIBUTEIDS}.		

#	Action	Expected Result	Notes
25	<i>V-2 Verify that the ESDT Maintenance GUI displays a message indicating number of ESDTs deleted.</i>		#comment
26	Verify the ESDT Maintenance GUI displays a message indicating 3 ESDTs were deleted.		
27	<i>V-3 Verify that the ESDT Maintenance GUI logs the processing activities, including time of request, action requested, ESDT ShortName, Version ID, descriptor file name, and the result of the operation, in a configured log file.</i>		#comment
28	ssh to the ESDT Maintenance GUI host.		
29	Search the ESDT GUI log for the first mentions of C1, C2, and C3 on or after the time t0.  /usr/ecs/\${MODE}/CUSTOM/logs/ESDTMaintenanceGUI.*.log*		
30	Verify each of the following is logged for each of C1, C2, C3:  request time action requested (delete) ShortName Version ID descriptor file name operation result		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**4.2 DMS**

**4.2.1 Single Granule External Processing Order (ECS-ECSTC-610)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Submit a scp order and a HTTPS order through EWOC.		
2	<i>Verify the following occurred:</i>		#comment
3	a) The Submit message contained the order ID for each order.		

#	Action	Expected Result	Notes
4	b) The orders showed up on the Order Manager GUI with scp as the Media for the scp request and HTTPS for the HTTPS request.		
5	c) The orders shipped in OMS.		

TEST DATA:

EXPECTED RESULTS:

#### 4.2.2 Cancelled and Failed Granules (ECS-ECSTC-611)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Submit an order through EWOC and then cancel the order in the OMS GUI.		
2	After the order has been canceled, verify that the EWOC sends an order update status message to ECHO indicating that the order has been canceled within 5 minutes of order status update in ECS.		
3	Repeat above test but this time fail the order in the OMS GUI.		

TEST DATA:

EXPECTED RESULTS:

### 4.3 DPL Ingest

#### 4.3.1 Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-612)

DESCRIPTION:

---

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Data Providers can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[View Data Providers]</i>		#comment
3	As the 'ingest admin' operator, navigate to the Data Provider page on the Data Pool Ingest GUI.		
4	Verify that all provider information in the INGST CI database appears correctly on the Data Provider page.		
5	<i>[Configure Data Providers]</i>		#comment
6	Edit existing provider information (if any) and define new providers to fulfill the following requirements:  At least one data provider should have an FTP notification method,  at least one data provider should have an scp notification method,  at least one data provider should have an http notification method,  at least one data provider should have an email notification method,  at least one data provider should have a combination FTP/email notification method,  and at least one data provider should have a combination scp/email notification method.  and at least one data provider should have a combination http/email notification method. Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des. At least one provider must use active mode. At least one should use passive mode.		
7	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit data provider information.		
8	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing data providers before saving this information.		
9	Verify that the ingest admin operator has the authorization to define new data providers (i.e., has access to the data provider pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
10	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16110 to be entered.		
11	Verify that all of the notification methods in S-DPL-16150 can be entered or selected on the Data Pool Ingest GUI, as appropriate for the selected transfer method as per S-DPL-16110.		

#	Action	Expected Result	Notes
12	For one provider with an FTP Notification method, verify that the Data Pool Ingest GUI allows the related FTP information to be entered (FTP host, destination directory, login id, password)		
13	Verify that the FTP password entered is not shown or stored in the clear.		
14	Verify that the FTP password entered is not shown in the Data Pool Ingest GUI log.		
15	For one provider with an email notification method, verify that the Data Pool Ingest GUI allows the related email address to be entered.		
16	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new data providers before saving this information.		
17	For one provider with an scp notification method, verify that the Data Pool Ingest GUI allows the related scp information to be entered (scp host, destination directory, login id, password, scp type, and cipher).		
18	Verify that the scp password entered is not shown or stored in the clear.		
19	Verify that the scp password entered is not shown in the Data Pool Ingest GUI log.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**4.3.2 Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-613)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Polling Locations can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure Polling Locations]</i>		#comment
3	As the Ingest Admin operator use the Data Pool Ingest GUI to define all polling locations that will be used for testing this ticket. (Table of data		

#	Action	Expected Result	Notes
	provider to polling location mappings, and related S-DPL-16230 information for each polling location TBS as part of ITP).  At least one polling location should have an FTP polling method,  at least one polling location should have a local polling method,  and at least one polling location should have an scp polling method.  and at least one polling location should have an http polling method. At least one polling location using FTP shall be for a provider using local transfers.		
4	Verify that the ingest admin operator has the authorization to define the polling locations (i.e., has access to the polling location pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
5	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16230 to be entered.		
6	Verify that all of the polling methods in S-DPL-16250 can be entered or selected on the Data Pool Ingest GUI.		
7	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new polling locations before saving this information.		
8	<i>[Edit Polling Locations]</i>		#comment
9	For one polling location, as the 'ingest admin' operator, edit all of its existing configuration parameters. (NOTE: After this criterion is complete, values of these configuration parameters should be reset to appropriate values for processing all PDRs in criterion 300).		
10	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit polling location information.		
11	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing polling locations before saving this information.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

4.3.3 Use the DPL Ingest GUI to configure Remote Transfer Hosts (ECS-ECSTC-614)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Remote Transfer Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure FTP hosts]</i>		#comment
3	As the 'ingest admin' operator, navigate to the FTP Host page on the Data Pool Ingest GUI.		
4	Edit existing ftp host information (if any, from the INGST CI), per S-DPL-16260, and define new ftp hosts such that all ftp hosts which will be used for testing this ticket are defined. (Table of ftp hosts and related S-DPL-16260 information TBS as part of ITP). At least two ftp hosts should be defined that are not configured for the INGST subsystem.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit ftp host information.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing ftp hosts and definitions of new ftp hosts before saving this information.		
7	Verify that the ingest admin operator has the authorization to define the ftp hosts (i.e., has access to the ftp host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
8	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16260 to be entered.		
9	<i>[Configure HTTP hosts]</i>		
10	Configure a HTTP host use the step similar to 'Configure FTP hosts'.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

4.3.4 Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-615)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies SCP Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure hosts for scp access]</i>		#comment
3	As the 'ingest admin' operator, navigate to the host page on the Data Pool Ingest GUI.		
4	Edit existing information for hosts accessed via scp (if any, from the INGST CI), per S-DPL-16290, and define new scp hosts such that all scp hosts which will be used for testing this ticket are defined. (Table of hosts accessed via scp, and related S-DPL-16290 information TBS as part of ITP.) Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des, i.e., the test needs to involve several different providers.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to enter and edit the scp host information in S-DPL-16290.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing scp host parameters and definitions of new scp host parameters before saving this information.		
7	Verify that the ingest admin operator has the authorization to define the scp host parameters in S-DPL-16290 (i.e., has access to the host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		

TEST DATA:

Any current data types

EXPECTED RESULTS:

4.3.5 View and modify the DPL Ingest GUI configuration (ECS-ECSTC-616)

DESCRIPTION:

---

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies DPL Ingest GUI configuration can be viewed and modified correctly.</i>		#comment
2	<i>[View collection configuration]</i>		#comment
3	As an ‘ingest admin’ operator, use the Data Pool Ingest GUI to list all data types for which configuration parameters were entered in criterion 120.		
4	Verify that “Ignore Validation Warnings” and “Public in Data Pool” are displayed for each ESDT.		
5	Verify that the Data Pool Ingest GUI provides the operator a method to quickly select or scan for a subset of existing Data Pool collections without having to enter the full ESDT name and version.		Since release 8.2, newly installed data types will be automatically added to the DPL Ingest InDataType table and visible in 'Data Types with DPL Ingest Configurations' table.
6	<i>[Edit collection configuration]</i>		#comment
7	As an ‘ingest admin’ operator, use the Data Pool Ingest GUI to edit publication policy configuration parameters for a Data Pool collection.		
8	Verify that the edited configuration parameters are correctly updated in the database.		
9	<i>[Reconfigure data type parameters.]</i>		#comment
10	Log into the DPL Ingest GUI as an ‘ingest admin’ operator, and configure a data type (ESDT and Version) to be published in the public Data Pool upon insert.		
11	Submit several PDRs for the data type configured above. There needs to be a sufficient number of PDRs and granules such that some granules will complete ingest before the configuration change made during the test (see next step), and some granules will not start ingest until after the configuration change has been applied.		
12	After the first few granules completed ingest, re-configure the data type via the DPL Ingest GUI such that the data type no longer will be published in the Data Pool. Also change the minimum retention period sufficiently long so that the ingested granules are not cleaned up immediately after archiving, to allow time for the verification steps below.		

#	Action	Expected Result	Notes
13	Verify that the first few granules that were ingested before the re-configuration are queued with the Data Pool Insert Service for insertion into the public Data Pool area.		
14	Verify that all granules that started ingest one minute or more after the re-configuration are inserted into the non-public Data Pool area, and are NOT queued with the Data Pool Insert Service for insertion into the public Data Pool area. NOTE: if the Science granule has an associated Browse the Browse will be published even if the Science granule is &quot;hidden.&quot;		
15	Verify that ALL browse granules ingested as part of the test are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**4.3.6 Add & Modifying Volume Groups (ECS-ECSTC-617)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>&lt;i&gt;Description and Objectives: Verifies it is possible to add and modify volume groups using the DPL Ingest GUI and that the server correctly uses the new volume groups.&lt;/i&gt;</i>		#comment
2	<i>&lt;i&gt;[Add &amp; Modifying Volume Groups]&lt;/i&gt;</i>		#comment
3	Using the DPL Ingest GUI add a primary and backup Volume Group for a collection.		
4	Ingest a granule and verify that it goes to the primary and backup volume groups.		
5	Modify Volume group and give it a new path		
6	Ingest a granule and verify that it goes to the new path.		

#	Action	Expected Result	Notes
7	Click on reports and verified that the addition and modification from above are displayed in the report.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**4.4 DPL**

**4.4.1 Data Pool Maintenance GUI - Check Batch Insert Status (ECS-ECSTC-618)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Data Pool Maintenance GUI – Check Batch Insert Status]</i>		#comment
2	Use the Data Pool Maintenance GUI to check the status of the batch insert (by Publish Utility) using the Batch Summary tab and also using the List Insert Queue tab (filter by batch label).		
3	Verify that the GUI correctly reports the status in the DIInsertActionQueue.		

**TEST DATA:**

Any current data type

**EXPECTED RESULTS:**

## 4.5 CLS

### 4.5.1 OBSOLETE - Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-619)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request the status of a single, completed order that was handled by the OMS and that contains at least two requests by specifying a valid Order Id and user contact email address.  Each request in the order should contain at least 100 granules that have NOT been processed by HEG or the external subsetter.		
2	<i>Verify the following:</i>		#comment
3	a. Order Id is correctly displayed.		
4	b. Submission date/time is correctly displayed.		
5	c. Order state is correctly displayed.		
6	d. Order completion date/time is correctly displayed.		
7	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed		
8	f. Order state and request states are presented in terms that an end user can understand.		
9	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
10	h. There is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

### 4.5.2 OBSOLETE - Order Status GUI: Display Order Status based on History Range, 0D\_S6\_05, Criterion 180 (ECS-ECSTC-620)

DESCRIPTION:

---

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request order history by specifying a starting and ending date, user contact email address, and one valid Order Id for the user.  Ensure that at least 100 orders are displayed that represent a mix of orders for which request details are available for some orders and not available for others.  Ensure that the date range requires the Order Status Interface to retrieve information from both the Order Manager operational tables and archive tables.  Ensure that at least one of the orders contains granules that were processed by the external subsetter and one of the orders contains granules that were processed by HEG.		
2	Verify that the correct orders are returned and sorted by submission date/time.		
3	<i>For each order verify the following:</i>		#comment
4	a. Order Id is correctly displayed.		
5	b. Submission date/time is correctly displayed.		
6	c. Order state is correctly displayed.		
7	d. Order completion date/time is correctly displayed.		
8	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed, including the processing description for the granule to be processed by the external subsetter, in a manner which should be generally understandable by the user.		
9	f. Order state and request states are presented in terms that an end user can understand.		
10	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
11	h. An indication is provided when request details are not available for an order.		

TEST DATA:

EXPECTED RESULTS:

## 4.6 Data Access

### 4.6.1 Use DataAccess GUI to configure service for datatypes (ECS-ECSTC-623)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Go to the DataAccess Configuration Interface GUI		
2	Under the Collection Configuration tab, right click on an ESDT. Select &quot;Configure New Service for Collection&quot; Add the GDAL, HEG, GLAS, OPENDAP service.		
3	Verify the service is added successfully.		
4	Under the Service Configuration tab, right click on the same ESDT. Select &quot;Remove Collection from Service&quot; to remove the collection from the service.		
5	Go to the Collection Configuration tab to verify that the collection is no longer configured for the service.		

TEST DATA:

EXPECTED RESULTS:

### 4.6.2 Start and navigate through the Data Access Configuration Interface (ECS-ECSTC-627)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Start the Data Access Configuration Interface GUI.		
2	Go through each tab and verify that it displays correctly.		

---

TEST DATA:

EXPECTED RESULTS:

### 4.7 BMGT

#### 4.7.1 Connect to CMR (ECS-ECSTC-678)

DESCRIPTION:

PRECONDITIONS:

Pre-Conditions

- \*Verify configurations - config files, properties files or database settings validate correctly for mode, host, and application
- \*Verify database connections to ecs connected to ecs db successfully in the configured mode
- \*Verify ECHO REST API service connections to ECHO connected to ECHO REST API successfully
- \*Verify collections under /sotestdata/DROP\_802/BE\_82\_01/630 with ECS metadata
- \*Ensure collection C1 has been installed in the mode.
- \*Ensure collection C1 is enabled for Collection and Granule Export.
- \*Assume <user2> as a regular user without update privileges (default, read-only).
- \*Assume <user1> as a privileged user with configuration update privileges (password required; write access).
- \*Assume <EchoURL1> and <EchoURL2> are two Mock ECHO connections
- \*Ensure a mock ECHO or TCP proxy is configured to capture and log BMGT requests.

STEPS:

#	Action	Expected Result	Notes
1	#S-1 Go to the BMGT GUI without logging in as a privileged user. View and attempt to modify the CMR connection information (ECHO URL, username, password, provider, etc.) Login to BMGT GUI as , without a password.		
2	On the BMGT Configuration tab, attempt to change the value of the property BMGT.Exporter.IngestClient.URL to a different URL, and save the configuration.		
3	#V-1 Verify that in S-1, the GUI allows you to view the CMR connection configuration, but prevents its update.		

---

---

#	Action	Expected Result	Notes
	Verify that the password is not shown in plain text.		
4	Verify that user2 cannot update the CMR connection configuration information		
5	Verify that when entering the CMR password, the password is not displayed in plain text		
6	#S-2 Log in to the BMGT GUI as a privileged user and update the CMR connection information (ECHO URL, username, password, provider, etc.). Save the new configuration values. It may be useful to change to incorrect configuration, so it is easier to tell when the new values are being used.		
7	Login to the BMGT GUI with a privileged password.		
8	On the BMGT Configuration tab, attempt to change the value of the property BMGT.Exporter.IngestClient.URL to a different URL, and save the configuration.		
9	#V-2 Verify that in S-2, the GUI allows you to both view and update the CMR connection configuration. Verify that the password is never shown in plain text, even when it is being entered.		
10	Verify that user1 can update the CMR connection configuration information		
11	Verify that when entering the CMR password, the password is not displayed in plain text		
12	#S-3 Ensure that there are pending export requests for BMGT to work off. *Enable Automatic Export		
13	Ingest granules g1 and g2 into Collection C1 (or alternatively perform a manual export for 2 granules and a collection)		
14	Verify that the requests are shown pending on the GUI		
15	#V-3 Verify that any exports which are picked up by BMGT and begin generation after the new values are saved (plus a reasonable lag time), are exported using the new configuration.		

TEST DATA:

EXPECTED RESULTS:

#### 4.7.2 GUI (ECS-ECSTC-679)

##### DESCRIPTION:

##### PRECONDITIONS:

###### Pre-Conditions

- \*Ensure a TCP proxy or mock ECHO is capturing BMGT traffic.
- \*Ensure at there are at least 10 failed BMGT exports.

E.g., in the small file archive modify a granule's metadata so that the RangeBeginningDate occurs after the RangeEndingDate; manually export the granule 10 times.

Alternatively, configure mock ECHO to reject a specific granule, then manually export that granule at least 10 times.

- \*Ensure collections C1, C2 are installed.
- \*Ensure collection C1 is disabled for collection export.
- \*Ensure collection C2 is enabled for collection and granule export.
- \*Ensure granules G1, G2 are in AIM.

##### STEPS:

#	Action	Expected Result	Notes
1	#S-1 All of the following steps shall be performed using each of the browsers and operating systems listed in “Technical Document 910-TDA-042, Browsers Baseline” *[FIXME] Identify browsers an operating systems in 910-TDA-042. Repeat the test for each combination.		
2	#S-2 Navigate to the BMGT GUI. Enter as an operator, without administrative privileges. *Open the BMGT GUI in a browser. Do not log in with a password.  The BMGT GUI does not use roles (such as operator or administrator). If a password is supplied, read-write access is granted. Without a password, access is limited to read-only.		
3	#S-3 Attempt to suspend BMGT processing via the GUI. *In the system status tab, click the 'Pause' button next to 'Dispatcher'		
4	#S-4 Navigate to the collection configuration page, and attempt to enable or disable a collection for export.		

#	Action	Expected Result	Notes
	*In the BMGT GUI, on the collection configuration page, attempt to enable or disable a collection for export by clicking the collection export check box next to it.		
5	#S-5 Navigate to the BMGT configuration page, and attempt to modify a configuration parameter. *In the BMGT GUI, on the BMGT configuration page, attempt to modify a configuration parameter.		
6	#S-6 Navigate to the current/history export page. Constrain the listed export requests to cover only a specific time period. *In the BMGT GUI, on the export request tab, modify the time range for which to display export requests.		
7	#S-7 Navigate to the failed export page. Constrain the errors to cover only a specific time period. Ensure that there are failed export requests in the system prior to doing this. *In the BMGT GUI, on the export activity/error tab, filter for failed activities only and then modify the time range for which to display errors.		
8	#S-8 Log in to the BMGT GUI as an administrator. *Log in to the BMGT GUI as an administrator.		
9	#S-9 On the main GUI page, suspend BMGT processing, both overall, and for one type of export (e.g. automatic). *In the BMGT GUI, on the main page, suspend overall BMGT processing by pressing the 'Pause' button next to 'Dispatcher'.		
10	In the BMGT GUI, on the main page, suspend automatic BMGT processing by pressing the 'Pause' button next to 'EVENT'.		
11	Request a manual export of granule G1:  EcBmBMGTManualStart --mode --metg --granules		
12	*Update granule G2 by changing its DayNightFlag:  update amgranule set daynightflag = 'Both' where granuleid = ##This is not used by BMGT. BMGT uses the value in the native XML. though it may generate a GRUPDATE event... not sure 3/21/2013 -- Goff, Timothy		
13	#S-10 Navigate to the collection configuration page. Enable a currently disabled collection for export. Disable a currently enabled collection.		

---

---

#	Action	Expected Result	Notes
	*In the BMGT GUI, on the collection configuration page, enable collection C1 for export.		
14	In the BMGT GUI, on the collection configuration page, disable collection C2 for export.		
15	#S-11 Navigate to the BMGT configuration page. Modify and save the email address to which alerts are sent. *In the BMGT GUI, on the BMGT configuration page, modify the email address to which alerts are sent.		
16	#S-12 On the main GUI page, resume all BMGT processing. *In the BMGT GUI, on the main GUI page, resume all BMGT processing.		
17	Log out of the BMGT GUI.		
18	Clear the browser's history, including cache, cookies, and saved sessions.		
19	Restart the browser.		
20	V-1 Verify that all verification steps pass for each of the browsers and operating systems referred to in S-1.		
21	V-2 Verify that the main GUI page in S-2 provides the number of completed metadata exports, number of pending metadata exports, and statistics about retries, warning, and errors.		
22	Verify the BMGT GUI "Export Requests" page lists the number of completed metadata exports.		
23	Verify the BMGT GUI "Export Requests" page lists the number of pending metadata exports.		
24	Verify the BMGT GUI "Export Activity/Errors" page lists the number of errors.		
25	V-3 Verify that in S-3, it is not possible to suspend BMGT processing. Verify the BMGT GUI prevents read-only access from suspending BMGT processing.		
26	V-4 Verify that in S-4, all collections are displayed, grouped by their datapool group. Verify the BMGT GUI collection configuration page displays all collections, grouped by datapool group.		
27	V-5 Verify that in S-4, it is not possible to enable or disable a collection. Verify the BMGT GUI prevents read-only access from enabling or disabling a collection.		
28	V-6 Verify that in S-5, it is not possible to modify any configuration parameters. Verify however that the notification email address is visible on this page.		

---

---

#	Action	Expected Result	Notes
29	Verify the BMGT GUI prevents read-only access from changing configuration parameters.		
30	Verify the BMGT GUI BMGT configuration page displays the notification email address.		
31	V-7 Verify that in S-6, information is displayed for recent and pending export requests (it is allowable for recent and current request to be displayed on separate pages). Verify that each export request is clearly marked as to which type of export initiated it (e.g. automatic, manual, verification).		
32	Verify the BMGT GUI "Export Requests" page displays recent and pending export requests.		
33	Verify the BMGT GUI "Export Requests" page marks each export as one of automatic, manual, or verification.		
34	V-8 Verify that in S-7, information is displayed for recent export failures and errors. Verify the BMGT GUI "Export Activity/Errors" page displays recent export failures and errors.		
35	V-9 Verify that in S-6 and S-7, it is possible to constrain the items displayed on the page by time. Verify the BMGT GUI current/history allows filtering to display only those export requests withing a certain time range.		
36	Verify the BMGT GUI failed export page allows filtering to display only those failed exports within a certain time range.		
37	V-10 Verify that after logging in as an administrator, suspending processing in S-9 results in a halting of BMGT processing. Verify that the manual export request of granule G1 does not get processed.		
38	Verify that granule G2's update does not get automatically processed. V-11 Verify that in S-10, it is possible to enable and disable export for collections, both for granule and collection metadata export. *Verify the BMGT GUI collection configuration page allows enabling collection C1 for export. *Verify the BMGT GUI collection configuration page allows disabling collection C2 for export.		
39	V-11 Verify that in S-10, it is possible to enable and disable export for collections, both for granule and collection metadata export. Verify the BMGT GUI collection configuration page allows enabling collection C1 for export.		

---

---

#	Action	Expected Result	Notes
40	Verify the BMGT GUI collection configuration page allows disabling collection C2 for export.		
41	V-12 Verify that in S-11, the change made to the notification email address is saved to the database. Verify after restarting the browser, the BMGT GUI BMGT configuration page displays the new email address to which alerts are sent.		
42	V-13 Verify that after requesting resumption of BMGT processing in S-12, export requests resume being picked up and processed. Verify after resuming BMGT processing, granule G1 is processed to completion as a manual export.		
43	Verify after resuming BMGT processing, granule G2 is processed to completion as an automatic export.		

**TEST DATA:**

**EXPECTED RESULTS:**

**4.7.3 Configuring Collections for Export (ECS-ECSTC-680)**

**DESCRIPTION:**

**PRECONDITIONS:**

**Pre-Conditions**

- \*Ensure collection C1 is not installed.
- \*Ensure bg\_collection\_configuration has no row for collection C1.
- \*Ensure granules g1, g2, g3 are not in AIM.
- \*Ensure collections C2, C3, C4, C5, C6 are installed.
- \*Ensure granules g4 ... g18 are in AIM
- \*Ensure collection C2 is disabled for both collection and granule export.
- \*Ensure collection C3 is enabled for collection but not for granule export.
- \*Ensure collection C4 is enabled for both collection and granule export.
- \*Ensure collection C5 is disabled for collection or and granule export.
- \*Ensure collection C6 is enabled for both collection and granule export.
- \*Ensure a TCP proxy or mock ECHO is capturing BMGT HTTP traffic.

STEPS:

#	Action	Expected Result	Notes
1	S-1 Add a new ESDT to the ECS system and ingest some granules in that collection. Install collection C1.		
2	Add a row for collection C1 to the bg_collection_configuration table, setting both export flags to 'N'.  See EcBgPopulateCollections.ksh for examples.		
3	Ingest granules g1, g2, g3.		
4	S-2 Go to the GUI collection configuration page. Visit the BMGT GUI Collection Configuration page.		
5	V-1 On the collection configuration page in S-2, verify that all currently installed collections are listed with their current enabled/disabled status.		
6	Verify the BMGT GUI's collection configuration page lists collections C1 ... C6.		
7	Verify the BMGT GUI indicates C1 is disabled for both collection and granule export.		
8	Verify the BMGT GUI indicates C2 is disabled for both collection and granule export.		
9	Verify the BMGT GUI indicates C3 is enabled collection export but disabled for granule export.		
10	Verify the BMGT GUI indicates C4 is enabled for both collection and granule export.		
11	Verify the BMGT GUI indicates C5 is disabled for both collection and granule export.		
12	Verify the BMGT GUI indicates C6 is enabled for both collection and granule export.		
13	V-2 On the collection configuration page in S-2, verify that the new collection is listed but not enabled for collection or granule export.		
14	Verify collection C1 appears on the BMGT GUI's collection configuration page.		
15	Verify the BMGT GUI indicates C1 disabled for both collection and granule export.		
16	S-3 Enable the collection for collection export. Enable C1 for collection export by clicking the collection export check box next to the collection.		

#	Action	Expected Result	Notes
17	V-3 After enabling the collection for collection export in S-3, verify that the collection metadata for the collection is automatically exported.		
18	Verify that after C1 is enabled for collection export, the BMGT GUI indicates C1's metadata is successfully exported.		
19	Verify that after C1 is enabled for collection export, a BMGT log records the export of C1's metadata.		
20	Verify that after C1 is enabled for collection export, the TCP proxy logs an HTTP PUT whose body contains C1's metadata.	<p>+The request should begin with a line such as</p> <pre>PUT /catalog-rest/providers/\${PROVIDERID}/datasets/\${DATASETID} HTTP/1.1</pre> <p>where \${DATASETID} is the URL-encoded longname + ' V' + versionid, e.g.,</p> <pre>MODIS%2FAqua%20Gross%20Primary%20Productivity%208-Day%20L4%20Global%201km%20SIN%20Grid%20V005</pre> <p>#S-4 Allow the export of the collection metadata to complete. *Wait for collection C1's export request to be marked complete:</p> <pre>select r.completiontime, r.status from bg_export_request r join amcollection c on c.collectionid = r.collectionid where c.ShortName = " and c.VersionId = and itemtype = 'CL';</pre> <p>Or, wait for the TCP proxy to log C1's HTTP PUT request.</p>	
21	S-5 Enable the collection for granule export.		

#	Action	Expected Result	Notes
	In the BMGT GUI, enable C1 for granule export by checking the granule export check box next to the collection in the collection configuration tab.		
22	V-4 After enabling the collection for granule export in S-5, verify that the granule metadata for every granule in the collection is automatically exported.		
23	Verify that after C1 is enabled for granule export, the BMGT GUI indicates all metadata for all granules belonging to C1 is successfully exported (except logically deleted granules).		
24	Verify that after C1 is enabled for granule export, a BMGT log records the export of C1 granules' metadata (except logically deleted granules).		
25	Verify that after C1 is enabled for granule export, the TCP proxy logs an HTTP PUT for each of C1's granules, containing granule metadata (except logically deleted granules).	<p>Each granule request should begin with a line such as</p> <p>PUT /catalog-rest/providers/\${PROVIDERID}/granules/\${GRANULEUR} HTTP/1.1</p> <p>where \${GRANULEUR} is a URL-encoded geoid, such as</p> <p>SC%3AMYD17A2.005%3A20062</p> <p>#S-6 Find a collection which is not enabled for collection or granule export. Request the manual export of collection and granule metadata for this collection. *Request manual export of C2:</p> <p>EcBmBMGTManualStart --mode --metg --metc --collections .</p>	
26	#S-6 Find a collection which is not enabled for collection or granule export. Request the manual export of collection and granule metadata for this collection.		
27	Request manual export of C2:  EcBmBMGTManualStart --mode --metg --metc --collections .		
28	V-5 For the export attempt in S-6, verify that nothing is exported.		

#	Action	Expected Result	Notes
29	Verify the TCP proxy logs no request for C2's collection or granule metadata.		
30	Verify a BMGT log indicates no request is attempted for C2's collection or granule metadata.		
31	S-7 Find a collection which is enabled for collection, but not granule export. Request the manual export of collection and granule metadata for this collection.		
32	Request manual export of C3:  EcBmBMGTManualStart --mode --metg --metc --collections .		
33	V-6 For the export attempt in S-7, verify that only collection metadata is exported.		
34	Verify the TCP proxy logs a single HTTP PUT request, containing C3's collection metadata.  Get the request ID for the collection export from the GUI. Look in the BMGT manual log for pattern like  &quot;requestId&quot;:18629,&quot;batchId&quot;:76		
35	Verify the TCP proxy logs no HTTP PUT requests for C3's granules.		
36	Verify a BMGT log records a single export request for C3's collection metadata.		
37	Verify a BMGT log records no export attempts for any C3 granule metadata.		
38	S-8 Find a collection which is enabled for collection and granule export. Request the manual export of collection and granule metadata for this collection.		
39	Request manual export of C4:  EcBmBMGTManualStart --mode --metg --metc --collections .		
40	V-7 For the export attempt in S-8, verify that both collection and granule metadata is exported.		
41	Verify the TCP proxy logs a single HTTP PUT request, containing C4's collection metadata.		
42	Verify the TCP proxy logs a single HTTP PUT request for each of C4's granules, containing granule metadata (excepting any granules which are logically deleted).		
43	Verify a BMGT log records a single export request for C4's collection metadata.		

#	Action	Expected Result	Notes
44	Verify a BMGT log records a single export request for each of C4's granules (excepting those which are logically deleted).		
45	S-9 Find a collection which is not enabled for collection or granule export, but for which there is another ESDT with the same short name, but different version, which is. Request the manual export of collection and granule metadata for this collection.		
46	Request manual export of C5:  EcBmBMGTManualStart --mode --metg --metc --collections .		
47	V-8 For the export attempt in S-9, verify that nothing is exported.		
48	Verify the TCP proxy logs no request for C5's collection or granule metadata.		
49	Verify a BMGT log indicates no request is attempted for C5's collection or granule metadata.		

TEST DATA:

EXPECTED RESULTS:

4.7.4 Incremental Verification – Nominal (ECS-ECSTC-681)

DESCRIPTION:

PRECONDITIONS:

Pre-Conditions

- \*Ensure no activity other than the test will occur in the mode for the duration of the test.
- \*Ensure BMGT configuration is current and correct (config files, properties files, database settings).
- \*Ensure a PostgreSQL prompt is available and connected to the ecs database.
- \*Ensure test data is available under /sotestdata/DROP\_802/BE\_82\_01/Criteria/520.
- \*Ensure collections C1..C10 are installed. E.g., ensure the DPL Ingest GUI shows C1..C10 as configured datatypes.
- \*Ensure collections C1..C10 are enabled for collection and granule export:

```
update bg_collection_configuration  
set collectionexportflag = 'Y', granuleexportflag = 'Y'  
where (shortname = '<C1_SHORTNAME>' and versionid = <C1_VERSIONID>)
```

or (shortname = '<C2\_SHORTNAME>' and versionid = <C2\_VERSIONID>)

...

or (shortname = '<C10\_SHORTNAME>' and versionid = <C10\_VERSIONID>)

\*Ensure collections C1..C1 exist in ECHO.

\*Ensure all test granules exist in ECHO.

\*Ensure a TCP proxy or mock ECHO is capturing BMGT traffic.

**STEPS:**

#	Action	Expected Result	Notes
1	#S-1 Choose an ECS inventory that includes at least one hundred thousand (100,000) science granules that are eligible for ECHO export and covers at least three different collections. Configure a time period increment for automatic verification that is not in excess of one month and no less than one week. Configure the maximum incremental export operation size such that none of the incremental verifications performed during the test will encounter the export size limit, i.e., the number of inserted/updated granules within the time increment never exceeds the maximum export size. Ensure that all collections in the mode are completely unverified, resetting the verification status if necessary.		
2	Ensure that the collections C1..C10 have a large number of granules in AIM (1000 in the EDF; about 100 000 in the PVC)		
3	Configure BMGT.Incremental.Duration = 10 ( the interval is in days).		
4	Configure BMGT.Verification.MaxGranules to 100 in the EDF or 10 000 in the PVC.		
5	In the BMGT GUI, reset verification status for all collections to be unverified or 0%.		
6	#V-1 Inspect the verification report in the BMGT GUI before performing any incremental verification, and verify that the overall verification percentage as well as that for each collection and group, is equal to 0%.		
7	In the BMGT GUI System Status tab, verify that the overall verification percentage is displayed as 0%		
8	In the BMGT GUI System Status tab, verify that verification percentage is displayed as 0% for each collection and group in the mode		
9	#S-2 Initiate incremental verification repeatedly until there are no more granules to verify.		
10	Ensure that no other operations are going on in the system which would cause the update of any granules.		

#	Action	Expected Result	Notes
11	Ensure that the incremental start utility requires the ECS mode as a command line option. It is acceptable to use a cron job or script to automate the initiation of incremental verification.		
12	Ensure that there are no ingest, delete, or update operations occurring in the mode		
13	Ensure the TCP proxy log will have only requests exported during this test:  Stop the TCP proxy. Move the log to a new name. Start the proxy.		
14	./EcBmBMGTManualStart --incremental		
15	V-2 After each of the first two incremental iterations, inspect the logs and export request queue to ensure that all granules whose last update falls within the time period covered by the iteration were added to the request queue.		
16	Verify that for the first two incremental iterations, any granule that falls between min_last_update_time and min_last_update_time + BMGT.Incremental.Duration (up to the maximum defined by BMGT.Verification.MaxGranules and bg_collection_configuration.maxgranulestoverify for each collection)		
17	Verify that after running multiple incremental iterations, each of the granules in each of the Collections C1..C10 in the mode have been added to the export request queue - check bg_export_request table.		
18	V-3 Inspect the log file to verify that for each Incremental verification initiation, no more than the configured number of granules is enqueued.		
19	Verify that for each incremental iteration, the total number of granules is less than or equal to BMGT.Verification.MaxGranules		
20	Verify that for each incremental iteration, the total number of granules for each collection is less than or equal to bg_collection_configuration.maxgranulestoverify for the given collection.		
21	V-4 Verify that when each verification process is started, a message is printed to the log, followed by another message when all requests have been added to the queue. Verify that the logs are written to the conventional ECS location (i.e. under /usr/ecs//CUSTOM/logs) and indicate the time at which the verification export driver started and completed, as well as how many items were enqueued for export, how many were added per collection, and the time span represented by the update times of the added granules.		

#	Action	Expected Result	Notes
22	Verify that the bmgmt log is written under /usr/ecs/CUSTOM/logs/EcBmBMGTManualDriver.log		
23	Verify that the bmgmt log indicates the time when the incremental verification was started.		
24	Verify that the bmgmt log indicates the time when all request have been added to the queue		
25	Verify that the bmgmt log indicated the time when verification export was started.		
26	Verify that the bmgmt log indicated the time when verification export was completed		
27	Verify that the bmgmt log indicates the time span of the verified granules in the iteration.		
28	V-5 Inspect the verification report in the BMGT GUI after the first two incremental verification iterations to verify that the overall verification percentage increases as well as that for the collections and groups which were selected for export.		
29	Verify the overall verification percentage increases after the first two iterations		
30	Verify the verification percentage for collections C1 - C10 increases after the first two iterations		
31	Verify the overall verification percentage = 100% after running multiple incremental iterations		
32	Verify that the verification percentage for Collections C1..C10 = 100% after running multiple incremental iterations.		
33	V-7 Verify that when verification is complete, in S-2, across all incremental exports, every eligible granule has been exported exactly once. This can be done at a coarse level, ensuring that the number of exports per collection is as expected.		
34	Save the TCP proxy log, and start a new one:  Stop the TCP proxy. Move the log to a new name. Start the proxy.		
35	Save eligible granule counts to a file:  select count(g.granuleid), esdt(g.shortname, g.versionid) from amgranule g		

#	Action	Expected Result	Notes
	join DsGeESDTCConfiguredType t on (g.shortname = t.configuredname and g.versionid = t.versionid) join bg_collection_configuration bcc on g.collectionid = bcc.collectionid where g.deleteeffectivedate is null and g.deletefromarchive != 'Y' and t.esdtstate = 'installed' and bcc.collectionexportflag = 'Y' and bcc.granuleexportflag = 'Y' group by g.shortname, g.versionid order by g.shortname, g.versionid		
36	Save the exported granule counts to a file:  sed -n 's/^PUT \V[^\ ]*\Vgranules\VC%3A\([^\ ]*\)\%3A.* \1/p' tcp.log   sort   uniq -c &gt; exported_counts.txt		
37	Verify the number of granules eligible for export per collection matches the number found in the TCP proxy log:  diff -w eligible_counts.txt exported_counts.txt		
38	V-11 Verify that each of the verification export requests results in exactly one export to ECHO (or an ECHO stand-in). Each Export shall take the form of a single HTTP PUT request containing the full granule metadata and an HTTP query parameter which indicates that the request is for verification purposes.		
39	Verify that the TCP proxy shows that each verification export is a single HTTP PUT request.		
40	Verify that the TCP proxy shows that the request has the HTTP query parameter &quot;xmldiff=true&quot; indicating it is for verification purposes.		
41	V-12 Verify that each incremental interval also queues and exports for each collection which has granules included in the incremental interval, the export of the associated collection metadata. This export shall take the form of a single HTTP PUT request per collection, containing the full collection metadata and an HTTP query parameter indicating that the request is for verification purposes.		
42	Verify that the TCP proxy shows a HTTP PUT request with Collection metadata for Collections C1..C10.		
43	Verify that the TCP proxy shows that the request is for verification purposes.		

#	Action	Expected Result	Notes
44	S-3 Once incremental verification has completed and verified the entire inventory, perform some granule inserts and updates. Pause automatic export before making these updates so that the events are not automatically exported. Ensure that at least some of the updated granules are each of the following: a) In the public datapool b) Have browse links c) Are restricted.		
45	Pause the EVENT queue via the BMGT GUI, so requests are queued and in PENDING state and will not be picked up by incremental verification.		
46	# a) In the public datapool *Ingest Science granules g1..g10 into Collection C1 with default publishing on.		
47	# b) Have browse links *Ingest Science granules g11..g20 into Collection C2 with default publishing on.		
48	# c) Are restricted. *Choose or create a restriction flag:  select * from dsmdrestrictionflag -OR- insert into dsmdrestrictionflag values(128, 'BE_82_01 Crit 520')		
49	*Add the restriction flag to granules g21, g22:  insert into dsmdgranulerestriction values(, 128)		
50	#S-4 Initiate another incremental verification export. * ./EcBmBMGTManualStart -mode --incremental		
51	V-8 Verify that the export attempt in S-4 results in a warning message indicating that there are no granules eligible for incremental verification and that no granules are added to the queue.		
52	Verify that there are no granules queued for verification export		
53	S-5 Resume Automatic export and allow it to pick up and export the events in S-3.		
54	Resume the Event queue in the BMGT GUI.		

#	Action	Expected Result	Notes
55	Verify that each of the granules in each of the g1..g10 in Collection C1 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
56	Verify that the bmgt logs also include the export of each of the granules in C1..C10 in Collection C1.		
57	Verify that each of the granules in each of the g11..g20 in Collection C2 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
58	Verify that the bmgt logs also include the export of each of the granules in g11..g20 in Collection C2.		
59	Verify that each of the granules in each of the g21..g22 in Collection C3 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
60	Verify that the bmgt logs also include the export of each of the granules in g21..g22 in Collection C3.		
61	S-6 Initiate another incremental verification export. *./EcBmBMGTManualStart --incremental		
62	V-2 After each of the first two incremental iterations, inspect the logs and export request queue to ensure that all granules whose last update falls within the time period covered by the iteration were added to the request queue.		
63	Verify that each of the granules in each of the g1..g10 in Collection C1 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
64	Verify that the bmgt logs also include the export of each of the granules in C1..C10 in Collection C1.		
65	Verify that each of the granules in each of the g11..g20 in Collection C2 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
66	Verify that the bmgt logs also include the export of each of the granules in C11..C20 in Collection C2.		
67	Verify that each of the granules in each of the g21..g22 in Collection C3 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
68	Verify that the bmgt logs also include the export of each of the granules in g21..g22 in Collection C3.		
69	#V-3 Inspect the log file to verify that for each Incremental verification initiation, no more than the configured number of granules is enqueued.		

#	Action	Expected Result	Notes
70	Verify that for each incremental iteration, the total number of granules is less than or equal to <code>BMGT.Verification.MaxGranules</code>		
71	Verify that for each incremental iteration, the total number of granules for each collection is less than or equal to <code>bg_collection_configuration.maxgranulestoverify</code> for the given collection.		
72	#V-4 Verify that when each verification process is started, a message is printed to the log, followed by another message when all requests have been added to the queue. Verify that the logs are written to the conventional ECS location (i.e. under <code>/usr/ecs//CUSTOM/logs</code> ) and indicate the time at which the verification export driver started and completed, as well as how many items were enqueued for export, how many were added per collection, and the time span represented by the update times of the added granules.		
73	Verify that the <code>bmg</code> log is written under <code>/usr/ecs//CUSTOM/logs/EcBmBMGTManualDriver.log</code>		
74	Verify that the <code>bmg</code> log indicates the time when the incremental verification was started.		
75	Verify that the <code>bmg</code> log indicates the time when all request have been added to the queue		
76	Verify that the <code>bmg</code> log indicated the time when verification export was started.		
77	Verify that the <code>bmg</code> log indicated the time when verification export was completed.		
78	Verify that the <code>bmg</code> log indicates the time span of the verified granules in the cycle.		
79	V-5 Inspect the verification report in the <code>BMGT</code> GUI after the first two incremental verification iterations to verify that the overall verification percentage increases as well as that for the collections and groups which were selected for export.		
80	Verify the overall verification percentage = 100%		
81	Verify that the verification percentage for Collections <code>C1..C10</code> = 100%		
82	V-9 Verify that the export attempt in <code>S-6</code> results in the queueing and export of the granules which were updated in <code>S-3</code>		
83	V-10 Verify that the bodies of the verification exports in <code>S-6</code> are exactly the same as the bodies of the automatic exports in <code>S-5</code> , including datapool URLs and restriction flags. Note that it is allowable for the verification to include additional exports not in the automatic export,		

---

---

#	Action	Expected Result	Notes
	e.g. the collection metadata for all collections for which there is a granule export.		
84	Compare the granule metadata in the tcp log between the automatic export (S-5) and verification export (S-6) to verify that they are identical.  NOTE: The verification exports will have collection exports for any granule exports in the collection.		
85	If there are extra collection exports in the verification, these can be ignored.		
86	V-11 Verify that each of the verification export requests results in exactly one export to ECHO (or an ECHO stand-in). Each Export shall take the form of a single HTTP PUT request containing the full granule metadata and an HTTP query parameter which indicates that the request is for verification purposes.		
87	Verify that the TCP proxy shows that each verification export is a single HTTP PUT request.		
88	Verify that the TCP proxy shows that the request has the HTTP query parameter &quot;xmldiff=true&quot; indicating it is for verification purposes.		
89	V-12 Verify that each incremental interval also queues and exports for each collection which has granules included in the incremental interval, the export of the associated collection metadata. This export shall take the form of a single HTTP PUT request per collection, containing the full collection metadata and an HTTP query parameter indicating that the request is for verification purposes.		
90	Verify that the TCP proxy shows a HTTP PUT request with Collection metadata for Collections C1...C10.		
91	Verify that the TCP proxy shows that the request is for verification purposeses (the url will contain a query parameter &quot;xmldiff=true&quot;).		
92	V-13 Verify that the database and the log files contain information on the process of each request through the system such that it is possible to identify when the metadata was generated, when the export was sent to ECHO, and when the response was received, etc.		
93	Verify that the bmgmt logs and database show when the request was generated for each verification request.		
94	Verify that the bmgmt logs and database show when the metadata was generated for each verification request.		

---

---

#	Action	Expected Result	Notes
95	Verify that the bmgmt logs and database show when the export was sent to ECHO.		
96	Verify that the bmgmt logs and database show when the response was received for each verification request.		
97	V-14 Verify that the BMGT GUI displays the completed export events, indicating that they were the result of an incremental verification export. Verify that it indicates that the requests were successfully exported and indicates the time of export as well as granule or ID.		
98	In the BMGT GUI export request tab, filter the request to view only those items on the INCR queue		
99	Verify that each request is listed in the SUCCESS state.		
100	Verify that each request has the associated granule or collection id listed.		
101	Verify that each request has its completion time listed.		
102	V-15 Verify that the BMGT GUI displays the following incremental verification metrics overall, or for a particular time frame: a) Number of collections/granules which were exported for verification. b) Number of collections/granules which were successfully verified c) Number of collections/granules which failed verification but were automatically repaired. d) Number of collections/granules which failed verification and could not be automatically repaired. e) Number of collections/granules skipped during export due to errors.		
103	In the BMGT GUI Export Request tab, filter to view only requests on the INCR queue.		
104	select the "Batch Job Summary" sub tab		
105	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items exported for verification.		
106	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items successfully verified (success column).		
107	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items which failed verification, but were automatically repaired by ECHO (warning column).		
108	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items which failed verification, but were not automatically repaired by ECHO.		

#	Action	Expected Result	Notes
109	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items which were skipped.		

TEST DATA:

EXPECTED RESULTS:

#### 4.8 URS Authentication

##### 4.8.1 URS Authentication - HTTP download (ECS-ECSTC-682)

DESCRIPTION:

<p>ASDC is currently only support wu-ftp. This test case in not applicable to production. </p>

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Using EDF URS Authentication (HTTP download)</i>		#comment
2	<i>To use Apache URS authentication in EDF DEV09 mode you must have a Systems Integration and Test (SIT) URS4 User Profile. Typically the Testbed systems associated with the EDF get integrated with SIT applications. You will also need to allow the URS LANCE application access to your profile and then you will be ready to use Datapool drill down to access the MISR LANCE data.</i>		#comment
3	1) Create a URS4 profile if you don't already have one:   a) Navigate to the SIT URS login page: <a href="https://sit.urs.earthdata.nasa.gov">https://sit.urs.earthdata.nasa.gov</a>   b) Click the Register button beneath the Password text entry box.  c) Fill in the required information on the page "Register for a URS Profile". Note that you need a valid email address to register for a URS profile. Also the Username, Password, Password Confirmation, First Name, Last Name, Email, Country and Affiliation fields are required as marked by a red asterisk. When you are finished filling in the fields, click on the "Register for URS" green button at the bottom of the page.  d) After registering for your URS profile you will be taken to the URS Profile screen indicating that "Your URS Profile has been successfully created":  e) You will need to go to your email inbox and look for a message similar to the following: urs-		

#	Action	Expected Result	Notes
	<p>noreply@urs.eosdis.nasa.gov &lt;br /&gt;&lt;br /&gt; User Registration System - Profile registration&lt;br /&gt;Hello, Kenneth Cockerill.&lt;br /&gt;A profile in the EOSDIS User Registration System has been requested&lt;br /&gt;for kengcmail, using this email address.To complete the registration, please follow this link:&lt;br /&gt;                      /&gt;http://sit.urs.earthdata.nasa.gov/activations/MXMcN5GhjD2X%2ByLXQvM6%2BLRdzO%2FzF67K0wc3TzR0zldBov3eehiT58NOzixR%0A&lt;br /&gt;&lt;br /&gt;If your email client does not show the URL as a link, just copy and&lt;br /&gt;paste it into your Web browser's Location field. If the URL has&lt;br /&gt;wrapped, you will need to edit it to be on a single line.&lt;br /&gt;&lt;br /&gt;If you did not request this profile, you may do nothing, and no further&lt;br /&gt;action will be taken. You will receive no further communications from EOSDIS URS.&lt;br /&gt;&lt;br /&gt;The login page can be found at the following link:&lt;br /&gt;&lt;br /&gt;http://sit.urs.earthdata.nasa.gov/&lt;br /&gt;&lt;br /&gt;Thanks,&lt;br /&gt;The EOSDIS User Registration Team&lt;br /&gt;&lt;br /&gt; f) You will need to click on the link provided in your email to activate your profile. Once you click on the link you will be taken to the acceptance page as follows: Click on the "Go to your user profile" link in order to navigate to the URS Profile Home page prior to moving on to the next steps.&lt;br /&gt;</p>		
4	<p>2) Once your SIT URS Profile is activated you will need to allow the URS LANCE application access to your profile:&lt;br /&gt;&lt;br /&gt; a) Once you are on the Profile Home page click the "My Applications" tab:&lt;br /&gt; b) Under the "Approved Applications" section of the "My Applications" page, click on the "Approve More Applications" button:&lt;br /&gt; c) In the "Search for Applications to Approve" text field, enter "URS4_EDF_LANCE" and click the "Search for Applications" green button:&lt;br /&gt; d) Under the "Application Search Results/Select the applications you wish to approve." , click the radio box next to "URS4_EDF_LANCE" so a blue check mark appears in the box:&lt;br /&gt; e) Click the "Approve Selected Applications" green button that will allow the URS4_EDF_LANCE application access to your profile.&lt;br /&gt; f) The preceding action will transition you back to the My Applications screen with the message at the top stating "The applications you have selected have been added to your applications." Also the application URS4_EDF_LANCE will appear under your Approved Applications:&lt;br /&gt;</p>		
5	<p>3) Now you are ready to use the Data Pool download to download MISR LANCE data:&lt;br /&gt; a) Navigate to the data pool root in EDF DEV09 mode: http://f5eil01v.edn.ecs.nasa.gov:46519&lt;br /&gt; b) Click on the link FS2/:&lt;br /&gt; c) Click on the link MISR_LANCE/:&lt;br /&gt; d) Click on any of the MISR Near Real Time (NRT) data collections:&lt;br /&gt; e) Click on the date</p>		

#	Action	Expected Result	Notes
	directory:  f) Choose any science file (suffix .hdf) to download by clicking on it. You will be presented with an authentication challenge box that requires you to enter your URS4 SIT credentials (User Name and Password). Then click the OK button.  g) You will be presented with the typical browser download panel. Click OK to download your data: 		
6	4) To test ECHO/Reverb access to the LANCE NRT data via the online URLs, navigate to the ECHO/Reverb testbed main screen: <a href="https://testbed.echo.nasa.gov/reverb">https://testbed.echo.nasa.gov/reverb</a>    a) Login using your URS SIT (System Integration and Test environment) profile that you created in step 1.  b) In the upper right hand Search Terms text box, enter EDF_DEV09 MI1B2_ELLIPSOID_NRT. The MISR Near Real Time (NRT) Level 1B2 Ellipsoid Data V001 should be displayed in the ECHO/Reverb Step 2: Select Datasets. Select the checkbox next to the dataset entry.  c) On the ECHO/Reverb Step 3: Discover Granules section of the screen, the granules should already have a checkmark next to them. Click the Search for Granules button at the bottom of the screen.  d) On the following screen click the information icon that appears right next to the shopping cart for one of the granules.  e) The following screen should provide information about the granule that includes links (next to Online Access:). Click the http link for the “.hdf” science file. Because you have previously signed on to URS, clicking on the link should immediately take you to the download popup for the granule.   f) Click the OK button to download the granule science file. 		

TEST DATA:

EXPECTED RESULTS:

4.8.2 [OBSOLETE] - NCR8052187 test instruction - MISR LANCE for ASDC requires URS authentication for users accessing via FTP (ECS-ECSTC-683)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	1. Run wuftp rollup script against the old xferlog with no URS authentication userid field.	verify that the data are rolled up correctly. This is to test the backward compatibility.   	
2	2. Run wuftp rollup script against the new xferlog with URS authentication userid field. 		
3	a. ftp one LANCE file, using urs login		
4	b. ftp one non-LANCE file, using regular linux login	c. verify in /var/log/xferlog that the entry related to the LANCE file contains urs userid in the last field and the entry related to the non-LANCE file contains &quot;-&quot; in the last field.	
5	d. RunEcDIRollupWuFtpLogs.pl against the time period, at least a day later.	e. Verify DIGranuleAccess table contains the entries.	
6	f. Run EcDbEMSdataExtractor.pl	verify that the *DistFTP_*DataPool.flt LANCE report contains the URS userid and the non-LANCE report contains the &quot;-&quot; as the first column 	This is not what EMS wants. EMS does not want to have to deal with '-' in the URS field. They want URS reports and non-URS reports.  How do we know which report is the LANCE report an which is the non-LANCE report?
7	<i></i>		#comment

TEST DATA:

EXPECTED RESULTS:

4.8.3 [OBSOLETE] - NCR8052188 test instruction - MISR LANCE for ASDC requires URS authentication for users accessing via HTTPS (ECS-ECSTC-684)

DESCRIPTION:

<p>ASDC does not have HTTP configured for downloads. This Test case in not currently a real life scenario. </p>

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<p>&lt;i&gt;Background:&lt;br /&gt;Currently EcDIHttpRollup.pl processes the error log to get the information needed to populate DIGranuleAccess table.&lt;br /&gt;In order to add URS authentication information, we have to create a new log, which has different format as the error log. Look for the CustomLog that makes use of the http_rollup&lt;br /&gt;LogFormat in httpd.conf under the mode you use.&lt;br /&gt;To make the script to be backward compatible, I added the following in the EcDIHttpRollup.CFG:&lt;br /&gt;&lt;br /&gt;USE_URS_LOG_PATH=Y&lt;br /&gt;LOG_PATH=/home/sxu/testftplog (this one existed before)&lt;br /&gt;URS_LOG_PATH=/home/sxu/testftplog1&lt;br /&gt;&lt;br /&gt;Also, in order to process LANCE data(the only one that's URS authenticated), we have to have this line in the EcDbEMSdataExtractor.CFG&lt;br /&gt;LANCEPROVIDER =&lt;br /&gt;&lt;br /&gt;We can choose to use either one. The LOG_PATH points to the error log we currently use, the URS_LOG_PATH points to the new log file that contains the URS info.&lt;br /&gt;This allows the DAACs to go back to run the old report, before the new log was created. The old log contains no URS info, the default is &amp;quot;-&amp;quot; which will appear in the report as the first column.&lt;br /&gt;&lt;br /&gt;Test Steps:&lt;/i&gt;</p>		#comment
2	1. Pick a mode that has the URS authentication configured.		
3	2. Configure the EcDIHttpRollup.CFG and EcDbEMSdataExtractor.CFG so that they contain the right info for the above lines. test with both the URS and non URS cases.		
4	3. Make sure the AIM database are patched for the mode for this ncr		
5	4. Do a http download on both the LANCE data and non-LANCE data.		
6	5. Run the rollup and EMS extractor	6. Verify that the LANCE report contains the UrsUserId in the first column and non LANCE report contains &quot;-&quot; in the first column, which indicates there's no urs authentication. *DistFTP_&lt;mode&gt;DataPool.flr is the output file to check. Note: if use the old log file, the LANCE report will contain &quot;-&quot; as the UrsUserId as well. 	

TEST DATA:

EXPECTED RESULTS:

#### 4.9 Tomcat Conf Checkout

##### 4.9.1 Apache Tomcat Load Balance and Failover Test (ECS-ECSTC-685)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	When tomcat is up in both oml and dpl - requests should be load balanced to both hosts		
2	When tomcat and apache are down on dpl, any edsc requests should still continue to go to oml.		
3	When both dpl and oml tomcat and apache are down - the egi requests should stay pending stop dispatching		
4	When tomcat and apache oml are down - requests should continue to go to dpl		
5	When dpl /oml is brought back up with pending requests - they should resume dispatching if a new request came in or if the dispatching is resumed through the GUI ( in this case looks like dispatching never got suspended).		
6	Verify esi.processing.url.failover is set to oml in the egi properties file on eil.		

TEST DATA:

EXPECTED RESULTS:

#### 4.9.2 Long URL Test (ECS-ECSTC-686)

##### DESCRIPTION:

##### PRECONDITIONS:

##### STEPS:

#	Action	Expected Result	Notes
1	Ensure that the Limit is set in the Apache configuration on f7eil01, f7dpl01, and f7oml01.		
2	Do curl on any egi endpoint first (making sure you use the cookie options, -b/-c). Note: It doesn't even have to be a valid request - something as simple as 'https://f7eil01.edn.ecs.nasa.gov/ops/egi/request' and this should establish a session. Subsequent requests won't then be redirected to EDL and should go through. e.g. curl -b -url https://f7eil01.edn.ecs.nasa.gov/ops/egi/request		
3	Do a long url test on eil host e.g. https://f7eil01.edn.ecs.nasa.gov/ops/egi/request?SUBAGENT_ID=ICESAT2&REQUEST_MODE=ASYNC&FILE_IDS=598137&SUBSET_DATA_LAYERS=/ATL03/ancillary_data/atlas_sdp_gps_epoch,/ATL03/ancillary_data/control,/ATL03/ancillary_data/data_end_utc,/ATL03/ancillary_data/data_start_utc,/ATL03/ancillary_data/end_cycle,/ATL03/ancillary_data/end_geoseg,/ATL03/ancillary_data/end_gpssow,/ATL03/ancillary_data/end_gpsweek,/ATL03/ancillary_data/end_latitude,/ATL03/ancillary_data/end_longitude,/ATL03/ancillary_data/end_orbit,/ATL03/ancillary_data/end_region,/ATL03/ancillary_data/end_rgt,/ATL03/ancillary_data/granule_end_utc,/ATL03/ancillary_data/granule_start_utc,/ATL03/ancillary_data/podppd_pad,/ATL03/ancillary_data/proc_interval,/ATL03/ancillary_data/release,/ATL03/ancillary_data/start_cycle,/ATL03/ancillary_data/start_geoseg,/ATL03/ancillary_data/start_gpssow,/ATL03/ancillary_data/start_gpsweek,/ATL03/ancillary_data/start_latitude,/ATL03/ancillary_data/start_longitude,/ATL03/ancillary_data/start_orbit,/ATL03/ancillary_data/start_region,/ATL03/ancillary_data/start_rgt,/ATL03/ancillary_data/version,/ATL03/ancillary_data/gt11/photon_classification_input/alpha_inc,/ATL03/ancillary_data/gt11/photon_classification_input/alpha_max,/ATL03/ancillary_data/gt11/photon_classification_input/delta_eslw,/ATL03/ancillary_data/gt11/photon_classification_input/delta_eslw_v,/ATL03/ancillary_data/gt11/photon_classification_input/delta_t_gap_min,/ATL03/ancillary_data/gt11/photon_classification_input/delta_t_lin_fit,/ATL03/ancillary_data/gt11/photon_classification_input/delta_t_max,/ATL03/ancillary_data/gt11/photon_classification_input/delta_t_min,/ATL03/ancillary_data/gt	The request should be successful	



#	Action	Expected Result	Notes
	cation_input/n_delta_z1,/ATL03/ancillary_data/gt1r/photon_classification_in put/n_delta_z2,/ATL03/ancillary_data/gt1r/photon_classification_input/nphot _min,/ATL03/ancillary_data/gt1r/photon_classification_input/out_edit_flag,/ATL03/ancillary_data/gt1r/photon_classification_input/pc_bckgrd_flag,/ATL03/ancillary_data/gt1r/photon_classification_input/r,/ATL03/ancillary_data/gt1r/photon_classification_input/r2,/ATL03/ancillary_data/gt1r/photon_classification_input/sig_find_t_inc,/ATL03/ancillary_data/gt1r/photon_classification_input/snrmed,/ATL03/ancillary_data/gt1r/photon_classification_input/t_gap_big,/ATL03/ancillary_data/gt2l/photon_classification_input/alpha_inc,/ATL03/ancillary_data/gt2l/photon_classification_input/alpha_max,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_eslw,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_eslw_v,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_t_gap_min,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_t_lin_fit,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_t_max,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_t_min,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_z_bg,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_zmax2,/ATL03/ancillary_data/gt2l/photon_classification_input/delta_zmin,/ATL03/ancillary_data/gt2l/photon_classification_input/dtime,/ATL03/ancillary_data/gt2l/photon_classification_input/e_a,/ATL03/ancillary_data/gt2l/photon_classification_input/e_gap,/ATL03/ancillary_data/gt2l/photon_classification_input/e_lfit,/ATL03/ancillary_data/gt2l/photon_classification_input/e_m,/ATL03/ancillary_data/gt2l/photon_classification_input/e_m_mult,/ATL03/ancillary_data/gt2l/photon_classification_input/htspanmin,/ATL03/ancillary_data/gt2l/photon_classification_input/lsant_flag,/ATL03/ancillary_data/gt2l/photon_classification_input/min_fit_time,/ATL03/ancillary_data/gt2l/photon_classification_input/nbin_min,/ATL03/ancillary_data/gt2l/photon_classification_input/n_delta_z1,/ATL03/ancillary_data/gt2l/photon_classification_input/n_delta_z2,/ATL03/ancillary_data/gt2l/photon_classification_input/nphot_min,/ATL03/ancillary_data/gt2l/photon_classification_input/out_edit_flag,/ATL03/ancillary_data/gt2l/photon_classification_input/pc_bckgrd_flag,/ATL03/ancillary_data/gt2l/photon_classification_input/r,/ATL03/ancillary_data/gt2l/photon_classification_input/r2,/ATL03/ancillary_data/gt2l/photon_classification_input/sig_find_t_inc,/ATL03/ancillary_data/gt2l/photon_classification_input/snrmed,/ATL03/ancillary_data/gt2l/photon_classification_input/t_gap_big,/ATL03/ancillary_data/gt2r/photon_classification_input/alpha_inc,/ATL03/ancillary_data/gt2r/photon_classification_input/alpha_max,/ATL03/ancillary_data/gt2r/photon_classification_		

#	Action	Expected Result	Notes
	input/delta_eslw,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_eslw_v,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_t_gap_min,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_t_lin_fit,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_t_max,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_t_min,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_z_bg,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_zmax2,/ATL03/ancillary_data/gt2r/photon_classification_input/delta_zmin,/ATL03/ancillary_data/gt2r/photon_classification_input/dtime,/ATL03/ancillary_data/gt2r/photon_classification_input/e_a,/ATL03/ancillary_data/gt2r/photon_classification_input/e_gap,/ATL03/ancillary_data/gt2r/photon_classification_input/e_lfit,/ATL03/ancillary_data/gt2r/photon_classification_input/e_m,/ATL03/ancillary_data/gt2r/photon_classification_input/e_m_mult,/ATL03/ancillary_data/gt2r/photon_classification_input/htspanmin,/ATL03/ancillary_data/gt2r/photon_classification_input/lslant_flag,/ATL03/ancillary_data/gt2r/photon_classification_input/min_fit_time,/ATL03/ancillary_data/gt2r/photon_classification_input/nbin_min,/ATL03/ancillary_data/gt2r/photon_classification_input/n_delta_z1,/ATL03/ancillary_data/gt2r/photon_classification_input/n_delta_z2,/ATL03/ancillary_data/gt2r/photon_classification_input/nphot_min,/ATL03/ancillary_data/gt2r/photon_classification_input/out_edit_flag,/ATL03/ancillary_data/gt2r/photon_classification_input/pc_bckgrd_flag,/ATL03/ancillary_data/gt2r/photon_classification_input/r,/ATL03/ancillary_data/gt2r/photon_classification_input/r2,/ATL03/ancillary_data/gt2r/photon_classification_input/sig_find_t_inc,/ATL03/ancillary_data/gt2r/photon_classification_input/snr_low,/ATL03/ancillary_data/gt2r/photon_classification_input/snrmed,/ATL03/ancillary_data/gt2r/photon_classification_input/t_gap_big		

TEST DATA:

EXPECTED RESULTS:

4.9.3 Package/Zip file checkout (ECS-ECSTC-687)

DESCRIPTION:

PVC configuration:  
 (06:13:13 PM) ddclarke: Concerning the "PackageEnable" Feature (When true use compression algorithm for .z or gz ....)  
 (06:13:19 PM) ddclarke: Alias /FS1 "/datapool/OPS/user/FS1"

```
<Directory "/datapool/OPS/user/FS1">
PackageEnable true
Options Indexes FollowSymLinks
IndexOptions +Charset=UTF-8 +FancyIndexing +IgnoreCase +FoldersFirst +XHTML +HTMLTable +SuppressRules +SuppressDescription +NameWidth=*
+IconsAreLinks
AllowOverride None
Require all granted
</Directory>
Alias /FS2 "/datapool/OPS/user/FS2"
<Directory "/datapool/OPS/user/FS2">
PackageEnable false
Options Indexes FollowSymLinks
IndexOptions +Charset=UTF-8 +FancyIndexing +IgnoreCase +FoldersFirst +XHTML +HTMLTable +SuppressRules +SuppressDescription +NameWidth=*
+IconsAreLinks
AllowOverride None
Require all granted
</Directory>
<Directory "/datapool/OPS/user/FS1/ACRM">
PackageEnable false
Options Indexes FollowSymLinks
IndexOptions +Charset=UTF-8 +FancyIndexing +IgnoreCase +FoldersFirst +XHTML +HTMLTable +SuppressRules +SuppressDescription +NameWidth=*
+IconsAreLinks
AllowOverride None
Require all granted
</Directory>
<Directory "/datapool/OPS/user/FS1/ASTT">
PackageEnable true
Options Indexes FollowSymLinks
IndexOptions +Charset=UTF-8 +FancyIndexing +IgnoreCase +FoldersFirst +XHTML +HTMLTable +SuppressRules +SuppressDescription +NameWidth=*
+IconsAreLinks
AllowOverride None
Require all granted
```

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
---	--------	-----------------	-------

TEST DATA:

EXPECTED RESULTS:

**4.10 Apache 2.2.23 - SSL**

4.10.1 C12500-1 Verify Apache 2.2.23-SSL/tomcat Web Services (ECS-ECSTC-835)

DESCRIPTION:

PRECONDITIONS:

- JDK 1.6.0\_24 and higher
- tomcat-6.0.32 is running

STEPS:

#	Action	Expected Result	Notes
1	Verify Apache 2.2.23-SSL/tomcat Web Services have been started: ps -ef   grep httpd   grep SSL	It should have several returns similar to the following: /usr/ecs/OPS/COTS/apache-2.2.23-ssl/bin/httpd -D SSL	

TEST DATA:

EXPECTED RESULTS:

#### 4.10.2 C12500-2 Verifying the HTTP port numbers for Apache 2.2.23 with SSL can be used to access the specific modes (ECS-ECSTC-836)

**DESCRIPTION:**

**PRECONDITIONS:**

- Apache 2.2.23 with SSL is running
- An Internet browser with network access to the machine running the above Apache

Note: The HTTP port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf - production

mod\_jk\_SSL\_test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	Open the following URL in a Web browser: />http://&lt;machine&gt;.&lt;domain&gt;:&lt;http TS1 port&gt;/WebAccess_TS1	The Data Pool home page should be displayed.	

**TEST DATA:**

**EXPECTED RESULTS:**

#### 4.10.3 C12500-3 Testing the mode-specific SSL port number can only be used to access one specific single mode (ECS-ECSTC-837)

**DESCRIPTION:**

**PRECONDITIONS:**

- C12500-1 has been run

- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	Open the web browser and open the following URL with OPS-specific SSL port: https://&lt;machine&gt;.&lt;domain&gt;.&lt;SSL OPS port&gt;/WebAccess_TS1	404 error not found	
2	Open the web browser and open the following URL with TS1 specific SSL port: https://&lt;machine&gt;.&lt;domain&gt;.&lt;SSL TS1 port&gt;/WebAccess_TS1	The Data Pool home page should be displayed.	

**TEST DATA:**

**EXPECTED RESULTS:**

**4.10.4 Apache 2.2.23-SSL: AIM Regression Test 1 (ECS-ECSTC-838)**

**DESCRIPTION:**

**PRECONDITIONS:**

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Install ESDTs with a variety of spatial search types]</i>		#comment
2	Prepare descriptor files for several (at least 5) ESDTs covering a variety of ESDT types described below, such that the descriptors conform to the ECS data model and the XML schema validation rules, guaranteeing successful ESDT installation.  1. ESDT descriptor with a spatial search attribute of Rectangle  2. ESDT descriptor with a spatial search attribute of GPolygon  3. ESDT descriptor with a spatial search attribute of NotSupported  4. ESDT descriptor with spatial search attribute of Orbit		
3	From the ESDT Maintenance GUI, select the ESDTs in the source directory and initiate installation.		
4	Upon completion of the ESDT installation, verify that the GUI displays a message indicating number of ESDTs successfully installed.		
5	Verify that the metadata elements provided in the descriptor files were validated by the XML Services library against the ESDR common schema, which conforms to the ECS data model.		
6	Verify by inspection that the element types and lengths in the descriptors match those stated in the ESDT common schema which in turn matches the ECS data model. The ESDT common schema is under: /usr/ecs/MODE/CUSTOM/WWW/DSS/ESDTMaint/WEB-INF/lib/EcDsAmEsdtCommonSchemas.jar.		
7	Verify that an MCF file is generated for each ESDT being installed, and stored into the configured MCF directory location		
8	Verify that the generated MCF files are correct.		
9	Verify that an XML schema file is generated for each ESDT being installed, and stored into the configured location.		

#	Action	Expected Result	Notes
10	Verify that the generated XML schema files contain validation rules based upon the contents of the Inventory section of the Descriptor file and the default element rules.		
11	Verify that the descriptor ODL files for the ESDT being installed are moved from the installation source directory into the configured descriptor directory.		
12	Verify that basic collection identification information, proper spatial search type, and the PSA associations if any have been added into the AIM Inventory database.		
13	Verify that an insert event, and event qualifiers are inserted into the Spatial Subscription Server database for each ESDT being installed.		
14	Verify that the ESDT Maintenance GUI logs the processing activities including time of request, action requested, ESDT ShortName, Version ID, descriptor file name, and the result of the operation, in a configured log file.		
15	<i>Verification Instructions</i>		#comment
16	Verify that there are at least 5 descriptors that match the setup requirements in directory /sotestdata/DROP_721/DS_7E_01/Criteria/2030/. Verify that the value of the OBJECT = SpatialSearchType group is “rectangle”, “GPolygon”, “not supported”, “point”, or “orbit” accordingly:  ESDT   SpatialSearchType   Archiving DAAC   ACR3L2SC.001   NotSupported   ASDC   AEPOE7W.001   NotSupported   NSIDC   AE_Land.002   Orbit   NSIDC   AE_Rain.002   Orbit   NSIDC   MB2LME.198   Orbit   ASDC   MIANRCCH.198   NotSupported   ASDC   MIL2ASOS.198   Orbit   ASDC   MIL3DAE.198   Rectangle   ASDC   MOD09CMG.005   Rectangle   LP DAAC   MOD14.005   GPolygon   LP DAAC   MOP02.003   Rectangle   ASDC   MYD09CMG.005   Rectangle   LP_DAAC   MYD14.005   GPolygon   LP DAAC   PM1GBAD1.001   NotSupported   NSIDC   TL3ATD.002   Rectangle   ASDC   g3assp.004   Point   ASDC   g3atb.004   Point   ASDC		
17	Return to the ESDT List Page. Select the check box next to the descriptor names set up in setup step 1. Select the “Install new ESDTs/Update existing ESDTs” button. Note the wall clock time of the start of the installation for later verification in the logs. Note the number of descriptors selected.		
18	Verify that the GUI displays the number of descriptors installed is equal to the number of descriptors selected.		

#	Action	Expected Result	Notes
19	Verify that the metadata elements provided in the descriptor files were validated by the XML validation utility by viewing its application log in the mode. Verify for each descriptor that no failures occurred.  Select one descriptor from each part of setup step 1.  Go to the directory where the generic descriptor schema is installed and view it. Verify by inspection that rules for elements in the generic schema are executed correctly for corresponding elements in the descriptor. A similar comparison should be performed between the elements of the generic descriptor schema and the ECS data model.		
20	Find the mcf target directory (parameter mcf.target.dir) in the EcAmMaintenanceGui.properties file.  Change the directory to the ESDT specific subdirectory (probably /stornext/smallfiles/&lt;mode&gt;/mcf/).  Perform an ls to verify that the mcf file is found. Repeat for each MCF file generated from the ESDT installation.		
21	Compare the descriptor file to the mcf file to verify that the MCF is correct.  Change the directory to this directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/). Perform an ls to verify that the schema file is found.		
22	Make a one to one comparison of the groups in the Inventory Section of the Descriptor file with the corresponding groups in the schema. Verify that the values of the descriptor are valid according to the rules in the schema.		
23	Find the descriptor source and target directory (parameters descriptor.source.dir and descriptor.target.dir) in the EcAmMaintenanceGui.properties file.  Change to target directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/). Perform an ls to verify that the descriptor file DsESDT&lt;ShortName&gt;.&lt;VersionId&gt;.desc is found. Repeat for each part descriptor installed in the setup step 1. Now verify that the descriptor file was deleted in the source directory in the mode. Change to directory descriptor.source.dir. Verify that the descriptor was deleted by using the ls command and finding no files matching the descriptor's name. Also, repeat for each descriptor installed.		
24	In the AIM Inventory database, verify that the ShortName, VersionID, and insertTime are populated in the DsMdCollections table for each ESDT installed.   select ShortName, VersionID, insertTime  from DsMdCollections  where ShortName = "&lt;ShortName&gt;"  and VersionID= "&lt;VersionID&gt;" 		

#	Action	Expected Result	Notes
	<p>Also, verify that the spatial searchSearchType matches the one specified in the descriptor.  <pre> select configuredName, VersionID, spatialSearchType from DsMdESDTCConfiguredType where configuredName = "&amp;lt;shortName&amp;gt;" and VersionID = "VersionID" </pre> Repeat for each installed ESDT.</p>		
25	<p>In another terminal log into the Spatial Subscription Server (SSS) database.  View the &amp;lt;ShortName&amp;gt;.&amp;lt;VersionId&amp;gt; descriptor in the mode and find the INSERT object under the EVENT group in the descriptor.  Under the EVENTPARMS object are a list of parameters which should be populated in the SSS database.  Verify that the event was inserted into the EcNbEventDefinition table.  <pre> select * from EcNbEventDefinition where ESDT_Id in ( &amp;lt;a quoted list of ShortNames of installed ESDTs&amp;gt;); </pre> Verify that each parameter EVENTPARM parameter is populated in the SSS database.  <pre> select * from EcNbEventAttrXref where ESDT_Id = "ShortName" </pre> Repeat this step for each descriptor installed</p>		
26	<p>Change directory to the log.dir directory of the mode.  With the vi command view the ESDT_Maint.log and search for the first ESDT installed.  Verify the time of the installation request, the ShortName, Version ID, descriptor file name and the result of the operation (successful) is listed in the log file.  Repeat for each ESDT installed.</p>		

TEST DATA:

EXPECTED RESULTS:

4.10.5 Apache 2.2.23-SSL: AIM Regression Test 2 (ECS-ECSTC-839)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running

- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Update multiple ESDTs, success case]</i>		#comment
2	Modify the ESDT descriptor ODL files for several (at least 6) previously installed ESDTs, such that the modifications include the following variety of ESDT updates:  1. removing one or more existing collection level attributes that are not restricted  2. removing DLLName parameter from the Collection metadata group  3. changing the ShortName in the CollectionAssociation group  4. changing a mandatory inventory attribute to optional  5 adding one or more optional inventory metadata attributes  6. adding one or more new qualifiers on existing events		
3	Place the descriptor files in a source directory accessible by the ESDT Maintenance GUI.		
4	Upon completion of the ESDT update, verify that the GUI displays a message indicating number of ESDTs successfully updated.		
5	Verify that an MCF file is created for each ESDT being updated, and stored into the configured MCF directory location replacing the previous version.		
6	Verify that the MCF files are correct.		
7	Verify that the descriptor file for each ESDT is replaced by the updated descriptor file, and that the updated descriptor files are consistent with the ECS data model.		
8	Verify that an XML schema file is generated for each ESDT and stored into the configured location replacing the previous version		
9	Verify that the generated XML schema files contain validation rules based upon the contents of the Inventory section of the Descriptor file and the default element rules.		

#	Action	Expected Result	Notes
10	Verify that collection based tables in the AIM Inventory database as identified in the Operations Concept are populated with correct information.		
11	Verify that the insert event, and event qualifiers are updated correctly in the Spatial Subscription Server database for each ESDT being updated.		
12	Verify that, upon completion of updates, the ESDT Maintenance GUI displays a message saying that the ESDT changes will take effect only after the Ingest service is re-started.		
13	<i>Verification Instructions</i>		#comment
14	Open a terminal.		
15	For each ESDT update perform the following setup and verification:  Save the original versions of files generated by the already installed ESDT selected for update. Change directory (cd) to /stornext/smallfiles/&lt;mode&gt;/descriptor/. Copy the descriptor ODL and xml files to a temporary location. Similarly, copy the mcf file to a temporary location. Change directory to /sotestdata/DROP_721/DS_7E_01/Criteria/2120, and vi the descriptor file, which will contain the added attributes. Compare the original ODL descriptor file in the temporary location to the staged one to verify which collection metadata has been added (a new container). Verify that in its container the mandatory flag is set to FALSE. Similarly, compare the original and staged mcf files.		
16	Copy the descriptors in setup step 1 to the directory specified as descriptor.source.dir in the EcAmMaintenanceGui.properties file.		
17	Verify that the GUI displays the number of descriptors updated is equal to the number of descriptors selected (6).		
18	Find the mcf target directory (parameter mcf.target.dir) in the EcAmMaintenanceGui.properties file. Change the directory to this directory (probably /stornext/smallfiles/&lt;mode&gt;/mcf/). Perform an ls -l to verify that the mcf file &lt;ShortName&gt;. &lt;VersionID&gt;.MCF is found. The date should match approximately the date that the update began. Repeat for each ESDT updated.		
19	Change directory to the MCF file used for sub criteria 1 in setup step 1. Compare this MCF with the original and verify that the differences were the attributes not found in the updated MCF. Repeat this step for the remaining subcriteria to verify the attributes are omitted or modified attributes are updated.		

#	Action	Expected Result	Notes
20	Perform an ls -l command on the descriptor file that was updated and verify that its modification time is about the time of this update. Compare the updated descriptor and the original one stored in a temporary location and verify that it was added and consistent with the ECS data model. Repeat for each updated ESDT.		
21	Cd back to the descriptor directory and find the ESDT specific schema file. Perform an ls -l and verify that the time for this file is approximately when this update started. Repeat this step for each updated ESDT.		
22	Compare the schema stored in the temporary location with the newly generated one and verify that the correct rules have been generated for the new attribute that was added. Repeat this step for each updated ESDT.		
23	Use a db client to login to the AIM inventory database. Verify that the lastUpdate in the DsMdCollections table has been updated for this &lt;ShortName&gt;. &lt;VersionID&gt; ESDT for each ESDT. In the DsMdAdditionalAttributes table, verify that the entry for the attribute removed from the descriptor is not found.(Setup step 1 part 1) Verify that an entry for this attribute is not found in the DsMdCollectionAddnlAttribsXref.(Setup step 1 part 1). For setup step 1 part 2. Verify as in the verify step 7 of criteria 2100. For setup step 1 part 3. Change made in the updated descriptor. See step verification step 4. For setup step 1 part 4. Verify that in the XML Validation Utility log that there are no error messages caused by changing a mandatory flag from TRUE to false in the descriptor. For setup step 1 part 5. Verify that the added optional AdditionalAttribute was added to the DsMdAdditionalAttributes table.		
24	Verify that the event was inserted into the EcNbEventDefinition table.   select *  from EcNbEventDefinition  where ESDT_Id = “&lt;ShortName&gt;”  and VersionID = &lt;VersionID&gt;		
25	Record the EventId to be used in the next test.		
26	Verify that each parameter added to the EVENTPARM parameter section is populated in the SSS database for each ESDT installed.   select *  from EcNbEventAttrXref  where ESDT_Id = “&lt;ShortName&gt;”  and VersionID = &lt;VersionID&gt;		
27	Return to the ESDT Maintenance GUI. Verify that it displays a message saying that the ESDT changes will teak effect only after the Ingest service is re-started.		

TEST DATA:

EXPECTED RESULTS:

4.10.6 Apache 2.2.23-SSL: AIM Regression Test 3 (ECS-ECSTC-840)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

STEPS:

#	Action	Expected Result	Notes
1	<i>[Delete multiple ESDTs, success case]</i>		#comment
2	From the ESDT Maintenance GUI select a group of ESDTs for deletion. Each of the ESDTs selected must satisfy the following conditions to guarantee successful deletion:  1. the ESDT does not contain any granules in the AIM inventory database  2. the ESDT is not defined within the Data Pool  3. the Spatial Subscription service does not contain active or inactive subscriptions referencing the ESDT		
3	Verify that the ESDTs were deleted successfully.  1. The MCF files were deleted  2. The ESDT XML schema files were deleted  3. The ESDT descriptor files were deleted.  4. The ESDT basic		

#	Action	Expected Result	Notes
	collections and PSA associations were deleted from the Inventory database 5 The insert events and event qualifiers for the ESDTs were deleted from the Subscription Server database 6. The XML metadata file directories associated with the ESDTs were removed from the XML archive 7. PSA definitions are removed if there are no other ESDTs associated with them		
4	Verify that the ESDT Maintenance GUI displays a message indicating number of ESDTs deleted.		
5	Verify that the ESDT Maintenance GUI logs the processing activities including time of request, action requested, ESDT ShortName, Version ID, descriptor file name, and the result of the operation, in a configured log file.		
6	<i>Verification Instructions</i>		#comment
7	Open a terminal.		
8	Use a database client to log into the AIM database for this mode.		
9	1. Verify that there no rows returned for any granules in the DsMdGranules table (in case they were ingested previously). select dbID, ShortName, VersionID from DsMdGranules where ShortName = "&lt;ShortName&gt;" and VersionID = "&lt;VersionID&gt;" Repeat for each ESDT.		
10	2. Perform a query on the collections table in the Data Pool to verify that it is not defined in the DataPool database select * from DataPool_DEV04..DICollections where ShortName = "&lt;ShortName&gt;" and VersionID = &lt;VersionID&gt; Repeat for each ESDT.		
11	3. Similarly, for the Spatial Subscription Server database, verify that that no rows returned indicating a subscription with this event: select * from EcNbDb_DEV04..EcNbSubscription where ESDT_Id = &quot;&lt;ShortName&gt;&quot; and VersionID = "&lt;VersionID&gt;" Repeat for each ESDT.		
12	On the ESDT List Page of the ESDT Maintenance GUI, select the checkbox to delete and select the "Delete Selected ESDTs" button at the bottom of the page to begin the ESDT deletion.		
13	Select OK for the confirmation prompt of 1 ESDT to be deleted.		
14	Apply the verification in verification step 1 of Criteria 2200 here for each ESDT Refer to ESDT Maintenance GUI ITP <a href="http://dmserver.gsfc.nasa.gov/release721/ESDT_Maint_GUI/ITP_DS_7E_01_ESDTGUI.doc">http://dmserver.gsfc.nasa.gov/release721/ESDT_Maint_GUI/ITP_DS_7E_01_ESDTGUI.doc</a>		

#	Action	Expected Result	Notes
15	Apply the verification of verification step 2 of Criteria 2200 here for each ESDT		
16	Apply the verification of verification step 3 of Criteria 2200 here for each ESDT		

TEST DATA:

EXPECTED RESULTS:

4.10.7 Apache 2.2.23-SSL: DPLIngest Regression Test 1 (ECS-ECSTC-841)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Data Providers can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[View Data Providers]</i>		#comment

---

#	Action	Expected Result	Notes
3	As the 'ingest admin' operator, navigate to the Data Provider page on the Data Pool Ingest GUI.		
4	Verify that all provider information in the INGST CI database appears correctly on the Data Provider page.		
5	<i>[Configure Data Providers]</i>		#comment
6	Edit existing provider information (if any) and define new providers to fulfill the following requirements:  At least one data provider should have an FTP notification method,  at least one data provider should have an scp notification method,  at least one data provider should have an email notification method,  at least one data provider should have a combination FTP/email notification method,  and at least one data provider should have a combination scp/email notification method. Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des. At least one provider must use active mode. At least one should use passive mode.		
7	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit data provider information.		
8	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing data providers before saving this information.		
9	Verify that the ingest admin operator has the authorization to define new data providers (i.e., has access to the data provider pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
10	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16110 to be entered.		
11	Verify that all of the notification methods in S-DPL-16150 can be entered or selected on the Data Pool Ingest GUI, as appropriate for the selected transfer method as per S-DPL-16110.		
12	For one provider with an FTP Notification method, verify that the Data Pool Ingest GUI allows the related FTP information to be entered (FTP host, destination directory, login id, password)		
13	Verify that the FTP password entered is not shown or stored in the clear.		
14	Verify that the FTP password entered is not shown in the Data Pool Ingest GUI log.		
15	For one provider with an email notification method, verify that the Data Pool Ingest GUI allows the related email address to be entered.		

#	Action	Expected Result	Notes
16	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new data providers before saving this information.		
17	For one provider with an scp notification method, verify that the Data Pool Ingest GUI allows the related scp information to be entered (scp host, destination directory, login id, password, scp type, and cipher).		
18	Verify that the scp password entered is not shown or stored in the clear.		
19	Verify that the scp password entered is not shown in the Data Pool Ingest GUI log.		

TEST DATA:

EXPECTED RESULTS:

4.10.8 Apache 2.2.23-SSL: DPLIngest Regression Test 2 (ECS-ECSTC-842)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Polling Locations can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure Polling Locations]</i>		#comment
3	As the Ingest Admin operator use the Data Pool Ingest GUI to define all polling locations that will be used for testing this ticket. (Table of data provider to polling location mappings, and related S-DPL-16230 information for each polling location TBS as part of ITP).  At least one polling location should have an FTP polling method,  at least one polling location should have a local polling method,  and at least one polling location should have an scp polling method. At least one polling location using FTP shall be for a provider using local transfers.		
4	Verify that the ingest admin operator has the authorization to define the polling locations (i.e., has access to the polling location pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
5	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16230 to be entered.		
6	Verify that all of the polling methods in S-DPL-16250 can be entered or selected on the Data Pool Ingest GUI.		
7	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new polling locations before saving this information.		
8	<i>[Edit Polling Locations]</i>		#comment
9	For one polling location, as the 'ingest admin' operator, edit all of its existing configuration parameters. (NOTE: After this criterion is complete, values of these configuration parameters should be reset to appropriate values for processing all PDRs in criterion 300).		
10	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit polling location information.		
11	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing polling locations before saving this information.		

TEST DATA:

EXPECTED RESULTS:

#### 4.10.9 Apache 2.2.23-SSL: DPLIngest Regression Test 3 (ECS-ECSTC-843)

##### DESCRIPTION:

##### PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_ssl.conf – production

mod\_jk\_ssl.test.conf - test

##### STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies FTP Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure FTP hosts]</i>		#comment
3	As the 'ingest admin' operator, navigate to the FTP Host page on the Data Pool Ingest GUI.		
4	Edit existing ftp host information (if any, from the INGEST CI), per S-DPL-16260, and define new ftp hosts such that all ftp hosts which will be used for testing this ticket are defined. (Table of ftp hosts and related S-DPL-16260 information TBS as part of ITP). At least two ftp hosts should be defined that are not configured for the INGEST subsystem.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit ftp host information.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing ftp hosts and definitions of new ftp hosts before saving this information.		

#	Action	Expected Result	Notes
7	Verify that the ingest admin operator has the authorization to define the ftp hosts (i.e., has access to the ftp host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
8	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16260 to be entered.		

TEST DATA:

EXPECTED RESULTS:

4.10.10Apache 2.2.23-SSL: DPLIngest Regression Test 4 (ECS-ECSTC-844)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies SCP Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure hosts for scp access]</i>		#comment

#	Action	Expected Result	Notes
3	As the 'ingest admin' operator, navigate to the host page on the Data Pool Ingest GUI.		
4	Edit existing information for hosts accessed via scp (if any, from the INGST CI), per S-DPL-16290, and define new scp hosts such that all scp hosts which will be used for testing this ticket are defined. (Table of hosts accessed via scp, and related S-DPL-16290 information TBS as part of ITP.) Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des, i.e., the test needs to involve several different providers.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to enter and edit the scp host information in S-DPL-16290.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing scp host parameters and definitions of new scp host parameters before saving this information.		
7	Verify that the ingest admin operator has the authorization to define the scp host parameters in S-DPL-16290 (i.e., has access to the host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		

TEST DATA:

EXPECTED RESULTS:

4.10.11Apache 2.2.23-SSL: DPLIngest Regression Test 2 (ECS-ECSTC-845)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies DPL Ingest GUI configuration can be viewed and modified correctly.</i>		#comment
2	<i>[View collection configuration]</i>		#comment
3	As an ‘ingest admin’ operator, use the Data Pool Ingest GUI to list all data types for which configuration parameters were entered in criterion 120.		
4	Verify that ”Ignore Validation Warnings” and “Public in Data Pool” are displayed for each ESDT.		
5	Verify that the Data Pool Ingest GUI provides the operator a method to quickly select or scan for a subset of existing Data Pool collections without having to enter the full ESDT name and version.		
6	<i>[Edit collection configuration]</i>		#comment
7	As an ‘ingest admin’ operator, use the Data Pool Ingest GUI to edit publication policy configuration parameters for a Data Pool collection.		
8	Verify that the edited configuration parameters are correctly updated in the database.		
9	<i>[Reconfigure data type parameters.]</i>		#comment
10	Log into the DPL Ingest GUI as an ‘ingest admin’ operator, and configure a data type (ESDT and Version) to be published in the public Data Pool upon insert.		
11	Submit several PDRs for the data type configured above. There needs to be a sufficient number of PDRs and granules such that some granules will complete ingest before the configuration change made during the test (see next step), and some granules will not start ingest until after the configuration change has been applied.		
12	After the first few granules completed ingest, re-configure the data type via the DPL Ingest GUI such that the data type no longer will be published in the Data Pool. Also change the minimum retention period sufficiently long so that the ingested granules are not cleaned up immediately after archiving, to allow time for the verification steps below.		

#	Action	Expected Result	Notes
13	Verify that the first few granules that were ingested before the re-configuration are queued with the Data Pool Insert Service for insertion into the public Data Pool area.		
14	Verify that all granules that started ingest one minute or more after the re-configuration are inserted into the non-public Data Pool area, and are NOT queued with the Data Pool Insert Service for insertion into the public Data Pool area. NOTE: if the Science granule has an associated Browse the Browse will be published even if the Science granule is &quot;hidden.&quot;		
15	Verify that ALL browse granules ingested as part of the test are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

TEST DATA:

EXPECTED RESULTS:

4.10.12Apache 2.2.23-SSL: DPLIngest Regression Test 24 (ECS-ECSTC-846)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies it is possible to add and modify volume groups using the DPL Ingest GUI and that the server correctly uses the new volume groups.</i>		#comment
2	<i>[Add & Modifying Volume Groups]</i>		#comment
3	Using the DPL Ingest GUI add a primary and backup Volume Group for a collection.		
4	Ingest a granule and verify that it goes to the primary and backup volume groups.		
5	Modify Volume group and give it a new path		
6	Ingest a granule and verify that it goes to the new path.		
7	Click on reports and verified that the addition and modification from above are displayed in the report.		

**TEST DATA:**

**EXPECTED RESULTS:**

4.10.13Apache 2.2.23-SSL: BMGT Regression Test 1 (ECS-ECSTC-847)

**DESCRIPTION:**

**PRECONDITIONS:**

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[BMGT GUI Configuration Page]</i>		#comment
2	Log in to the BMGT GUI as an administrator to perform the following configuration tasks. Verify that this is possible.		
3	Configure the automatic export cycle length to 1 hour.		
4	Configure the FTP destination, directory, and login parameters with which to transmit the export packages.		
5	Configure the location where BMGT is to save the received ingest summary report and an e-mail address for alert and error notifications. The email address should be a valid address which you have access to.		
6	Log in to the BMGT GUI as an operator, and verify that the configuration page looks the same, but cannot be edited.		
7	In the BMGT configuration file, set the logging level to the highest verbosity.		

**TEST DATA:**

**EXPECTED RESULTS:**

4.10.14Apache 2.2.23-SSL: BMGT Regression Test 2 (ECS-ECSTC-848)

**DESCRIPTION:**

**PRECONDITIONS:**

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[BMGT GUI Monitor Packages Page]</i>		#comment
2	For all remaining test steps, use the BMGT operator GUI to display audit trail information when possible. For any information not available in the GUI, consult the log files. Verify that all remaining tests are verifiable in this way.		
3	Verify that you can change the number of cycles that appear on each page of the Recent Packages page, and that if you log out and return, that configuration has persisted.		
4	Verify that clicking on a cycleId in the Recent Packages GUI page provides a page containing information about the cycle.		
5	Verify that all cycles processed through the expected statuses, ie. NEW, STARTED, PRODUCT_GENERATED_PACKAGE_GENERATED, TRANSFERRING, EXPORTED, and COMPLETE.		

**TEST DATA:**

**EXPECTED RESULTS:**

**4.10.15Apache 2.2.23-SSL: BMGT Regression Test 3 (ECS-ECSTC-849)**

**DESCRIPTION:**

**PRECONDITIONS:**

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running

- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[Automatic Export]</i>		#comment
2	Allow 2 or more automatic export cycles to pass without being generated. For instance, do not run the auto start utility between 8AM and 10AM. Ensure that at least 2 of these cycles contain a reasonable number of events. Ensure that some of these events will result in the export of browse binary files. The rest of the cycles can be empty. Designate 2 cycles which have events as A1 and A2 in order of start time.		To allow cycles to pass without being generated you need to bring down BMGT. This is done by going to the utilities folder and running EcBmBMGTAutoStop &lt;MODE&gt;. Example: EcBmBMGTAutoStop TS2 . This will bring down BMGT and all for time to pass. Allow however long the auto cycle time is set to pass. So if its set for 1, allow for 2 hours to pass. Once that has happened bring up Bmgt by going to that same directory and running EcBmBMGTAutoStart &lt;MODE&gt;
3	Cause a granule event (GRINSERT or GRDELETE) for at least one public granule that precedes the end of the most recent complete export cycle. It is acceptable to manually modify the time stamp of some event to achieve this.		To do this you can go to the DsMdGrEventHistory table. Find a granule event that has already happened and you can modify the eventTime by running this query: &quot;Update

#	Action	Expected Result	Notes
			DsMdGrEventHistory Set eventTime = '05/12/2011 2:15:55 PM' Where eventKey = &lt;eventKey&gt;&quot;
4	Start the BMGT auto start utility and allow it to pick up the unexported cycles.		Start by going to utilities folder and running EcBmBMGTAutoStar &lt;MODE&gt;
5	Verify that only one automatic cycle is generating at a time, and that each cycle is given a unique, sequential package id, in the order of its start time.		
6	Verify that the first Automatic package generated above contains the metadata for the events that preceded the end of the last completed automatic export cycle.		
7	Verify that the generator log contains a message for each such event indicating that an event from a previous cycle was exported.		
8	Verify that A1 and A2 contain the expected metadata.		
9	Verify that there is no duplicate metadata and that every granule or collection for which there was an event is properly represented.		
10	Verify that the packages are successfully exported to the configured FTP endpoint. This should include the zip file for each of the two automatic cycles as well as the appropriate number of browse binary files.		

TEST DATA:

EXPECTED RESULTS:

4.10.16Apache 2.2.23-SSL: Data Pool Regression Test 2 (ECS-ECSTC-850)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run

- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	Use the Data Pool Web Access GUI to perform a drill down search for the granule inserted in Test Case 1.		
2	View the XML for that granule by clicking on the XML metadata icon for that granule in the drill down results set.		
3	Verify that the drill down search criteria used to find the granule correspond to the information in the XML file.		
4	View the browse image for the granule by clicking on the browse icon for the granule in the drill down results set.		
5	Verify that the browse image is correctly displayed.		
6	<i></i>		#comment
7	Insert a granule into the hidden Data Pool.		
8	Use the Web Access GUI to perform a drill down search for the hidden granule just inserted.		
9	Verify the hidden granule is not visible via Web Access.		

**TEST DATA:**

**EXPECTED RESULTS:**

#### 4.10.17 Apache 2.2.23-SSL: DMS Regression Test 1 (ECS-ECSTC-851)

**DESCRIPTION:**

**PRECONDITIONS:**

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_ssl.conf – production

mod\_jk\_ssl.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Single Granule Order</i>		#comment
2	Submit a single granule order through EWOC with the following media types: FtpPull, FtpPush, CDROM, DVD, and DLT.	Verify that the order is successfully registered in the OMS.	

**TEST DATA:**

**EXPECTED RESULTS:**

#### 4.10.18 Apache 2.2.23-SSL: DMS Regression Test 3 (ECS-ECSTC-852)

**DESCRIPTION:**

**PRECONDITIONS:**

---

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Single Granule External Processing Order</i>		#comment
2	Submit an order through EWOC with a granule that requires spatial subsetting.		
3	Verify a Submit message was received by the spatial subsetting external processor.		
4	Verify the Submit message contained this order's order ID.		
5	Verify the Submit message contained the subsetting request's request ID.		
6	Verify the MSS request state has been updated to 'pending'.		
7	Verify a secondary Submit message was received by EWOC from an External Processor.		
8	Verify an order and request were registered with MSS, corresponding to the order submitted.		
9	Verify the MSS order is not marked as an ECHO order.		
10	Verify the MSS request is not marked as an External Processing request.		
11	Verify a request was placed with OMS, corresponding to the order submitted.		
12	Verify the OMS request is not marked as an External Processing request.		
13	Verify a Submit response is received at the client that contains the order ID recorded in the MSS database for this request and a successful status.		
14	Verify close Provider Order was not invoked in ECHO for the secondary Submit sent by an External Processor.		
15	Verify the subsetted granule is shipped to the customer via FTP PULL.		

#	Action	Expected Result	Notes
16	Verify that the granule is subsetted correctly.		

TEST DATA:

EXPECTED RESULTS:

4.10.19Apache 2.2.23-SSL: Web Order Status Regression Test 1 (ECS-ECSTC-853)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

STEPS:

#	Action	Expected Result	Notes
1	<i>Order Status GUI: Display Order Status, OD_S6_05. Criterion 130</i>		#comment
2	Request the status of a single, completed order that was handled by the OMS and that contains at least two requests by specifying a valid Order ID and user contact email address.  Each request in the order should contain at least 100 granules that have NOT been processed by HEG or the external subsetter.		
3	Verify order ID is correctly displayed.		

#	Action	Expected Result	Notes
4	Verify submission date/time is correctly displayed.		
5	Verify order state is correctly displayed.		
6	Verify order completion date/time is correctly displayed.		
7	Verify request information is correctly displayed:  request ID,  request state,  media type,  number of granules,  request completion date/time		
8	Verify order state and request states are presented in terms that an end user can understand.		
9	Verify order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
10	Verify there is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

4.10.20Apache 2.2.23-SSL: Web Order Status Regression Test 2 (ECS-ECSTC-854)

DESCRIPTION:

PRECONDITIONS:

- C12500-1 has been run
- Apache 2.2.23- SSL and Tomcat are still running
- An Internet browser with network access to the machine running the above Apache-SSL and Tomcat

Note: The SSL port numbers can be found in /usr/ecs/OPS/COTS/tomcat-6.0.32/conf/

mod\_jk\_SSL.conf – production

mod\_jk\_SSL.test.conf - test

STEPS:

#	Action	Expected Result	Notes
1	<i>Order Status GUI: Display Order Status based on History Range, 0D_S6_05, Criterion 180</i>		#comment
2	Request order history by specifying a starting and ending date, user contact email address, and one valid Order ID for the user.  Ensure that at least 100 orders are displayed that represent a mix of orders for which request details are available for some orders and not available for others.  Ensure that the date range requires the Order Status Interface to retrieve information from both the Order Manager operational tables and archive tables.  Ensure that at least one of the orders contains granules that were processed by the external subsetter and one of the orders contains granules that were processed by HEG.		
3	Verify that the correct orders are returned and sorted by submission date/time.		
4	For each order verify the following:  a. Order Id is correctly displayed.  b. Submission date/time is correctly displayed.  c. Order state is correctly displayed.  d. Order completion date/time is correctly displayed.  e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed, including the processing description for the granule to be processed by the external subsetter, in a manner which should be generally understandable by the user.  f. Order state and request states are presented in terms that an end user can understand.  g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.  h. An indication is provided when request details are not available for an order.		

TEST DATA:

EXPECTED RESULTS:

## 5 PVC LITTLE LANCE REGRESSION TESTS

### 5.1 Apache

#### 5.1.1 BMGT

##### 5.1.1.1 GUI (ECS-ECSTC-700)

DESCRIPTION:

	S	270	1	<b>[GUI]</b> All of the following steps shall be performed using each of the browsers and operating systems listed in “Technical Document 910-TDA-042, Browsers Baseline”		
	S	270	2	Navigate to the BMGT GUI. Enter as an operator, without administrative privileges.		
	S	270	3	Attempt to suspend BMGT processing via the GUI.		
	S	270	4	Navigate to the collection configuration page, and attempt to enable or disable a collection for export.		
	S	270	5	Navigate to the BMGT configuration page, and attempt to modify a configuration parameter.		
	S	270	6	Navigate to the current/history export page. Modify the number of items to display per page.		
	S	270	7	Navigate to the failed export page. Modify the number of items to display per page. Ensure that there are failed export requests in the system prior to doing this.		
	S	270	8	Log in to the BMGT GUI as an administrator.		
	S	270	9	On the main GUI page, suspend BMGT processing, both overall, and for one type of export (e.g. automatic).		
	S	270	10	Navigate to the collection configuration page. Enable a currently disabled collection for export. Disable a currently enabled collection.		
	S	270	11	Navigate to the BMGT configuration page. Modify and save the email address to which alerts are sent.		
	S	270	12	On the main GUI page, resume all BMGT processing.		

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure a TCP proxy or mock ECHO is capturing BMGT traffic.		
3	Ensure at there are at least 10 failed BMGT exports.  E.g., in the small file archive modify a granule's metadata so that the RangeBeginningDate occurs after the RangeEndingDate; manually export the granule 10 times.  Alternatively, configure mock ECHO to reject a specific granule, then manually export that granule at least 10 times.		
4	Ensure collections C1, C2 are installed.		
5	Ensure collection C1 is disabled for collection export.		
6	Ensure collection C2 is enabled for collection and granule export.		
7	Ensure granules G1, G2 are in AIM.		
8	<i>Setup</i>		#comment
9	<i>S-1 All of the following steps shall be performed using each of the browsers and operating systems listed in “Technical Document 910-TDA-042, Browsers Baseline”</i>		#comment
10	[FIXME] Identify browsers an operating systems in 910-TDA-042. Repeat the test for each combination.		
11	<i>S-2 Navigate to the BMGT GUI. Enter as an operator, without administrative privileges.</i>		#comment
12	Open the BMGT GUI in a browser. Do not log in with a password.  The BMGT GUI does not use roles (such as operator or administrator). If a password is supplied, read-write access is granted. Without a password, access is limited to read-only.		
13	<i>S-3 Attempt to suspend BMGT processing via the GUI.</i>		#comment
14	In the system status tab, click the 'Pause' button next to 'Dispatcher'		
15	<i>S-4 Navigate to the collection configuration page, and attempt to enable or disable a collection for export.</i>		#comment
16	In the BMGT GUI, on the collection configuration page, attempt to enable or disable a collection for export by clicking the collection export check box next to it.		
17	<i>S-5 Navigate to the BMGT configuration page, and attempt to modify a configuration parameter.</i>		#comment

#	Action	Expected Result	Notes
18	In the BMGT GUI, on the BMGT configuration page, attempt to modify a configuration parameter.		
19	<i>S-6 Navigate to the current/history export page. Constrain the listed export requests to cover only a specific time period.</i>		#comment
20	In the BMGT GUI, on the export request tab, modify the time range for which to display export requests.		
21	<i>S-7 Navigate to the failed export page. Constrain the errors to cover only a specific time period. Ensure that there are failed export requests in the system prior to doing this.</i>		#comment
22	In the BMGT GUI, on the export activity/error tab, filter for failed activities only and then modify the time range for which to display errors.		
23	<i>S-8 Log in to the BMGT GUI as an administrator.</i>		#comment
24	Log in to the BMGT GUI as an administrator.		
25	<i>S-9 On the main GUI page, suspend BMGT processing, both overall, and for one type of export (e.g. automatic).</i>		#comment
26	In the BMGT GUI, on the main page, suspend overall BMGT processing by pressing the 'Pause' button next to 'Dispatcher'.		
27	In the BMGT GUI, on the main page, suspend automatic BMGT processing by pressing the 'Pause' button next to 'EVENT'.		
28	Request a manual export of granule G1:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --granules &lt;GRANULE_ID&gt;		
29	Update granule G2 by changing its DayNightFlag:  update amgranule set daynightflag = 'Both' where granuleid = &lt;GRANULE_ID&gt;		This is not used by BMGT. BMGT uses the value in the native XML. though it may generate a GRUPDATE event... not sure 3/21/2013 -- Goff, Timothy
30	<i>S-10 Navigate to the collection configuration page. Enable a currently disabled collection for export. Disable a currently enabled collection.</i>		#comment
31	In the BMGT GUI, on the collection configuration page, enable collection C1 for export.		
32	In the BMGT GUI, on the collection configuration page, disable collection C2 for export.		

#	Action	Expected Result	Notes
33	<i>S-11 Navigate to the BMGT configuration page. Modify and save the email address to which alerts are sent.</i>		#comment
34	In the BMGT GUI, on the BMGT configuration page, modify the email address to which alerts are sent.		
35	<i>S-12 On the main GUI page, resume all BMGT processing.</i>		#comment
36	In the BMGT GUI, on the main GUI page, resume all BMGT processing.		
37	Log out of the BMGT GUI.		
38	Clear the browser's history, including cache, cookies, and saved sessions.		
39	Restart the browser.		
40	<i>Verification</i>		#comment
41	<i>V-1 Verify that all verification steps pass for each of the browsers and operating systems referred to in S-1.</i>		#comment
42	<i>V-2 Verify that the main GUI page in S-2 provides the number of completed metadata exports, number of pending metadata exports, and statistics about retries, warning, and errors.</i>		#comment
43	Verify the BMGT GUI &quot;Export Requests&quot; page lists the number of completed metadata exports.		
44	Verify the BMGT GUI &quot;Export Requests&quot; page lists the number of pending metadata exports.		
45	Verify the BMGT GUI &quot;Export Activity/Errors&quot; page lists the number of errors.		
46	<i>V-3 Verify that in S-3, it is not possible to suspend BMGT processing.</i>		#comment
47	Verify the BMGT GUI prevents read-only access from suspending BMGT processing.		
48	<i>V-4 Verify that in S-4, all collections are displayed, grouped by their datapool group.</i>		#comment
49	Verify the BMGT GUI collection configuration page displays all collections, grouped by datapool group.		
50	<i>V-5 Verify that in S-4, it is not possible to enable or disable a collection.</i>		#comment
51	Verify the BMGT GUI prevents read-only access from enabling or disabling a collection.		
52	<i>V-6 Verify that in S-5, it is not possible to modify any configuration parameters. Verify however that the notification email address is visible on this page.</i>		#comment

#	Action	Expected Result	Notes
53	Verify the BMGT GUI prevents read-only access from changing configuration parameters.		
54	Verify the BMGT GUI BMGT configuration page displays the notification email address.		
55	<i>V-7 Verify that in S-6, information is displayed for recent and pending export requests (it is allowable for recent and current request to be displayed on separate pages). Verify that each export request is clearly marked as to which type of export initiated it (e.g. automatic, manual, verification).</i>		#comment
56	Verify the BMGT GUI "Export Requests" page displays recent and pending export requests.		
57	Verify the BMGT GUI "Export Requests" page marks each export as one of automatic, manual, or verification.		
58	<i>V-8 Verify that in S-7, information is displayed for recent export failures and errors.</i>		#comment
59	Verify the BMGT GUI "Export Activity/Errors" page displays recent export failures and errors.		
60	<i>V-9 Verify that in S-6 and S-7, it is possible to constrain the items displayed on the page by time.</i>		#comment
61	Verify the BMGT GUI current/history allows filtering to display only those export requests withing a certain time range.		
62	Verify the BMGT GUI failed export page allows filtering to display only those failed exports within a certain time range.		
63	<i>V-10 Verify that after logging in as an administrator, suspending processing in S-9 results in a halting of BMGT processing.</i>		#comment
64	Verify that the manual export request of granule G1 does not get processed.		
65	Verify that granule G2's update does not get automatically processed.		
66	<i>V-11 Verify that in S-10, it is possible to enable and disable export for collections, both for granule and collection metadata export.</i>		#comment
67	Verify the BMGT GUI collection configuration page allows enabling collection C1 for export.		
68	Verify the BMGT GUI collection configuration page allows disabling collection C2 for export.		
69	<i>V-12 Verify that in S-11, the change made to the notification email address is saved to the database.</i>		#comment
70	Verify after restarting the browser, the BMGT GUI BMGT configuration page displays the new email address to which alerts are sent.		

#	Action	Expected Result	Notes
71	<i>V-13 Verify that after requesting resumption of BMGT processing in S-12, export requests resume being picked up and processed.</i>		#comment
72	Verify after resuming BMGT processing, granule G1 is processed to completion as a manual export.		
73	Verify after resuming BMGT processing, granule G2 is processed to completion as an automatic export.		

**TEST DATA:**

Crit id	Crit ccr no	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location	Readiness Status
				2 collections (C1, C2).				
				2 granules (G2, G2).				

**EXPECTED RESULTS:**

	V	270	1	Verify that all verification steps pass for each of the browsers and operating systems referred to in S-1.		
	V	270	2	Verify that the main GUI page in S-2 provides the number of completed metadata exports(over some period), number of pending metadata exports, and statistics about retries, warning, and errors.		
	V	270	3	Verify that in S-3, it is not possible to suspend BMGT processing.		
	V	270	4	Verify that in S-4, all collections are displayed, grouped by their datapool group.		
	V	270	5	Verify that in S-4, it is not possible to enable or disable a collection.		
	V	270	6	Verify that in S-5, it is not possible to modify any configuration parameters. Verify however that the notification email address is visible on this page.		
	V	270	7	Verify that in S-6, information is displayed for recent and pending export requests (it is allowable for recent and current request to be displayed on separate pages). Verify that each export request is clearly marked as to which type of export initiated it (e.g. automatic, manual, verification).		

	V	270	8	Verify that in S-7, information is displayed for recent export failures and errors.		
	V	270	9	Verify that in S-6 and S-7, it is possible to change the number of items displayed on the page.		
	V	270	10	Verify that after logging in as an administrator, suspending processing in S-9 results in a halting of BMGT processing.		
	V	270	11	Verify that in S-10, it is possible to enable and disable export for collections, both for granule and collection metadata export.		
	V	270	12	Verify that in S-11, the change made to the notification email address is saved to the database.		
	V	270	13	Verify that after requesting resumption of BMGT processing in S-12, export requests resume being picked up and processed.		

### 5.1.1.2 Configuring Collections For Export (ECS-ECSTC-701)

#### DESCRIPTION:

	S	280	1	<b>[Configuring Collections For Export]</b> Add a new ESDT to the ECS system and ingest some granules in that collection.		
	S	280	2	Go to the GUI collection configuration page.		
	S	280	3	Enable the collection for collection export.		
	S	280	4	Allow the export of the collection metadata to complete.		
	S	280	5	Enable the collection for granule export.		
	S	280	6	Find a collection which is not enabled for collection or granule export. Request the manual export of collection and granule metadata for this collection.		
	S	280	7	Find a collection which is enabled for collection, but not granule export. Request the manual export of collection and granule metadata for this collection.		

	S	280	8	Find a collection which is enabled for collection and granule export. Request the manual export of collection and granule metadata for this collection.		
	S	280	9	Find a collection which is not enabled for collection or granule export, but for which there is another ESDT with the same short name, but different version, which is. Request the manual export of collection and granule metadata for this collection.		

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure collection C1 is not installed.		
3	Ensure bg_collection_configuration has no row for collection C1.		
4	Ensure granules g1, g2, g3 are not in AIM.		
5	Ensure collections C2, C3, C4, C5, C6 are installed.		
6	Ensure granules g4 ... g18 are in AIM		
7	Ensure collection C2 is disabled for both collection and granule export.		
8	Ensure collection C3 is enabled for collection but not for granule export.		
9	Ensure collection C4 is enabled for both collection and granule export.		
10	Ensure collection C5 is disabled for collection or and granule export.		
11	Ensure collection C6 is enabled for both collection and granule export.		
12	Ensure a TCP proxy or mock ECHO is capturing BMGT HTTP traffic.		
13	<i>Setup and Verification</i>		#comment
14	<i>S-1 Add a new ESDT to the ECS system and ingest some granules in that collection.</i>		#comment
15	Install collection C1.		
16	Add a row for collection C1 to the bg_collection_configuration table, setting both export flags to 'N'.  See EcBgPopulateCollections.ksh for examples.		
17	Ingest granules g1, g2, g3.		
18	<i>S-2 Go to the GUI collection configuration page.</i>		#comment
19	Visit the BMGT GUI Collection Configuration page.		
20	<i>V-1 On the collection configuration page in S-2, verify that all currently installed collections are listed with their current enabled/disabled status.</i>		#comment

#	Action	Expected Result	Notes
21	Verify the BMGT GUI's collection configuration page lists collections C1 ... C6.		
22	Verify the BMGT GUI indicates C1 is disabled for both collection and granule export.		
23	Verify the BMGT GUI indicates C2 is disabled for both collection and granule export.		
24	Verify the BMGT GUI indicates C3 is enabled collection export but disabled for granule export.		
25	Verify the BMGT GUI indicates C4 is enabled for both collection and granule export.		
26	Verify the BMGT GUI indicates C5 is disabled for both collection and granule export.		
27	Verify the BMGT GUI indicates C6 is enabled for both collection and granule export.		
28	<i>V-2 On the collection configuration page in S-2, verify that the new collection is listed but not enabled for collection or granule export.</i>		#comment
29	Verify collection C1 appears on the BMGT GUI's collection configuration page.		
30	Verify the BMGT GUI indicates C1 disabled for both collection and granule export.		
31	<i>S-3 Enable the collection for collection export.</i>		#comment
32	Enable C1 for collection export by clicking the collection export check box next to the collection.		
33	<i>V-3 After enabling the collection for collection export in S-3, verify that the collection metadata for the collection is automatically exported.</i>		#comment
34	Verify that after C1 is enabled for collection export, the BMGT GUI indicates C1's metadata is successfully exported.		
35	Verify that after C1 is enabled for collection export, a BMGT log records the export of C1's metadata.		
36	Verify that after C1 is enabled for collection export, the TCP proxy logs an HTTP PUT whose body contains C1's metadata.	The request should begin with a line such as  PUT /catalog-rest/providers/\${PROVIDERID}/datasets/\${DATASETID} HTTP/1.1  where \${DATASETID} is the URL-encoded longname + ' V' + versionid, e.g.,  MODIS%2FAqua%20Gross%20Pri	

#	Action	Expected Result	Notes
		mary%20Productivity%208-Day%20L4%20Global%201km%20SIN%20Grid%20V005	
37	<i>S-4 Allow the export of the collection metadata to complete.</i>		#comment
38	Wait for collection C1's export request to be marked complete:  select r.completiontime, r.status from bg_export_request r join amcollection c on c.colelctionid = r.collectionid where c.ShortName = '&lt;SHORT_NAME&gt;'; and c.VersionId = &lt;VERSION_ID&gt;; and itemtype = 'CL';  Or, wait for the TCP proxy to log C1's HTTP PUT request.		
39	<i>S-5 Enable the collection for granule export.</i>		#comment
40	In the BMGT GUI, enable C1 for granule export by checking the granule export check box next to the collection in the collection configuration tab.		
41	<i>V-4 After enabling the collection for granule export in S-5, verify that the granule metadata for every granule in the collection is automatically exported.</i>		#comment
42	Verify that after C1 is enabled for granule export, the BMGT GUI indicates all metadata for all granules belonging to C1 is successfully exported (except logically deleted granules).		
43	Verify that after C1 is enabled for granule export, a BMGT log records the export of C1 granules' metadata (except logically deleted granules).		
44	Verify that after C1 is enabled for granule export, the TCP proxy logs an HTTP PUT for each of C1's granules, containing granule metadata (except logically deleted granules).	Each granule request should begin with a line such as  PUT /catalog-rest/providers/{PROVIDERID}/granules/{GRANULEUR} HTTP/1.1  where {GRANULEUR} is a URL-encoded geoid, such as  SC%3AMYD17A2.005%3A20062	
45	<i>S-6 Find a collection which is not enabled for collection or granule export. Request the manual export of collection and granule metadata for this collection.</i>		#comment
46	Request manual export of C2:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C2_SHORT_NAME&gt;.&lt;C2_VERSION_ID&gt;		
47	<i>V-5 For the export attempt in S-6, verify that nothing is exported.</i>		#comment
48	Verify the TCP proxy logs no request for C2's collection or granule metadata.		

#	Action	Expected Result	Notes
49	Verify a BMGT log indicates no request is attempted for C2's collection or granule metadata.		
50	<i>S-7 Find a collection which is enabled for collection, but not granule export. Request the manual export of collection and granule metadata for this collection.</i>		#comment
51	Request manual export of C3:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C3_SHORT_NAME&gt;.&lt;C3_VERSION_ID&gt;		
52	<i>V-6 For the export attempt in S-7, verify that only collection metadata is exported.</i>		#comment
53	Verify the TCP proxy logs a single HTTP PUT request, containing C3's collection metadata.  Get the request ID for the collection export from the GUI. Look in the BMGT manual log for pattern like  &quot;requestId&quot;:18629,&quot;batchId&quot;;76		
54	Verify the TCP proxy logs no HTTP PUT requests for C3's granules.		
55	Verify a BMGT log records a single export request for C3's collection metadata.		
56	Verify a BMGT log records no export attempts for any C3 granule metadata.		
57	<i>S-8 Find a collection which is enabled for collection and granule export. Request the manual export of collection and granule metadata for this collection.</i>		#comment
58	Request manual export of C4:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C4_SHORT_NAME&gt;.&lt;C4_VERSION_ID&gt;		
59	<i>V-7 For the export attempt in S-8, verify that both collection and granule metadata is exported.</i>		#comment
60	Verify the TCP proxy logs a single HTTP PUT request, containing C4's collection metadata.		
61	Verify the TCP proxy logs a single HTTP PUT request for each of C4's granules, containing granule metadata (excepting any granules which are logically deleted).		
62	Verify a BMGT log records a single export request for C4's collection metadata.		
63	Verify a BMGT log records a single export request for each of C4's granules (excepting those which are logically deleted).		
64	<i>S-9 Find a collection which is not enabled for collection or granule export, but for which there is another ESDT with the same short name, but		#comment

#	Action	Expected Result	Notes
	different version, which is. Request the manual export of collection and granule metadata for this collection.</i>		
65	Request manual export of C5:  EcBmBMGTManualStart --mode &lt;MODE&gt; --metg --metc --collections &lt;C5_SHORT_NAME&gt;,&lt;C5_VERSION_ID&gt;		
66	<i>V-8 For the export attempt in S-9, verify that nothing is exported.</i>		#comment
67	Verify the TCP proxy logs no request for C5's collection or granule metadata.		
68	Verify a BMGT log indicates no request is attempted for C5's collection or granule metadata.		

**TEST DATA:**

Crit id	Crit ccr no	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location	Readiness Status
280				6 collections (C1 ... C6)			/sotestdata/DROP_802/BE_82_01/Criteria/280	
280				3 granules for each collection (g1 ... g18)			/sotestdata/DROP_802/BE_82_01/Criteria/280	

**EXPECTED RESULTS:**

	V	280	1	On the collection configuration page in S-2, verify that all currently installed collections are listed with their current enabled/disabled status.		
	V	280	2	On the collection configuration page in S-2, verify that the new collection is listed but not enabled for collection or granule export.		
	V	280	3	After enabling the collection for collection export in S-3, verify that the collection metadata for the collection is automatically exported.		
	V	280	4	After enabling the collection for granule export in S-5, verify that the granule metadata for every granule in the collection is automatically exported.		

	V	280	5	For the export attempt in S-6, verify that nothing is exported.		
	V	280	6	For the export attempt in S-7, verify that only collection metadata is exported.		
	V	280	7	For the export attempt in S-8, verify that both collection and granule metadata is exported.		
	V	280	8	For the export attempt in S-9, verify that nothing is exported.		

### 5.1.1.3 Connecting to ECHO (ECS-ECSTC-702)

#### DESCRIPTION:

	S	630	1	<b>[Connecting to ECHO]</b> Go to the BMGT GUI without logging in as a privileged user. View and attempt to modify the ECHO connection information (ECHO URL, username, password, provider, etc.)		
	S	630	2	Log in to the BMGT GUI as a privileged user and update the ECHO connection information (ECHO URL, username, password, provider, etc.). Save the new configuration values. It may be useful to change to incorrect configuration, so it is easier to tell when the new values are being used.		
	S	630	3	Ensure that there are pending export requests for BMGT to work off.		

#### PRECONDITIONS:

#### STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Verify configurations - config files, properties files or database settings validate correctly for mode, host, and application		
3	Verify database connections to ecs connected to ecs db successfully in the configured mode		
4	Verify ECHO REST API service connections to ECHO connected to ECHO REST API successfully		
5	Verify collections under /sotestdata/DROP_802/BE_82_01/630 with ECS metadata		
6	Ensure collection C1 has been installed in the mode.		

#	Action	Expected Result	Notes
7	Ensure collection C1 is enabled for Collection and Granule Export.		
8	Assume &lt;user2&gt; as a regular user without update privileges (default, read-only).		
9	Assume &lt;user1&gt; as a privileged user with configuration update privileges (password required; write access).		
10	Assume &lt;EchoURL1&gt; and &lt;EchoURL2&gt; are two Mock ECHO connections		
11	Ensure a mock ECHO or TCP proxy is configured to capture and log BMGT requests.		
12	<i>Setup</i>		#comment
13	<i>S-1 Go to the BMGT GUI without logging in as a privileged user. View and attempt to modify the ECHO connection information (ECHO URL, username, password, provider, etc.)</i>		#comment
14	Login to BMGT GUI as &lt;user2&gt;, without a password.		
15	On the BMGT Configuration tab, attempt to change the value of the property BMGT.Exporter.IngestClient.URL to a different URL, and save the configuration.		
16	<i>V-1 Verify that in S-1, the GUI allows you to view the ECHO connection configuration, but prevents its update. Verify that the password is not shown in plain text.</i>		#comment
17	Verify that user2 cannot update the ECHO connection configuration information		
18	Verify that when entering the ECHO password, the password is not displayed in plain text		
19	<i>S-2 Log in to the BMGT GUI as a privileged user and update the ECHO connection information (ECHO URL, username, password, provider, etc.) Save the new configuration values. It may be useful to change to incorrect configuration, so it is easier to tell when the new values are being used.</i>		#comment
20	Login to the BMGT GUI with a privileged password.		
21	On the BMGT Configuration tab, attempt to change the value of the property BMGT.Exporter.IngestClient.URL to a different URL, and save the configuration.		
22	<i>V-2 Verify that in S-2, the GUI allows you to both view and update the ECHO connection configuration. Verify that the password is never shown in plain text, even when it is being entered.</i>		#comment
23	Verify that user1 can update the ECHO connection configuration information		

#	Action	Expected Result	Notes
24	Verify that when entering the ECHO password, the password is not displayed in plain text		
25	<i>S-3 Ensure that there are pending export requests for BMGT to work off.</i>		#comment
26	Enable Automatic Export		
27	Ingest granules g1 and g2 into Collection C1 (or alternatively perform a manual export for 2 granules and a collection)		
28	Verify that the requests are shown pending on the GUI		
29	<i>V-3 Verify that any exports which are picked up by BMGT and begin generation after the new values are saved (plus a reasonable lag time), are exported using the new configuration.</i>		#comment
30	Verify that the TCP proxy on ECHO URL1 does not show any HTTP PUT requests for granule g1 and g2		
31	Verify that the TCP proxy on ECHO URL2 (if it exists) shows 2 HTTP PUT requests for granule g1 and g2		

TEST DATA:

Crit id	Crit ccr no	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location	Readiness Status
			1 Collection with 2 science granules					

EXPECTED RESULTS:

	V	630	1	Verify that in S-1, the GUI allows you to view the ECHO connection configuration, but prevents its update. Verify that the password is not shown in plain text.		
	V	630	2	Verify that in S-2, the GUI allows you to both view and update the ECHO connection configuration. Verify that the password is never shown in plain text, even when it is being entered.		
	V	630	3	Verify that any exports which are picked up by BMGT and begin generation after the new values are saved (plus a reasonable lag time), are exported using the new configuration.		

#### 5.1.1.4 Configuring Collections For Export for Lance (ECS-ECSTC-703)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Ensure a collection is installed.		
2	Add a row for the collection to the bg_collection_configuration table, setting both export flags to 'Y'. See EcBgPopulateCollections.ksh for examples.		
3	Verify the BMGT GUI's collection configuration page lists the collection.		
4	Verify the BMGT GUI indicates that the collection is enabled for both collection and granule export.		
5	Request the manual export of collection and granule metadata for this collection. EcBmBMGTManualStart --mode <MODE> --metg --metc --collections <SHORT_NAME>.<VERSION_ID>		
6	Verify that both collection and granule metadata is exported.		
7	Verify a BMGT log records a single export request for the collection metadata.		
8	Verify a BMGT log records a single export request for each of collection granules (excepting those which are logically deleted).		

TEST DATA:

EXPECTED RESULTS:

#### 5.1.2 AIM

##### 5.1.2.1 Install ESDTs with a variety of spatial search types (ECS-ECSTC-704)

DESCRIPTION:

---

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1			comment
2	[Install ESDTs with a variety of spatial search types]		
3	Verify that the descriptor ODL files for the ESDT being installed are moved from the installation source directory into the configured descriptor directory.		
4	Prepare descriptor files for several (at least 5) ESDTs covering a variety of ESDT types described below, such that the descriptors conform to the ECS data model and the XML schema validation rules, guaranteeing successful ESDT installation. Examples of such can be found in /sotestdata/DROP_721/DS_7E_01/Criteria/2030, and are listed as follows: 1. ESDT descriptor with a spatial search attribute of Rectangle MOD09CMG.005 MYD09CMG.005 2. ESDT descriptor with a spatial search attribute of GPolygon MOD14.005 MYD14.005 3. ESDT descriptor with a spatial search attribute of NotSupported AEPOE7W.001 PM1GBAD1.001 4. ESDT descriptor with a spatial search attribute of Orbit AE_Land.002 AE_Rain.002 5. ESDT descriptor with a spatial search attribute of Point g3bssp.007 g3bt.007		
5	Ensure collections are not already installed.		
6	Copy the descriptor files to the source directory (specified in the GUI).		
7	Verify that the descriptors are listed in the GUI.		
8	From the ESDT Maintenance GUI, select the ESDTs in the source directory and initiate installation.		
9	Upon completion of the ESDT installation, verify that the GUI displays a message indicating number of ESDTs successfully installed.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

### 5.1.2.2 Update multiple ESDTs, success case (ECS-ECSTC-705)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1			comment
2	Preconditions		comment
3	Modify the ESDT descriptor ODL files for several (at least 6) previously installed ESDTs, such that the modifications include the following variety of ESDT updates: 1. removing one or more existing collection level attributes that are not restricted 2. removing DLLName parameter from the Collection metadata group 3. changing the ShortName in the CollectionAssociation group 4. changing a mandatory inventory attribute to optional 5 adding one or more optional inventory metadata attributes 6. adding one or more new qualifiers on existing events		comment
4	Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2060/Replacement as the initial descriptors. Use only GLA01.013 ... GLA06.013; do not use GLA07.013.		comment
5	Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2120 as the replacement descriptors.		
6	Compare each replacement descriptor with its initial counterpart, and note the differences.		
7	Ensure none of the test collections are installed, according to the ESDT Maintenance GUI.		
8	Note the values associated with the following names in EcDsAmESDTMaint.properties: descriptor.target.dir descriptor.source.dir mcf.target.dir e.g., descriptor.target.dir=/stornext/smallfiles/OPS/descriptor descriptor.source.dir=/usr/ecs/OPS/CUSTOM/data/ESSmcf.target.dir=/stornext/smallfiles/OPS/mcf		
9	Install the initial test collections: Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}. Log in to the ESDT Maintenance GUI. Click the "Install New ESDTs/Update Existing ESDTs" button. Select all the test descriptors. Click the "Proceed with installation/update" button. Ensure that all descriptors install		comment

#	Action	Expected Result	Notes
	successfully.Ensure the descriptor files are removed from the source directory.		
10	Setup: Testing ESDT Updates		
11	Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}.		
12	Log in to the ESDT Maintenance GUI.		
13	Click the &quot;Install New ESDTs/Update Existing ESDTs&quot; button.		
14	Select all the test descriptors.		
15	Click the &quot;Proceed with installation/update&quot; button.		
16	Upon completion of the ESDT update, verify that the GUI displays a message indicating the number of ESDTs successfully updated.		
17	Verify that, upon completion of updates, the ESDT Maintenance GUI displays a message saying that the ESDT changes will take effect only after the Ingest service is re-started.		

**TEST DATA:**

Any current data types. Initial ESDTs: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2060/Replacement Updated ESDTs: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2120 The 'update' ESDT descriptor files have been modified as follows: 1. removing one or more existing collection level attributes that are not restricted DsESDTGIGLA01.013.desc 2. removing DLLName parameter from the Collection metadata group DsESDTGIGLA02.013.desc 3. changing the ShortName in the CollectionAssociation group DsESDTGIGLA03.013.desc 4. changing a mandatory inventory attribute to optional DsESDTGIGLA04.013.desc 5 adding one or more optional inventory metadata attributes DsESDTGIGLA05.013.desc 6. adding one or more new qualifiers on existing events DsESDTGIGLA06.013.desc QAPercentInterpolatedData

**EXPECTED RESULTS:**

5.1.2.3 Delete multiple ESDTs, success case (ECS-ECSTC-706)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1			comment

#	Action	Expected Result	Notes
2	DS_7E_01 ESDT Maintenance GUI ITP:http://dmsserver.gsfc.nasa.gov/release721/ESDT_Maint_GUI/ITP_DS_7E_01_ESDTGUI.doc.		comment
3	Preconditions		
4	Ensure a database client is connected to the ecs database.		
5	Identify 3 ESDTs for deletion (C1, C2, C3).		
6	On the ESDT Maintenance GUI &quot;ESDT List&quot; page, select C1, C2, and C3.		
7	Click the “Delete Selected ESDTs” button at the bottom of the page to begin the ESDT deletion.		
8	Select OK at the confirmation prompt.		
9	Verify the ESDT Maintenance GUI displays a message indicating 3 ESDTs were deleted.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**5.1.3 CLS**

**5.1.3.1 Order Status GUI: Display Order Status, OD\_S6\_05. Criterion 130 (ECS-ECSTC-720)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Request the status of a single, completed order that was handled by the OMS and by specifying a valid Order Id and user contact email address.		comment
2	Verify the following:		
3	a. Order Id is correctly displayed.		
4	b. Submission date/time is correctly displayed.		
5	c. Order state is correctly displayed.		

---

---

#	Action	Expected Result	Notes
6	d. Order completion date/time is correctly displayed.		
7	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed		
8	f. Order state and request states are presented in terms that an end user can understand.		
9	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
10	h. There is an indication that additional request details are available.		

TEST DATA:

EXPECTED RESULTS:

5.1.3.2 Order Status GUI: Display Order Status based on History Range, 0D\_S6\_05, Criterion 180 (ECS-ECSTC-721)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request order history by specifying a starting and ending date, user contact email address, and one valid Order Id for the user.Ensure that at least 100 orders are displayed that represent a mix of orders for which request details are available for some orders and not available for others.Ensure that the date range requires the Order Status Interface to retrieve information from both the Order Manager operational tables and archive tables.Ensure that at least one of the orders contains granules that were processed by the external subsetter and one of the orders contains granules that were processed by HEG.		
2	Verify that the correct orders are returned and sorted by submission date/time.		comment
3	For each order verify the following:		
4	a. Order Id is correctly displayed.		
5	b. Submission date/time is correctly displayed.		
6	c. Order state is correctly displayed.		

---

---

---

---

#	Action	Expected Result	Notes
7	d. Order completion date/time is correctly displayed.		
8	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed, including the processing description for the granule to be processed by the external subsetter, in a manner which should be generally understandable by the user.		
9	f. Order state and request states are presented in terms that an end user can understand.		
10	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
11	h. An indication is provided when request details are not available for an order.		

TEST DATA:

EXPECTED RESULTS:

5.1.3.3 Order Status GUI: Display Order Status based on History Range, 0D\_S6\_05, Criterion 180 - Specific to PVC Little Lance (ECS-ECSTC-722)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Request order history by specifying a starting and ending date, user contact email address, and one valid Order Id for the user. Ensure that the date range requires the Order Status Interface to retrieve information from both the Order Manager operational tables and archive tables.		
2	Verify that the correct orders are returned and sorted by submission date/time.		
3	For each order verify the following:		
4	a. Order Id is correctly displayed.		
5	b. Submission date/time is correctly displayed.		

---

---

#	Action	Expected Result	Notes
6	c. Order state is correctly displayed.		
7	d. Order completion date/time is correctly displayed.		
8	e. Request information (Request Id, request state, media type, number of granules, request completion date/time) is correctly displayed, in a manner which should be generally understandable by the user.		
9	f. Order state and request states are presented in terms that an end user can understand.		
10	g. Order and request information are properly segmented across html pages in accordance with the user-configurable number of items per page.		
11	h. An indication is provided when request details are not available for an order.		

TEST DATA:

EXPECTED RESULTS:

#### 5.1.4 DMS

##### 5.1.4.1 Single Granule FtpPull and FtpPush Order (ECS-ECSTC-719)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Submit a FtpPull and FtpPush order through EWOC.		
2	Verify that both the FtpPull order and the FtpPush order shipped.		

TEST DATA:

EXPECTED RESULTS:

---

### 5.1.5 DPLIngest

#### 5.1.5.1 Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-713)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1			comment
2	Description and Objectives: Verifies Data Providers can be configured using the DPL Ingest GUI		comment
3	[View Data Providers]		
4	As the 'ingest admin' operator, navigate to the Data Provider page on the Data Pool Ingest GUI.		
5	Verify that all provider information in the INGST CI database appears correctly on the Data Provider page.		comment
6	[Configure Data Providers]		
7	Edit existing provider information (if any) and define new providers to fulfill the following requirements: At least one data provider should have an FTP notification method, at least one data provider should have an scp notification method, at least one data provider should have an http notification method, at least one data provider should have an email notification method, at least one data provider should have a combination FTP/email notification method, and at least one data provider should have a combination scp/email notification method. and at least one data provider should have a combination http/email notification method.Scp type/cipher combinations to include in the test are: F-secure/None; OpenSSH/aes128; OpenSSH/3des.At least one provider must use active mode.At least one should use passive mode.		
8	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit data provider information.		
9	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing data providers before saving this information.		
10	Verify that the ingest admin operator has the authorization to define new data providers (i.e., has access to the data provider pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		

#	Action	Expected Result	Notes
11	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16110 to be entered.		
12	Verify that all of the notification methods in S-DPL-16150 can be entered or selected on the Data Pool Ingest GUI, as appropriate for the selected transfer method as per S-DPL-16110.		
13	For one provider with an FTP Notification method, verify that the Data Pool Ingest GUI allows the related FTP information to be entered (FTP host, destination directory, login id, password)		
14	Verify that the FTP password entered is not shown or stored in the clear.		
15	Verify that the FTP password entered is not shown in the Data Pool Ingest GUI log.		
16	For one provider with an email notification method, verify that the Data Pool Ingest GUI allows the related email address to be entered.		
17	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new data providers before saving this information.		
18	For one provider with an scp notification method, verify that the Data Pool Ingest GUI allows the related scp information to be entered (scp host, destination directory, login id, password, scp type, and cipher).		
19	Verify that the scp password entered is not shown or stored in the clear.		
20	Verify that the scp password entered is not shown in the Data Pool Ingest GUI log.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

5.1.5.2 Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-714)

**DESCRIPTION:**

**PRECONDITIONS:**

STEPS:

#	Action	Expected Result	Notes
1			comment
2	Description and Objectives: Verifies Polling Locations can be configured using the DPL Ingest GUI		comment
3	[Configure Polling Locations]		
4	As the Ingest Admin operator use the Data Pool Ingest GUI to define all polling locations that will be used for testing this ticket.(Table of data provider to polling location mappings, and related S-DPL-16230 information for each polling location TBS as part of ITP). At least one polling location should have an FTP polling method, at least one polling location should have a local polling method, and at least one polling location should have an scp polling method. and at least one polling location should have an http polling method.At least one polling location using FTP shall be for a provider using local transfers.		
5	Verify that the ingest admin operator has the authorization to define the polling locations (i.e., has access to the polling location pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
6	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16230 to be entered.		
7	Verify that all of the polling methods in S-DPL-16250 can be entered or selected on the Data Pool Ingest GUI.		
8	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new polling locations before saving this information.		comment
9	[Edit Polling Locations]		
10	For one polling location, as the 'ingest admin' operator, edit all of its existing configuration parameters.(NOTE: After this criterion is complete, values of these configuration parameters should be reset to appropriate values for processing all PDRs in criterion 300).		
11	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit polling location information.		
12	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing polling locations before saving this information.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

5.1.5.3 Use the DPL Ingest GUI to configure Remote Transfer Hosts (ECS-ECSTC-715)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1			comment
2	Description and Objectives: Verifies Remote Transfer Hosts can be configured using the DPL Ingest GUI		comment
3	[Configure FTP hosts]		
4	As the 'ingest admin' operator, navigate to the FTP Host page on the Data Pool Ingest GUI.		
5	Edit existing ftp host information (if any, from the INGST CI), per S-DPL-16260, and define new ftp hosts such that all ftp hosts which will be used for testing this ticket are defined.(Table of ftp hosts and related S-DPL-16260 information TBS as part of ITP).At least two ftp hosts should be defined that are not configured for the INGST subsystem.		
6	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit ftp host information.		
7	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing ftp hosts and definitions of new ftp hosts before saving this information.		
8	Verify that the ingest admin operator has the authorization to define the ftp hosts (i.e., has access to the ftp host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
9	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16260 to be entered.		
10	[Configure HTTP hosts]		
11	Configure a HTTP host use the step similar to 'Configure FTP hosts'.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

5.1.5.4 Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-716)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1			comment
2	Description and Objectives: Verifies SCP Hosts can be configured using the DPL Ingest GUI		comment
3	[Configure hosts for scp access]		
4	As the 'ingest admin' operator, navigate to the host page on the Data Pool Ingest GUI.		
5	Edit existing information for hosts accessed via scp (if any, from the INGST CI), per S-DPL-16290, and define new scp hosts such that all scp hosts which will be used for testing this ticket are defined.(Table of hosts accessed via scp, and related S-DPL-16290 information TBS as part of ITP.)Scp type/cipher combinations to include in the test are: F-secure/None; OpenSSH/aes128; OpenSSH/3des,i.e., the test needs to involve several different providers.		
6	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to enter and edit the scp host information in S-DPL-16290.		
7	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing scp host parameters and definitions of new scp host parameters before saving this information.		
8	Verify that the ingest admin operator has the authorization to define the scp host parameters in S-DPL-16290 (i.e., has access to the host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		

TEST DATA:

Any current data types

EXPECTED RESULTS:

5.1.5.5 View and modify the DPL Ingest GUI configuration (ECS-ECSTC-717)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1			comment
2	Description and Objectives: Verifies DPL Ingest GUI configuration can be viewed and modified correctly.		comment
3	[View collection configuration]		
4	As an 'ingest admin' operator, use the Data Pool Ingest GUI to list all data types for which configuration parameters were entered in criterion 120.		
5	Verify that "Ignore Validation Warnings" and "Public in Data Pool" are displayed for each ESDT.		
6	<i>Since release 8.2, newly installed data types will be automatically added to the DPL Ingest InDataType table and visible in 'Data Types with DPL Ingest Configurations' table.</i>		#comment
7	Verify that the Data Pool Ingest GUI provides the operator a method to quickly select or scan for a subset of existing Data Pool collections without having to enter the full ESDT name and version.		comment
8	[Edit collection configuration]		
9	As an 'ingest admin' operator, use the Data Pool Ingest GUI to edit publication policy configuration parameters for a Data Pool collection.		
10	Verify that the edited configuration parameters are correctly updated in the database.		comment
11	[Reconfigure data type parameters.]		
12	Log into the DPL Ingest GUI as an 'ingest admin' operator, and configure a data type (ESDT and Version) to be published in the public Data Pool upon insert.		

#	Action	Expected Result	Notes
13	Submit several PDRs for the data type configured above. There needs to be a sufficient number of PDRs and granules such that some granules will complete ingest before the configuration change made during the test (see next step), and some granules will not start ingest until after the configuration change has been applied.		
14	After the first few granules completed ingest, re-configure the data type via the DPL Ingest GUI such that the data type no longer will be published in the Data Pool. Also change the minimum retention period sufficiently long so that the ingested granules are not cleaned up immediately after archiving, to allow time for the verification steps below.		
15	Verify that the first few granules that were ingested before the re-configuration are queued with the Data Pool Insert Service for insertion into the public Data Pool area.		
16	Verify that all granules that started ingest one minute or more after the re-configuration are inserted into the non-public Data Pool area, and are NOT queued with the Data Pool Insert Service for insertion into the public Data Pool area. NOTE: if the Science granule has an associated Browse the Browse will be published even if the Science granule is "hidden."		
17	Verify that ALL browse granules ingested as part of the test are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**5.1.5.6 Add & Modifying Volume Groups (ECS-ECSTC-718)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1			comment
2	Description and Objectives: Verifies it is possible to add and modify volume groups using the DPL Ingest GUI and that the server correctly uses the new volume groups.		comment
3	[Add & Modifying Volume Groups]		
4	Using the DPL Ingest GUI add a primary and backup Volume Group for a collection.		
5	Ingest a granule and verify that it goes to the primary and backup volume groups.		
6	Modify Volume group and give it a new path		
7	Ingest a granule and verify that it goes to the new path.		
8	Click on reports and verified that the addition and modification from above are displayed in the report.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**5.2 WuFtp**

**5.2.1 WuFTP Regression Test (PVC Little Lance) (ECS-ECSTC-712)**

**DESCRIPTION:**

Perform the following regression tests used for previous WU-FTP release testing for compatibility with ECS release 8.x.x custom code and databases:

1. Download an uncompressed .xml file with no checksum using (.zip, .gz, .Z), and verify that the download succeeds without a problem.
2. Download a .hdf file with no checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem.
3. Download a .hdf file with an MD5 checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem.
4. Download a folder using (.zip, tar, tar.gz, tar.Z), and verify that the download succeeds without problem.
5. Submit an FtpPull order for public granules whose files are not retrievable by anonymous/ftp users (configurable via /etc/ftpd/ftppass noretrieve directives).
  - Submit another order for hidden granules.
  - Verify that the orders (non-compressed and compressed) are only retrievable with earth data login.

- Submit an FtpPull order for granules, some granules whose files are marked as noretrieve, some whose aren't. Verify that the files that are marked as noretrieve can only be retrieved with earth data login, and other can also retrieved by anonymous/ftp user, and that the compressed files can only be retrieved with earth data login.
- Verify that the compressed file download is reported to EMS (This is not part of this ncr, but we want to know this type of download is reported).

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Download an uncompressed .xml file with no checksum using (.zip, .gz, .Z), and verify that the download succeeds without a problem. 		
2	Download a .hdf file with no checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem. 		
3	Download a .hdf file with an MD5 checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem.  		
4	Download a folder using (.zip, tar, tar.gz, tar.Z), and verify that the download succeeds without problem.		
5	Submit an FtpPull order for public granules whose files are not retrievable by anonymous/ftp users (configurable via /etc/ftpd/ftpaccess noretrieve directives).  Submit another order for hidden granules.  Verify that the orders (non-compressed and compressed) are only retrievable with earth data login.  Submit an FtpPull order for granules, some granules whose files are marked as noretrieve, some whose aren't. Verify that the files that are marked as noretrieve can only be retrieved with earth data login, and other can also retrieved by anonymous/ftp user, and that the compressed files can only be retrieved with earth data login.  Verify that the compressed file download is reported to EMS (This is not part of this ncr, but we want to know this type of download is reported).   		

TEST DATA:

EXPECTED RESULTS:

## 6 HYRAX REGRESSION TESTS

### 6.1 EDSC test - Reformatting L1/L2 collections for NetCDF-3 output format (ECS-ECSTC-723)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Ensure that the test SMAP L1 collections are installed in the mode and and the collections and granules are exported to CMR.		
2	Ensure that the SMAPL1L2 is configured in the mode and enabled for the test collections.		
3	Ensure that dataObjects config script has been run for the collection and the bands are displayed for the collectons in edf testbed GUI.		
4	Ensure that the NetCDF-3 formats is enabled for SMAPL1L2		
5	Ensure that the spatial and band subsetting is enabled for the collection for the SMAPL1L2		
6	Upload the service form for the collection to earthdata/cmr using the Data Access Config GUI or with the FormUpload.rb Script		
7	Ensure that the form uploaded and the options are assigned in testbed pump		
8	Navigate to the Earthdata Search Client (EDSC) GUI. <a href="https://search.uat.earthdata.nasa.gov">https://search.uat.earthdata.nasa.gov</a>		
9	Search for granules for the test SMAP L1/L2 collections.		
10	Click on Perform Service. Verify that the service options for the collection are displayed correctly		
11	Enter a valid email address		
12	Select the netcdf format - NetCDF-3		
13	Submit Service Request. Verify that the service request succeeded.		
14	Verify that the subsetting request, constraints, statistics, outputs are logged.	Some of the logging is in EcDIDaRqs.debug0.log and Egi logs.	
15	Verify that an email is sent to the requested email address that the service request is complete with links to download the subsetted files		
16	Download the subsetted file.		
17	Verify that there is a link to download the request summary		

#	Action	Expected Result	Notes
18	Verify that the output file has all the datasets in the subset request. Panoply can be used to view netcdf outputs. ncdump can also be used to verify the output data		
19	Verify that the request summary link displays the request summary with all the subsetting parameter information, the input and the output granule list		
20	Download the files from the output links and verify that the output file format is the output format selected in the request.		

TEST DATA:

EXPECTED RESULTS:

**6.2 EDSC test - Reformatting SMAP L1/L2 collections for ASCII output format (ECS-ECSTC-724)**

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Ensure that the test SMAP L1/L2 collections are installed in the mode and and the collections and granules are exported to CMR.		
2	Ensure that the SMAPL1L2 service is configured in the mode and enabled for the test collections.		
3	Ensure that dataObjects config script has been run for the collection and the bands are displayed for the collectons in edf testbed GUI.		
4	Ensure that the ASCII format is enabled for SMAPL1L2 service		
5	Ensure that the spatial and band subsetting is enabled for the collection for the SMAPL1L2		
6	Upload the service form for the collection to earthdata/cmr using the Data Access Config GUI or with the FormUpload.rb Script		
7	Ensure that the form uploaded and the options are assigned in testbed pump		
8	Navigate to the Earthdata Search Client (EDSC). <a href="https://search.uat.earthdata.nasa.gov">https://search.uat.earthdata.nasa.gov</a>		

#	Action	Expected Result	Notes
9	Search for granules for the test SMAP L1/L2 collections.		
10	Click on Perform Service. Verify that the service options for the collection are displayed correctly		
11	Enter a valid email address		
12	Select the supported ascii format		
13	Submit Service Request. Verify that the service request succeeded.		
14	Verify that the subsetting request, constraints, statistics, outputs are logged.	Some of the logging is in EcDIDaRqs.debug0.log and Egi logs.	
15	Verify that an email is sent to the requested email address that the service request is complete with links to download the subsetted files		
16	Download the subsetted file.		
17	Verify that there is a link to download the request summary		
18	Verify that the output file has all the datasets in the subset request.		
19	Verify that the request summary link displays the request summary with all the subsetting parameter information, the input and the output granule list		
20	Download the files from the output links and verify that the output file format is the output format selected in the request.		

TEST DATA:

EXPECTED RESULTS:

### 6.3 Manual Export - Opendap URL (ECS-ECSTC-725)

#### DESCRIPTION:

S 180 1 Request the manual export of metadata for a single granule, specifying the granule ID. S 180 2 Request the manual export of metadata fro granules in a specified collection. Ensure that the collection contains at least one logically deleted granule. S 180 3 Request, in a single manual operation, the export of metadata for the following: a) Normal granule b) Logically deleted granule. c) Physically deleted granule (specify Short Name, VersionId and GranuleId – NOTE: it is easiest to use a valid ShortName and VersionID but a made up granuleID) S 180 4 Request, in a single manual operation, specifying the export of inserts-only, the export of metadata for the following: a) Normal granule b) Logically deleted granule. c) Physically deleted granule (specify Short Name, VersionId and GranuleId– NOTE: it is easiest to use a valid ShortName and VersionID but a made up granuleID) S 180 5 Request, in a single manual operation, specifying the export of deletes-only, the export of metadata for the following: a) Normal granule b) Logically deleted granule. c) Physically deleted granule (specify Short Name, VersionId and GranuleId– NOTE: it is easiest to use a valid ShortName and VersionID but a made up granuleID)

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Verify configurations - config files, properties files or database settings validate correctly for mode, host, and application		This test case is blocked by ECS-162
3	Verify database connections to ecs connected to ecs db successfully in the configured mode		
4	Verify ECHO REST API service connections to ECHO connected to ECHO REST API successfully		
5	Verify collections under /sotestdata/DROP_802/BE_82_01/10 with ECS metadata		
6	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic. or use the bmgmt exporter log		
7	<i>Setup</i>		#comment
8	Ensure Collections C1, C2, C3 has been installed in the mode.		
9	Ensure Collections C1, C2, C3 are enabled for Collection, Granule, and OPeNDAP URL Export.		
10	Ensure the Collections C1, C2, C3 have a few ingested granules.		
11	Ensure collections C1, C2, C3 have been exported to ECHO.		
12	Ensure granules in collections C1, C2, C3 have been exported to ECHO.		
13	<i>S-1 Request the manual export of metadata for a single granule, specifying the granule ID.</i>		#comment
14	Identify a granule g1 in Collection C1 that can be exported.		
15	EcBmBMGTManualStart &lt;MODE&gt; --metg -g &lt;g1 GRANULEID&gt;		
16	<i>V-1 Verify that the operation in S-1 results in the export of a single HTTP PUT containing the full granule metadata and opendap url of the requested granule.</i>		#comment
17	Verify that the TCP proxy log (/tools/common/test/BE_82_01/proxy/&lt;MODE&gt;/tcp.log) shows an HTTP PUT request for granule g1. (There may be more than one HTTP request, e.g., if there are network issues.) or check the bmgmt exporter log		

#	Action	Expected Result	Notes
18	Verify that the TCP proxy log shows that the request contains the full granule metadata and opendap url for granule g1.  or check the bmgmt exporter log		
19	<i>S-2 Request the manual export of metadata for granules in a specified collection. Ensure that the collection contains at least one logically deleted granule.</i>		#comment
20	Identify granules g2, g3 in Collection C2 that can be logically deleted.		
21	Logically delete a granule g2, g3: ./EcDsBulkDelete.pl -physical -user &lt;dbuser&gt; -password &lt;passwd&gt; -geoidfile /path/to/geoids_g2_g3.txt		
22	Manually Export granules in Collection C2 ./EcBmBMGTManualStart &lt;MODE&gt; --metg -c &lt;C2_SHORTNAME&gt;.&lt;C2_VERSIONID&gt;		
23	<i>V-2 Verify that the operation in S-2 results in the export of a single HTTP PUT containing the full granule metadata and opendap url of the requested granule for each granule in the specified collection. Verify that an HTTP DELETE is also exported for any logically deleted granules in the collection.</i>		#comment
24	Verify that the TCP proxy log shows an HTTP PUT request for each granule in Collection C2 except for g2 and g3. (There may be more than one HTTP request, e.g., if there are network issues.)  or check the bmgmt exporter log		
25	Verify that the TCP proxy shows that the request contains the full granule metadata and opendap url for each granule in Collection C2 except for g2 and g3.  or check the bmgmt exporter log		
26	Verify that an HTTP DELETE is exported for each of granules g2 and g3 (There may be more than one HTTP request, e.g., if there are network issues.)		
27	<i>S-3 Request, in a single manual operation, the export of metadata for the following:  a) Normal granule  b) Logically deleted granule.  c) Physically deleted granule (specify Short Name, VersionId and GranuleId – NOTE: it is easiest to use a valid ShortName and VersionID but a made up granuleID)</i>		#comment
28	Identify granule g4 in Collection C3 as a normal science granule that can be exported.		

#	Action	Expected Result	Notes
29	Identify granule g5 that is logically deleted in Collection C3 ./EcDsBulkDelete.pl -physical -user &lt;dbuser&gt; -password &lt;passwd&gt; -geoidfile /path/to/geoids_g5.txt		
30	Identify granule g6 which is physically deleted in Collection C3 or make up a nonexistent granuleid in Collection C3 ./EcDsBulkDelete.pl -physical -user &lt;dbuser&gt; -password &lt;passwd&gt; -geoidfile /path/to/geoids_g6.txt ./EcDIUnpublishStart.pl -mode &lt;MODE&gt; -granules &lt;g6&gt; ./EcDeletionCleanup.pl -user &lt;db_user&gt; ./EcBmBMGTManualStart --mode &lt;MODE&gt; --metg -g &lt;g4_granuleid&gt;,&lt;g5_granuleid&gt;,&lt;g6_granuleid&gt;		
31	<i>V-3 Verify that the operation in S-3 is in a complete state and the following are exported for each respective operation:  a) HTTP PUT for containing the full granule metadata and opendap url.  b) HTTP DELETE  c) HTTP DELETE</i>		#comment
32	Verify that the TCP proxy log shows an HTTP PUT request for granule g4. (There may be more than one HTTP request, e.g., if there are network issues.)  or check the bmgmt exporter log		
33	Verify that the TCP proxy log shows that the request contains the full granule metadata and opendap url for granule g4.  or check the bmgmt exporter log		
34	Verify that the TCP proxy log shows an HTTP DELETE exported for granule g5. (There may be more than one HTTP request, e.g., if there are network issues.)  or check the bmgmt exporter log		
35	Verify that the TCP proxy log shows an HTTP DELETE exported for granule g6. (There may be more than one HTTP request, e.g., if there are network issues.)  or check the bmgmt exporter log		
36	<i>S-4 Request, in a single manual operation, specifying the export of inserts-only, the export of metadata for the following:  a) Normal granule  b) Logically deleted granule.  c) Physically deleted granule (specify Short Name, VersionId and GranuleId- NOTE: it is easiest to use a valid ShortName and VersionID but a made up granuleID)</i>		#comment
37	./EcBmBMGTManualStart --mode &lt;MODE&gt; --insertonly --metg -g &lt;g4_granuleid&gt;,&lt;g5_granuleid&gt;,&lt;g6_granuleid&gt;		
38	<i>V-4 Verify that the operation in S-4 is in a complete state and the following are exported for each respective operation:  a) HTTP PUT for containing the full granule metadata.  b) No export  c) No export</i>		#comment

#	Action	Expected Result	Notes
39	Verify that the TCP proxy log shows an HTTP PUT request for granule g4. (There may be more than one HTTP request, e.g., if there are network issues.)  or check the bmgt exporter log		
40	Verify that the TCP proxylog shows that the request contains the full granule metadata and opendir url for granule g4.  or check the bmgt exporter log		
41	Verify that the TCP proxy log does not show any export request for granule g5.  or check the bmgt exporter log		
42	Verify that the TCP proxy log does not show any export request for granule g6.  or check the bmgt exporter log		
43	<i>S-5 Request, in a single manual operation, specifying the export of deletes-only, the export of metadata for the following:  a) Normal granule  b) Logically deleted granule.  c) Physically deleted granule (specify Short Name, VersionId and GranuleId- NOTE: it is easiest to use a valid ShortName and VersionID but a made up granuleID)</i>		#comment
44	./EcBmBMGTManualStart &lt;MODE&gt; --deleteonly --metg -g &lt;g4_granuleid&gt;,&lt;g5_granuleid&gt;,&lt;g6_granuleid&gt;;		
45	<i>V-5 Verify that the operation in S-5 is in a complete state and the following are exported for each respective operation:  a) No export  b) HTTP DELETE  c) HTTP DELETE</i>		#comment
46	Verify that the TCP proxy log shows no HTTP export request for granule g4.  or check the bmgt exporter log		
47	Verify that the TCP proxy log shows an HTTP DELETE exported for granule g5. (There may be more than one HTTP request, e.g., if there are network issues.)  or check the bmgt exporter log		
48	Verify that the TCP proxy log shows an HTTP DELETE exported for granule g6. (There may be more than one HTTP request, e.g., if there are network issues.)  or check the bmgt exporter log		

**TEST DATA:**

Crit id Crit ccr no Test Data Description Data Type Requirements Metadata Requirements Volume Requirements Size Requirements Data Location Readiness Status 3 collections with at least 3 granules in each

**EXPECTED RESULTS:**

V 180 1 Verify that the operation in S-1 results in the export of a single HTTP PUT containing the full granule metadata and opendir url of the requested granule. V 180 2 Verify that the operation in S-2 results in the export of a single HTTP PUT containing the full granule metadata and opendir url of the

requested granule for each granule in the specified collection. Verify that an HTTP DELETE is also exported for any logically deleted granules in the collection. V 180 3 Verify that the operation in S-3 is in a complete state and the following are exported for each respective operation: a) HTTP PUT for containing the full granule metadata and opendap url. b) HTTP DELETE c) HTTP DELETE V 180 4 Verify that the operation in S-4 is in a complete state and the following are exported for each respective operation: a) HTTP PUT for containing the full granule metadata and opendap url. b) No export c) No export V 180 5 Verify that the operation in S-5 is in a complete state and the following are exported for each respective operation: a) No export b) HTTP DELETE c) HTTP DELETE

#### 6.4 ESI tests - SMAPL1L2 collections - formatting (ECS-ECSTC-726)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Login as cmshared on f5oml01v and set view to baseline or the latest view		
2	setenv MODE OPS		
3	cd /ecs/formal/COMMON/automation/autotest directory		
4	<i>Run each of the following SMAPL1L2 cucumber tests</i>		#comment
5	Run  cucumber --profile feature_log features/DataAccess/esi/smap/esi_smapl1_format.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed.	
6	Run  cucumber --profile feature_log features/DataAccess/esi/smap/esi_smapl2_format.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed.	

TEST DATA:

EXPECTED RESULTS:

#### 6.5 EDSC test - Reformatting L3/L4 collections for NetCDF-3 output format (ECS-ECSTC-727)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Ensure that the test SMAP L3/L4 collections are installed in the mode and and the collections and granules are exported to CMR.		
2	Ensure that the HEG service is configured in the mode and enabled for the test collections.		
3	Ensure that dataObjects config script has been run for the collection and the bands are displayed for the collectons in edf testbed GUI.		
4	Ensure that the NetCDF-3 formats is enabled for HEG Service		
5	Ensure that the spatial and band subsetting is enabled for the collection for the HEG service		
6	Upload the service form for the collection to earthdata/cmr using the Data Access Config GUI or with the FormUpload.rb Script		
7	Ensure that the form uploaded and the options are assigned in testbed pump		
8	Navigate to the Earthdata Search Client (EDSC). <a href="https://search.uat.earthdata.nasa.gov">https://search.uat.earthdata.nasa.gov</a>		
9	Search for granules for the test SMAP L3/L4 collections.		
10	Click on Perform Service. Verify that the service options for the collection are displayed correctly		
11	Enter a valid email address		
12	Select the netcdf format - NetCDF-3		
13	Submit Service Request. Verify that the service request succeeded.		
14	Verify that the subsetting request, constraints, statistics, outputs are logged.	Some of the logging is in EcDIDaRqs.debug0.log and Egi logs.	
15	Verify that an email is sent to the requested email address that the service request is complete with links to download the subsetted files		
16	Download the subsetted file.		
17	Verify that there is a link to download the request summary		
18	Verify that the output file has all the datasets in the subset request. Panoply can be used to view netcdf outputs. ncdump can also be used to verify the output data		
19	Verify that the request summary link displays the request summary with all the subsetting parameter information, the input and the output granule list		
20	Download the files from the output links and verify that the output file format is the output format selected in the request.		

TEST DATA:

EXPECTED RESULTS:

### 6.6 EDSC test - Reformatting SMAP L3/L4 collections for ASCII output format (ECS-ECSTC-728)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Ensure that the test SMAP L3/L4 collections are installed in the mode and the collections and granules are exported to CMR.		
2	Ensure that the HEG service is configured in the mode and enabled for the test collections.		
3	Ensure that dataObjects config script has been run for the collection and the bands are displayed for the collectons in edf testbed GUI.		
4	Ensure that the ASCII format is enabled for HEG service		
5	Ensure that the spatial and band subsetting is enabled for the collection for HEG service		
6	Upload the service form for the collection to earthdata/cmr using the Data Access Config GUI or with the FormUpload.rb Script		
7	Ensure that the form uploaded and the options are assigned in testbed pump		
8	Navigate to the Earthdata Search Client (EDSC). <a href="https://search.uat.earthdata.nasa.gov">https://search.uat.earthdata.nasa.gov</a>		
9	Search for granules for the test SMAP L3/L4 collections.		
10	Click on Perform Service. Verify that the service options for the collection are displayed correctly		
11	Enter a valid email address		
12	Select the supported ascii format		
13	Submit Service Request. Verify that the service request succeeded.		
14	Verify that the subsetting request, constraints, statistics, outputs are logged.	Some of the logging is in EcDIDaRqs.debug0.log and Egi logs.	

#	Action	Expected Result	Notes
15	Verify that an email is sent to the requested email address that the service request is complete with links to download the subsetted files		
16	Download the subsetted file.		
17	Verify that there is a link to download the request summary		
18	Verify that the output file has all the datasets in the subset request.		
19	Verify that the request summary link displays the request summary with all the subsetting parameter information, the input and the output granule list		
20	Download the files from the output links and verify that the output file format is the output format selected in the request.		

TEST DATA:

EXPECTED RESULTS:

### 6.7 EGI tests for SMAPL1L2 collections - formatting and exclusions (ECS-ECSTC-729)

#### DESCRIPTION:

S 40 1 [Nominal Granule Export] Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections. S 40 2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Login as cmshared on f5oml01 and set view to the latest autotest view or the baseline		
2	setenv MODE OPS		
3	cd /ecs/formal/COMMON/automation/autotest directory if runnign from the view or the appropriate autoetst folder 		
4	<i>Run each of the following SMAPL1L2 cucumber tests</i>		#comment

#	Action	Expected Result	Notes
5	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl1_format.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
6	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl2_smp_format.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
7	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl2_sma_format.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
8	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl2_smap_format.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed.  	

**TEST DATA:**

See Test Case 603.

**EXPECTED RESULTS:**

V 40 1 Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID). V 40 2 Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata. V 40 3 Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body. V 40 4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <https://api.echo.nasa.gov/ingest/schema/Granule.xsd> ). V 40 5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database V 40 6 Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255. V 40 7 Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule. V 40 8 Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent. V 40 9 Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed. V 40 10 Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule. V 40 11 Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs. V 40 12 Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs. V 40 13 Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs. V 40 14 Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection. V 40 15 Verify that the action performed in S-2 subclause n results in the

export of full granule metadata for every granule in the collection affected. V 40 16 Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.

### 6.8 EGI tests for SMAPL3 collections - NetCDF3 and ASCII format (ECS-ECSTC-730)

#### DESCRIPTION:

S 40 1 [Nominal Granule Export] Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections. S 40 2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.

#### PRECONDITIONS:

#### STEPS:

#	Action	Expected Result	Notes
1	Login as cmshared on f5oml01v and set view to latest AUTOTEST view or baseline		
2	setenv MODE OPS		
3	cd /ecs/formal/COMMON/automation/autotest directory or the folder where it is installed		
4	<i>Run each of the following SMAP L3 cucumber tests</i>		#comment
5	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_fta_format_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
6	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_ftp_format_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
7	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_sma_format_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	

#	Action	Expected Result	Notes
8	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_sma_parameter_subset_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
9	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_sma_parameter_subset_reprojection_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
10	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_sma_reprojection_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
11	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smap_format_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
12	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smap_parameter_subset_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
13	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smap_parameter_subset_reprojection_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
14	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smap_reprojection_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
15	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smp_format_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
16	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smp_parameter_subset_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	

#	Action	Expected Result	Notes
17	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smp_parameter_subset_reprojecti on_opendap.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed. 	
18	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl3_smp_reprojection_opendap.featur e	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed.   	

**TEST DATA:**

See Test Case 603.

**EXPECTED RESULTS:**

V 40 1 Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID). V 40 2 Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata. V 40 3 Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body. V 40 4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <https://api.echo.nasa.gov/ingest/schema/Granule.xsd> ). V 40 5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database V 40 6 Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255. V 40 7 Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule. V 40 8 Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent. V 40 9 Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed. V 40 10 Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule. V 40 11 Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs. V 40 12 Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs. V 40 13 Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs. V 40 14 Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection. V 40 15 Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected. V 40 16 Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.

## 6.9 EGI tests for SMAPL1L2 collections parameter subsetting, formatting and exclusions (ECS-ECSTC-731)

### DESCRIPTION:

S 40 1 [Nominal Granule Export] Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections. S 40 2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.

### PRECONDITIONS:

### STEPS:

#	Action	Expected Result	Notes
1	Login as cmshared on f5oml01v and set view to AUTOTEST		
2	setenv MODE OPS		
3	cd /ecs/formal/COMMON/automation/autotest directory		
4	<i>Run each of the following SMAPL1L2 cucumber tests</i>		#comment
5	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl1_parameter_subset.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed. 	
6	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl2_smp_parameter_subset.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed. 	
7	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl2_sma_parameter_subset.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed. 	
8	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl2_smap_parameter_subset.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed.   	

**TEST DATA:**

See Test Case 603.

**EXPECTED RESULTS:**

V 40 1 Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID). V 40 2 Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata. V 40 3 Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body. V 40 4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <https://api.echo.nasa.gov/ingest/schema/Granule.xsd> ). V 40 5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database V 40 6 Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255. V 40 7 Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule. V 40 8 Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent. V 40 9 Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed. V 40 10 Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule. V 40 11 Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs. V 40 12 Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs. V 40 13 Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs. V 40 14 Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection. V 40 15 Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected. V 40 16 Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.

**6.10 EGI tests for SMAPL4 collections - NetCDF-3 and ASCII formatting and exclusions (ECS-ECSTC-732)****DESCRIPTION:**

S 40 1 [Nominal Granule Export] Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections. S 40 2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.

**PRECONDITIONS:**

STEPS:

#	Action	Expected Result	Notes
1	Login as cmshared on f5oml01 and set view to latest AUTOTEST view or baseline		
2	setenv MODE OPS		
3	cd /ecs/formal/COMMON/automation/autotest directory or the folder where it is installed		
4	<i>Run each of the following SMAP L4 cucumber tests</i>		#comment
5	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_cmdl_format_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed. 	
6	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smgp_format_opendap.feature 		
7	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_cmdl_parameter_subset_opendap.feature  		
8	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smgp_parameter_subset_opendap.feature 		
9	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_cmdl_reprojection_opendap.feature  		
10	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smgp_reprojection_opendap.feature 		
11	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_cmdl_spatial_subset_opendap.feature  		
12	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smlm_format_opendap.feature 		
13	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smau_format_opendap.feature  		

#	Action	Expected Result	Notes
14	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smlm_parameter_subset_opendap.feature 		
15	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smau_parameter_subset_opendap.feature  		
16	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smlm_reprojection_opendap.feature 		
17	Run  cucumber --profile feature_log features/DataAccess/egi/smap/egi_smapl4_smau_reprojection_opendap.feature	Check /workingdata/cucumber/OPS/&lt;feature file log&gt; and Verify that the scenario is passed.	

**TEST DATA:**

See Test Case 603.

**EXPECTED RESULTS:**

V 40 1 Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID). V 40 2 Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata. V 40 3 Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body. V 40 4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <https://api.echo.nasa.gov/ingest/schema/Granule.xsd> ). V 40 5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database V 40 6 Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255. V 40 7 Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule. V 40 8 Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent. V 40 9 Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed. V 40 10 Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule. V 40 11 Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs. V 40 12 Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs. V 40 13 Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs. V 40 14 Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection. V 40 15 Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected. V 40 16 Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.

### 6.11 ESI tests - SMAPL3L4 collections - reformatting (ECS-ECSTC-733)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Login as cmshared on f5oml01v and set view to baseline or latest view		
2	setenv MODE OPS		
3	cd /ecs/formal/COMMON/automation/autotest directory		
4	<i>Run each of the following SMAPL3L4 cucumber tests</i>		#comment
5	Run  cucumber --profile feature_log features/DataAccess/esi/smap/esi_smapl3_formati.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed.	
6	Run  cucumber --profile feature_log features/DataAccess/esi/smap/esi_smapl4_format.feature	Check /workingdata/cucumber/OPS/&lt;featu re file log&gt; and Verify that the scenario is passed.	

TEST DATA:

EXPECTED RESULTS:

### 6.12 Configure OPENDAP collection - configure single collection with multiple formats (ECS-ECSTC-734)

DESCRIPTION:

Run the EcDIDaOpendapColFmtCfgStart utility to configure all formats for one collection. The utility will populate the AmDa datatables with the appropriate formats for the chosen collection.

PRECONDITIONS:

There is a granule in DEV07 mode of datatype MOD10A1.006 (MODIS Snow data set for Terra) that is public.

STEPS:

#	Action	Expected Result	Notes
1	Login to f5dpl01v host as cmshared.  cd /usr/ecs/DEV07/CUSTOM/utilities		OpenDAPMatrix is not used by the DAACs.
2	./EcDIDaOpendapColFmtCfgStart --mode DEV07 -s f4dbl03 -p 5432 --singleCollection MOD10A1.006 --timeout 600	-- Log file gets created in /usr/ecs/DEV07/CUSTOM/logs/Open dapMatrix.&lt;YYMMDDHHMMSS&gt;.log -- output file generated in /usr/ecs/DEV07/CUSTOM/data/DPL/ OpendapMatrix.&lt;YYMMDDHHMMSS&gt;.out containing the result of the opendap conversions. -- Result data set gets created in /usr/ecs/DEV07/CUSTOM/data/DPL/ OPENDAP/&lt;granuleid&gt;.dods -- Granule log file gets created in /usr/ecs/DEV07/CUSTOM/data/DPL/ OPENDAP_OUT/&lt;granuleid&gt;_log.out -- AIM database tables are populated with correct entries for MOD10A1.006 collection with the appropriate formats enabled	
3	cd /usr/ecs/DEV07/CUSTOM/data/DPL vi OpendapMatrix.&lt;YYMMDDHHMMSS&gt;.out	Output file should have: MOD10A1.006:96120 : ASCII : Success : 0 : 48836 ms MOD10A1.006:96120 : NetCDF-3 : Success : 0 : 5121 ms MOD10A1.006:96120 : NetCDF4- classic : Success : 0 : 10035 ms MOD10A1.006:96120 : DODS : Success : 0 : 3430 ms  	
4	cd /usr/ecs/DEV07/CUSTOM/data/DPL/OPENDAP_OUT vi &lt;granuleid&gt;_log.out	Log file should show:  ----- Format: ASCII ----- 200 ----- Format: NetCDF-3 ----- 200 ----- Format: NetCDF4-	

#	Action	Expected Result	Notes
		classic ----- 200 ----- Format: DODS ----- 200 	
5	Check that the amdagranuleformatconfig has the results of the run of opendap Matrix: select * from amdagranuleformatconfig where granuleid in (select granuleid from amgranule where collectionid = (select collectionid from amcollection where shortname = 'MOD10A1' and Versionid = 6));	Verify that the amgranuleformatconfig table has entries similar to:  kcockeri@f4dbl03:5432 ecs# select * from amdagranuleformatconfig where granuleid in (select granuleid from amgranule where collectionid = (select collectionid from amcollection where shortname = 'MOD10A1' and Versionid = 6));  granuleid   serviceid   formatid   success   starttime   duration  ----- --+-+-+-+-+-----  -----+-----  96120   20   153   t   2016- 03-15 15:11:49.976744   10035  > 96120   20   8   t   2016-03-15 15:11:53.447427   3430  96120   20   155   t   2016-03-15 15:11:34.570283   48836  96120   20   152   t   2016-03-15 15:11:39.837705   5121 (4 rows)  	
6	Check amdaformatsrvxref for the formats:  select * from amdaformatsrvxref where serviceid in (select serviceid from amdaservice where name = 'OPENDAP'); 	formatid   serviceid  -----+--- -----  155   20  152   20  8   20  > 153   20 (4 rows) 	
7	select * from amdadatasetconfig where collectionid in (select collectionid from amcollection where shortname = 'MOD10A1' and versionid = 6) and serviceid = (select serviceid from amdaservice where name = 'OPENDAP');	collectionid   serviceid   allowprojectionflag   allowformatflag   allowresamplingflag   allowinterpolationflag   allowbandflag   allowspatialflag   allowtemporalflag	

#	Action	Expected Result	Notes
		<pre> enabledflag   asynonlyflag &lt;br /&gt;----- -----+-----+----- +-----+-----+----- -----+-----+----- -----+-----+----- +-----&lt;br /&gt;      88540   20   N            Y            N   N            Y            N   N            N            N&lt;br /&gt;(1 row)&lt;br /&gt;&lt;br /&gt;Verify that the amdadatasetconfig has an entry for the collectionid and opendap serviceid and enabledflag = 'N'. Verify that the allowspatialflag, allowbandflag and enableformatFlag = 'Y'. Verify that all the other flags are set to 'N' </pre>	
8	Click on the DataAccess GUI, Service Configuration tab, open OPENDAP folder.	Verify that the collection is displayed under the OPENDAP folder.	
9	Click on the Service Configuration tab, when you click on the collection under OPENDAP=> Configure Service: OPENDAP for Collection. Select Enabled formats.	Verify that the successful opendap formats for the collection are displayed under Selected Formats	

TEST DATA:

EXPECTED RESULTS:

Data tables in the data access table set will be populated with appropriate formats for Opendap.

**6.13 NCR 8053130:Support parameter subsetting for the Opendap ESI service (ECS-ECSTC-735)**

DESCRIPTION:

Extract parameters from a granule (via script that invokes the OPeNDAP OLFS) and use the output to configure bands for a collection. Configure the OPeNDAP back-end server as a service in the mode. Perform an end-to-end test with the newly-configured dataset.

**PRECONDITIONS:**

OPeNDAP front-end and back-end servers must be active in the target mode; OPeNDAP BES service must be installed in the mode; EcDlDaOpendapParamConfig.rb and RubyCustomModules/ must be installed.

**STEPS:**

#	Action	Expected Result	Notes
1	Select a collection not natively in *.nc format.  In a chosen mode, execute this database query:  &gt; select DirectoryPath    OnlineFileName AS path from AmDataFile where onlinefilename like '%nc';  From the resulting list, choose a collection.  Make sure this collection has been exported using BMGT.	select DirectoryPath    OnlineFileName AS path from AmDataFile where onlinefilename like '%nc';  path  ----- -----  /NSIDC- 0484.001/1996.01.01/antarctica_ice_v elocity_900m.nc  /NSIDC- 0530.001/1999.01.01/nhtsd25e2_1999 0101_v01r01.nc  /TL2O3LN.006/2004.09.01/TES- Aura_L2-O3-Nadir_2004- 09_v006_Litev01.00.nc  /MIL3DAEN.004/2010.10.02/MISR_ AM1_CGAS_0_5_DEG_OCT_02_20 10_F08_0031.nc  /MIL3DAEN.004/2010.10.01/MISR_ AM1_CGAS_0_5_DEG_OCT_01_20 10_F08_0031.nc 	
2	Configure the OPENDAP_BES service.  Open the Data Access GUI; in the &quot;Service Configuration&quot; tab, click the &quot;Add New Service&quot; button.  Configure the new service as follows:  Service Name: OPENDAP_BES  Description: Service that bypasses theOLFS to access the Back-End Server directly.  Request Type: Both  Max Async Jobs: 10  Max Sync Jobs: 10  Timeout (Sec): 3600  Host: &lt;Host&gt;:&lt;Port&gt;:&lt;URL/Srv Path&gt; /usr/ecs/&lt;MODE&gt;/CUSTOM/bin/DPL/BesService  <input type="checkbox"/> Enable Spatial Subsetting  <input type="checkbox"/> Enable Temporal Subsetting  <input checked="" type="checkbox"/> Enable Band Subsetting	SELECT * FROM AmDaService WHERE name = 'OPENDAP_BES';  serviceid   name   description   allowtemporalflag   allowspatialflag   maxactiveasyncjobs   maxactivesyncjobs   allowbandflag  -----+-----+----- -----+-----+----- -----+-----+----- ---+-----  58   OPENDAP_BES   Service that	

#	Action	Expected Result	Notes
		bypasses the OLFS to access the Back-End Server directly.   N   N   10   10   Y                       />(1 row) 	
3	Configure data_objects/parameters for the collection.  &gt; EcDIDaOpendapParmCfg.rb --mode &lt;MODE&gt; --host-fqdn &lt;HOST&gt; --esi-port &lt;PORT&gt; --collection &lt;SNVI/ESDT&gt; -- output-dir &lt;WRITABLE_DIR&gt;  The output will be a URL; copy it; in a browser, log into the Data Access GUI; paste and visit the URL.  Use the query: SELECT procdagetalobjflddimbands('&lt;SHORTNAME&gt;', &lt;VERSIONID&gt;, 'OPENDAP_BES'); to verify that at least two bands exist for the collection. 	http://f5dpl01.edn.ecs.nasa.gov:22500/ DataAccessGui_DEV02/config/import ?configLevel=Dataset_Service&amp;d ataType=Data_Objects&amp;serviceI d=OPENDAP_BES&amp;collectionId =AE_Rain.002&amp;filePath=/home/ mbrown/Data/Test/AE_Rain.002.json   { status : &quot;Success&quot;, collections_processed : [ &quot;AE_Rain.002&quot; ], results : &quot;import DATASET_SERVICE DATA_OBJECTS to/from: /home/mbrown/Data/Test/AE_Rain.00 2.json&quot;, messages : [ ] }    select procdagetalobjflddimbands('NSIDC- 0484', 1, 'OPENDAP_BES'); procdagetalobjflddimbands ----- ----- (58,199,/,UNKNOWN,2424,Data_cita tion,,,,,Y,N,N,,,) (58,199,/,UNKNOWN,2412,Ellipsoid, ,,,,Y,N,N,,,) (58,199,/,UNKNOWN,2414,Latidue _of_origin,,,,,Y,N,N,,,) (58,199,/,UNKNOWN,2413,Longitud e_of_origin,,,,,Y,N,N,,,)	

#	Action	Expected Result	Notes
		(58,199,/,UNKNOWN,2425,More_inform ation,,,,,Y,N,N,,,) )	
4	Return to the &quot;Collection Configuration&quot; tab of the Data Access GUI.  Right-click on the chosen collection and &quot;Configure New Service for Collection&quot; (Choose OPENDAP_BES.)  (If GDAL is enabled for the collection, &quot;Unmap All Services,&quot; then remap the OPENDAP_BES service.)	SELECT esdt(shortname,versionid), s.name FROM AmDaDatasetConfig dc  INNER JOIN AmDaService s ON s.serviceid = dc.serviceid  INNER JOIN AmCollection c ON c.collectionid = dc.collectionid WHERE s.name = 'OPENDAP_BES';  esdt   name  -----+----- -----  SPL2SMA.003   OPENDAP_BES  NSIDC- 0484.001   OPENDAP_BES  VNP10.001   OPENDAP_BES  /> AE_Ocean.002   OPENDAP_BES  MOD10CM.005   OPENDAP_BES (5 rows) 	
5	Perform an EGI request using the inventory drilldown.  Choose the OPENDAP_BES service.  Choose one or two of the available bands.  View the outputs to make sure only the selected bands have been included in the output.	http://f5eil01:22500/egi_DEV02/reque st?FILE_IDS=3000595354&amp;SUB AGENT_ID=OPENDAP_BES&amp; DATASET_ID=MODIS%2FTerra%2 0Snow%20Cover%20Monthly%20L3 %20Global%200.05Deg%20CMG%2 0V005&amp;FORMAT=ASCII&amp; EMAIL=michael.a.brown- 2@nasa.gov&amp;CLIENT=ESI&am p;REQUEST_MODE=sync&amp;SU BSET_DATA_LAYERS=%2F%3AS now_Cover_Monthly_CMG,%2F%3A Snow_Spatial_QA&amp;	
6	Upload forms for the new collection:  Use the GUI, or FormUpload.rb:  &gt; FormUpload.rb --mode &lt;MODE&gt; -- host-fqdn &lt;HOST&gt; --esi-port &lt;PORT&gt; --output-dir &lt;WRITABLE_DIR&gt; --associate-service &lt;PROVIDER&gt; EOSDIS	FormUpload.rb --mode DEV02 --host- fqdn f5dpl01.edn.ecs.nasa.gov --esi- port 22500 --use-token /home/mbrown/Data/Forms/brownm0	



**EXPECTED RESULTS:**

User should be able to perform a parameter subsetting request using the OPeNDAP back-end server via Earthdata Search Client.

## 7 WUFTP REGRESSION TEST

### 7.1 WuFTP Regression Test (ECS-ECSTC-711)

**DESCRIPTION:**

Perform the following tests used for previous WU-FTPD release testing for compatibility with ECS release 8.x.x custom code and databases:

1. Download an uncompressed .xml file with no checksum using (.zip, .gz, .Z), and verify that the download succeeds without a problem.
2. Download a .hdf file with no checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem. NOTE:- (you might need to change the ingest checksumming percentage to 0 for the provider that you use for ingesting the granule. You should be able to use a PDR from /sotestdata/DROP\_723/sample\_full\_size/ACRIM/L2/ACR3L2DM.001 for this).
3. Download a .hdf file with an ECS checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem. (NOTE:- You should be able to use the PDR from /sotestdata/DROP\_801/DP\_81\_05/Criteria/100/MOD29P1D.005 for this).
4. Download a .hdf file with a CKSUM using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem. (Note:- You should be able to use a PDR from /sotestdata/DROP\_723/sample\_full\_size/MODIS/L2/MOD10\_L2.005 for this).
5. Download a .hdf file with MD5 checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem. (Notes:- You should be able to use a PDR from /sotestdata/DROP\_723/sample\_full\_size/TES/L3/TL3ATD.003 for this).
6. Download a file with an ECS checksum, but with files that have corrupt checksums using (.zip, .gz, .Z), and verify that the download succeeds without a problem. You should be able to use the PDR from /sotestdata/DROP\_801/DP\_81\_05/Criteria/100/MCD12Q1.005 for this if you change the checksum in the database after ingesting the granule.
7. Download a folder using (.zip, tar, tar.gz, tar.Z), and verify that the download succeeds without a problem.
8. Run the wuftp rollup script to verify that Wu-ftp includes and properly sets a URS authentication userid field in its "xferlog" log file.
  - ftp one LANCE file using a login registered in URS
  - ftp one non-LANCE file using a Linux login not registered in URS.
  - verify that the entry in /var/log/xferlog related to the LANCE file contains the URS userid in the last field and that the entry related to the non-LANCE file contains "-" in the last field. ftp one non-LANCE file using a Linux login not registered in URS
9. Submit an FtpPull order for public granules whose files are not retrievable by anonymous/ftp users (configurable via /etc/ftpd/ftppass noretrieve directives).
  - i. Submit another order for hidden granules.
  - ii. Verify that the orders (non-compressed and compressed) are only retrievable with earth data login.

- iii. Submit an FtpPull order for granules, some granules whose files are marked as noretrieve, some whose aren't. Verify that the files that are marked as noretrieve can only be retrieved with earth data login, and other can also retrieved by anonymous/ftp user, and that the compressed files can only be retrieved with earth data login.
- iv. Verify that the compressed file download is reported to EMS (This is not part of this ncr, but we want to know this type of download is reported).

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Download an uncompressed .xml file with no checksum using (.zip, .gz, .Z), and verify that the download succeeds without a problem. 		
2	Download a .hdf file with no checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem.   NOTE:- (you might need to change the ingest checksumming percentage to 0 for the provider that you use for ingesting the granule. You should be able to use a PDR from /sotestdata/DROP_723/sample_full_size/ACRIM/L2/ACR3L2DM.001 for this). 		
3	Download a .hdf file with an ECS checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem.  (NOTE:- You should be able to use the PDR from /sotestdata/DROP_801/DP_81_05/Criteria/100/MOD29PID.005 for this). 		
4	Download a .hdf file with a CKSUM using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem. (Note:- You should be able to use a PDR from /sotestdata/DROP_723/sample_full_size/MODIS/L2/MOD10_L2.005 for this). 		
5	Download a .hdf file with MD5 checksum using (uncompressed, .zip, .gz, .Z), and verify that the download succeeds without a problem.  (Notes:- You should be able to use a PDR from /sotestdata/DROP_723/sample_full_size/TES/L3/TL3ATD.003 for this). 		
6	Download a file with an ECS checksum, but with files that have corrupt checksums using (.zip, .gz, .Z), and verify that the download succeeds		

#	Action	Expected Result	Notes
	without a problem. You should be able to use the PDR from /sotestdata/DROP_801/DP_81_05/Criteria/100/MCD12Q1.005 for this if you change the checksum in the database after ingesting the granule. 		
7	Download a folder using (.zip, tar, tar.gz, tar.Z), and verify that the download succeeds without a problem.  		
8	Run the wuftp rolup script to verify that Wu-ftp includes and properly sets a URS authentication userid field in its &quot;xferlog&quot; log file.   i) ftp one LANCE file using a login registered in URS ii) ftp one non-LANCE file using a Linux login not registered in URS.  iii) verify that the entry in /var/log/xferlog related to the LANCE file contains the URS userid in the last field and that the entry related to the non-LANCE file contains &quot;-&quot; in the last field. ftp one non-LANCE file using a Linux login not registered in URS 		
9	Submit an FtpPull order for public granules whose files are not retrievable by anonymous/ftp users (configurable via /etc/ftpd/ftpaccess noretrieve directives).   i) Submit another order for hidden granules.  ii) Verify that the orders (non-compressed and compressed) are only retrievable with earth data login. iii) Submit an FtpPull order for granules, some granules whose files are marked as noretrieve, some whose aren't. Verify that the files that are marked as noretrieve can only be retrieved with earth data login, and other can also retrieved by anonymous/ftp user, and that the compressed files can only be retrieved with earth data login. iv) Verify that the compressed file download is reported to EMS (This is not part of this ncr, but we want to know this type of download is reported).  		

TEST DATA:

EXPECTED RESULTS:

## 8 IMAGEMAGICK REGRESSION TESTS

### 8.1 Make sure imagemagick can annotate a PNG file with text (ECS-ECSTC-855)

DESCRIPTION:

In order to add legends to KML outputs we need to annotate the PNG files with text. This test should allow the tester to begin with a PNG file which is just data and add text to it using imagemagick's convert

**PRECONDITIONS:**

User has PNG file they want to add text to. One of these files can be ordered in DEV06 by request PNG output format. David should have a DEV06 URL that allows this to happen.

**STEPS:**

#	Action	Expected Result	Notes
1	Generate a PNG file using DataAccess (see David Auty to add specific steps here)	The PNG file is viewable in a browser	
2	run <code>&amp;quot;convert ~akhan/npp_cell_tb_v_aft.tif.png -draw &amp;quot;gravity south fill white text 0,100 'Test Legend' &amp;quot; out.png</code> Note: Substitute your image for the first parameter to convert, and you may have to play with the other options to get it to work.	The new PNG file out.png is viewable in the browser and you can see the text <code>&amp;quot;Test Legend&amp;quot;</code> in it.	

**TEST DATA:**

**EXPECTED RESULTS:**

output PNG file has added text in it

## 9 PYTHON REGRESSION TESTS

### 9.1 SMAP L1/L2 reformatting to HDF-EOS5 format (ECS-ECSTC-707)

**DESCRIPTION:**

Python scripts are used as part of Data Access processing. One particular use case for exercising the python interpreter is running a SMAP L1/L2 data access service request that converts the original h5 data to HDF-EOS5 format. This can be tested in the following steps (See Steps tab):

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Find a collection (such as SPL2SMA.003) which is enabled for both SMAPL1L2 processing in Data Access.		
2	Enter the following EGI command on the x5dpl01 server assuming that the requested granule is in the datapool repository: <code>#wget -O - --server-response --post-data 'FILE_IDS=403112&amp;FORMAT=HDF-</code>		

#	Action	Expected Result	Notes
	EOS5&amp;EMAIL=kenneth.l.cockerill@nasa.gov&amp;CLIENT=ESI&amp;SUBAGENT_ID=SMAPL1L2&amp;PROJECTION=GEOGRAPHIC&amp;DATASET_ID=SMAP L1C Radiometer Half-Orbit 36 km EASE-Grid Brightness TemperaturesV003&amp;REQUEST_MODE=async&amp;SUBSET_DATA_LAYERS=/SPL1CTB/South_Polar_Projection/cell_boresight_incidence_fore,/SPL1CTB/South_Polar_Projection/cell_lat,/SPL1CTB/South_Polar_Projection/cell_lon,/SPL1CTB/North_Polar_Projection/cell_boresight_incidence_fore,/SPL1CTB/North_Polar_Projection/cell_lat,/SPL1CTB/North_Polar_Projection/cell_lon,/SPL1CTB/Global_Projection/cell_boresight_incidence_fore,/SPL1CTB/Global_Projection/cell_lat,/SPL1CTB/Global_Projection/cell_lon&amp;INTERPOLATION=NN&amp;' http://f5eil01:22500/egi/request		
3	Verify that the result file contains the hdf-eos5 formatted granule.		

TEST DATA:

EXPECTED RESULTS:

## 9.2 Run test script to test the OGC Web Service python library. (ECS-ECSTC-708)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	 The OGC Web Service python library is used to make Web Coverage Service calls to support the Service Gateway functionality. The following script can be exercised on any external host that will produce a canned tiff file from a central repository using WCS calls:    #!/tools/python/bin/python # Very simple script demonstrating how to interact with a THREDDS based WCS. # --- # # The GetCapabilities and DescribeCoverage requests for this dataset are:  # http://cida.usgs.gov/thredds/wcs/prism?service=WCS&amp;version=1.0.0&amp;request=GetCapabilities # http://cida.usgs.gov/thredds/wcs/prism?service=WCS&amp;version=1.0.0&a		Save the script in a file with the .py extension and run it.

#	Action	Expected Result	Notes
	<pre>mp;request=DescribeCoverage&lt;br /&gt;#&lt;br /&gt;# The equivalent GetCoverage request that is equivalent of hte example is:&lt;br /&gt;# http://cida.usgs.gov/thredds/wcs/prism?request=GetCoverage&amp;amp;version=1.0.0&amp;amp;service=WCS&amp;amp;format=GeoTIFF&amp;amp;coverage=tmx&amp;amp;time=1895-01-01T00:00:00Z&amp;amp;bbox=-90,40,-89,41&lt;br /&gt;# ---&lt;br /&gt;# &lt;br /&gt;# Example to find the equivalent information using OWSLib:&lt;br /&gt;# &lt;br /&gt;from __future__ import absolute_import&lt;br /&gt;from __future__ import print_function&lt;br /&gt;from owslib.wcs import WebCoverageService&lt;br /&gt;wcs=WebCoverageService('http://cida.usgs.gov/thredds/wcs/prism',version='1.0.0')&lt;br /&gt;# Take a look at the contents (coverages) of the wcs.&lt;br /&gt;print(wcs.contents)&lt;br /&gt;tmax=wcs['tmx']&lt;br /&gt;# Take a look at the attributes of the coverage&lt;br /&gt;dir(tmax)&lt;br /&gt;print(tmax.boundingBoxWGS84)&lt;br /&gt;print(tmax.timepositions)&lt;br /&gt;print(tmax.supportedFormats)&lt;br /&gt;# mock up a simple GetCoverage request.&lt;br /&gt;output=wcs.getCoverage(identifier='tmx',time=['1895-01-01T00:00:00Z'],bbox=(-90,40,-89,41),format='GeoTIFF')&lt;br /&gt;# Write the file out to disk.&lt;br /&gt;f=open('foo.tif','wb')&lt;br /&gt;f.write(output.read())&lt;br /&gt;f.close()&lt;br /&gt;&lt;br /&gt;</pre>		

TEST DATA:

EXPECTED RESULTS:

### 9.3 Run test script to exercise the python numerical analysis library (numPy) and hdf 5 library (h5py) (ECS-ECSTC-709)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<pre>To exercise the python numerical analysis library (NumPy) and hdf 5 library (h5py), the following script can be run on the x5dpl01 host to get a listing of the contents of an hdf5 file:&lt;br /&gt;&lt;br /&gt;#!/tools/python/bin/python&lt;br /&gt;import numpy as np&lt;br /&gt;import h5py&lt;br /&gt;file=&amp;quot;GLAH13_633_2111_001_1261_0_01_0001.H5&amp;quot;&lt;br /&gt;</pre>		Have a hdf5 file in the local directory. Save the script in a file with the extension .py and run it.

#	Action	Expected Result	Notes
	<pre> /&gt;with h5py.File(file,'r') as hf:&lt;br /&gt;#with h5py.File('SMAP_L2_SM_A_01250_D_20150427T010648_R13080_001.h5 ','r') as hf:&lt;br /&gt;  print("&amp;quot;List of arrays in this file: \n&amp;quot;, hf.keys())&lt;br /&gt;  for array in hf.keys():&lt;br /&gt;    gp = hf.get(array)&lt;br /&gt; print ('\nlist of items in', array, gp.items()) &lt;br /&gt;    data = hf.get(array)&lt;br /&gt;    np_data = np.array(data)&lt;br /&gt;    print('Shape of the array', array, np_data.shape) </pre>		

TEST DATA:

EXPECTED RESULTS:

### 9.4 Run test script to exercise the python shape file library (ECS-ECSTC-710)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<pre> To exercise the python shape file library (pyShp), run the following script on x5dp101 to write a test shape file to the current directory:&lt;br /&gt;&lt;br /&gt;&lt;br /&gt;#!/tools/python/bin/python&lt;br /&gt;# -*- coding: utf-8 -*-&lt;br /&gt;&amp;quot;&amp;quot;&amp;quot;&lt;br /&gt;Script to create a sample Shape File for simulated Satelite Science Data,&lt;br /&gt;low-level, point data.&lt;br /&gt;&lt;br /&gt;Created: 2016-11-06&lt;br /&gt;&lt;br /&gt;@author: dauty&lt;br /&gt;&amp;quot;&amp;quot;&amp;quot;&lt;br /&gt;&lt;br /&gt;import shapefile&lt;br /&gt;&lt;br /&gt;shp_writer = shapefile.Writer(shapeType=21) &lt;br /&gt; # 21 =&amp;gt; shapefile with point data and with Measured value (PointM)&lt;br /&gt;shp_writer.autoBalance = 1&lt;br /&gt;&lt;br /&gt;for lon in range(-150,151,10):&lt;br /&gt;  for lat in range (-80,81,10):&lt;br /&gt;    shp_writer.point(lon,lat,m=(lon*1000+lat))&lt;br /&gt;&lt;br /&gt;target_name = shp_writer.save()&lt;br /&gt;&lt;br /&gt;if __name__ == &amp;quot;__main__&amp;quot;:&lt;br /&gt; print("&amp;quot;Hello from Shape_Pts&amp;quot;) </pre>		

TEST DATA:

EXPECTED RESULTS:

## 10 POSTGRES REGRESSION TEST

### 10.1 AIM

10.1.1 Test Case 1 - Install ESDTs with a variety of spatial search types (ECS-ECSTC-862)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Install ESDTs with a variety of spatial search types]</i>		#comment
2	Verify that the descriptor ODL files for the ESDT being installed are moved from the installation source directory into the configured descriptor directory.		
3	Prepare descriptor files for several (at least 5) ESDTs covering a variety of ESDT types described below, such that the descriptors conform to the ECS data model and the XML schema validation rules, guaranteeing successful ESDT installation. Examples of such can be found in /sotestdata/DROP_721/DS_7E_01/Criteria/2030, and are listed as follows:   1. ESDT descriptor with a spatial search attribute of Rectangle  MOD09CMG.005  MYD09CMG.005   2. ESDT descriptor with a spatial search attribute of GPolygon  MOD14.005  MYD14.005   3. ESDT descriptor with a spatial search attribute of NotSupported  AEPOE7W.001  PM1GBAD1.001   4. ESDT descriptor with a spatial search attribute of Orbit  AE_Land.002  AE_Rain.002   5. ESDT descriptor with a spatial search attribute of Point  g3bssp.007  g3bt.007 		
4	Ensure collections are not already installed.		
5	Ensure EcDsAmESDTMaint.properties has log.debug.level=XVERBOSE (for verification).		
6	Copy the descriptor files to the source directory (specified in the GUI).		
7	Verify that the descriptors are listed in the GUI.		

#	Action	Expected Result	Notes
8	Get the current time to help search the logs.		
9	From the ESDT Maintenance GUI, select the ESDTs in the source directory and initiate installation.		
10	Upon completion of the ESDT installation, verify that the GUI displays a message indicating number of ESDTs successfully installed.		
11	Verify that the metadata elements provided in the descriptor files were validated by the XML Services library against the ESDR common schema, which conforms to the ECS data model. If you have the ESDT Maintenance GUI debug level set to XVERBOSE, you should get a message something like the following in the ESDT Maintenance GUI log:   EcsFileProcessor.validateXmlFile DsESDTMoMOD09A1.005.xml  Other than that, it might require looking at the actual code.		
12	Verify by inspection that the element types and lengths in the descriptors match those stated in the ESDT common schema which in turn matches the ECS data model. The ESDT common schema is under: /usr/ecs/MODE/CUSTOM/WWW/DSS/ESDTMaint/WEB-INF/lib/EcDsAmEsdtCommonSchemas.jar. 		Are the descriptors supposed to be compared to XML schema? Should verify that the descriptors in the configured descriptors directory diff with the originals. Jon Pals:  &quot; For Step 5, Yes, it is asking that you compare what is in the ODL ESDT descriptor file against the XML schema files. For this, I suggest doing some random sampling. Just take a few attributes from the ESDT descriptor file and find how those attributes are defined in the XML schema files.&quot; Need an automated way to compare ODL descriptor files to XML schemas. 
13	Verify that an MCF file is generated for each ESDT being installed, and stored into the configured MCF directory location ( locations pulled from /custom/ecs/TS3/CUSTOM/cfg/EcDsAmESDTMaint.properties ) 		20130731t124819::f5dpl01v::cmshared::stornext/smallfiles/TS3/mcf  \$ find . -

#	Action	Expected Result	Notes
	/> descriptor.target.dir = /stornext/smallfiles/TS3/descriptor descriptor.source.dir = /usr/ecs/TS3/CUSTOM/data/ESS mcf.target.dir = /stornext/smallfiles/TS3/mcf archive.metadata.dir = /stornext/smallfiles/TS3/metadata 		type f -mtime -1   sed 's/..([^\#]*)#([^\.]*).*\1.\2/' MOD09CMG.005 MOD14.005 MYD14.005 MYD09CMG.005 AE_Land.002 AE_Rain.002 g3bssp.007 PM1GBAD1.001 g3bt.007 AEPOE7W.001 
14	Verify that the generated MCF files are correct. 		( Define &quot;correct&quot;; The MCFs should be a pretty close match with the INVENTORYMETADAT A section and the ARCHIVEDMETADATA section in the ESDT descriptor file. The place where there should be a difference in in the AdditionalAttributes group. The ProductSpecificMetadata group in the ESDT descriptor file is replaced with a generic AdditionalAttributes group in the MCF. )  Within the INVENTORYMETADAT A group, the ProductSpecificMetadata group from the descriptor file should be replaced with this AdditionalAttributes block in the MCF file: 

#	Action	Expected Result	Notes
			/>  GROUP = AdditionalAttributes  OBJECT = AdditionalAttributesContain er  Data_Location = &quot;NONE&quot;<br > Mandatory = &quot;FALSE&quot;  CLASS = &quot;M&quot;  OBJECT = AdditionalAttributeName<b r /> Mandatory = &quot;FALSE&quot;  CLASS = &quot;M&quot;  Data_Location = &quot;PGE&quot;  NUM_VAL = 1  TYPE = &quot;STRING&quot;<br > END_OBJECT = AdditionalAttributeName<b r /> GROUP = InformationContent  CLASS = &quot;M&quot;  OBJECT = ParameterValue  Mandatory = &quot;FALSE&quot;  Data_Location = &quot;PGE&quot;  NUM_VAL = 1  TYPE = &quot;STRING&quot;<br > END_OBJECT = ParameterValue  END_GROUP =

#	Action	Expected Result	Notes
			InformationContent  END_OBJECT = AdditionalAttributesContain er  END_GROUP = AdditionalAttributes<br />MODIS MCF files should include the descriptor file's ARCHIVEDMETADATA group.<br />AE_Land#002.MCF, AE_Rain#002.MCF, and PM1GBAD1#001.MCF do not have this group 
15	Verify that an XML schema file (*.xsd) is generated for each ESDT being installed, and stored into the descriptors directory.		
16	Verify that the generated XML schema files contain validation rules based upon the contents of the Inventory section of the Descriptor file and the default element rules. 		The &quot;Inventory section&quot; probably starts here: GROUP = INVENTORYMETADATA  Jon Pals:   For Step 9, take a look at the 'Mandatory =' lines in the INVENTORYMETADATA section of the ESDT descriptor file and compare that with the ESDT's .xsd file in the /stornext/smallfiles/&lt;MO DE&gt;/descriptor directory. The INVENTORYMETADATA attributes with 'Mandatory = &quot;FALSE&quot;' lines should be listed in the .xsd file as having 'minOccurs=&quot;0&quot;' . The INVENTORYMETADATA attributes with 'Mandatory =

#	Action	Expected Result	Notes
			&quot;TRUE&quot;' lines should be listed in the .xsd file as having no 'minOccurs' or having a 'minOccurs=&quot;1&quot;'. . 
17	Verify that basic collection identification information and proper spatial search type have been added into the AIM Inventory database. 		
18	Verify that the PSA associations, if any, have been added into the AIM Inventory database. 		<pre> select aa.additionalattributename&lt;br /&gt; from amcollection c&lt;br /&gt; join dsmdcollectionaddnlattrbsx ref ax&lt;br /&gt; on c.collectionid = ax.collectionid&lt;br /&gt; join dsmdadditionalattributes aa&lt;br /&gt; on ax.attributeid = aa.attributeid&lt;br /&gt; where c.shortname = '\${SHORTNAME}'&lt;br /&gt; and c.versionid = '\${VERSIONID}';&lt;br /&gt;                     </pre>
19	Verify that an insert event, and event qualifiers are inserted into the Spatial Subscription Server database for each ESDT being installed. 		Event qualifiers appear restricted to those defined in ecnbeventmetadataattrdef.attributename.                      />EVENTPARMS: Extracted from descriptor file                       Online EVENTPARMS known by SSS will be added.                       />-- What qualifiers does SSS know about?                      />select *                      from ecnbeventmetadataattrdef                      where attributename in (                      \${EVENTPARMS});                      />                      />-- What qualifiers

#	Action	Expected Result	Notes
			<p>were added? <code>select attributename from ecnbeventattrxref where esdt_id = '\${SHORTNAME}' and versionid = \$VERSIONID; 'RangeBeginningTime' and 'RangeEndingTime' appear to get combined into 'GranuleTimeRange'. &lt;br&gt; 'RangeBeginningDate' and 'RangeEndingDate' appear to get combined into 'GranuleDateRange'. &lt;br&gt;g3bssp.007 and g3bt.007 both lack date and time ranges, but both have GranuleTimeRange and GranuleDateRange in ecnbeventattrxref. &lt;br&gt;example: select * from ecnbeventdefinition where (esdt_id = 'MOD14' and versionid = 5) or (esdt_id = 'MYD14' and versionid = 5) or (esdt_id = 'AEPOE7W' and versionid = 1) or (esdt_id = 'PM1GBAD1' and versionid = 1) or (esdt_id = 'AE_Land' and versionid = 2) or (esdt_id = 'AE_Rain' and versionid = 2) or (esdt_id = 'MOD09CMG' and versionid = 5)</code></p>

#	Action	Expected Result	Notes
			<pre>(esdt_id = 'MYD09CMG' and versionid = 5)&lt;br /&gt;or (esdt_id = 'g3bssp' and versionid = 7)&lt;br /&gt;or (esdt_id = 'g3bt' and versionid = 7);&lt;br /&gt;&lt;br /&gt; eventtype   esdt_id   versionid   eventid&lt;br /&gt;----- -----+-----+----- ----+-----&lt;br /&gt; INSERT   AE_Land   2  &lt;br /&gt; INSERT   AE_Rain   2  &lt;br /&gt; INSERT   AEPOE7W   1  &lt;br /&gt; INSERT   MOD09CMG  5  &lt;br /&gt; INSERT   MOD14   5  &lt;br /&gt; INSERT   MYD09CMG   5  &lt;br /&gt; INSERT   MYD14   5  &lt;br /&gt; INSERT   PM1GBAD1   1  &lt;br /&gt; INSERT   g3bssp   7  &lt;br /&gt; INSERT   g3bt   7  &lt;br /&gt;(10 rows)&lt;br /&gt;</pre>
20	Verify that the ESDT Maintenance GUI logs the processing activities including time of request, action requested, ESDT ShortName, Version ID, descriptor file name, and the result of the operation, in a configured log file.		<p>The ESDTMaintenanceGUI.* logs include a timestamp in each log entry, but it's not clear when a request actually began.&lt;br /&gt;&lt;br /&gt;These lines from the debug log may be the beginning of a request:&lt;br /&gt;&lt;br /&gt;07.31.2013 12:02:09.563 : Thread ID</p>

#	Action	Expected Result	Notes
			<p>[41] : VERBOSE : /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTAmAE_Land. 002.desc dataModelType is: ECS&lt;br /&gt;07.31.2013 12:02:09.563 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTAmAE _Land.002.desc&lt;br /&gt;The action is not stated explicitly. In the debug log, one entry includes the word &amp;quot;installing&amp;quot;;&lt;br<br &gt;&lt;br="" &gt;07.31.2013<br=""/>12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br<br &gt;the="" and<br="" shortname=""/>Version ID are part of the descriptor filename. They appear in several other log entries, but it's not clear why:&lt;br /&gt;&lt;br /&gt;07.31.2013 12:02:09.651 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObj ct4CallableStmt() Param Index:10 String Value:AE_Land.002 Type:12&lt;br /&gt;&lt;br<br &gt;07.31.2013="" 12:02:09.702="" :<br=""/>Thread ID [41] : VERBOSE : AbstractServiceImpl.setObj ct4CallableStmt() Param Index:1 String</p>

#	Action	Expected Result	Notes
			<p>Value:AE_Land Type:1&lt;br /&gt;07.31.2013 12:02:09.702 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:2 String Value:2 Type:4&lt;br /&gt;The debug log reports the successful installation like this:&lt;br /&gt;&lt;br /&gt;07.31.2013 12:02:10.358 : Thread ID [41] : INFORMATION : InstallESDTPage:Successfully installed ESDT DsESDTAmAE_Land.002.desc&lt;br /&gt;The ESDTMaintenanceGUI.ops0.log contains very little information.&lt;br /&gt;Example:&lt;br /&gt;From /usr/ecs/TS3/CUSTOM/logs/ESDTMaintenanceGUI.debug0.log:&lt;br /&gt;&lt;br /&gt;time of the installation request:&lt;br /&gt;07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br /&gt;&lt;br /&gt;action requested: (this shows the action taken)&lt;br /&gt;07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br /&gt;</p>

#	Action	Expected Result	Notes
			/> ShortName: 07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:1 String Value:MOD09CMG Type:1  Version ID: 07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:4 String Value:5 Type:4  descriptor file name: 07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc  result of the operation (successful): 07.31.2013 12:02:12.215 : Thread ID [41] : INFORMATION : InstallESDTPage:Successfully installed ESDT DsESDTMoMOD09CMG.005.desc
21	<i>Verification Instructions</i>		#comment
22	Verify that there are at least 5 descriptors that match the setup requirements in directory /sotestdata/DROP_721/DS_7E_01/Criteria/2030/ Verify that the value of the OBJECT = SpatialSearchType group is “rectangle”, “GPolygon”, “not supported”, “point”, or “orbit” accordingly:  ESDT   SpatialSearchType   Archiving DAAC   ACR3L2SC.001   NotSupported   ASDC   		

#	Action	Expected Result	Notes
	AEPOE7W.001   NotSupported   NSIDC    AE_Land.002   Orbit   NSIDC     AE_Rain.002   Orbit   NSIDC     MB2LME.198   Orbit   ASDC     MIANRCCH.198   NotSupported   ASDC     MIL2ASOS.198   Orbit   ASDC    MIL3DAE.198   Rectangle   ASDC    MOD09CMG.005   Rectangle   LP DAAC    MOD14.005   GPolygon   LP DAAC     MOP02.003   Rectangle   ASDC     MYD09CMG.005   Rectangle   LP_DAAC     MYD14.005   GPolygon   LP DAAC     PM1GBAD1.001   NotSupported   NSIDC    TL3ATD.002   Rectangle   ASDC     g3assp.004   Point   ASDC     g3atb.004   Point   ASDC		
23	Return to the ESDT List Page. Select the check box next to the descriptor names set up in setup step 1. Select the “Install new ESDTs/Update existing ESDTs” button. Note the wall clock time of the start of the installation for later verification in the logs. Note the number of descriptors selected.		Login to the ESDT Maintenance GUI. Ensure no test ESDT is currently installed. If it is, delete its granules, clean up orphans, remove from the data pool (DPL GUI), and delete it (ESDT GUI). Click the “Install new ESDTs/Update existing ESDTs” button. Note the ESDT source directory. Copy the test ESDTs to the ESDT source directory.  Login to the ESDT Maintenance GUI. Click the “Install new ESDTs/Update existing ESDTs” button. Select the test ESDTs. Click the “Proceed with installation/update”

#	Action	Expected Result	Notes
			button. Wait for the screen to refresh.
24	Verify that the GUI displays the number of descriptors installed is equal to the number of descriptors selected.		
25	Verify that the metadata elements provided in the descriptor files were validated by the XML validation utility by viewing its application log in the mode. Verify for each descriptor that no failures occurred. 		
26	Select one descriptor from each part of setup step 1. 		
27	Go to the directory where the generic descriptor schema is installed and view it.		
28	Verify by inspection that rules for elements in the generic schema are executed correctly for corresponding elements in the descriptor. A similar comparison should be performed between the elements of the generic descriptor schema and the ECS data model. 		What is the XML validation utility used, and where is its log? Where is &quot;the directory where the generic descriptor schema is installed&quot;? 
29	Verify by inspection that rules for elements in the generic schema are executed correctly for corresponding elements in the descriptor. 		
30	Verify each element in the descriptor file that has a matching element in the generic schema gets added to the descriptor's .xsd file. 		&quot;A similar comparison should be performed between the elements of the generic descriptor schema and the ECS data model.&quot;  What does this mean? /usr/ecs/TS3/CUSTOM/logs/ESDTMaintenanceGUI.ddebug0.log shows the descriptor .xml and .xsd files are generated but not whether they are validated:  07.31.2013 12:02:12.021 : Thread ID [41] : xmlsvcs.schemagen.EcAmDescSchemaGenImpl : VERBOSE : START

#	Action	Expected Result	Notes
			generateDescriptorSchemaFromOdlDescriptor odlDescriptor: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc outputSchema: /stornext/smallfiles/TS3/descriptor/DsESDTMoMOD09CMG.005.xsd  >07.31.2013 12:02:12.021 : Thread ID [41] : xmlsvcs.schemagen.EcAmDescSchemaGenImpl : VERBOSE : START generateDescriptorXmlFile inputODLFile: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc outputXMLFile: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc.xml  >07.31.2013 12:02:12.050 : Thread ID [41] : xmlsvcs.schemagen.EcAmDescSchemaGenImpl : VERBOSE : END generateDescriptorXmlFile inputODLFile: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc outputXMLFile: /usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoMOD09CMG.005.desc.xml  >07.31.2013 12:02:12.051 :

#	Action	Expected Result	Notes
			<p>Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : START generateDescriptorSchemaF romXmlDescriptor xmlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br &gt;07.31.2013 12:02:12.051 : Thread ID [41] : xmlsvcs.schemagen.EcAmS chemaGenerator : INFORMATION : createESDTSchema START: XML descriptor - /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml, Schema - /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd&lt;br &gt;07.31.2013 12:02:12.154 : Thread ID [41] : xmlsvcs.schemagen.EcAmS chemaGenerator : INFORMATION : createESDTSchema DONE: XML descriptor - /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml, Schema - /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09</p>

#	Action	Expected Result	Notes
			CMG.005.xsd 07.31.2013 12:02:12.155 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : END generateDescriptorSchemaF romXmlDescriptor xmlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc.xml outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd 07.31.2013 12:02:12.155 : Thread ID [41] : xmlsvcs.schemagen.EcAmD escSchemaGenImpl : VERBOSE : END generateDescriptorSchemaF romOdlDescriptor odlDescriptor: /usr/ecs/TS3/CUSTOM/data /ESS/DsESDTMoMOD09C MG.005.desc outputSchema: /stornext/smallfiles/TS3/des criptor/DsESDTMoMOD09 CMG.005.xsd
31	Find the mcf target directory (parameter mcf.target.dir) in the EcAmMaintenanceGui.properties file.  Change the directory to the ESDT specific subdirectory (probably /stornext/smallfiles/&lt;mode&gt;/mcf/).  Perform an ls to verify that the mcf file is found. Repeat for each MCF file generated from the ESDT installation.		/custom/ecs/\${MODE}/CU STOM/cfg/EcDsAmESDT Maint.properties  >mcf.target.dir=/stornext/s mallfiles/\${MODE}/mcf
32	Compare the descriptor file to the mcf file to verify that the MCF is correct. 		

#	Action	Expected Result	Notes
33	Change the directory to this directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/).		
34	Perform an ls to verify that the schema file is found. 		/custom/ecs/\${MODE}/CUSTOM/cfg/EcDsAmESDT Maint.properties descriptor.target.dir=/stornext/smallfiles/\${MODE}/descriptor
35	Make a one to one comparison of the groups in the Inventory Section of the Descriptor file with the corresponding groups in the schema.		
36	Verify that the values of the descriptor are valid according to the rules in the schema.		
37	Find the descriptor source and target directory (parameters descriptor.source.dir and descriptor.target.dir) in the EcAmMaintenanceGui.properties file. 		
38	Change to target directory (probably /stornext/smallfiles/&lt;mode&gt;/descriptor/).		
39	Perform an ls to verify that the descriptor file DsESDT&lt;ShortName&gt;.&lt;VersionId&gt;.desc is found. Repeat for each part descriptor installed in the setup step 1.		
40	Verify that the descriptor file was deleted in the source directory in the mode.		
41	Change to directory descriptor.source.dir.		
42	Verify that the descriptor was deleted by using the ls command and finding no files matching the descriptor's name. Also, repeat for each descriptor installed.		/custom/ecs/\${MODE}/CUSTOM/cfg/EcDsAmESDT Maint.properties descriptor.source.dir=/usr/ecs/\${MODE}/CUSTOM/data/ESS descriptor.target.dir=/stornext/smallfiles/\${MODE}/descriptor This should be at least 2 steps.
43	In the AIM Inventory database, verify that the ShortName, VersionID, and insertTime are populated in the DsMdCollections table for each ESDT installed.   select ShortName, VersionID, insertTime  from DsMdCollections  where ShortName = "&lt;ShortName&gt;"  and VersionID= "&lt;VersionID&gt;" 		

#	Action	Expected Result	Notes
44	Verify that the spatial searchSearchType matches the one specified in the descriptor. <code>select configuredName, VersionID, spatialSearchType from DsMdESDTCConfiguredType where configuredName = '&lt;shortName&gt;' and VersionID = 'VersionID'</code>		
45	Repeat for each installed ESDT.		<code>select c.inserttime, esdt(c.shortname,c.versionid), c.collectionid, c.spatialsearchtype, e.spatialsearchtype from amcollection c join dsgeesdtconfiguredtype e on c.shortname = e.configuredname and c.versionid = e.versionid where (c.shortname = '{SHORTNAME}' and c.versionid = '{VERSIONID}) or (c.shortname = '{SHORTNAME}' and c.versionid = '{VERSIONID})</code> ... <code>order by c.spatialsearchtype, c.shortname, c.versionid; dsgeesdtconfiguredtype replaces DsMdESDTCConfiguredType e.</code> <code>select c.inserttime, esdt(c.shortname,c.versionid), c.collectionid, c.spatialsearchtype, e.spatialsearchtype from amcollection c join dsgeesdtconfiguredtype e on c.shortname =</code>

#	Action	Expected Result	Notes
			e.configuredname and c.versionid = e.versionid where (c.shortname = 'MOD14' and c.versionid = 5) or (c.shortname = 'MYD14' and c.versionid = 5) or (c.shortname = 'AEPOE7W' and c.versionid = 1) or (c.shortname = 'PMIGBAD1' and c.versionid = 1) or (c.shortname = 'AE_Land' and c.versionid = 2) or (c.shortname = 'AE_Rain' and c.versionid = 2) or (c.shortname = 'MOD09CMG' and c.versionid = 5) or (c.shortname = 'MYD09CMG' and c.versionid = 5) or (c.shortname = 'g3bssp' and c.versionid = 7) or (c.shortname = 'g3bt' and c.versionid = 7) order by c.spatialsearchtype, c.shortname, c.versionid;  inserttime   esdt   collectionid   spatialsearchtype   spatialsearchtype ----- -----+----- -----+----- +-----+----- -----  2013-07-31 12:02:12.560421   MOD14.005   198330   GPolygon

#	Action	Expected Result	Notes
			GPolygon  2013-07-31 12:02:13.716981   MYD14.005   198332   GPolygon   GPolygon  2013-07-31 12:02:11.403874   AEPOE7W.001   198328   NotSupported   NotSupported  2013- 07-31 12:02:14.194661   PM1GBAD1.001   198333   NotSupported   NotSupported  2013- 07-31 12:02:09.656776   AE_Land.002   198326   Orbit   Orbit  2013-07-31 12:02:10.765544   AE_Rain.002   198327   Orbit   Orbit  2013-07-31 12:02:14.865666   g3bssp.007   198334   Point   Point<br /> 2013-07-31 12:02:15.367958   g3bt.007   198335   Point   Point  2013-07-31 12:02:11.866531   MOD09CMG.005   198329   Rectangle   Rectangle  2013-07-31 12:02:13.12891   MYD09CMG.005   198331   Rectangle   Rectangle (10 rows)
46	In another terminal log into the Spatial Subscription Server (SSS) database. View the &lt;ShortName&gt;.&lt;VersionId&gt; descriptor in the mode and find the INSERT object under the EVENT group in the		

#	Action	Expected Result	Notes
	descriptor. Under the EVENTPARMS object are a list of parameters which should be populated in the SSS database.		
47	Verify that the event was inserted into the EcNbEventDefinition table.   select * from EcNbEventDefinition  where ESDT_Id in ( &lt;a quoted list of ShortNames of installed ESDTs&gt;) 		
48	Verify that each parameter EVENTPARM parameter is populated in the SSS database .   select * from EcNbEventAttrXref  where ESDT_Id = "ShortName" 		
49	Repeat this step for each descriptor installed		
50	Change directory to the log.dir directory of the mode.		
51	View the ESDT_Maint.log and search for the first ESDT installed.		
52	Verify the time of the installation request, the ShortName, Version ID, descriptor file name and the result of the operation (successful) is listed in the log file.		
53	Repeat for each ESDT installed.		time of the installation request: 07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTOM/data/ESS/DsESDTMoM/OD09CMG.005.desc  ShortName: 07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:1 String Value:MOD09CMG Type:1  Version ID: 07.31.2013 12:02:11.862 : Thread ID [41] : VERBOSE : AbstractServiceImpl.setObject4CallableStmt() Param Index:4 String Value:5 Type:4  descriptor file name: 

---

---

#	Action	Expected Result	Notes
			<pre>&gt;07.31.2013 12:02:11.821 : Thread ID [41] : VERBOSE : installing ESDT:/usr/ecs/TS3/CUSTO M/data/ESS/DsESDTMoM OD09CMG.005.desc&lt;br &lt;br /&gt;result of the operation (successful):&lt;br &gt;07.31.2013 12:02:12.215 : Thread ID [41] : INFORMATION : InstallESDTPage:Successful ly installed ESDT DsESDTMoMOD09CMG.0 05.desc</pre>

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

10.1.2 Test Case 4 - Search Granules for Deletion (ECS-ECSTC-863)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	Select a test collection.		
2	Find a datetime T_BEGIN that's later than the earliest beginningdatetime of granules in the test collection.		
3	Find a datetime T_END that's earlier than the latest beginningdatetime of granules in the test collection.		

---

---

#	Action	Expected Result	Notes
4	Find the granules with beginningdatetime between T_BEGIN and T_END: <pre>select granuleid from amgranule where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID} and beginningdatetime &gt;= '\${T_BEGIN}' and beginningdatetime &lt;= '\${T_END}';</pre>		
5	To simplify the command line, set and export the following environment variables before running the search utility: <pre>MODE &amp;MODE</pre>		
6	Execute the EcDsBulkSearch utility with the following command line parameters: <pre>EcDsBulkSearch.pl -mode &amp;MODE -geoidfile &amp;path/geoidfile_name -name &amp;ShortName -version &amp;VersionId -acquirebegin &amp;mm/dd/yyyy [&amp;hh:mm:ss] -acquireend &amp;mm/dd/yyyy [&amp;hh:mm:ss]</pre>		
7	<i>See AIM Granule Deletion 609 document for details.</i>		#comment
8	Verify the utility runs without errors.		
9	Verify the geoid file file is created in the specified directory.		
10	Verify the geoid file contains the expected granule GEOIDs (i.e., all granules for the specified Shortname.versionId which have a BeginningDateTime in the specified range).		

TEST DATA:

Any current data types

EXPECTED RESULTS:

10.1.3 Test Case 5 - Granule Logical Deletion -- physical (ECS-ECSTC-865)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Logical delete is combined with Unpublish and BMGT for an end to end test.</i>		#comment
2	<i>Preconditions</i>		#comment
3	Identify several granules in a single collection to delete. Ensure some granules have associated browse.		
4	To simplify the command line, set and export the following environment variables before running the delete utility: #csh setenv MODE &lt;MODE&gt;  #bash export MODE=&lt;MODE&gt;  See AIM Granule Deletion 609 document for details.		
5	Ensure a TCP proxy or mock ECHO is capturing BMGT traffic.		
6	Ensure BMGT is running:  ./EcBmBMGTAppStart \${MODE}		
7	Ensure the granules' collection is enabled for BMGT collection and granule export.  update bg_collection_configuration set collectionexportflag = 'Y', granuleexportflag = 'Y' where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID}  If the collection was newly enabled for export in this step, wait for all of its granules to finish exporting.		
8	<i>Setup and Verification</i>		#comment
9	Note the current time as t0.		
10	Run the EcDsBulkDelete utility with the -physical option as follows:  EcDsBulkDelete.pl -physical  -geoidfile &lt;path/geoidfile_name&gt;  -log &lt;log_file_name.log&gt;		
11	<i>Verify the utility runs successfully and the output is correct:</i>		#comment
12	Verify all granule(s) in the path/geoidfile_name have been marked for deletion, i.e., each has a non-null deleteEffectiveDate in the amgranule table.		
13	Verify the deleted granule(s) have been logged in the specified log file.		
14	Verify the TCP proxy log shows that after time t0 BMGT exported a DELETE request for each granule.		
15	Note the current time as t1.		
16	Run the Unpublish utility with -aim option in the mode:  ./EcDIUnpublishStart.pl -mode \${MODE} -aim -offset 1		
17	Verify the granules marked for deletion are moved to the hidden Data Pool.		
18	Verify the TCP proxy shows no requests for the unpublished granules after time t1.		ncr8051949 Unpublished logical deleted a science granule associated with a

#	Action	Expected Result	Notes
			browse causes BMGT exports the logical deleted a science granule to ECHO twice.

TEST DATA:

Any current data types

EXPECTED RESULTS:

10.1.4 Test Case 9 - Granule Deletion Phase II - DeletionCleanup (ECS-ECSTC-866)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Granule Deletion Phase II - DeletionCleanup] (including browse)</i>		#comment
2	<i>Preconditions</i>		#comment
3	Select 2 ESDTs with granules to be deleted. Save the ESDTs to a file, one per line.		
4	Ensure granules for at least 2 ESDTs have been marked for deletion, using the BulkDelete utility.  At least one of these ESDTs should be a MISR ESDT in the DsMdMisrProcessingCriteria table.  Some granules should be marked for deletion using the -physical option, and some should be marked using the -dfa option.		
5	Ensure either that no other granules in the mode are marked for deletion or that it is OK with the lab lead and everyone else in the mode to physically delete them.		
6	Ensure the granules to be deleted are unpublished, e.g., by running the Unpublish utility with the -aim option.  Note that Granule Deletion Phase II will skip granules that are in public Data Pool or on order.		
7	To simplify the command line, set and export the following environment variables before running the search utility:  #csh setenv		

#	Action	Expected Result	Notes
	MODE &lt;MODE&gt;  #bash export MODE=&lt;MODE&gt;		
8	<i>Set up and verification</i>		#comment
9	<i>See AIM Granule Deletion 609 document for details.  1. All command line parameters are optional, if not supplied, required parameters will be prompted in runtime;  2. The batch size command line parameters can take default values if they are not specified by the operator;  3. If there are any granule(s) that have not been cleaned up in a previous run of the DeletionCleanup utility, the operator will be prompted to 1) complete the unfinished run only, or 2) complete the previous unfinished run and start a new run;  4. On the lag time menu, choose either option 1 or option 2.  5. On the Data Type menu, choose option 1, and use an input file of the ESDTs for which you have marked granules for deletion.</i>		#comment
10	Run the EcDsDeletionCleanup utility with the following options:  EcDsDeletionCleanup.pl -mode &lt;MODE&gt;		
11	On the lag time menu, choose either option 1 or option 2.		
12	On the Data Type menu, choose option 1, and use the input file of the ESDTs prepared earlier.		
13	<i>Verify that the utility runs successfully and that the output is correct:</i>		#comment
14	Verify all database entries have been cleaned up for granules marked for deletion with the -physical option.  For the MISR granules this includes the DsMdMsrAttributes, DsMdMsrCamera and DsMdMsrBrowseGranuleXref tables.		
15	Verify that the database entries have NOT been cleaned up for granules marked for deletion with the -dfa option.		
16	Verify XML metadata files have been physically removed from the XML archive for granules marked for deletion with the -physical option.		
17	Verify XML metadata files have NOT been physical removed from the XML archive for granules marked for delete with the -dfa option.		
18	Verify data files have been physically removed from the archive for all granules marked for deletion (with either the -physical or -dfa options).		
19	Verify that the deleted granule(s) have been logged in the specified log file.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

10.1.5 Test Case 16 - DPL - AIM Inventory Validation (ECS-ECSTC-867)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>This criterion was adapted from criterion 200 in ticket DP_7F_01.</i>		#comment
2	<i>S-1 Execute a Data Pool inventory validation Ensure that the AIM Inventory Catalog includes at least two of each of the following cases:  a. Collections defined in the DPL inventory and configured to be published during ingest as well as collections not so configured that contain ECS granules in the AIM inventory, including granules that are: logically deleted (non-NULL deleteEffectiveDate), flagged as deleted from archive (DeleteFromArchive set to 'Y') and flagged as hidden (DeleteFromArchive set to 'H'), as well as granules that do not meet any of these conditions. In addition to granules from science collections, include at least one granule each from the DAP, QA, and PH collections that is logically deleted and one that is not. These granules are not meant to represent data integrity issues and are only used to supply conditions that should be ignored by the validation.  b. Non-ECS granules in the DPL inventory (also not a data integrity issue).  c. Collections defined in the DPL inventory and configured to be published during ingest that contain ECS granules in the AIM inventory whose Inventory entries indicate they are members of the Hidden Data Pool (this simulates a Publication failure or possibly the removal of a granule restriction or deletion). In addition to granules from science collections, include at least one granule each from the QA and PH collections.  d. Collections with ECS granules that are public in the Data Pool and are currently flagged as logically deleted (i.e., have a non-NULL deleteEffectiveDate), deleted from archive (i.e., DeleteFromArchive set to 'Y'), or hidden from normal users (i.e., DeleteFromArchive set to 'H') in the AIM Inventory Catalog, as well as granules that do not meet any of these conditions (this simulates the result of running the AIM Granule Deletion utilities without running the DPL Un-Publish utility).  e. Granules in the Public Data Pool that contain Inventory Catalog Restriction entries that		#comment

#	Action	Expected Result	Notes
	indicate the Granules should not be Public.  f. Science Granules in the Public Data Pool that should have been replaced by a replacement granule in the Hidden Data Pool that should be public (the database may be manually updated to create this situation). This simulates a publication failure on a replacement granule.  g. Granules that are registered in the AIM Inventory Catalog but are not listed as Archived (the Inventory Catalog ArchiveTime is NULL). This could be the result of an Ingest failure prior to archiving. The state of the request/granule in Ingest (if it exists) should indicate that Ingest is not currently processing it (it should be in a terminal state). In addition to granules from science collections, include at least one granule each from the DAP, QA, and PH collections.  h. Public and Hidden Science Granules that were manually deleted from the Data Pool using the EcDICleanupGranules utility (both should be represented as hidden after the utility completes).  i. Public Science Granules with associated Browse granules where the Browse Inventory Catalog entries indicate they are not in the Public Data Pool.  j. Public Browse granules that are associated only with hidden Science Granules.  k. Science Granules in the Public Data Pool that are replacements for granules in the Hidden Data Pool. This represents a normal replacement that should not be reported.  l. Hidden granules with &quot;not null&quot; version numbers that were replaced but their file names are not consistent with their version numbers recorded in the Inventory Catalog.   Specify the output location for the validation report.  Perform the test while there are concurrent Ingest and Distribution activities, with at least twenty granules ingested and twenty granules being newly ordered and distributed during the test.</i>		
3	a) Ensure the AllowPublishFlag = 'Y' for datatype MOD29P1D.005 and in AllowPublishFlag = 'N' for datatype MOD29P1N.005 in the AmCollection table.		
4	Ingest the PDRs in the test requirement 100_A/MD29P1D.005 above.		
5	a.1) Use the query below to find geoids:  select 'SC:'    esdt(shortname, versionid)    ':'    granuleid from AmGranule where ShortName = 'MOD29P1D' and VersionId = 5 and RegistrationTime > now()::date;		
6	a.2) Pick up the first granule to save in /usr/ecs/&lt;MODE&gt;/CUSTOM/utilities/Delete_Physical.txt  Pick up the second granule to save in /usr/ecs/&lt;MODE&gt;/CUSTOM/utilities/Delete_DFA.txt  Pick		

#	Action	Expected Result	Notes
	up the third granule to save in /usr/ecs/&lt;MODE&gt;/CUSTOM/utilities/Hidden.txt  Pick up the third granule to save in /usr/ecs/&lt;MODE&gt;/CUSTOM/utilities/Hidden.txt		
7	a.3) Ingest the the PDRs in the test requirement 100_A/MYD29P1D.005 above then use the query in step a.1 for MYD29P1D.005 data type to obtain granules.  Edit the Delete_Physical.txt in step a.2 to add the first granule above and save the file.  Edit the Delete_DFA.txt in step a.2 above to add the second granule above and save the file.  Edit the Hidden.txt in step a.2 above to add to add the third granule above and save the file.		
8	a.4) Ingest the the PDRs in the test requirement 100_A/MOD29P1N.005 above then use the query in step a.1 for MOD29P1N.005 data type to obtain granules.  Edit the Delete_Physical.txt in step a.2 to add the first granule above and save the file.  Edit the Delete_DFA.txt in step a.2 above to add the second granule above and save the file.  Edit the Hidden.txt in step a.2 above to add to add the third granule above and save the file		
9	a.5) Ingest the the PDRs in the test requirement 100_A/MYD29P1N.005 above then use the query in step a.1 for MYD29P1N.005 data type to obtain granules.  Edit the Delete_Physical.txt in step a.2 to add the first granule above and save the file.  Edit the Delete_DFA.txt in step a.2 above to add to add the second granule above and save the file. <br >Edit the Hidden.txt in step a.2 above to add to add the third granule above and save the file		
10	Run EcDsBulkDelete.pl &lt;MODE&gt; -physical -geoidfile delete_physical.txt -server &lt;server&gt; -database &lt;database name&gt; - U &lt;username&gt; -P &lt;user password&gt;		
11	Run EcDsBulkDelete.pl &lt;MODE&gt; -dfa -geoidfile delete_dfa.txt - server &lt;server&gt; -database &lt;database name&gt; -U &lt;username&gt; -P &lt;user password&gt;  Use the GranuleId in the Hidden.txt file to set DFA = 'H'.		
12	b) EcDIPublishUtilityStart &lt;MODE&gt; -nonecs -file &lt;filename&gt;		
13	c) Ingest the the PDRs in the test requirement 100_C above create one inputfile which contains granuleId in S-1c.		
14	Run unpublish script below.  EcDIUnpublishStart.pl -mode &lt;MODE&gt; -f inputfile  Inputfile contains QA and PH granuleIDs.		

#	Action	Expected Result	Notes
15	d) Ingest the PDRs in the test requirement 100_D above.  To logically delete 2 granules, run BulkDelete -physical.  To DFA 2 granules, run BulkDelete -dfa.  To hide 2 granules, use SQL to set DeleteFromArchive = 'H'.  Leave 2 granules as is.		
16	e) Ingest the PDRs in the test requirement 100_e.		
17	Login to the ecs database.  Exec InsertGranuleRestriction granuleId, RestrictionFlag, UnpublishFlag  For example, Exec InsertGranuleRestriction 12345, 255, 'Y'		
18	Run EcDIUnpublishStart.pl script EcDIUnpublishStart.pl -mode &lt;MODE&gt; -g granuleId &lt;step S-1e&gt;		
19	f) Turn replacement on or off for an ESDT using the DPL maintenance GUI. Set ReplacementOn = 'N' for ESDT MOD29P1D.005 Ingest granule f1 and allow it to be published.		
20	Re-ingest d2. It will should publication and therefore remain in the hidden datapool.		
21	Set ReplacementOn = 'Y' for ESDT MOD29P1D.005 when the operator runs IVT it should identify the granule the operator ingested as a missing replacement.		
22	g) Set non-exist archive directory for data type in 100_G above then ingest the PDRs in the test requirement 100_G.		
23	h) Ingest granules.		
24	Get granuleId then run the script below.  EcDICleanupGranules.pl &lt;MODE&gt; -ecs -grans &lt;granuleId&gt;  The ecsId_file contains ecsId.		
25	select ShortName, VersionId, granuleId, isOrderOnly, PublishTime, from AmGranule where GranuleId in (granuleId)		
26	j) Ingest PDRs in the test requirement 100_J for sc granules with browse. Manually update the IsOrderOnly to 'H' for the browse granule.		
27	k) Ingest sc granules with browse.		
28	Run the publish the browse using granuleIds above. The filename contains ecs GranuleId  EcDIPublishUtilityStart MODE -ecs -file filename		
29	a. Ingest granules		
30	Using the SSS GUI, create a subscription order for MOD29P1D.005. Ingest all PDRs in directory the below.  />stotestdata/DROP_801/DP_81_01/Criteria/100/100_Concurrent		

#	Action	Expected Result	Notes
31	While the 40 granules are being ingested and 40 granules are being distributed, execute the validation utility below.  EcDIInventoryValidationTool.pl &lt;MODE&gt;  and save off the log and output files for verification later.		
32	<i>V-1 Verify that non-ECS granules (S-1b) are not reported as missing from the Archive.</i>		#comment
33	Open the EcDIInventoryValidationTool.log to verify that non-ECS granules are not reported as missing from Archive.		
34	<i>V-2 Verify ECS granules that are eligible to be public but that are members of the Hidden Data Pool (S-1c) are logged by the utility and that the log includes their status in the AIM inventory and the time they were inserted into ECS.</i>		#comment
35	Open the EcDIInventoryValidationTool.log to verify public ECS granules that are logged hidden (S-1c) are logged includes their status and ecsInsertTime for QA and PH granules.  2011/06/02 07:15:40.760 [GranuleId, DeleteEffectiveDate, DeleteFromArchive, ArchiveTime, ReplacementOnFlag]: 3002000469, , N, Jun 1 2011 3:20PM, Y		
36	<i>V-3 Verify Public ECS granules that are logically deleted, DFA'ed, or hidden (S-1d) are logged by the utility and that the log includes their status in the AIM inventory and the time they were inserted into ECS.</i>		#comment
37	Open the EcDIInventoryValidationTool.log to verify public ECS granules that are logged logically deleted, DFA'ed or hidden (S-1d) are logged includes their status and ecsInsertTime.  Logically deleted: 2011/06/02 08:05:22.879 (GranuleId, DeleteEffectiveDate, DeleteFromArchive, ArchiveTime): 3002000471, Jun 2 2011 8:00AM, N, Jun 2 2011 7:49AM.  DFA'ed: 2011/06/02 08:05:22.879 (GranuleId, DeleteEffectiveDate, DeleteFromArchive, ArchiveTime): 3002000470, , Y, Jun 2 2011 7:49AM.  Hidden: 2011/06/02 08:05:22.879 (GranuleId, DeleteEffectiveDate, DeleteFromArchive, ArchiveTime): 3002000472, , H, Jun 2 2011 7:57AM. 		
38	<i>V-4 Verify Public ECS granules that are Restricted with an option to un-publish (S-1e) are logged by the utility.</i>		#comment
39	Open the EcDIInventoryValidationTool.log to verify public ECS granules that are Restricted with an option to un-publish (S-1e) are logged.  2011/06/02 07:51:43.122 (GranuleId, DeleteEffectiveDate, DeleteFromArchive, ArchiveTime): 3002000462, , N, May 31 2011 10:18AM.		

#	Action	Expected Result	Notes
40	<i>V-5 Verify Public ECS granules that have eligible replacement granules in the Hidden Data Pool (S-1f) are logged by the utility and that granules were replaced (S-1K) are not logged.</i>		#comment
41	Open the EcDIInventoryValidationTool.log to verify granules that have eligible replacement granules in the Hidden Data Pool (S-1f) are logged and that granules were replaced (S-1K) are not logged.		
42	<i>V-6 Verify that granules with NULL ArchiveTime (S-1g) are logged by the utility. Notes, granules that are currently being processed by Ingest (not in a “terminal” state) should not be reported.</i>		#comment
43	Open the EcDIInventoryValidationTool.log to verify that granules with NULL ArchiveTime (S-1g) are logged.		
44	<i>V-7 Verify that granules that were manually deleted from the Data Pool (S-1h) are not logged. They should still be represented as Hidden granules in the Inventory Catalog.</i>		#comment
45	Open the EcDIInventoryValidationTool.log to verify that granules that were manually deleted from the Data Pool (S-1h) are not logged.		
46	<i>V-8 Verify that browse granules that are missing from the Public Data Pool (S-1i) are logged by the utility.</i>		#comment
47	Open the EcDIInventoryValidationTool.log to verify that browse granules that are missing from the Public Data Pool (S-1i) are logged. For example, Start CheckInventoryI: Browse Granules that should be in public Datapool.  2011/05/31 16:03:28.409 (BrowseId, DeleteEffectiveDate, ArchiveTime): 3002000158, , Apr 21 2011 2:12PM.		
48	<i>V-9 Verify that Public browse granules that are associated with Hidden Science granules only (S-1j) are logged by the utility.</i>		#comment
49	EcDIPublishUtilityStart OPS -ecs -g browseId (298972) Check the EcDIPublishUtility.log to make sure there is browse id which was published from the command line.		
50	<i>V-10 Verify that no DPL-AIM inventory discrepancies are logged other than those mentioned in V-2 through V-9.</i>		#comment
51	Verify that no other discrepancy (from this data) is logged in this run.		
52	<i>V-11 Verify that the inventory validation creates one or several output files reporting the discrepancies in the specified location.</i>		#comment
53	The VALIDATION_OUTPUT_DIR defined in the EcDIInventoryValidationTool.CFG file. Go to the VALIDATION_OUTPUT_DIR to verify that Inventory Validation creates one or more output file suitable as input to the various repair utilities.		

#	Action	Expected Result	Notes
54	<i>V-12 Verify that the output report(s) include all the discrepancies that were logged and no others.</i>		#comment
55	Verify that an output report is prepared that includes the above errors and no other errors.		
56	<i>V-13 Verify that the validation exits with an exit code indicating that errors occurred.</i>		#comment
57	Open the EcDIInventoryValidationTool.log to verify that the Inventory Validation Tool exits with an exit code that indicates some errors occurred.		
58	<i>V-14 Verify that granules with version numbers that don't match their file names (S-11) are logged</i>		#comment
59	Verify that granules with version numbers that don't match their file names (S-11) are logged.		

TEST DATA:

Crit id	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location
100_A/AE_Land 100_A/DAP 100_A/MOD29P1D 100_A/MYD29P1D 100_A/MOD29P1N 100_A/MYD29P1N		AE_Land.002 DAP.001 MOD29P1D.005 MYD29P1D.005 MOD29P1N.005 MYD29P1N.005 PH.001 QA.001		2 granules 2 daps 4 granules 4 granules 4 non- public granules 4 non-public granules 2 ph 2 qa		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_A/AE_Land.002 /sotestdata/DROP_801/DP_81_01/Criteria/100/100_A/DAP.001 /sotestdata/DROP_801/DP_81_01/Criteria/100/100_A/MOD29P1D.005 /sotestdata/DROP_801/DP_81_01/Criteria/100/100_A/MYD29P1D.005 /sotestdata/DROP_801/DP_81_01/Criteria/100/100_A/MOD29P1N.005 /sotestdata/DROP_801/DP_81_01/Criteria/100/100_A/MYD29P1N.005
100_B		ATSM2LSF.001		2 granules		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_B
100_C		AE_Land.002 MOD29P1D.005 MYD29P1D.005		2 granules 2 granules 2 ph 2 qa		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_C

Crit id	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location
		PH.001 QA.001				
100_D		MOD29P1D.005 MYD29P1D.005		4 granules 4 granules		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_D
100_E		MOD29P1D.005		2 granules		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_E
100_F		MOD29P1D.005		2 granules + 2 replacement		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_F
100_G		AE_Land.002 DAP MOD29P1D.005 MYD29P1D.005 PH.001 QA.001		1 granules 1 granules 2 granules 2 non –public granules 2 ph 2 qa		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_G
100_H		MOD29P1D.005 MYD29P1D.005		2 granules 2 non –public granules		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_H
100_I		MOD29P1D.005 Browse.001		2 granules 2 browse		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_I
100_J		MOD29P1D.005 Browse.001		2 granules 2 browse		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_J
100_K		MOD29P1D.005		2 granules + 2 replacement		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_K
100_Concurrent		MOD29P1D.005 Browse.001		40 granules 40 browse		/sotestdata/DROP_801/DP_81_01/Criteria/100/100_Concurrent

EXPECTED RESULTS:

10.1.6 Test Case 18 - On-line Archive Repair from non-resident tape: DP\_7F\_01 Criterion 320 (ECS-ECSTC-868)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[On-line Archive Repair from non-resident tape: DP_7F_01 Criterion 320] Create phantoms in the public and hidden Data Pool that satisfy the following conditions:  a. Turn at least ten (10) public science granules into phantoms including MISR, MODIS and ASTER granules by removing their science and XML files. At least one of these granules shall be in use for order purposes; remove its hidden links, as well.  b. At least one but not all of the science granules shall have a browse that is present in the public Data Pool.  c. At least two of the phantom science granules with browse shall also be missing their browse links from the Data Pool directories, including a MISR and a non-MISR granule.  d. One of the science granules shall have experienced a file name collision in the hidden Data Pool, that is, its file names shall have been suffixed during Data Pool insert.</i>		#comment
2	Also turn at least five (5) public Browse granules, including MISR, MODIS and ASTER browse into phantoms by removing their browse files from the public Data Pool.		
3	Save the removed XML and browse files for comparison during the test.		
4	Submit an on-line archive repair for these granules via the command line using the RestoreOlaFromTape utility, specifying the granules to be repaired in an input file; and request restoration of browse links that may be missing for granules whose files were restored (using the -restorelinks command line parameter).		
5	Ensure that the test requires access to granules on a tape that is resident in the silo as well as a tape that is not resident in the silo.		
6	Verify that all science granules that resided on tapes resident in the silo were repaired, i.e., their files are now in the correct public Data Pool locations and have the original file names		

---

---

#	Action	Expected Result	Notes
7	Verify that the science granules that resided on tapes that were not resident in the silo were not repaired.		
8	Verify that the utility logs each of the granules that were not repaired, the reason for not repairing it, and the tape label.		
9	Verify that the utility exits with an exit code indicating the occurrence of errors.		
10	Verify that the utility sends a notice to the specified e-mail address indicating that a repair failed due to error, the nature of the repair, and the name and location of the log file.		
11	Make the missing tape resident in the silo and re-run the repair for the skipped granules. Verify that the remaining granules are now repaired		
12	Verify that the contents of all the restored XML files (including those referenced in V-6) match the contents of the original XML files.		
13	Verify that all browse granules were repaired and are present as jpeg files in their original disk location and with their original file names.		
14	Verify that the repaired jpeg files are identical to the original jpeg files.		
15	Verify that the missing browse links are restored.		
16	Verify that the hidden links required by the public granules in condition (a) that are also referenced by orders were restored.		
17	Verify that any files that were copied from the ECS tape archive into the Data Pool during the repair were checksummed and that the checksum is recorded in the Data Pool inventory.		
18	Verify that any files that were copied from the ECS browse archive into the Data Pool during the repair had their checksum verified.		

TEST DATA:

EXPECTED RESULTS:

10.1.7 Test Case 26 - Run Archive Checksum Validation Utility (ACVU) providing a file with a list of dbIds without the -calculate option : CK\_7F\_01, Criterion 670 (ECS-ECSTC-869)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Run Archive Checksum Validation Utility (ACVU) providing a file with a list of dbIds without the -calculate option : CK_7F_01, Criterion 670]</i>		#comment
2	Choose a list of dbIds that span all the possible checksum status and checksum verification time combinations. Run ACVU specifying a file of dbIds and days since last checksum without the calculate option.		
3	Verify the last checksum times of the appropriate granules were updated.		
4	Verify that none of the dbIds with null last checksum times were updated.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

10.1.8 Test Case 27 - AIM XML Check Utility (ECS-ECSTC-870)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[AIM XML Check Utility]</i>		#comment
2	<i>Note: Refer to the AIM XML Check Utility 609 document for detailed instructions.</i>		#comment
3	<i>Usage of EcDsAmXcu.pl:  EcDsAmXcu.pl &lt;MODE&gt; [-days &lt;NUMBER OF DAYS&gt;] [-percent &lt;PERCENT 1-100&gt;] (-ESDT &lt;ESDTS ex. MOD29.005&gt;) [-startdate &lt;STARTDATE ex: Jan 1 2008&gt;] [-enddate &lt;ENDDATE ex: Dec 25 2008&gt;]]  -granuleid &lt;GRANULEIDS&gt;    -file &lt;FILENAME&gt;) [-outputDir &lt;DIRECTORY&gt;]</i>		#comment

#	Action	Expected Result	Notes
4	Run XCU to check the well-formedness of all the xml files or only the specified xml files (by using the optional parameters) in the small file archive in the mode.		
5	Verify that the utility reports the validation results and list the xml files which are not well-formed.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

10.1.9 Test Case 3 - Delete multiple ESDTs, success case (ECS-ECSTC-904)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>DS_7E_01 ESDT Maintenance GUI ITP: http://dmserver.gsfc.nasa.gov/release721/ESDT_Maint_GUI/ITP_DS_7E_01_ESDTGUI.doc&para;</i>		#comment
2	<i>Preconditions</i>		#comment
3	Ensure a database client is connected to the ecs database.		
4	Identify 3 ESDTs for deletion (C1, C2, C3).		
5	Save the collection IDs of C1, C2, C3. For each,  select collectionid from amcollection where shortname = '{SHORTNAME}' and versionid = {VERSIONID}		
6	Save the attribute IDs associated with each of C1, C2, C3. For each,  select attributeid from amcollection c join dsmdcollectionaddnlattrbsxref x on c.collectionid = x.collectionid where shortname = '{SHORTNAME}' and versionid = {VERSIONID}		

#	Action	Expected Result	Notes
7	Ensure C1, C2, C3 have no granules. For each,  select esdt(shortname, versionid), granuleid from amgranule where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID}		
8	Ensure C1, C2, C3 are absent from the data pool. For each,  select * from amcollection where groupid is null and shortname = '\${SHORTNAME}' and versionid = \${VERSIONID}		
9	Ensure C1, C2, C3 are absent from the Spatial Subscription Server database. For each,  select * from EcNbSubscription where esdt_id = '\${C1_SHORTNAME}' and versionid = \${C1_VERSIONID}		
10	<i>S-1 From the ESDT Maintenance GUI select a group of ESDTs for deletion. Each of the ESDTs selected must satisfy the following conditions to for deletion to be possible:  1. the ESDT contain no granules in the AIM inventory database  2. the ESDT is not defined within the Data Pool  3. the Spatial Subscription service contains no active or inactive subscriptions referencing the ESDT</i>		#comment
11	Note the current time as t0.		
12	On the ESDT Maintenance GUI &quot;ESDT List&quot; page, select C1, C2, and C3.		
13	Click the “Delete Selected ESDTs” button at the bottom of the page to begin the ESDT deletion.		
14	Select OK at the confirmation prompt.		
15	<i>V-1 Verify that each ESDT was deleted successfully. 1. The MCF file was deleted 2. The ESDT XML schema file was deleted 3. The ESDT descriptor file was deleted. 4. The ESDT basic collections and PSA associations were deleted from the Inventory database 5. The insert events and event qualifiers for the ESDT were deleted from the Subscription Server database 6. The XML metadata file directories associated with the ESDT were removed from the XML archive 7. PSA definitions are removed if there are no other ESDTs associated with them</i>		#comment
16	1. Verify no MCF files exist for C1, C2, or C3. For each ESDT,  ls -l /stornext/smallfiles/\${MODE}/mcf/\${SHORTNAME}#\${VERSIONID}.MCF  For example,  ls -l /stornext/smallfiles/OPS/mcf/MOD29PID#005.MCF		
17	2. Verify no ESDT XML schema files exist for C1, C2, or C3. For each ESDT,  ls /stornext/smallfiles/\${MODE}/descriptor/*\${SHORTNAME}.\${VERSIO		

#	Action	Expected Result	Notes
	NID}.xsd  For example,  ls -l /stornext/smallfiles/OPS/descriptor/*MOD29.003.xsd		
18	3. Verify no ESDT descriptor files exist for C1, C2, or C3. For each ESDT,   /stornext/smallfiles/\${MODE}/descriptor/*\${SHORTNAME}.\${VERSIONID}.desc  For example,  ls -l /stornext/smallfiles/OPS/descriptor/*MOD29.003.desc		
19	4. Verify the collections table has no rows for C1, C3, or C3. For each ESDT,  select * from amcollection where shortname = '\${C1_SHORTNAME}' and versionid = \${VERSIONID}		
20	4. Verify additional attribute associations no longer exist for C1, C2, C3.  select * from dsmdcollectionaddnlattrbsxref where collectionid in (\${COLLECTIONIDS})		
21	5. Verify Subscription Server database has no event definitions for C1, C2, or C3. For each,  select * from ecnbeventdefinition where esdt_id = '\${SHORTNAME}' and versionid = \${VERSIONID}		
22	5. Verify Subscription Server database has no event qualifiers for C1, C2, or C3. For each,  select * from ecnbeventattrxref where esdt_id = '\${SHORTNAME}' and versionid = \${VERSIONID}		
23	6. Verify no XML metadata directory exists in the small file archive for C1, C2, or C3. For each,  ls -ld /stornext/smallfiles/\${MODE}/metadata/\${SHORTNAME}.\${VERSIONID}   For example,  ls -ld /stornext/smallfiles/OPS/metadata/MOD29.003		
24	7. Verify the additional attribute definitions for C1, C2, C3 no longer exist:  select attributeid from dsmdadditionalattributes where attributeid in (\${ATTRIBUTEIDS})  If any previously associated attributes remain, verify they are associated with existing collections:  select distinct(attributeid) from dsmdcollectionaddnlattrbsxref where attributeid in (\${REMAINING_ATTRIBUTEIDS})  should return the same list of \${REMAINING_ATTRIBUTEIDS}.		
25	<i>V-2 Verify that the ESDT Maintenance GUI displays a message indicating number of ESDTs deleted.</i>		#comment
26	Verify the ESDT Maintenance GUI displays a message indicating 3 ESDTs were deleted.		
27	<i>V-3 Verify that the ESDT Maintenance GUI logs the processing activities, including time of request, action requested, ESDT ShortName,		#comment

#	Action	Expected Result	Notes
	Version ID, descriptor file name, and the result of the operation, in a configured log file.</i>		
28	ssh to the ESDT Maintenance GUI host.		
29	Search the ESDT GUI log for the first mentions of C1, C2, and C3 on or after the time t0.  /usr/ecs/\${MODE}/CUSTOM/logs/ESDTMaintenanceGUI.*.log*		
30	Verify each of the following is logged for each of C1, C2, C3:  request time action requested (delete) ShortName Version ID descriptor file name operation result		The descriptor file name is not logged in the ESDTMaintenanceGUI.debug0.log file. There is a existence ncr8051131.Test Run - 1705:Delete multiple ESDTs, success case - Multiple Issues Found

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

10.1.10Test Case 2 - Update multiple ESDTs, success case (ECS-ECSTC-864)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Preconditions</i>		#comment
2	<i>Modify the ESDT descriptor ODL files for several (at least 6) previously installed ESDTs, such that the modifications include the following variety of ESDT updates:  1. removing one or more existing collection level attributes that are not restricted  2. removing DLLName parameter from the Collection metadata group  3. changing the ShortName in the CollectionAssociation group  4. changing a mandatory inventory		#comment

#	Action	Expected Result	Notes
	attribute to optional 5 adding one or more optional inventory metadata attributes 6. adding one or more new qualifiers on existing events		
3	Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2060/Replacement as the initial descriptors. Use only GLA01.013 ... GLA06.013; do not use GLA07.013.		#comment
4	Use descriptors in /sotestdata/DROP_721/DS_7E_01/Criteria/2120 as the replacement descriptors.		#comment
5	Compare each replacement descriptor with its initial counterpart, and note the differences.		
6	Ensure none of the test collections are installed, according to the ESDT Maintenance GUI.		
7	Note the values associated with the following names in EcDsAmESDTMaint.properties: descriptor.target.dir descriptor.source.dir mcf.target.dir E.g., descriptor.target.dir=/stornext/smallfiles/OPS/descriptor descriptor.source.dir=/usr/ecs/OPS/CUSTOM/data/ESS mcf.target.dir=/stornext/smallfiles/OPS/mcf		
8	Install the initial test collections: Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}. Log in to the ESDT Maintenance GUI. Click the "Install New ESDTs/Update Existing ESDTs" button. Select all the test descriptors. Click the "Proceed with installation/update" button. Ensure that all descriptors install successfully. Ensure the descriptor files are removed from the source directory.		
9	Verify an MCF file is created in \${mcf.target.dir} for each descriptor.		
10	Copy the MCF files to a separate directory for later verification.		
11	Verify each descriptor was moved to \${descriptor.target.dir}.		
12	Copy the descriptor files to a separate directory for later verification.		
13	Verify an XML schema file (*.xsd) was created for each descriptor in \${descriptor.target.dir}.		
14	Copy the schema files to a separate directory for later verification.		
15	Extract /usr/ecs/TS3/CUSTOM/lib/DSS/EcDsAmEsdtCommonSchemas.jar for later verification.		
16	Setup		#comment

#	Action	Expected Result	Notes
17	Copy the descriptor files to the the ESDT Maintenance GUI host in \${descriptor.source.dir}.		
18	Log in to the ESDT Maintenance GUI.		
19	Click the &quot;Install New ESDTs/Update Existing ESDTs&quot; button.		
20	Select all the test descriptors.		
21	Note the current time as t0.		
22	Click the &quot;Proceed with installation/update&quot; button.		
23	Upon completion of the ESDT update, verify that the GUI displays a message indicating the number of ESDTs successfully updated.		
24	<i>Verify that an MCF file is created for each ESDT being updated, and stored into the configured MCF directory location replacing the previous version.</i>		#comment
25	Verify for each updated descriptor an MCF file exists in \${mcf.target.dir} with a timestamp more recent than time t0.		
26	<i>Verify that the MCF files are correct.</i>		#comment
27	For each descriptor file, note the contents of the INVENTORYMETADATA section and the ARCHIVEMETADATA section.		
28	Verify the MCF file has the same INVENTORYMETADATA group as the descriptor, with the exception that ProductSpecificMetadata group from the descriptor file is replaced with this static AdditionalAttributes block in the MCF file:   GROUP = AdditionalAttributes  OBJECT = AdditionalAttributesContainer  Data_Location = &quot;NONE&quot;  Mandatory = &quot;FALSE&quot;  CLASS = &quot;M&quot;  OBJECT = AdditionalAttributeName  Mandatory = &quot;FALSE&quot;  CLASS = &quot;M&quot;  Data_Location = &quot;PGE&quot;  NUM_VAL = 1  TYPE = &quot;STRING&quot;  END_OBJECT = AdditionalAttributeName  GROUP = InformationContent  CLASS = &quot;M&quot;  OBJECT = ParameterValue  Mandatory = &quot;FALSE&quot;  Data_Location = &quot;PGE&quot;  NUM_VAL = 1  TYPE = &quot;STRING&quot;  END_OBJECT = ParameterValue  END_GROUP = InformationContent  END_OBJECT = AdditionalAttributesContainer  END_GROUP = AdditionalAttributes		
29	Verify in the MCF file INVENTORYMETADATA section is followed by the descriptor's ARCHIVEMETADATA section.		

#	Action	Expected Result	Notes
30	<i>Verify the descriptor file for each ESDT is replaced by the updated descriptor file, and that the updated descriptor files are consistent with the ECS data model.</i>		#comment
31	Verify each ESDT's updated descriptor file replaced the initial descriptor file in the \${descriptor.source.dir} directory, using diff.		
32	Verify each ESDT validates against the extracted schema. For each ESDT,  Log in to the ESDT Maintenance GUI. Click on the descriptor file in the list. Copy the XML version of the descriptor. Save the XML to a file. Validate the file against the extracted schema:  xmllint --noout --schema /path/to/schema/Descriptor.xsd ../descriptor_file.xml		
33	<i>Verify that an XML schema file is generated for each ESDT and stored into the configured location replacing the previous version</i>		#comment
34	Verify that each ESDT has an XML schema file (*.xsd) in \${descriptor.source.dir} with a timestamp on or after time t0.		
35	<i>Verify that the generated XML schema files contain validation rules based upon the contents of the Inventory section of the Descriptor file and the default element rules.</i>		#comment
36	Diff the updated XML schema files against the initial XML schema files saved earlier.		
37	Verify the differences between updated and initial XML schema files correspond to the differences between updated and initial descriptor files.		
38	<i>Verify that collection-based tables in the AIM Inventory database as identified in the Operations Concept are populated with correct information.</i>		#comment
39	For each updated ESDT, in the ecs database, in the amcollection table, verify lastupdate is after time t0 and each of LongName, CollectionDescription, SpatialSearchType, and RevisionDate match those in the descriptor file:  select lastupdate, esdt(shortname,versionid), LongName, CollectionDescription, SpatialSearchType, RevisionDate from amcollection where shortname = '\${SHORTNAME}' and versionid = \${VERSIONID};		
40	For each updated ESDT, in the ecs database, in the DsGeESDTCfiguredType table, verify descriptorFileNameBase matches the descriptor file name and spatialSearchType matches that in the descriptor file:  select descriptorFileNameBase, spatialSearchType from DsGeESDTCfiguredType where configuredName = '\${SHORTNAME}' and versionid = \${VERSIONID};		

#	Action	Expected Result	Notes
41	For each updated ESDT, in the ecs database, verify the AdditionalAttributes are correctly cross-referenced and that each value matches that in the descriptor file in the appropriate AdditionalAttributesContainer section:  select aa.attributeid, aa.additionalattributename, aa.additionalattributedescription, aa.additionalattributedatatype, aa.parameterunitsofmeasure, aa.parameterrangebegin, aa.parameterrangeend, aa.parametervalueaccuracy, aa.valueaccuracyexplanation, aa.measurementresolution from amcollection c join dsmdcollectionadnlattrbsxref ax on c.collectionid = ax.collectionid join dsmdadditionalattributes aa on ax.attributeid = aa.attributeid where c.shortname = '{SHORTNAME}' and c.versionid = {VERSIONID}; Note that if an additional attribute existed in the database prior to installing an ESDT, it will not be updated to reflect the new definition.		
42	<i>Verify that the insert event and event qualifiers are updated correctly in the Spatial Subscription Server database for each ESDT being updated.</i>		#comment
43	Verify each ESDT has an INSERT event definition:  select * from EcNbEventDefinition where ESDT_Id = '{SHORTNAME}' and VersionID = {VERSIONID}		
44	From each descriptor file, get a list of EVENTPARM names:  sed -n '/^GROUP\s*=\s*EVENT\s*\$/,/^END_GROUP\s*=\s*EVENT\s*\$/ { /^OBJECT\s*=\s*INSERT\s*\$/,/^END_OBJECT\s*=\s*INSERT\s*\$/ { /^OBJECT\s*=\s*EVENTPARMS\s*\$/,/^END_OBJECT\s*=\s*EVENTPARMS\s*\$/ { /^OBJECT\s*=\s*EVENTPARMS\s*\$/ d s/^OBJECT\s*=\s*//p } } } }' {DESCRIPTOR_FILE}		
45	For each ESDT, find the EVENTPARMS that are eligible to be added to the Spatial Subscription Server database:  select attributename from ecnbeventmetadataattrdef where attributename in ({EVENTPARMS}) order by attributename;		
46	For each ESDT, find the EVENTPARMS that were associated with that ESDT as event qualifiers:  select attributename from ecnbeventattrxref where esdt_id = '{SHORTNAME}' and versionid = {VERSIONID} order by attributename;		
47	Verify the associated event qualifiers match the eligible list with the exception that 'RangeBeginningTime' and 'RangeEndingTime' from the descriptor get combined into 'GranuleTimeRange' in the		

#	Action	Expected Result	Notes
	database;  'RangeBeginningDate' and 'RangeEndingDate' from the descriptor get combined into 'GranuleDateRange' in the database.		
48	Verify that, upon completion of updates, the ESDT Maintenance GUI displays a message saying that the ESDT changes will take effect only after the Ingest service is re-started.		

**TEST DATA:**

Any current data types.

Initial ESDTS: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2060/Replacement

Updated ESDTS: /sotestdata/DROP\_721/DS\_7E\_01/Criteria/2120

The 'update' ESDT descriptor files have been modified as follows:

1. removing one or more existing collection level attributes that are not restricted  
DsESDTGIGLA01.013.desc
2. removing DLLName parameter from the Collection metadata group  
DsESDTGIGLA02.013.desc
3. changing the ShortName in the CollectionAssociation group  
DsESDTGIGLA03.013.desc
4. changing a mandatory inventory attribute to optional  
DsESDTGIGLA04.013.desc
- 5 adding one or more optional inventory metadata attributes  
DsESDTGIGLA05.013.desc
6. adding one or more new qualifiers on existing events  
DsESDTGIGLA06.013.desc  
QAPercentInterpolatedData

**EXPECTED RESULTS:**

**10.2 DPLIngest**

10.2.1 Test Case 1 - Use the DPL Ingest GUI to configure Data Providers (ECS-ECSTC-871)

**DESCRIPTION:**

**PRECONDITIONS:**

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Data Providers can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[View Data Providers]</i>		#comment
3	As the 'ingest admin' operator, navigate to the Data Provider page on the Data Pool Ingest GUI.		
4	Verify that all provider information in the INGST CI database appears correctly on the Data Provider page.		
5	<i>[Configure Data Providers]</i>		#comment
6	Edit existing provider information (if any) and define new providers to fulfill the following requirements:  At least one data provider should have an FTP notification method,  at least one data provider should have an scp notification method,  at least one data provider should have an http notification method,  at least one data provider should have an email notification method,  at least one data provider should have a combination FTP/email notification method,  and at least one data provider should have a combination scp/email notification method.  and at least one data provider should have a combination http/email notification method. Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des. At least one provider must use active mode. At least one should use passive mode.		
7	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit data provider information.		
8	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing data providers before saving this information.		
9	Verify that the ingest admin operator has the authorization to define new data providers (i.e., has access to the data provider pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
10	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16110 to be entered.		
11	Verify that all of the notification methods in S-DPL-16150 can be entered or selected on the Data Pool Ingest GUI, as appropriate for the selected transfer method as per S-DPL-16110.		
12	For one provider with an FTP Notification method, verify that the Data Pool Ingest GUI allows the related FTP information to be entered (FTP host, destination directory, login id, password)		

#	Action	Expected Result	Notes
13	Verify that the FTP password entered is not shown or stored in the clear.		
14	Verify that the FTP password entered is not shown in the Data Pool Ingest GUI log.		
15	For one provider with an email notification method, verify that the Data Pool Ingest GUI allows the related email address to be entered.		
16	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new data providers before saving this information.		
17	For one provider with an scp notification method, verify that the Data Pool Ingest GUI allows the related scp information to be entered (scp host, destination directory, login id, password, scp type, and cipher).		
18	Verify that the scp password entered is not shown or stored in the clear.		
19	Verify that the scp password entered is not shown in the Data Pool Ingest GUI log.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**10.2.2 Test Case 2 - Use the DPL Ingest GUI to configure Polling Locations (ECS-ECSTC-872)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Polling Locations can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure Polling Locations]</i>		#comment
3	As the Ingest Admin operator use the Data Pool Ingest GUI to define all polling locations that will be used for testing this ticket. (Table of data provider to polling location mappings, and related S-DPL-16230 information for each polling location TBS as part of ITP).  At least one polling location should have an FTP polling method,  at least one polling		

#	Action	Expected Result	Notes
	location should have a local polling method.  and at least one polling location should have an scp polling method.  and at least one polling location should have an http polling method. At least one polling location using FTP shall be for a provider using local transfers.		
4	Verify that the ingest admin operator has the authorization to define the polling locations (i.e., has access to the polling location pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
5	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16230 to be entered.		
6	Verify that all of the polling methods in S-DPL-16250 can be entered or selected on the Data Pool Ingest GUI.		
7	Verify that the Data Pool Ingest GUI prompts the operator to confirm definitions of new polling locations before saving this information.		
8	<i>[Edit Polling Locations]</i>		#comment
9	For one polling location, as the 'ingest admin' operator, edit all of its existing configuration parameters. (NOTE: After this criterion is complete, values of these configuration parameters should be reset to appropriate values for processing all PDRs in criterion 300).		
10	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit polling location information.		
11	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing polling locations before saving this information.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**10.2.3 Test Case 3 - Use the DPL Ingest GUI to configure Remote Transfer Hosts (ECS-ECSTC-873)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Remote Transfer Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure FTP hosts]</i>		#comment
3	As the 'ingest admin' operator, navigate to the FTP Host page on the Data Pool Ingest GUI.		
4	Edit existing ftp host information (if any, from the INGST CI), per S-DPL-16260, and define new ftp hosts such that all ftp hosts which will be used for testing this ticket are defined. (Table of ftp hosts and related S-DPL-16260 information TBS as part of ITP). At least two ftp hosts should be defined that are not configured for the INGST subsystem.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to edit ftp host information.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing ftp hosts and definitions of new ftp hosts before saving this information.		
7	Verify that the ingest admin operator has the authorization to define the ftp hosts (i.e., has access to the ftp host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		
8	Verify that the Data Pool Ingest GUI allows all information in S-DPL-16260 to be entered.		
9	<i>[Configure HTTP hosts]</i>		
10	Configure a HTTP host use the step similar to 'Configure FTP hosts'.		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**10.2.4 Test Case 4 - Use the DPL Ingest GUI to configure SCP Hosts (ECS-ECSTC-874)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies SCP Hosts can be configured using the DPL Ingest GUI</i>		#comment
2	<i>[Configure hosts for scp access]</i>		#comment
3	As the 'ingest admin' operator, navigate to the host page on the Data Pool Ingest GUI.		
4	Edit existing information for hosts accessed via scp (if any, from the INGST CI), per S-DPL-16290, and define new scp hosts such that all scp hosts which will be used for testing this ticket are defined. (Table of hosts accessed via scp, and related S-DPL-16290 information TBS as part of ITP.) Scp type/cipher combinations to include in the test are:  F-secure/None;  OpenSSH/aes128;  OpenSSH/3des, i.e., the test needs to involve several different providers.		
5	Verify that the Data Pool Ingest GUI allows the 'ingest admin' operator to enter and edit the scp host information in S-DPL-16290.		
6	Verify that the Data Pool Ingest GUI prompts the operator to confirm changes to existing scp host parameters and definitions of new scp host parameters before saving this information.		
7	Verify that the ingest admin operator has the authorization to define the scp host parameters in S-DPL-16290 (i.e., has access to the host pages on the Data Pool Ingest GUI, and that information entered by this operator is stored in the database.)		

**TEST DATA:**

Any current data types

**EXPECTED RESULTS:**

**10.2.5 Test Case 5 - View and modify the DPL Ingest GUI configuration (ECS-ECSTC-875)**

**DESCRIPTION:**

**PRECONDITIONS:**

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies DPL Ingest GUI configuration can be viewed and modified correctly.</i>		#comment
2	<i>[View collection configuration]</i>		#comment
3	As an 'ingest admin' operator, use the Data Pool Ingest GUI to list all data types for which configuration parameters were entered in criterion 120.		
4	Verify that "Ignore Validation Warnings" and "Public in Data Pool" are displayed for each ESDT.		
5	Verify that the Data Pool Ingest GUI provides the operator a method to quickly select or scan for a subset of existing Data Pool collections without having to enter the full ESDT name and version.		Since release 8.2, newly installed data types will be automatically added to the DPL Ingest InDataType table and visible in 'Data Types with DPL Ingest Configurations' table.
6	<i>[Edit collection configuration]</i>		#comment
7	As an 'ingest admin' operator, use the Data Pool Ingest GUI to edit publication policy configuration parameters for a Data Pool collection.		
8	Verify that the edited configuration parameters are correctly updated in the database.		
9	<i>[Reconfigure data type parameters.]</i>		#comment
10	Log into the DPL Ingest GUI as an 'ingest admin' operator, and configure a data type (ESDT and Version) to be published in the public Data Pool upon insert.		
11	Submit several PDRs for the data type configured above. There needs to be a sufficient number of PDRs and granules such that some granules will complete ingest before the configuration change made during the test (see next step), and some granules will not start ingest until after the configuration change has been applied.		
12	After the first few granules completed ingest, re-configure the data type via the DPL Ingest GUI such that the data type no longer will be published in the Data Pool. Also change the minimum retention period sufficiently long so that the ingested granules are not cleaned up immediately after archiving, to allow time for the verification steps below.		
13	Verify that the first few granules that were ingested before the re-configuration are queued with the Data Pool Insert Service for insertion into the public Data Pool area.		

#	Action	Expected Result	Notes
14	Verify that all granules that started ingest one minute or more after the re-configuration are inserted into the non-public Data Pool area, and are NOT queued with the Data Pool Insert Service for insertion into the public Data Pool area. NOTE: if the Science granule has an associated Browse the Browse will be published even if the Science granule is &quot;hidden.&quot;		
15	Verify that ALL browse granules ingested as part of the test are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

TEST DATA:

Any current data types

EXPECTED RESULTS:

10.2.6 Test Case 8 - Cross-DAAC Ingest (ECS-ECSTC-876)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies Cross-DAAC Ingest.</i>		#comment
2	<i>[Perform Cross-DAAC Ingest.]</i>		#comment
3	This criterion may be verified by a separate test procedure or as part of the test procedure for criteria 300 to 395.		
4	This test uses the new capability that allows OMS to distribute metadata in .met file format.		
5	Identify collections that are transferred from one DAAC to another via cross DAAC ingest. Include at least two collections that represent ancillary granules and two collections that do not.		
6	Distribute at least one granule for each collection via OMS to a directory that is polled for Data Pool Ingest. Ensure that the corresponding DN includes a checksum and that the distribution includes an ODL metadata file (configure OMS accordingly).		

#	Action	Expected Result	Notes
7	Verify that the corresponding Distribution Notices include a checksum.		
8	Verify that the distribution includes an ODL metadata file (configure OMS accordingly)		
9	Verify that the granules are ingested successfully.		
10	Reconfigure OMS to distribute metadata files to the destination in XML file format.		
11	Distribute at least one granule for each collection via OMS to a directory that is polled for Data Pool Ingest. Ensure that the corresponding DN includes a checksum and that the distribution includes a XML metadata file (configure OMS accordingly).		
12	Verify that the granules are ingested successfully.		
13	Verify that the AIM inventory contains the correct metadata.		

**TEST DATA:**

Any DDIST data type

**EXPECTED RESULTS:**

**10.2.7 Test Case 13 - Verifies MISBR Ingest and linkages are handled correctly (ECS-ECSTC-877)**

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies MISBR Ingest and linkages are handled correctly.</i>		#comment
2	<i>[Ingest MISBR, MISR L1, MISR L2]</i>		#comment
3	Ingest one L1, one L2, and one MISBR which should link together.		
4	<i>Verify that the MISR granules are successfully ingested into ECS. Specifically, verify for at least one MISR Level 1 granule, one MISR Level 2 granule, and one MISBR granule that:</i>		#comment

#	Action	Expected Result	Notes
5		1. the granule information is recorded correctly in the AIM inventory database	
6		2. the granule XML metadata files are stored in XML Archive and their contents are correct	
7		3. the science data files are stored into archive	
8		4. the links between the MISR science granules and the matching MISBR granules are correctly recorded in the AIM database.	
9	Verify that ALL browse granules that were created (because the MISBR was configured as public and the MISBR matched the criteria of a L1 or L2 granule and thus generated browse granules) are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

TEST DATA:

MISR Level 1, MISR Level 2, and MISBR.005

EXPECTED RESULTS:

10.2.8 Test Case 18 - Browse granule(s) in separate PDR from science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-878)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies correct linking when ingesting a browse granule with a linkage file.</i>		#comment

---

---

#	Action	Expected Result	Notes
2	<i>[Browse granule(s) in separate PDR from science granule, where science granule is to be inserted in public Data Pool]</i>		#comment
3	Submit two PDRs, the first for ingest of a science granule, and the second for ingest of (an) associated Browse granule(s), where the science collection is configured for public Data Pool insert (e.g., AMSR Level 2). The PDRs should be submitted far enough apart in time such that the science granule is archived and queued for insertion into the public Data Pool before the browse PDR is submitted.		
4	Verify that the browse linkage information is included in the AIM metadata (database) for the science granule.		
5	Verify that the science granule(s) are queued with the Data Pool Insert Service for insertion into in the public Data Pool.		
6	Verify that the original (hdfeos) version(s) of the associated browse granule(s) are stored in the Data Pool in the hidden directory structure.		
7	Verify that the science granule(s) and the original version(s) of the associated browse granule(s) are copied to the archive.		
8	Verify that the browse file name in the archive includes the ECS id that was pre-assigned to the browse granule.		
9	Verify that the science granule(s) and the original version(s) of the associated browse granule(s) are removed from the Data Pool hidden directory structure after their ingest request completes, subject to the rules in S-DPL-18455 and S-DPL-18460.		
10	Verify that the start and completion of preprocessing operations, and all preprocessing information in S-DPL-18365, is included in the Data Pool Ingest Service application log for this request.		
11	Verify that ALL browse granules ingested as part of the test are recorded as public in the AIM schema and that the required browse JPEG files are created in the correct location in the public data pool (as recorded in the aim schema).		

TEST DATA:

Any current data types with Browse

EXPECTED RESULTS:

10.2.9 Test Case 19 - QA granule(s) in same PDR as science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-879)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies correct linking when ingesting a QA granule and science granule in the same PDR.</i>		#comment
2	<i>[QA granule(s) in same PDR as science granule, where science granule is to be inserted in public Data Pool]</i>		#comment
3	Select one successful request from the filtered list where one or more science granules in the request have associated QA granule(s) in the same PDR, and the science collection is configured to be archived and to be inserted in the public Data Pool. QA collection is configured to be inserted into public Data Pool as well.		
4	Verify that the QA linkage information is included in the AIM metadata (database) for the science granule.		
5	Verify that the science granule(s) are queued with the Data Pool Insert Service for insertion into the public Data Pool.		
6	Verify that the science granule(s) and the associated QA granule(s) are copied to the archive.		
7	Verify that the QA file name in the archive includes the ECS id that was pre-assigned to the QA granule.		
8	Verify that the science granule(s) and the QA granule(s) are removed from the Data Pool hidden directory structure after their ingest request completes, subject to the rules in S-DPL-18455.		

TEST DATA:

Any current data types with associated QA

EXPECTED RESULTS:

10.2.10 Test Case 20 - PH granule(s) in same PDR as science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-880)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies correct linking when ingesting a PH granule and a science granule in the same PDR.</i>		#comment
2	<i>[PH granule(s) in same PDR as science granule, where science granule is to be inserted in public Data Pool]</i>		#comment
3	Select one successful request from the filtered list where one or more science granules in the request have associated PH granule(s) in the same PDR, and the science collection is configured to be inserted in the public Data Pool. Assume PH will not be enabled for public Data Pool insert.		
4	Verify that the PH linkage information is included in the AIM metadata (database) for the science granule.		
5	Verify that the science granule(s) and associated PH granule(s) are stored in the Data Pool hidden directory structure.		
6	Verify that the science granule(s) and associated PH granule(s) are copied to the archive.		
7	Verify that the PH file name in the archive includes the ECS id that was pre-assigned to the PH granule.		
8	Verify that the science granule(s) and the PH granule(s) are removed from the Data Pool hidden directory structure after their ingest request completes, subject to the rules in S-DPL-18455.		

TEST DATA:

Any current data types with associated PH

EXPECTED RESULTS:

10.2.11 Test Case 21 - PH granule(s) in different PDR from science granule, where science granule is to be inserted in public Data Pool (ECS-ECSTC-881)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Description and Objectives: Verifies correct linking when ingesting a PH granule with a linkage file.</i>		#comment
2	<i>[PH granule(s) in different PDR from science granule, where science granule is to be inserted in public Data Pool]</i>		#comment
3	Submit two PDRs, the first for ingest of a science granule, and the second for ingest of (an) associated PH granule(s), where the science collection is configured for public Data Pool insert. The PDRs should be submitted far enough apart in time such that the science granule is archived and queued for insertion into the public Data Pool before the PH PDR is submitted.		
4	Verify that the PH linkage information is included in the AIM metadata (database) for the science granule.		
5	Verify that the science granule(s) and associated PH granule(s) are stored in the Data Pool hidden directory structure.		
6	Verify that the science granule(s) and associated PH granule(s) are copied to the archive.		
7	Verify that the PH file name in the archive includes the ECS id that was pre-assigned to the PH granule.		
8	Verify that the science granule(s) and the PH granule(s) are removed from the Data Pool hidden directory structure after their ingest request completes, subject to the rules in S-DPL-18455.		

TEST DATA:

Any current data types with associated PH

EXPECTED RESULTS:

### 10.3 DPL

#### 10.3.1 Test Case 3 - Publish with Theme (ECS-ECSTC-882)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Publish with Theme]</i>		#comment
2	Ingest ICEBRIDGE data.  First verify that the ThemeID value is not in the themeaname column of AmCollection table for the collection so that the theme association would not happen upon ingest.		For example IRPAR2.001
3	Run the Publish Utility. Use the -theme option to associate these granules with an existing theme. (Use the Data Pool Maintenance GUI Manage Themes tab to see the list of existing themes.)		
4	Verify that the granules were successfully inserted into the Data Pool by checking the status of the insert actions in the Data Pool database (DIInsertActionQueue).		
5	Verify that the granules were inserted into the AmGranule table.		
6	Verify, using Unix cd and ls commands, that the files for the granules were inserted into the appropriate Data Pool directories. select fs.absoluteFilePath    c.GroupId    df.DirectoryPath    df.OnlineFileName from DIFileSystems fs join AmCollection c on fs.fileSystemLabel = c.FileSystemLabel join AmGranule g on c.CollectionId = g.CollectionId join AmDataFile df on g.GranuleId = df.GranuleId where g.GranuleId = granuleid;		
7	Verify that the granules are associated with the specified theme, i.e., that appropriate rows for the granules have been inserted in the DIGranuleThemeXref table.		

TEST DATA:

Any nonECS data type

EXPECTED RESULTS:

10.3.2 OBSOLETE - Test Case 8 - Update Granule Utility (ECS-ECSTC-883)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Update Granule Utility]</i>		#comment
2	Use the Update Granule utility (EcDIUpdateGranule.pl) to update the retention priority and expiration date of nonECS granules.		
3	Verify using isql that the retention priority and expiration date for the granules was updated properly in the DIGranuleExpirationPriority table.		

TEST DATA:

nonECS data types

EXPECTED RESULTS:

10.3.3 Test Case 20 - Hidden Scrambler Utility (ECS-ECSTC-884)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Hidden Scrambler Utility]</i>		#comment
2	With Data Pool inserts shut down, use the Data Pool Hidden Scrambler utility with the -shortname/-versionid command line parameters to generate a new hidden directory name for a single collection, where there are pending orders for granules in that collection.		

#	Action	Expected Result	Notes
3	Verify that the new hidden directory is created, that all files for pending orders for the collection are moved to the new hidden directory, that all FTPPull links for existing orders referencing the old hidden directory now point to the new hidden directory, and that the old hidden directory is removed.		

**TEST DATA:**

Any current data type which is already in the DICollections table with Data Pool inserts enabled.

**EXPECTED RESULTS:**

10.3.4 Test Case 23 - DPL Maintenance GUI (ECS-ECSTC-885)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>[DPL Maintenance GUI]</i>		#comment
2	Open the DPL Maintenance GUI in a web browser.		
3	Verify that  a. A collection can be added 		
4	b. A collection can be deleted 		
5	c. A collection can be updated 		
6	d. A collection group can be added 		
7	e. A collection group can be updated 		
8	f. A theme can be added 		
9	g. A theme can be deleted 		
10	h. A theme can be updated		

**TEST DATA:**

Any current data types

EXPECTED RESULTS:

10.3.5 Test Case 34 - Un-publish Granules that were Reported by Data Pool Inventory Validation, DP\_7F\_01, Criterion 90 (ECS-ECSTC-886)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[Un-publish Granules that were Reported by Data Pool Inventory Validation, DP_7F_01, Criterion 90]</i>		#comment
2	Perform a Data Pool inventory validation that reports at least four granules as being public that are not eligible to be public as per S-DPL-08180f(listed below) and pick up the output of that run with an un-publishing run.		
3	Verify that the granules are indeed un-published, i.e., the granules are no longer flagged as public in the Data Pool inventory and their files are in the correct hidden Data Pool directory and no longer in the public Data Pool area.		
4	Verify that the start and completion of the un-publishing run are logged, as well as the un-publishing attempt/success for each of the granules reported by the Data Pool inventory validation run.		
5	Verify that the un-publishing run exits with an exit code indicating success.		
6	Verify that the next automatic BMGT export will include the URL deletions of the granules that were un-published.		
7	S-DPL-08180f - The DPL CI shall not make granules public that are currently flagged as logically deleted (i.e., have a non-NULL deleteEffectiveDate), deleted from archive (i.e., DeleteFromArchive set to 'Y'), or hidden from normal users (i.e., DeleteFromArchive set to 'H') in the AIM CI inventory.		

TEST DATA:

Any current data types

EXPECTED RESULTS:

---

### 10.3.6 Test Case 1 - Qualified Subscription Insert (ECS-ECSTC-916)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	[Qualified Subscription Insert] Insert a granule with browse into ECS for which there is an existing qualified Data Pool insert subscription.		
2	Verify that the granule and its browse file were successfully inserted into the Data Pool by checking the status of the insert action in the Data Pool database (DIInsertActionQueue).		
3	Verify, using isql, that the granule was inserted into the DIGranules table and the browse granule was inserted into the DIBrowse table.		
4	Verify, using UNIX 'cd' and 'ls' commands, that the files for the granule were inserted into the appropriate Data Pool directory and that the browse files were inserted into the appropriate Data Pool directory, and that a link to the browse file is present in the science file directory.		

TEST DATA:

EXPECTED RESULTS:

### 10.3.7 OBSOLETE - Test Case 6 - Download Granules via Web (ECS-ECSTC-917)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	[Download Granules via Web] Verify that anonymous ftp is configured for the mode in which you are testing.		
2	From outside of the firewall, use the Data Access Testbed to download the granules inserted in Test Cases 1 and 3.		

---

#	Action	Expected Result	Notes
3	Verify that the granules are correctly downloaded to the specified download directory.		

TEST DATA:

EXPECTED RESULTS:

### 10.3.8 OBSOLETE - Test Case 19 - HEG Order from Web Access (ECS-ECSTC-919)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	[HEG Order from Web Access] Insert an AST_L1B granule into the Data Pool, using the Batch Insert utility.		
2	Use the Data Pool Web Access GUI to drill down for this granule.		
3	Perform a HEG order for this granule.		
4	Verify that the band subsetting icon appears for this granule on the HEG order page (although you do not need to select it).		
5	Verify that the HEG order completes successfully, and that the HEG output files are placed in the /datapool//user//.orderdata/OUTPUTS/HEGOUT.001/HEG/. directory.		
6	Verify that the order status appears on the OMS GUI.		

TEST DATA:

EXPECTED RESULTS:

### 10.4 BMGT

#### 10.4.1 Test Case 3 - Nominal Granule Export[S-2a]: Automatic Export: Ingest Granule (ECS-ECSTC-907)

DESCRIPTION:

S	40	1	<b>[Nominal Granule Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections.		
S	40	2	For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.		

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure granule g1's collection is installed. E.g., the DPL Ingest GUI shows the collection as a configured datatype.		
3	Ensure the DPL Ingest GUI shows granule g1's collection is configured to be public on ingest.		
4	Ensure granule g1's collection is enabled for collection and granule export:  update bg_collection_configuration set		

#	Action	Expected Result	Notes
	granuleexportflag = 'Y', collectionexportflag = 'Y' where shortname = &lt;SHORTNAME&gt; and versionid = &lt;VERSIONID&gt;;		
5	Ensure ECHO has granule g1's collection metadata:  curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/\${URL_ENCODED_DATASETID}  If ECHO is missing the collection, export it:  EcBmBMGTManualStart \${MODE} --metc --collections \${SHORTNAME}.\${VERSIONID}		
6	Ensure a local copy of the ECHO 10 schema files is available for schema validation (e.g., under /tools/common/test/BE_82_01/echo10/schema).  If needed, download *.xsd files from https://api.echo.nasa.gov/ingest/schema/.		
7	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
8	Ensure the BMGT Dispatcher is running. On the BMGT host:  ps auxww   grep \$MODE   grep EcBmDispatcher  If needed, start the dispatcher:  ./EcBmBMGTDispatcherStart \$MODE		
9	Ensure the BMGT Auto driver is running. On the BMGT host:  ps auxww   grep \$MODE   grep EcBmAuto  If needed, start the Auto driver:  ./EcBmBMGTAutoStart \$MODE		
10	<i>Setup</i>		#comment
11	<i>S-2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory.</i>		#comment
12	Ingest a new granule g1 into the ECS inventory.		
13	<i>Verification</i>		#comment
14	<i>V-2 Verify that the operations in S-2, except subclauses b – d, each result in the export of one or more HTTP PUTs containing the full granule metadata.</i>		#comment
15	<i>NOTE: Each action may result in multiple distinct events being triggered. The BMGT Auto driver will consolidate these events if they are picked up within the same polling interval. If they span multiple intervals, then multiple, redundant exports could result.</i>		#comment
16	Verify the TCP proxy log one or more PUTs for granule g1.  There could be 1 PUT after the granule is archived and another when the granule is published.		
17	Verify the TCP proxy log shows that each of granule g1's PUT requests contains the full granule metadata.		

#	Action	Expected Result	Notes
18	<i>V-4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).</i>		#comment
19	Verify granule g1's exported metadata validates against the ECHO granule metadata schema: xmllint --noout --schema /path/to/Granule.xsd g1.xml or /tools/common/test/BE_82_01/bin/echo10_validate_metadata g g1.xml		
20	<i>V-5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements:  a) Visible = true  b) Orderable = true  c) InsertTime = The insert time of the granule recorded in the AIM database.  d) LastUpdate = The last update time of the granule recorded in the AIM database</i>		#comment
21	<i>Verify that the exported metadata has the following elements, using the Ruby script /tools/common/test/BE_82_01/bin/xpath:</i>		#comment
22	a) Visible = true  xpath /Granule/Visible g1.xml	&lt;Visible&gt;true&lt;/Visible&gt;	
23	b) Orderable = true  xpath /Granule/Orderable g1.xml	&lt;Orderable&gt;true&lt;/Orderable&gt;	
24	c) InsertTime = The insert time of the granule recorded in the AIM database  xpath /Granule/InsertTime g1.xml	&lt;InsertTime&gt;YYYY-MM-DDTHH:mm:SSZ&lt;/InsertTime&gt;  where the date time string return is equal to what is returned from the query to the database  select archivetime from amgranule where granuleid = &lt;GRANULEID&gt;;	
25	d) LastUpdate = The last update time of the granule recorded in the AIM database.  xpath /Granule/LastUpdate g1.xml	&lt;LastUpdate&gt;YYYY-MM-DDTHH:mm:SSZ&lt;/LastUpdate&gt;  where the date time string returned is equal to what is returned from the query  select lastupdate from amgranule where granuleid = &lt;GRANULEID&gt;;	

TEST DATA:  
See Test Case 603.

EXPECTED RESULTS:

V	40	1	Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID).		
V	40	2	Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.		
V	40	3	Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.		
V	40	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).		
V	40	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database		
V	40	6	Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255.		
V	40	7	Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule.		
V	40	8	Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent.		
V	40	9	Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed.		
V	40	10	Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule.		
V	40	11	Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs.		

	V	40	12	Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs.		
	V	40	13	Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs.		
	V	40	14	Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection.		
	V	40	15	Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected.		
	V	40	16	Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.		

10.4.2 Test Case 4 - Nominal Collection Export[S-1]: Manual Export (ECS-ECSTC-908)

DESCRIPTION:

	S	10	1	<b>[Nominal Collection Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Request the manual export of collection metadata for one of these collections.		
	S	10	2	Find collections which have ECS Metadata: a) Insert a new collection into the ECS inventory (and enable for collection export). b) Delete a collection from the ECS inventory. c) Update an existing collection. d) For a collection which is currently disabled for collection export, enable it for collection (but not granule) export.		

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure BMGT configuration is current and correct (config files, properties files, database settings, etc.).		

#	Action	Expected Result	Notes
3	Ensure a PostgreSQL prompt is available and connected to the ecs database.  /tools/postgres/current32/bin/psql -U \$USERNAME -h f4dbl03 -d ecs select public.set_search_path('AIM', MODE); show search_path;	Should list the schemas installed in the mode. Should include 'aim_&lt;MODE&gt;'	
4	Ensure test collections C1, C2 exist under /sotestdata/DROP_802/BE_82_01/Criteria/010/010_1.		
5	Ensure that the collections C1 and C2 have been installed in the mode, e.g., the DPL Ingest GUI shows them as configured datatypes.		
6	Ensure a local copy of the ECHO 10 schema files is available for validation (from https://api.echo.nasa.gov/ingest/schema).		
7	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
8	<i>Setup</i>		#comment
9	<i>S-1 [Nominal Collection Export] Find two collections with ECS metadata and which share the same short name, but have different version IDs. Request the manual export of collection metadata for one of these collections.</i>		#comment
10	Ensure that the two collections C1 and C2 have the same short name but different version IDs.		
11	Ensure collections C1 and C2 enabled for collection and granule export:  update bg_collection_configuration set collectionexportflag = 'Y', granuleexportflag = 'Y' where shortname = &lt;C1_SHORTNAME&gt; and versionid in (&lt;C1_VERSIONID&gt;, &lt;C2_VERSIONID&gt;);		
12	Request manual export of collection metadata for collection C1:  EcBmBMGTManualStart &lt;MODE&gt; --metc --collections &lt;C1_SHORTNAME&gt;.&lt;C1_VERSIONID&gt;		
13	<i>V-1 Verify that the manual export in S-1 results in a single HTTP PUT request containing the full collection metadata for the requested collection (but no other collections sharing a short name but with different version IDs).</i>		#comment
14	Verify the bg_export_error table shows no errors referring to collection C1 on or after the time of the manual export.		
15	Verify that the TCP proxy log reports a single HTTP PUT request for collection C1.		
16	Verify that the HTTP PUT request contains collection C1's full collection metadata.		
17	Verify the TCP proxy log reports no PUTs for collection C2.		

#	Action	Expected Result	Notes
18	<i>V-4 Verify that the metadata exported in S-1 and S-2, except S-2 subclause b, validates against the ECHO collection metadata schema (https://api.echo.nasa.gov/ingest/schema/Collection.xsd).</i>		#comment
19	Verify collection C1's exported metadata validates against the ECHO 10 collection schema. (https://api.echo.nasa.gov/ingest/schema/Collection.xsd).  xmllint --noout --schema Collection.xsd collection.xml		
20	<i>V-5 Verify that the metadata exported in S-1 and S-2, except S-2 subclause b has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the collection recorded in the AIM database. d) LastUpdate = The last update time of the collection recorded in the AIM database.</i>		#comment
21	<i>Use an xpath utility to verify the exported metadata has the following elements.</i>		#comment
22	a) Visible = true  xpath /Collection/Visible C1.xml	&lt;Visible&gt;true&lt;/Visible&gt;	
23	b) Orderable = false  xpath /Collection/Orderable C1.xml	&lt;Orderable&gt;false&lt;/Orderable&gt;	
24	c) InsertTime = The insert time of the collection recorded in the AIM database.  xpath /Collection/InsertTime C1.xml	&lt;InsertTime&gt;&lt;YYYY-MM-DDTHH:mm:SSZ&lt;/InsertTime&gt;   where the date time string returned is equal to what is returned from the query to the aim_&lt;MODE&gt; schema in the ecs database:  select inserttime from amcollection where shortname = &lt;C1_ShortName&gt; and versionid = &lt;C1_VersionId&gt;;	
25	d) LastUpdate = The last update time of the collection recorded in the AIM database.  xpath /Collection/LastUpdate C1.xml	&lt;LastUpdate&gt;&lt;YYYY-MM-DDTHH:mm:SSZ&lt;/LastUpdate&gt;   where the date time string returned is equal to what is returned from the query:  select lastupdate from amcollection where shortname = &lt;C1_ShortName&gt; and versionid = &lt;C1_VersionId&gt;;	

#	Action	Expected Result	Notes
26	<i>V-6 Verify that the metadata exported in S-1 and S-2 contains version numbers with no leading zeroes.</i>		#comment
27	Verify that collection C1's exported metadata contains version numbers with no leading zeroes:  xpath /Collection/VersionId C1.xml	&lt;VersionId&gt;\${VERSIONID}&lt;/VersionId&gt;	

TEST DATA:

Crit id	Crit ccr no	Test Data Description	Data Type Requirements	Metadata Requirements	Volume Requirements	Size Requirements	Data Location	Readiness Status
10-S1			2 Collections with same shortname, different versionids (C1, C2)	MOD44W.005 MOD44W.006		2 collections	/sotestdata/DROP_802/BE_82_01/Criteria/010/010_1	

EXPECTED RESULTS:

	V	10	1	Verify that the manual export in S-1 results in a single HTTP PUT request containing the full collection metadata for the requested collection (but no other collections sharing a short name but with different version IDs).		
	V	10	2	Verify that the operations in S-2 subclauses a, c, and d each result in the export of a single HTTP PUT containing the full collection metadata.		
	V	10	3	Verify that the operation in S-2 subclause b results in the export of a single HTTP DELETE, with the ID of the collection in the URL, but containing no collection metadata in the request body.		
	V	10	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclause b, validates against the ECHO collection metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Collection.xsd">https://api.echo.nasa.gov/ingest/schema/Collection.xsd</a> ).		
	V	10	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclause b has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the collection recorded in the AIM database.		

				d) LastUpdate = The last update time of the collection recorded in the AIM database.		
	V	10	6	Verify that the metadata exported in S-1 and S-2 contains version numbers with no leading zeroes.		

### 10.4.3 Nominal Granule Export[S-2b]: Automatic Export: Logically Delete Granule (ECS-ECSTC-909)

#### DESCRIPTION:

	S	40	1	<b>[Nominal Granule Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections.		
	S	40	2	For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.		

#### PRECONDITIONS:

#### STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure BMGT configuration is current and correct (config files, properties files, database settings, etc.).		

#	Action	Expected Result	Notes
3	Ensure a PostgreSQL prompt is available, connected to the ecs database.  /tools/postgres/9.2/bin/psql -U \${username} -h f4db103 -d ecs select public.set_search_path('AIM', MODE); show search_path;	Should list the schemas installed in the mode. should include 'aim_&lt;MODE&gt;';	
4	Ensure test data exists under /sotestdata/DROP_802/BE_82_01/Criteria/040/040_2_B.		
5	Ensure granule g2's collection is installed. E.g., the DPL Ingest GUI shows the collection as a configured datatype.		
6	Ensure the DPL Ingest GUI shows granule g2's collection is configured to be public on ingest.		
7	Ensure granule g2's collection is enabled for collection and granule export:  update bg_collection_configuration set granuleexportflag = 'Y', collectionexportflag = 'Y' where shortname = &lt;SHORTNAME&gt;; and versionid = &lt;VERSIONID&gt;;		
8	Ensure ECHO has granule g2's collection metadata:  curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/\${URL_ENCODED_DATASETID}  If ECHO is missing the collection, export it:  EcBmBMGTManualStart \${MODE} --metc --collections \${SHORTNAME}.\${VERSIONID}		
9	Ensure granule g2 is in AIM:  select shortname, versionid, granuleid from amgranule where localgranuleid = \${LOCALGRANULEID}		
10	Ensure ECHO has granule g2's metadata:  curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/granules/\${URL_ENCODED_GEOID}  If ECHO is missing the granule, export it:  EcBmBMGTManualStart \${MODE} --metg --granules \${GRANULEID}		
11	Ensure a local copy of the ECHO 10 schema files is available for schema validation (e.g., under /tools/common/test/BE_82_01/echo10/schema).  If needed, download *.xsd files from https://api.echo.nasa.gov/ingest/schema/.		
12	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
13	<i>Setup</i>		#comment
14	<i>S-2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export):  Logically delete a granule from the ECS inventory.</i>		#comment
15	Add granule g2's geoid to a text file (geoids.txt).		

#	Action	Expected Result	Notes
16	Logically delete granule g2 from AIM:  ./EcDsBulkDelete.pl -physical -server \${DBSERVER} -database ecs -user \${DBUSER} -password \${DBPASSWORD} -geoidfile /path/to/geoids.txt		
17	<i>V-3 Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.</i>		#comment
18	Verify the TCP proxy log shows a single DELETE for granule g2.		
19	Verify the TCP proxy log shows granule g2's delete has no body (no Content-Length header, no content following the HTTP headers).		
20	Verify the TCP proxy shows granule g2's DELETE includes the URL-encoded geoid in the URL.  Example of a DELETE request for provider EDF_DEV02 and geoid SC:MOD10A1.005:3000159312  DELETE /catalog-rest/providers/EDF_DEV02/granules/SC%3AMOD10A1.005%3A3000159312		

TEST DATA:  
 See Test Case 603.

EXPECTED RESULTS:

	V	40	1	Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID).		
	V	40	2	Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.		
	V	40	3	Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.		
	V	40	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).		
	V	40	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements:		

				a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database		
	V	40	6	Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255.		
	V	40	7	Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule.		
	V	40	8	Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent.		
	V	40	9	Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed.		
	V	40	10	Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule.		
	V	40	11	Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs.		
	V	40	12	Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs.		
	V	40	13	Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs.		
	V	40	14	Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection.		
	V	40	15	Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected.		
	V	40	16	Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.		

10.4.4 Nominal Granule Export[S-2c]: Automatic Export: Physically Delete Granule (ECS-ECSTC-910)

DESCRIPTION:

S	40	1	<b>[Nominal Granule Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections.		
S	40	2	For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.		

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure BMGT configuration is current and correct (config files, properties files, database settings, etc.).		
3	Ensure a PostgreSQL prompt is available, connected to the ecs database.  /tools/postgres/9.2/bin/psql -U \${username} -h f4db103 -d ecs />select public.set_search_path('AIM', MODE); show search_path;	Should list the schemas installed in the mode. should include 'aim_&lt;MODE&gt;';	
4	Ensure test data exists under /sotestdata/DROP_802/BE_82_01/Criteria/040/040_2_C.		

#	Action	Expected Result	Notes
5	Ensure granule g3's collection is installed. E.g., the DPL Ingest GUI shows the collection as a configured datatype.		
6	Ensure the DPL Ingest GUI shows granule g3's collection is configured to be public on ingest.		
7	Ensure granule g3's collection is enabled for collection and granule export: <code>update bg_collection_configuration set granuleexportflag = 'Y', collectionexportflag = 'Y' where shortname = &amp;lt;SHORTNAME&amp;gt; and versionid = &amp;lt;VERSIONID&amp;gt;;</code>		
8	Ensure ECHO has granule g3's collection metadata: <code>curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/\${URL_ENCODED_DATASETID}</code> If ECHO is missing the collection, export it: <code>EcBmBMGTManualStart \${MODE} --metc --collections \${SHORTNAME}.\${VERSIONID}</code>		
9	Ensure granule g3 is in AIM: <code>select shortname, versionid, granuleid from amgranule where localgranuleid = \${LOCALGRANULEID}</code>		
10	Ensure ECHO has granule g3's metadata: <code>curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/granules/\${URL_ENCODED_GEOID}</code> If ECHO is missing the granule, export it: <code>EcBmBMGTManualStart \${MODE} --metg --granules \${GRANULEID}</code>		
11	Ensure a local copy of the ECHO 10 schema files is available for schema validation (e.g., under /tools/common/test/BE_82_01/echo10/schema). If needed, download *.xsd files from <a href="https://api.echo.nasa.gov/ingest/schema/">https://api.echo.nasa.gov/ingest/schema/</a> .		
12	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
13	<i>Setup</i>		#comment
14	<i>S-2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): Physically delete a granule from the ECS inventory.</i>		#comment
15	<i>Physically delete a granule from the ECS inventory.</i>		#comment
16	Add granule g3's geoid to a text file (geoids.txt).		
17	Logically delete granule g3 from AIM: <code>./EcDsBulkDelete.pl -physical -server \${DBSERVER} -database ecs -user \${DBUSER} -password \${DBPASSWORD} -geoidfile /path/to/geoids.txt</code>		

#	Action	Expected Result	Notes
18	Unpublish granule g3: ./EcDIUnpublishStart.pl -mode \${MODE} -g \${GRANULEID}		
19	Physically delete granule g3 by running deletion cleanup and choosing the most destructive option at every opportunity. ./EcDsDeletionCleanup.pl -mode \${MODE} -user \${DBUSER} -server \${DBSERVER} -database ecs -batch 1000 -grbatch 100 -xmlbatch 1000 -databatch 10000 -logbatch 100		
20	<i>V-3 Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.</i>		#comment
21	Verify the TCP proxy log shows a single DELETE for granule g3.		
22	Verify the TCP proxy log shows granule g3's delete has no body (no Content-Length header, no content following the HTTP headers).		
23	Verify the TCP proxy shows granule g3's DELETE includes the URL-encoded geoid in the URL. Example of a DELETE request for provider EDF_DEV02 and geoid SC:MOD10A1.005:3000159312 DELETE /catalog-rest/providers/EDF_DEV02/granules/SC%3AMOD10A1.005%3A3000159312		

TEST DATA:  
See Test Case 603.

EXPECTED RESULTS:

V	40	1	Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID).		
V	40	2	Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.		
V	40	3	Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.		

V	40	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).		
V	40	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database		
V	40	6	Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255.		
V	40	7	Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule.		
V	40	8	Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent.		
V	40	9	Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed.		
V	40	10	Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule.		
V	40	11	Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs.		
V	40	12	Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs.		
V	40	13	Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs.		
V	40	14	Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection.		
V	40	15	Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected.		

	V	40	16	Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.		
--	---	----	----	--	--	--

10.4.5 Nominal Granule Export[S-2i]: Automatic Export: Publish Granule (ECS-ECSTC-911)

DESCRIPTION:

	S	40	1	<b>[Nominal Granule Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections.		
	S	40	2	For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs. n) Move a collection. o) Perform XML replacement on a granule.		

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure granule g9's collection is installed. E.g., the DPL Ingest GUI shows the collection as a configured datatype.		

#	Action	Expected Result	Notes
3	Ensure the DPL Ingest GUI shows granule g9's collection is configured to NOT be public on ingest.		
4	Ensure granule g9's collection is enabled for collection and granule export: export: update bg_collection_configuration set granuleexportflag = 'Y', collectionexportflag = 'Y' where shortname = &lt;SHORTNAME&gt; and versionid = &lt;VERSIONID&gt;;		
5	Ensure ECHO has granule g9's collection metadata: curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/\${URL_ENCODED_DATASETID} If ECHO is missing the collection, export it: EcBmBMGTManualStart \${MODE} --metc --collections \${SHORTNAME}.\${VERSIONID}		
6	Ensure granule g9 is in AIM: select shortname, versionid, granuleid from amgranule where localgranuleid = \${LOCALGRANULEID}		
7	Ensure granule g9 is not public: select publishtime from amgranule where granuleid = \${GRANULEID} If publishtime is not null, unpublish the granule: EcDIUnpublishStart.pl --mode \${MODE} -g \${GRANULEID}		
8	Ensure ECHO has granule g9's metadata: curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/granules/\${URL_ENCODED_GEOID} If ECHO is missing the granule, export it: EcBmBMGTManualStart \${MODE} --metg --granules \${GRANULEID}		
9	Ensure a local copy of the ECHO 10 schema files is available for schema validation (e.g., under /tools/common/test/BE_82_01/echo10/schema). If needed, download *.xsd files from https://api.echo.nasa.gov/ingest/schema/.		
10	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
11	Ensure the BMGT Dispatcher is running. On the BMGT host: ps auxww   grep \$MODE   grep EcBmDispatcher If needed, start the dispatcher: ./EcBmBMGTDispatcherStart \$MODE		
12	Ensure the BMGT Auto driver is running. On the BMGT host: ps auxww   grep \$MODE   grep EcBmAuto If needed, start the Auto driver: ./EcBmBMGTAutoStart \$MODE		
13	<i>Setup</i>		#comment

#	Action	Expected Result	Notes
14	<i>S-2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): i> Publish a granule in the datapool.</i>		#comment
15	Note the current time as t0.		
16	Publish granule g9:  EcDIPublishStart &lt;MODE&gt; -ecs -g \${GRANULEID}		
17	<i>V-2 Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.</i>		#comment
18	Verify the TCP proxy log shows a single PUT for granule g9 after time t0. E.g.,  PUT /catalog-rest/providers/\${PROVIDERID}/granules/SC%3MYD14.005%3A123456 HTTP/1.1		
19	Verify the TCP proxy log shows that the PUT request includes granule g9's full metadata.		
20	<i>V-4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema (https://api.echo.nasa.gov/ingest/schema/Granule.xsd).</i>		#comment
21	Verify granule g9's exported metadata validates against the ECHO granule metadata schema:  xmllint --noout --schema /path/to/Granule.xsd g9.xml or /tools/common/test/BE_82_01/bin/echo10_validate_metadata g g9.xml		
22	<i>V-5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database</i>		#comment
23	<i>Verify that the exported metadata has the following elements, using the Ruby script /tools/common/test/BE_82_01/bin/xpath:</i>		#comment
24	a) Visible = true  xpath /Granule/Visible g9.xml	&lt;Visible&gt;true&lt;/Visible&gt;	
25	b) Orderable = true  xpath /Granule/Orderable g9.xml	&lt;Orderable&gt;true&lt;/Orderable&gt;	
26	c) InsertTime = The insert time of the granule recorded in the AIM database  xpath /Granule/InsertTime g9.xml	&lt;InsertTime&gt;YYYY-MM-DDTHH:mm:SSZ&lt;/InsertTime&gt;   where the date time string return is equal to what is returned from the query to the database  select archivetime from	

#	Action	Expected Result	Notes
		amgranule where granuleid = \${GRANULEID};	
27	d) LastUpdate = The last update time of the granule recorded in the AIM database.  xpath /Granule/LastUpdate g9.xml	&lt;LastUpdate&gt;YYYY-MM-DDTHH:mm:ssZ&lt;/LastUpdate&gt; ;  where the date time string returned is equal to what is returned from the query  select lastupdate from amgranule where granuleid = \${GRANULEID};	
28	<i>V-10 Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule.</i>		#comment
29	Verify that granule g9's exported metadata contains URLs for its science file(s):  xpath '//OnlineAccessURL/URL/text()' granule.xml		
30	Verify the URL to the science file is valid by downloading it and comparing it to the science file in the data pool:  curl -O \${SCIENCE_URL}  diff \${SCIENCE_FILE} /datapool/\${MODE}/user\${SCIENCE_URL_PATH}  where \${SCIENCE_URL_PATH} is the part of the \${SCIENCE_URL} after http://\${HOST}.		
31	Verify that granule g9's exported metadata contains URLs for its metadata and associated ancillary file(s) (browse, QA, PH, HDF Map):  xpath '//OnlineResource/URL/text()' granule.xml		
32	Verify each OnlineResource URL is valid by downloading it and comparing it to the file in the data pool:  curl -O \${URL}  diff \${DOWNLOADED_FILE} /datapool/\${MODE}/user\${URL_PATH}  where \${URL_PATH} is the part of the \${URL} after http://\${HOST}.		

TEST DATA:  
See Test Case 603.

EXPECTED RESULTS:

V	40	1	Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID).		
V	40	2	Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.		
V	40	3	Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.		
V	40	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).		
V	40	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database		
V	40	6	Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255.		
V	40	7	Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule.		
V	40	8	Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent.		
V	40	9	Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed.		
V	40	10	Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule.		
V	40	11	Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs.		

V	40	12	Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs.		
V	40	13	Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs.		
V	40	14	Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection.		
V	40	15	Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected.		
V	40	16	Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.		

10.4.6 Nominal Granule Export[S-2j]: Automatic Export: Unpublish Granule (ECS-ECSTC-912)

DESCRIPTION:

S	40	1	<b>[Nominal Granule Export]</b> Find two collections with ECS metadata and which share the same short name, but have different version IDs. Ensure that both collections are enabled for collection and granule export. Request the manual export of granule metadata for all granules in one of these collections.		
S	40	2	For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): a) Ingest a new granule into the ECS inventory. b) Logically delete a granule from the ECS inventory. c) Physically delete a granule from the ECS inventory. d) DFA a granule. e) Hide a granule. f) Restrict a granule. g) Unrestrict a granule. h) Perform a QAUpdate on a granule. i) Publish a granule in the datapool. j) Unpublish a granule in the datapool. k) Link a granule to a browse granule. l) Unlink a granule from a browse granule. m) Change the collection to which a granule belongs.		

				n) Move a collection. o) Perform XML replacement on a granule.		
--	--	--	--	---	--	--

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure granule g10's collection is installed. E.g., the DPL Ingest GUI shows the collection as a configured datatype.		
3	Ensure the DPL Ingest GUI shows granule g10's collection is configured to be public on ingest.		
4	Ensure granule g10's collection is enabled for collection and granule export:  update bg_collection_configuration set granuleexportflag = 'Y', collectionexportflag = 'Y' where shortname = &lt;SHORTNAME&gt;&lt;br />and versionid = &lt;VERSIONID&gt;;		
5	Ensure ECHO has granule g10's collection metadata:  curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/\${URL_ENCODED_DATASETID}  If ECHO is missing the collection, export it:  EcBmBMGTManualStart \${MODE} --metc --collections \${SHORTNAME}.\${VERSIONID}		
6	Ensure granule g10 is in AIM:  select shortname, versionid, granuleid from amgranule where localgranuleid = \${LOCALGRANULEID}		
7	Ensure granule g10 is public:  select publishtime from amgranule where granuleid = \${GRANULEID}  If publishtime is null, publish the granule:  EcDIPublishUtilityStart \${MODE} -ecs -g \${GRANULEID}		
8	Ensure ECHO has granule g10's metadata:  curl -k -H Echo-Token:\${TOKEN} https://testbed.echo.nasa.gov/catalog-rest/providers/\${PROVIDERID}/granules/\${URL_ENCODED_GEOID}  If ECHO is missing the granule, export it:  EcBmBMGTManualStart \${MODE} --metg --granules \${GRANULEID}		
9	Ensure a local copy of the ECHO 10 schema files is available for schema validation (e.g., under /tools/common/test/BE_82_01/echo10/schema).<br		

#	Action	Expected Result	Notes
	/> If needed, download *.xsd files from https://api.echo.nasa.gov/ingest/schema/.		
10	Ensure a mock ECHO or TCP proxy is capturing BMGT traffic.		
11	Ensure the BMGT Dispatcher is running. On the BMGT host:  ps auxww   grep \$MODE   grep EcBmDispatcher  If needed, start the dispatcher:  ./EcBmBMGTDispatcherStart \$MODE		
12	Ensure the BMGT Auto driver is running. On the BMGT host:  ps auxww   grep \$MODE   grep EcBmAuto  If needed, start the Auto driver:  ./EcBmBMGTAutoStart \$MODE		
13	<i>Setup</i>		#comment
14	<i>S-2 For one of the collections in S-1 (or another collection which has ECS metadata, and is enabled for collection and granule export): j) Unpublish a granule in the datapool.</i>		#comment
15	Unpublish granule g10:  EcDIUnpublishStart.pl -mode \${MODE} -g \${GRANULEID}		
16	<i>V-2 Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.</i>		#comment
17	Verify the TCP proxy log shows a single PUT for granule g10 after time t0. E.g.,  PUT /catalog- rest/providers/\${PROVIDERID}/granules/SC%3MYD14.005%3A123456 HTTP/1.1		
18	Verify the TCP proxy log shows that the PUT request includes granule g10's full metadata.		
19	<i>V-4 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema (https://api.echo.nasa.gov/ingest/schema/Granule.xsd).</i>		#comment
20	Verify granule g10's exported metadata validates against the ECHO granule metadata schema:  xmllint --noout --schema /path/to/Granule.xsd g10.xml or  >/tools/common/test/BE_82_01/bin/echo10_validate_metadata g g10.xml		
21	<i>V-5 Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database</i>		#comment
22	<i>Verify that the exported metadata has the following elements, using the Ruby script /tools/common/test/BE_82_01/bin/xpath:</i>		#comment

#	Action	Expected Result	Notes
23	a) Visible = true  xpath /Granule/Visible g10.xml	&lt;Visible&gt;true&lt;/Visible&gt;	
24	b) Orderable = true  xpath /Granule/Orderable g10.xml	&lt;Orderable&gt;true&lt;/Orderable&gt;	
25	c) InsertTime = The insert time of the granule recorded in the AIM database  xpath /Granule/InsertTime g10.xml	&lt;InsertTime&gt;YYYY-MM-DDTHH:mm:SSZ&lt;/InsertTime&gt;   where the date time string return is equal to what is returned from the query to the database  select archivetime from amgranule where granuleid = \${GRANULEID};	
26	d) LastUpdate = The last update time of the granule recorded in the AIM database.  xpath /Granule/LastUpdate g10.xml	&lt;LastUpdate&gt;YYYY-MM-DDTHH:mm:SSZ&lt;/LastUpdate&gt;   where the date time string returned is equal to what is returned from the query  select lastupdate from amgranule where granuleid = \${GRANULEID};	
27	<i>V-11 Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs.</i>		#comment
28	Verify granule g10's exported metadata contains no OnlineAccessURLs:  xpath '//OnlineAccessURL' g8.xml		
29	Verify granule g10's exported metadata contains no OnlineResource URLs, except BROWSE:  xpath '//OnlineResource' g8.xml &gt; all.txt xpath '"//OnlineResource[Type='BROWSE']&quot;' g8.xml &gt; browse.txt  Verify no differences are found: diff all.txt browse.txt		

TEST DATA:  
See Test Case 603.

EXPECTED RESULTS:

V	40	1	Verify that the manual export in S-1 results in multiple HTTP PUT requests containing the full granule metadata for each granule in the requested collection (but no granules belonging to other collections sharing a short name but with a different version ID).		
V	40	2	Verify that the operations in S-2, except subclauses b – d each result in the export of a single HTTP PUT containing the full granule metadata.		
V	40	3	Verify that the operations in S-2 subclauses b – d each result in the export of a single HTTP DELETE, with the ID of the granule in the URL, but containing no granule metadata in the request body.		
V	40	4	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d, validates against the ECHO granule metadata schema ( <a href="https://api.echo.nasa.gov/ingest/schema/Granule.xsd">https://api.echo.nasa.gov/ingest/schema/Granule.xsd</a> ).		
V	40	5	Verify that the metadata exported in S-1 and S-2, except S-2 subclauses b – d has the following elements: a) Visible = true b) Orderable = false c) InsertTime = The insert time of the granule recorded in the AIM database. d) LastUpdate = The last update time of the granule recorded in the AIM database		
V	40	6	Verify that the metadata generated by the action in S-2 subclause e contains a Restriction Flag value of 255.		
V	40	7	Verify that the metadata generated by the action in S-2 subclause f contains the restriction flag value that was set in the database for that granule.		
V	40	8	Verify that the metadata generated by the action in S-2 subclause g (and any other subclauses for which the affected granule did not have a restriction flag set) the restriction flag element is absent.		
V	40	9	Verify that the metadata generated by the action in S-2 subclause h contains the QA values reflecting the update which was performed.		
V	40	10	Verify that the metadata generated by the action in S-2 subclause i contains URLs for the science and metadata files, as well as any other ancillary files (e.g. browse) associated with the affected granule.		
V	40	11	Verify that the metadata generated by the action in S-2 subclause j contains no datapool URLs.		

V	40	12	Verify that the metadata generated by the action in S-2 subclause k contains a browse linkage URL, in addition to any datapool URLs.		
V	40	13	Verify that the metadata generated by the action in S-2 subclause l contains no browse linkage URLs.		
V	40	14	Verify that the metadata generated by the action in S-2 subclause m contains a reference to the newly assigned collection.		
V	40	15	Verify that the action performed in S-2 subclause n results in the export of full granule metadata for every granule in the collection affected.		
V	40	16	Verify that the metadata generated by the action in S-2 subclause o contains XML reflecting the replacement which was performed.		

10.4.7 Test Case 23 - Incremental Verification - Nominal (ECS-ECSTC-913)

DESCRIPTION:

S	520	1	<b>[Incremental Verification – Nominal]</b> Choose an ECS inventory that includes at least one hundred thousand (100,000) science granules that are eligible for ECHO export and covers at least three different collections. Configure a time period increment for automatic verification that is not in excess of one month and no less than one week. Configure the maximum incremental export operation size such that none of the incremental verifications performed during the test will encounter the export size limit, i.e., the number of inserted/updated granules within the time increment never exceeds the maximum export size. Ensure that all collections in the mode are completely unverified, resetting the verification status if necessary.		
S	520	2	Initiate incremental verification repeatedly until there are no more granules to verify. Ensure that no other operations are going on in the system which would cause the update of any granules. Ensure that the incremental start utility requires the ECS mode as a command line option.  It is acceptable to use a cron job or script to automate the initiation of incremental verification		
S	520	3	Once incremental verification has completed and verified the entire inventory, perform some granule inserts and updates. Pause automatic export before making these updates so that the events are not automatically exported. Ensure that at least some of the updated granules are each of the following: a) In the public datapool b) Have browse links		

				c) Are restricted.		
	S	520	4	Initiate another incremental verification export.		
	S	520	5	Resume Automatic export and allow it to pick up and export the events in S-3.		
	S	520	6	Initiate another incremental verification export.		

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	<i>Pre-Conditions</i>		#comment
2	Ensure no activity other than the test will occur in the mode for the duration of the test.		
3	Ensure BMGT configuration is current and correct (config files, properties files, database settings).		
4	Ensure a PostgreSQL prompt is available and connected to the ecs database.		
5	Ensure test data is available under /sotestdata/DROP_802/BE_82_01/Criteria/520.		
6	Ensure collections C1..C10 are installed. E.g., ensure the DPL Ingest GUI shows C1..C10 as configured datatypes.		
7	Ensure collections C1..C10 are enabled for collection and granule export:  update bg_collection_configuration set collectionexportflag = 'Y', granuleexportflag = 'Y' where (shortname = '&lt;C1_SHORTNAME&gt;' and versionid = &lt;C1_VERSIONID&gt;) or (shortname = '&lt;C2_SHORTNAME&gt;' and versionid = &lt;C2_VERSIONID&gt;) ... or (shortname = '&lt;C10_SHORTNAME&gt;' and versionid = &lt;C10_VERSIONID&gt;)		
8	Ensure collections C1..C1 exist in ECHO.		
9	Ensure all test granules exist in ECHO.		
10	Update bg_collection_status set lastupdateverified = now() where collectionid in ( )		
11	Ensure a TCP proxy or mock ECHO is capturing BMGT traffic.		
12			
13	<i>Setup</i>		#comment

#	Action	Expected Result	Notes
14	<i>S-1 Choose an ECS inventory that includes at least one hundred thousand (100,000) science granules that are eligible for ECHO export and covers at least three different collections. Configure a time period increment for automatic verification that is not in excess of one month and no less than one week. Configure the maximum incremental export operation size such that none of the incremental verifications performed during the test will encounter the export size limit, i.e., the number of inserted/updated granules within the time increment never exceeds the maximum export size. Ensure that all collections in the mode are completely unverified, resetting the verification status if necessary.</i>		#comment
15	Ensure that the collections C1..C10 have a large number of granules in AIM (1000 in the EDF; about 100 000 in the PVC)		
16	Configure BMGT.Incremental.Duration = 10 ( the interval is in days).		
17	Configure BMGT.Verification.MaxGranules to 1000 in the EDF or 10 000 in the PVC. Also set maxgranulestoverify = 50 per collection in bg_collection_configuration table		
18	In the BMGT GUI, reset verification status for all collections to be unverified or 0%.		
19	<i>V-1 Inspect the verification report in the BMGT GUI before performing any incremental verification, and verify that the overall verification percentage as well as that for each collection and group, is equal to 0%.</i>		#comment
20	In the BMGT GUI System Status tab, verify that the overall verification percentage is displayed as 0%		
21	In the BMGT GUI System Status tab, verify that verification percentage is displayed as 0% for each collection and group in the mode		
22	<i>S-2 Initiate incremental verification repeatedly until there are no more granules to verify.</i>		#comment
23	Ensure that no other operations are going on in the system which would cause the update of any granules.		
24	Ensure that the incremental start utility requires the ECS mode as a command line option. It is acceptable to use a cron job or script to automate the initiation of incremental verification.		
25	Ensure that there are no ingest, delete, or update operations occurring in the mode		
26	Ensure the TCP proxy log will have only requests exported during this test:  Stop the TCP proxy. Move the log to a new name. Start the proxy.		

#	Action	Expected Result	Notes
27	./EcBmBMGTManualStart &lt;MODE&gt; --incremental		
28	<i>V-2 After each of the first two incremental iterations, inspect the logs and export request queue to ensure that all granules whose last update falls within the time period covered by the iteration were added to the request queue.</i>		#comment
29	Verify that for the first two incremental iterations, any granule that falls between min(lastupdateverified) and min(lastupdateverified) in bg_collection_status table + 10 * (BMGT.Incremental.Duration) in bg_configuration_property		
30	Verify that after running multiple incremental iterations, each of the granules in each of the Collections C1..C10 in the mode &lt;MODE&gt; have been added to the export request queue - check bg_export_request table.		
31	<i>V-3 Inspect the log file to verify that for each Incremental verification initiation, no more than the configured number of granules is enqueued.</i>		#comment
32	Verify that for each incremental iteration, the total number of granules is less than or equal to BMGT.Verification.MaxGranules		
33	Verify that for each incremental iteration, the total number of granules for each collection is less than or equal to 10 * bg_collection_configuration.maxgranulestoverify for the given collection.		
34	<i>V-4 Verify that when each verification process is started, a message is printed to the log, followed by another message when all requests have been added to the queue. Verify that the logs are written to the conventional ECS location (i.e. under /usr/ecs/&lt;MODE&gt;/CUSTOM/logs) and indicate the time at which the verification export driver started and completed, as well as how many items were enqueued for export, how many were added per collection, and the time span represented by the update times of the added granules.</i>		#comment
35	Verify that the bmgmt log is written under /usr/ecs/&lt;MODE&gt;/CUSTOM/logs/EcBmBMGTManualDriver.log		
36	Verify that the bmgmt log indicates the time when the incremental verification was started.		
37	Verify that the bmgmt log indicates the time when all request have been added to the queue		
38	Verify that the bmgmt log indicated the time when verification export was started.		
39	Verify that the bmgmt log indicated the time when verification export was completed.		

#	Action	Expected Result	Notes
40	Verify that the bmgmt log indicates the time span of the verified granules in the iteration.		
41	<i>&lt;i&gt;V-5 Inspect the verification report in the BMGT GUI after the first two incremental verification iterations to verify that the overall verification percentage increases as well as that for the collections and groups which were selected for export.&lt;/i&gt;</i>		#comment
42	Verify the overall verification percentage increases after the first two iterations		
43	Verify the verification percentage for collections C1 - C10 increases after the first two iterations		
44	Verify the overall verification percentage = 100% after running multiple incremental iterations		
45	Verify that the verification percentage for Collections C1..C10 = 100% after running multiple incremental iterations.		
46	<i>&lt;i&gt;V-7 Verify that when verification is complete, in S-2, across all incremental exports, every eligible granule has been exported exactly once. This can be done at a coarse level, ensuring that the number of exports per collection is as expected.&lt;/i&gt;</i>		#comment
47	Save the TCP proxy log, and start a new one:  Stop the TCP proxy. Move the log to a new name. Start the proxy.		
48	Save eligible granule counts to a file:  select count(g.granuleid), esdt(g.shortname, g.versionid) from amgranule g join DsGeESDTCConfiguredType t on (g.shortname = t.configuredname  and g.versionid = t.versionid) join bg_collection_configuration bcc on g.collectionid = bcc.collectionid where g.deleteeffectivedate is null and g.deletefromarchive != 'Y' and t.esdtstate = 'installed' and bcc.collectionexportflag = 'Y' and bcc.granuleexportflag = 'Y' group by g.shortname, g.versionid order by g.shortname, g.versionid		
49	Save the exported granule counts to a file:  sed -n 's/^PUT \[^[^*]*\]*/granules\SC%3A\[([^\%]*\)%3A.* \1/p' tcp.log   sort   uniq -c &gt; exported_counts.txt		
50	Verify the number of granules eligible for export per collection matches the number found in the TCP proxy log:  diff -w eligible_counts.txt exported_counts.txt		
51	<i>&lt;i&gt;V-11 Verify that each of the verification export requests results in exactly one export to ECHO (or an ECHO stand-in). Each Export shall take the form</i>		#comment

#	Action	Expected Result	Notes
	of a single HTTP PUT request containing the full granule metadata and an HTTP query parameter which indicates that the request is for verification purposes.		
52	Verify that the TCP proxy shows that each verification export is a single HTTP PUT request.		
53	Verify that the TCP proxy shows that the request has the HTTP query parameter "semantic_diff=true" indicating it is for verification purposes.		
54	<i>V-12 Verify that each incremental interval also queues and exports for each collection which has granules included in the incremental interval, the export of the associated collection metadata. This export shall take the form of a single HTTP PUT request per collection, containing the full collection metadata and an HTTP query parameter indicating that the request is for verification purposes.</i>		#comment
55	Verify that the TCP proxy shows a HTTP PUT request with Collection metadata for Collections C1..C10.		
56	Verify that the TCP proxy shows that the request is for verification purposes.		
57	<i>S-3 Once incremental verification has completed and verified the entire inventory, perform some granule inserts and updates. Pause automatic export before making these updates so that the events are not automatically exported. Ensure that at least some of the updated granules are each of the following:  a) In the public datapool  b) Have browse links  c) Are restricted.</i>		#comment
58	Pause the EVENT queue via the BMGT GUI, so requests are queued and in PENDING state and will not be picked up by incremental verification.		
59	<i>a) In the public datapool</i>		#comment
60	Ingest Science granules g1..g10 into Collection C1 with default publishing on.		
61	<i>b) Have browse links</i>		#comment
62	Ingest Science granules g11..g20 into Collection C2 with default publishing on.		
63	<i>c) Are restricted.</i>		#comment
64	Choose or create a restriction flag:  select * from dsmdrestrictionflag -OR- insert into dsmdrestrictionflag values(128, 'BE_82_01 Crit 520')		
65	Add the restriction flag to granules g21, g22:  insert into dsmdgranulerestriction values(&lt;GRANULE_ID&gt;, 128)		

#	Action	Expected Result	Notes
66	<i>S-4 Initiate another incremental verification export.</i>		#comment
67	./EcBmBMGTManualStart -mode &lt;MODE&gt; --incremental		
68	<i>V-8 Verify that the export attempt in S-4 results in a warning message indicating that there are no granules eligible for incremental verification and that no granules are added to the queue.</i>		#comment
69	Verify that there are no granules queued for verification export		
70	<i>S-5 Resume Automatic export and allow it to pick up and export the events in S-3.</i>		#comment
71	Resume the Event queue in the BMGT GUI.		
72	Verify that each of the granules in each of the g1..g10 in Collection C1 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
73	Verify that the bmgt logs also include the export of each of the granules in C1..C10 in Collection C1.		
74	Verify that each of the granules in each of the g11..g20 in Collection C2 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
75	Verify that the bmgt logs also include the export of each of the granules in g11..g20 in Collection C2.		
76	Verify that each of the granules in each of the g21..g22 in Collection C3 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
77	Verify that the bmgt logs also include the export of each of the granules in g21..g22 in Collection C3.		
78	<i>S-6 Initiate another incremental verification export.</i>		#comment
79	./EcBmBMGTManualStart &lt;MODE&gt; --incremental		
80	<i>V-2 After each of the first two incremental iterations, inspect the logs and export request queue to ensure that all granules whose last update falls within the time period covered by the iteration were added to the request queue.</i>		#comment
81	Verify that each of the granules in each of the g1..g10 in Collection C1 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		
82	Verify that the bmgt logs also include the export of each of the granules in C1..C10 in Collection C1.		
83	Verify that each of the granules in each of the g11..g20 in Collection C2 have been added to the export request queue. - check bg_export_request table or the bmgt GUI		

#	Action	Expected Result	Notes
84	Verify that the bmgmt logs also include the export of each of the granules in C11..C20 in Collection C2.		
85	Verify that each of the granules in each of the g21..g22 in Collection C3 have been added to the export request queue. - check bg_export_request table or the bmgmt GUI		
86	Verify that the bmgmt logs also include the export of each of the granules in g21..g22 in Collection C3.		
87	<i>V-3 Inspect the log file to verify that for each Incremental verification initiation, no more than the configured number of granules is enqueued.</i>		#comment
88	Verify that for each incremental iteration, the total number of granules is less than or equal to BMGT.Verification.MaxGranules		
89	Verify that for each incremental iteration, the total number of granules for each collection is less than or equal to bg_collection_configuration.maxgranulestoverify for the given collection.		
90	<i>V-4 Verify that when each verification process is started, a message is printed to the log, followed by another message when all requests have been added to the queue. Verify that the logs are written to the conventional ECS location (i.e. under /usr/ecs/&lt;MODE&gt;/CUSTOM/logs) and indicate the time at which the verification export driver started and completed, as well as how many items were enqueued for export, how many were added per collection, and the time span represented by the update times of the added granules.</i>		#comment
91	Verify that the bmgmt log is written under /usr/ecs/&lt;MODE&gt;/CUSTOM/logs/EcBmBMGTManualDriver.log		
92	Verify that the bmgmt log indicates the time when the incremental verification was started.		
93	Verify that the bmgmt log indicates the time when all request have been added to the queue		
94	Verify that the bmgmt log indicated the time when verification export was started.		
95	Verify that the bmgmt log indicated the time when verification export was completed.		
96	Verify that the bmgmt log indicates the time span of the verified granules in the cycle.		
97	<i>V-5 Inspect the verification report in the BMGT GUI after the first two incremental verification iterations to verify that the overall verification		#comment

#	Action	Expected Result	Notes
	percentage increases as well as that for the collections and groups which were selected for export.</i>		
98	Verify the overall verification percentage = 100%		
99	Verify that the verification percentage for Collections C1..C10 = 100%		
100	<i>V-9 Verify that the export attempt in S-6 results in the queueing and export of the granules which were updated in S-3</i>		#comment
101	<i>V-10 Verify that the bodies of the verification exports in S-6 are exactly the same as the bodies of the automatic exports in S-5, including datapool URLs and restriction flags. Note that it is allowable for the verification to include additional exports not in the automatic export, e.g. the collection metadata for all collections for which there is a granule export.</i>		#comment
102	Compare the granule metadata in the tcp log between the automatic export (S-5) and verification export (S-6) to verify that they are identical.  NOTE: The verification exports will have collection exports for any granule exports in the collection.		
103	If there are extra collection exports in the verification, these can be ignored.		
104	<i>V-11 Verify that each of the verification export requests results in exactly one export to ECHO (or an ECHO stand-in). Each Export shall take the form of a single HTTP PUT request containing the full granule metadata and an HTTP query parameter which indicates that the request is for verification purposes.</i>		#comment
105	Verify that the TCP proxy shows that each verification export is a single HTTP PUT request.		
106	Verify that the TCP proxy shows that the request has the HTTP query parameter &quot;semantic_diff=true&quot; indicating it is for verification purposes.		
107	<i>V-12 Verify that each incremental interval also queues and exports for each collection which has granules included in the incremental interval, the export of the associated collection metadata. This export shall take the form of a single HTTP PUT request per collection, containing the full collection metadata and an HTTP query parameter indicating that the request is for verification purposes.</i>		#comment
108	Verify that the TCP proxy shows a HTTP PUT request with Collection metadata for Collections C1...C10.		
109	Verify that the TCP proxy shows that the request is for verification purposeses (the url will contain a query parameter &quot;xmldiff=true&quot;).		

#	Action	Expected Result	Notes
110	<i>V-13 Verify that the database and the log files contain information on the process of each request through the system such that it is possible to identify when the metadata was generated, when the export was sent to ECHO, and when the response was received, etc.</i>		#comment
111	Verify that the bmgmt logs and database show when the request was generated for each verification request.		
112	Verify that the bmgmt logs and database show when the metadata was generated for each verification request.		
113	Verify that the bmgmt logs and database show when the export was sent to ECHO.		
114	Verify that the bmgmt logs and database show when the response was received for each verification request.		
115	<i>V-14 Verify that the BMGT GUI displays the completed export events, indicating that they were the result of an incremental verification export. Verify that it indicates that the requests were successfully exported and indicates the time of export as well as granule or ID.</i>		#comment
116	In the BMGT GUI export request tab, filter the request to view only those items on the INCR queue		
117	Verify that each request is listed in the SUCCESS state.		
118	Verify that each request has the associated granule or collection id listed.		
119	Verify that each request has its completion time listed.		
120	<i>V-15 Verify that the BMGT GUI displays the following incremental verification metrics overall, or for a particular time frame:  a) Number of collections/granules which were exported for verification.  b) Number of collections/granules which were successfully verified  c) Number of collections/granules which failed verification but were automatically repaired.  d) Number of collections/granules which failed verification and could not be automatically repaired.  e) Number of collections/granules skipped during export due to errors.</i>		#comment
121	In the BMGT GUI Export Request tab, filter to view only requests on the INCR queue.		
122	select the &quot;Batch Job Summary&quot; sub tab		
123	Verify that the &quot;Batch Job Summary&quot; sub tab lists for each incremental iteration, or batch, the number of items exported for verification.		
124	Verify that the &quot;Batch Job Summary&quot; sub tab lists for each incremental iteration, or batch, the number of items successfully verified (success column).		

#	Action	Expected Result	Notes
125	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items which failed verification, but were automatically repaired by ECHO (warning colum).		
126	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items which failed verification, but were not automatically repaired by ECHO.		
127	Verify that the "Batch Job Summary" sub tab lists for each incremental iteration, or batch, the number of items which were skipped.		
128	Ingest granules for any test collection		
129	Run incremental verification with the -metg option . ./EcBmBMGTManualStart -mode --incremental		
130	Verify that only granules metadata is exported.		
131	Verify that no collection metadata is exported		

TEST DATA:

Cri t id	Cri t ccr no	Test Data Descriptio n	Data Type Requirement s	Metadata Requirement s	Volume Requirement s	Size Requirement s	Data Location	Readines s Status
520			10 collections (C1..C10) with at least a total of 1000 granules				/sotestdata/DROP_802/BE_82_01/Criteria/520/README.txt	

EXPECTED RESULTS:

	V	520	1	Inspect the verification report in the BMGT GUI before performing any incremental verification, and verify that the overall verification percentage as well as that for each collection and group, is equal to 0%.		
	V	520	2	After each of the first two incremental iterations, inspect the logs and export request queue to ensure that all granules whose last update falls within the time period covered by the iteration were added to the request queue.		

V	520	3	Inspect the log file to verify that for each Incremental verification initiation, no more than the configured number of granules is enqueued.		
V	520	4	Verify that when each verification process is started, a message is printed to the log, followed by another message when all requests have been added to the queue. Verify that the logs are written to the conventional ECS location (i.e. under /usr/ecs/<MODE>/CUSTOM/logs) and indicate the time at which the verification export driver started and completed, as well as how many items were enqueued for export, how many were added per collection, and the time span represented by the update times of the added granules.		
V	520	5	Inspect the verification report in the BMGT GUI after the first two incremental verification iterations to verify that the overall verification percentage increases as well as that for the collections and groups which were selected for export.		
V	520	6	Inspect the verification report in the BMGT GUI after the final incremental verification iteration to verify that the overall verification percentage, as well as that of each enabled collection and group, is 100%.		
V	520	7	Verify that when verification is complete, in S-2, across all incremental exports, every eligible granule has been exported exactly once. This can be done at a coarse level, ensuring that the number of exports per collection is as expected.		
V	520	8	Verify that the export attempt in S-4 results in a warning message indicating that there are no granules eligible for incremental verification and that no granules are added to the queue.		
V	520	9	Verify that the export attempt in S-6 results in the queueing and export of the granules which were updated in S-3		
V	520	10	Verify that the bodies of the verification exports in S-6 are exactly the same as the bodies of the automatic exports in S-5, including datapool URLs and restriction flags.		
V	520	11	Verify that each of the verification export requests results in exactly one export to ECHO (or an ECHO stand-in). Each Export shall take the form of a single HTTP PUT request containing the full granule metadata and an HTTP query parameter which indicates that the request is for verification purposes.		
V	520	12	Verify that each incremental interval also queues and exports for each collection which has granules included in the incremental interval, the export of the associated collection metadata. This export shall take the form of a single HTTP PUT request per collection, containing the full collection metadata and an HTTP query parameter indicating that the request is for verification purposes.		

V	520	13	Verify that the database and the log files contain information on the process of each request through the system such that it is possible to identify when the metadata was generated, when the export was sent to ECHO, and when the response was received, etc.		
V	520	14	Verify that the BMGT GUI displays the completed export events, indicating that they were the result of an incremental verification export. Verify that it indicates that the requests were successfully exported and indicates the time of export as well as granule or ID.		
V	520	15	Verify that the BMGT GUI displays the following incremental verification metrics overall, or for a particular time frame: a) Number of collections/granules which were exported for verification. b) Number of collections/granules which were successfully verified c) Number of collections/granules which failed verification but were automatically repaired. d) Number of collections/granules which failed verification and could not be automatically repaired. e) Number of collections/granules skipped during export due to errors.		

**10.5 EMS**

10.5.1 Test Case 1 - EMS Extract Utility: Verify that flat files are created and exported for all datasets when Extract utility is run using default configuration (ECS-ECSTC-894)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	[EMS Extract Utility: Verify that flat files are created and exported for all datasets when Extract utility is run using default configuration]		#comment
2	L4 Requirement 10		#comment
3	The following datasets are extracted and exported using the extraction utility: a) Product attribute metadata: Meta b) Product attribute search: searchExp c) Data Ingest: Ing d) Data Archive: Arch e) Physical media		

#	Action	Expected Result	Notes
	distribution orders DistFTP (DataPool, FtpPush, FtpPull, Scp) DistHTTP (DataPool)		
4	Note: add a test for extract type ArchDel . EMS script -d option.  Before running this test, set the initial date that the default configuration of the utility should begin processing. See Initialize default running of EMS extract utility at end of document. A possible set of the initial default datasets could be one week earlier than the current date.		
5	Run the following EMS extract utility command with or without the -v flag (verbose):  EcDbEMSdataExtractor.pl -m -v		
6	Verify that output dataset files are created for datasets and that these files are scp'd to the configured destination. This test can be set up to run as a cron.		

TEST DATA:

EXPECTED RESULTS:

10.5.2 Test Case 1a - EMS Extract Utility: The EMS Dataset Extraction Utility shall maintain a record of the status and starting/ending reporting period of each execution. (ECS-ECSTC-895)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	EMS Extract Utility: The EMS Dataset Extraction Utility shall maintain a record of the status and starting/ending reporting period of each execution.		
2	L4 Requirement 20		

#	Action	Expected Result	Notes
3	For the default EMS script execution run above, verify that a record for these dataset runs has been entered in the EcEMSExtractRecord table and that the starting and ending reporting period is entered for each execution record.		

TEST DATA:

EXPECTED RESULTS:

10.5.3 Test Case 1b - EMS Extract Utility: The EMS Dataset Extraction Utility shall by default generate and export EMS flat file information for a 24-hour period (ECS-ECSTC-896)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	[EMS Extract Utility: The EMS Dataset Extraction Utility shall by default generate and export EMS flat file information for a 24-hour period.]		
2	L4 Requirement 30		
3	For the default EMS script execution run above, verify that the period of time used for each record is listed as 24 hours.		

TEST DATA:

EXPECTED RESULTS:

10.5.4 Test Case 1c - EMS Extract Utility: The EMS Dataset Extraction Utility shall by default use the ending date of the previous reporting period to determine the starting and ending date for the current execution. (ECS-ECSTC-897)

DESCRIPTION:

---

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	[EMS Extract Utility: The EMS Dataset Extraction Utility shall by default use the ending date of the previous reporting period to determine the starting and ending date for the current execution.]		
2	L4 Requirement 40		
3	For the default EMS script execution run above, verify that the period of time used for each record is consecutive periods of 24 hours.		

**TEST DATA:**

**EXPECTED RESULTS:**

10.5.5 Test Case 1d - EMS Extract Utility: The EMS Dataset Extraction Utility shall by default generate and export each of the following types of flat files per execution (ECS-ECSTC-898)

**DESCRIPTION:**

**PRECONDITIONS:**

**STEPS:**

#	Action	Expected Result	Notes
1	EMS Extract Utility: The EMS Dataset Extraction Utility shall by default generate and export each of the following types of flat files per execution:		
2	a) Product attribute metadata b) Product attribute search c) Data Ingest d) Data Archive e) Physical media distribution orders		
3	L4 Requirement 60		
4	The following datasets are extracted and exported using the extraction utility:  a) Product attribute metadata: Meta b) Product attribute search: searchExp		

---

#	Action	Expected Result	Notes
	c) Data Ingest:                    Ing d) Data Archive:                 Arch e) Physical media distribution orders   DistFTP (DataPool, FtpPush, FtpPull, Scp) DistHTTP (DataPool)		
5	For the default EMS script execution run above, verify that output dataset files are created for all datasets listed above and that these files have been scp'd to the configured destination.		

TEST DATA:

EXPECTED RESULTS:

10.5.6 Test Case 1e-g - EMS Extract Utility: The EMS Dataset Extraction Utility shall provide the operator the capability to specify an output directory for the flat files. (ECS-ECSTC-899)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	<i>[EMS Extract Utility: The EMS Dataset Extraction Utility shall provide the operator the capability to specify an output directory for the flat files.]</i>		#comment
2	L4 Requirement 80		
3	Verify that the directory listed for the EMSEXTRACTDIR tag in the EcDbEMSdataExtractor.CFG file is the directory where the EMS extract utility has extracted the dataset files to. The EcDbEMSdataExtractor.CFG file is located in the /usr/ecs/&lt;MODE&gt;/CUSTOM/cfg directory.		
4	DataSets: All [EMS Extract Utility:The EMS Dataset Extraction Utility shall apply filenames that uniquely distinguish each flat file based on the following: a)Beginning date for the reporting period b)DAAC name c)EMS flat file type]		
5	L4 Requirement 90		

#	Action	Expected Result	Notes
6	Verify that the Name given to a file is the same name listed in the EcEMSExtractRecord table for that extract run and that the name contains the Beginning date for the reporting period, the DAAC name, and the EMS flat file type.		
7	DataSets: All [EMS Extract Utility: A record of the Extract utility run shall be kept in a log file and in the EcEMSExtractRecord table]		
8	L4 Requirement 100		
9	Run the EMS extract utility using a command similar to the following:		
10	EcDbEMSDataExtractor.pl -m &lt;MODE&gt; -s &quot;start date&quot; -e &quot;end date&quot; -v -x DistFTP -o		
11	Verify that a record of the run can be found in the log that is created and that a record for the run can be found in the EcEMSExtractRecord table.		
12	DataSets: All [EMS Extract Utility: Verify that in extracted dataset files, there is one record per line and each record is divided into fields using ' &amp; ' as a delimiter.]		
13	Visually inspect the files that have been created by the extract script.		
14	DataSets: All		

TEST DATA:

EXPECTED RESULTS:

10.5.7 Test Case 1h - EMS Extract Utility: Verify that in extracted dataset files, there is one record per line and each record is divided into fields using '|&amp;|' as a delimiter. (ECS-ECSTC-901)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	[EMS Extract Utility: Verify that in extracted dataset files, there is one record per line and each record is divided into fields using ' &amp; ' as a delimiter.]		

#	Action	Expected Result	Notes
2	Visually inspect the files that have been created by the extract script.		

TEST DATA:

EXPECTED RESULTS:

### 10.6 SSS

#### 10.6.1 Place, Edit, Trigger Spatial Subscriptions (ECS-ECSTC-861)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Go to SSS GUI, click on the Manage Subscrptions/Add Subscriptions to create a SSS. fill in the UserId text box, set the status to &quot;Active&quot;, and Collection Short Name text box, then click 'OK'		
2	Select from the drop down list for SELECT Short Name / Version / Event Type, then click 'Apply' button		
3	Fill in list of text box in 'Acquire Inforamation' to create a SSS.		
4	Go DPL ingest GUI to place a PDR file to ingest a granule,		
5	Make sure the order is in 'shipped' in OMS order GUI.		

TEST DATA:

EXPECTED RESULTS:

## 10.7 OMS

### 10.7.1 Ftp Pull Acquire (ECS-ECSTC-859)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Submit a Ftp Pull request for multiple granules using the Spatial Subscription GUI, EWOC client, or the EcOmSrCliDriverStart utility.		
2	Verify that: a.) The request shows up on the OMS GUI Distribution Requests Page.		
3	b.) The “Request Status” column proceeds from Queued to Shipped.		
4	c.) The appropriate granule files are available for download from the ftp server and that other request directories are not visible.		
5	d.) DN emails are sent to the user specified in the request.		

TEST DATA:

EXPECTED RESULTS:

### 10.7.2 Ftp Push Acquire (ECS-ECSTC-858)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Submit a FtpPush request for multiple granules using EcOmSrCliDriverStart utility. Make sure the ftp destination is configured in the OMS GUI.		
2	Verify that: a.) The request shows up on the OMS GUI Distribution Requests Page.		
3	b.) The “Request Status” column proceeds from Queued to Shipped.		

---

#	Action	Expected Result	Notes
4	c.) The appropriate granule files are transferred to the destination.		
5	d.) DN emails are sent to the user specified in the request.		

TEST DATA:

EXPECTED RESULTS:

### 10.7.3 SCP Acquire (ECS-ECSTC-860)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Submit a scp request for multiple granules using the Spatial Subscription GUI, EWOC client, or the EcOmSrCliDriverStart utility.		
2	Make sure the scp destination is configured in the OMS GUI.		
3	Verify that: a.) The request shows up on the OMS GUI Distribution Requests Page.		
4	b.) The "Request Status" column proceeds from Queued to Shipped.		
5	c.) The appropriate granule files are transferred to the destination.		
6	d.) The distribution notice is left in the destination directory.		

TEST DATA:

EXPECTED RESULTS:

### 10.7.4 OMS Checkout (Track, Edit, Cancel, Resubmit Orders) (ECS-ECSTC-914)

DESCRIPTION:

---

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Track Request:  Submit a ftppush request using the OMS Scli driver or EWOC client.		
2	Verify the request shows up on the Order Manager GUI.		
3	Verify the request in the Order Manager debug log.		
4	Edit Request:  Go to the Distribution Requests page on the Order Manager GUI and click on the RequestID of a ftppush request.		
5	Click on the Edit ftppush Parameters button to edit the parameters. Click on the Change This Request button.		
6	Go back to the Distribution Request page and resubmit the request.		
7	Verify the request completes successfully.		
8	Cancel Request:  From the Distribution request page on the OMS GUI, cancel a request.		
9	Verify the request gets canceled.		
10	Resubmit Request:  From the Distribution request page on the OMS GUI, resubmit a request. The request will go into operator intervention.		
11	Click on the Operator Intervention link of the request. Click on assign to assign the omsadmin worker. Select Submit and then Apply and then Ok twice.		
12	Verify the request completes successfully.		

TEST DATA:

EXPECTED RESULTS:

## 10.8 DataAccess

### 10.8.1 EDSC Spatial Query (ECS-ECSTC-887)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Submit an Earthdata Search Client Spatial query. Go to the Earthdata Search Client and search for a collection.		
2	Click on Download All to go to the Edit options page		
3	Select Customize, fill in the email address, select an output file format, and the spatial subsetting option.		
4	For the Spatial Subsetting options, put in the spatial coordinates for the bounding box		
5	Click on "Download data" to submit the request.		
6	Verify that the request completes and returns a status of Complete on the Order Status page.		
7	Verify an email is sent to the email address entered on the edit options page.		

TEST DATA:

EXPECTED RESULTS:

### 10.8.2 EDSC Acquire (http) (ECS-ECSTC-888)

DESCRIPTION:

PRECONDITIONS:

STEPS:

#	Action	Expected Result	Notes
1	Submit an Order through the Earthdata Search Client. Go to the Earthdata Search Client and search for a collection.		

---

#	Action	Expected Result	Notes
2	Click on Download All to go to the Edit options page		
3	Select Staging for Delivery, select distribution type HTTPS		
4	Click on "Download data" to submit the request.		
5	Verify that the request completes and returns a status of Complete on the Order Status page.		

TEST DATA:

EXPECTED RESULTS: