



McIDAS-XCD Administrator's Guide

Version 2020.1

McIDAS-XCD Administrator's Guide

Version 2020.1 Table of Contents

[Copyright Statement](#)

Chapter 1 - [Introduction to McIDAS-XCD](#)

- [Terminology](#)
- [Data receiving and processing](#)
- [McIDAS-XCD Status Display Window](#)

Chapter 2 - [Installing and Configuring McIDAS-XCD](#)

- [System Requirements](#)
- [Obtaining McIDAS-XCD 2020.1](#)
- [Installing McIDAS-XCD 2020.1](#)
- [Configuring McIDAS-XCD](#)
- [Starting and Stopping McIDAS-XCD](#)
- [Checking the Filing Status of McIDAS-XCD Data](#)
- [Uninstalling the Previous Version of McIDAS-XCD](#)

Chapter 3 - [McIDAS-XCD Administrative Commands](#)

Chapter 4 - [Troubleshooting](#)

Chapter 5 - [Decoding GRIB Messages](#)

- [Processing the GRIB message](#)
 - [Converting GRIB codes](#)
 - [Creating McIDAS Grid Files with DMGRID](#)
 - [Creating GRIB Files with DMBIN](#)
-

Copyright© 1997-2020 Space Science and Engineering Center
(SSEC) University of Wisconsin - Madison
All Rights Reserved

Permission is granted to make and distribute verbatim copies of this document, provided the copyright notice and this permission are preserved on all copies.

Permission is further granted to modify, amend or otherwise alter this document, and to distribute, including electronically, the modified, amended or otherwise altered document provided the copyright notice and this permission are preserved on all copies and derivative works thereof. In addition, the following notice must be added to this copyright page by each individual or organization that modifies, amends or otherwise alters this document: "This is NOT a verbatim version of the original SSEC document. Portions have been modified, amended or otherwise altered by *[name and address of modifying individual or organization]*."

SSEC makes no warranty of any kind with regard to the software, hardware or accompanying documentation, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. SSEC does not indemnify any infringement of copyright, patent, or trademark through the use or modification of this software.

There is no expressed or implied warranty made to anyone as to the suitability of this software for any purpose. All risk of use is assumed by the user. Users agree not to hold SSEC, the University of Wisconsin-Madison, or any of its employees or assigns liable for any consequences resulting from the use of the McIDAS software.

Mention of any commercial company or product in this document does not constitute an endorsement by SSEC. Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this document, and SSEC was aware of the trademark claim, the designations are printed in caps or initial caps.

The information in this document is subject to change without notice. Considerable effort has been expended to make this document accurate and complete, but SSEC cannot assume responsibility for inaccuracies, omissions, manufacturers' claims or their representations.

Chapter 1

Introduction to McIDAS-XCD

The McIDAS-X Conventional data Decoder (McIDAS-XCD) enables workstations running McIDAS-X to directly receive and process data from the NCEP/NWSTG and NCEP/NWSTG2 channels of the National Weather Service NOAAPORT broadcast. All of the operational McIDAS-XCD client commands for accessing conventional data have been removed from the McIDAS-XCD package. These commands have been replaced with ADDE (Abstract Data Distribution Environment) commands distributed with McIDAS-X. See the [McIDAS User's Guide](#) for more information.

This introduction consists of the following topics:

- [Terminology](#)
 - [Data receiving and processing](#)
 - [McIDAS-XCD Status Display Window](#)
-

Terminology

The terms defined below are used throughout this manual.

Term	Definition
BUFR	Binary Universal Form for the Representation of meteorological data (the WMO standard format for point data)
client	workstation that requests and receives data from a server workstation
data block	WMO header description and text data
data monitor	process that runs one or more decoders
DDS	Domestic Data Service
decoder	program that converts raw data into McIDAS data files
GRIB	GRIdded Binary (the WMO standard format for gridded data); includes GRIB1 and GRIB2
GRIB1	GRIB version 1
GRIB2	GRIB version 2
HRS	High Resolution data Service
IDS	International Data Service
ingestor	program that receives data through a communications port
MySQL/MariaDB	free downloadable database that can be used by McIDAS-XCD and the McIDAS GRIB server
NEXRAD	NEXt generation weather RADar
NFS	Network File System
PPS	Public Products data Service
server	workstation that stores and supplies data to client workstations

Data receiving and processing

McIDAS-XCD uses ingestors and data monitors to receive and process asynchronous data from the National Weather Service (NWS) NOAAPORT broadcast. The data arrives via satellite broadcast by either an outside vendor or a dedicated phone line directly from the circuit source.

This section discusses types of:

- [Ingestors](#)
 - [Data monitors](#)
-

Ingestors

An ingestor is a program that reads data entering the system through a communications port. Ingestors read asynchronous data from conventional data circuits such as DDS, IDS and PPS.

Each circuit has a text formatted configuration file that the ingestor reads to configure the communications port. This configuration file resides in `~oper/mcidas/data` and is usually named with the circuit name followed by `.CFG`, for example, `DDS.CFG`. It contains information such as baud rate and the number of data bits or stop bits.

McIDAS-XCD supports these types of ingestors:

- [Text ingestors](#)
- [Binary ingestors](#)

Text ingestors

A text ingestor receives data from the NOAAPORT broadcast in ASCII format. Each incoming circuit has its own text ingestor that writes to a set of raw text files and index files. Index files contain the location information of data in the raw text file.

For example, [Figure 1](#) shows that the text ingestor INGETEXT ingests data from NOAAPORT. Each ingested data block is placed in a circuit-specific raw text file for that day. INGETEXT also files information about the data block into an index file which is used by text applications and decoders for locating data quickly.

The naming convention for the circuit-specific raw text file is `ccyyddd0.XCD`, where `cc` is the first two characters of the circuit name, and `yyddd` is the Julian day. The naming convention of the index file is `hhyyddd.IDX` or `hhhyddd.IDX` where `hh` is a 2-character WMO header, `hhh` is a 3-character AWIPS header and `yyddd` is the Julian day.

While only one text ingestor can write into a text file, any text ingestor can write to any index file. For example, terminal forecasts (FTs) arrive on the DDS and IDS circuits. The actual forecasts are filed in `DDyyddd0.XCD` and `IDyyddd0.XCD`, respectively. However, both ingestors write their directory information into the same index file, `FTyyddd.IDX`. This ensures that applications will work consistently on similar data formats regardless of the data's source.

Index files also store related data that arrives under more than one WMO header. For example, mandatory upper level RAOB reports come in under the headers UJ, US, UK, UL, etc. Rather than having a separate index file for each header, a routing table is created during installation to tell the ingestors where to file the headers from each circuit. For example, the index file for all RAOB WMO headers is `UJyyddd.IDX`. If a WMO header is not forced into a particular index file, it is filed in the `ZZyyddd.IDX` file, which is a miscellaneous index.

Binary ingestors

The binary data ingestor, INGEBIN, ingests a binary data stream regardless of the data format. INGEBIN writes the data to a circular spool file, `HRS.SPL`, where it can be processed by a data monitor. The data that INGEBIN ingests includes model output grids in GRIB1 or GRIB2 format, NEXRAD radar data in its native format, and various text products in BUFR format. For further information see Chapter 2, [Configuring the NEXRAD Decoder](#), and [Configuring the GRIB Decoders](#).

Figure 1. Conventional Data Ingesting Flow Diagram

Data monitors

A data monitor is a process that runs one or more decoders. After the data is stored in the raw text or spool files, data monitors process the raw data into McIDAS files such as surface hourly observations, synoptic reports, upper air reports, and grid files.

The following table lists the supported McIDAS-XCD data monitors.

Monitor	Index	Data type	Decoded Data location	Configuration file
DMBIN		GRIB1, GRIB2 and BUFR	Native GRIB1, GRIB2 and BUFR	GRBFILER.CFG
DMCOND		GRIB2	Native GRIB2	CONFILER.CFG
DMGRID		GRIB1	Grid files	GRIBDEC.CFG
DMMISC	FO	NAMMOS	MD file NAM	NAMMOS.CFG
	FO	GFSMOS	MD file GFS	GFSMOS.CFG
	FT	Terminal Forecast	Rapid-Access System	TERMDEC.CFG
	SD	MDR	Grid files	MDRDEC.CFG
	TB	TIROS NAV	SYSNAV1	TIRDEC.CFG
	UA	PIREP/AIREP	MD file PIRP	PIRPDEC.CFG
DMNEXR		Radar products	Compressed data files	NEXRFILE.CFG
DMRAOB	UJ	TEMP/PILOT	MD file IRAB/IRSG	IRABDEC.CFG
	UJ	TEMP SHIP	MD file ITSHP	ITSHPDEC.CFG
DMSFC	SA	SAO/METAR	MD file ISFC	ISFCDEC.CFG
DMSYN	SM	SYNOPTIC	MD file SYN	SYNDEC.CFG
	SM	SHIP/DRIBU	MD file ISHP	ISHPDEC.CFG

To make the data monitors more flexible, each is designed to use a text formatted configuration file similar to those used for circuit configuration. This configuration file contains the .IDX files to search, WMO headers to decode, decoder display number, MD file numbers to store data, etc.

Figure 2 shows an example of the configuration script file for the GFS MOS decoder.

Figure 2. GFS MOS Configuration File

```
# GFSMOS.CFG - Configuration file for the GFS MOS decoder # ----- Cross Reference List
(do not change) ----- : FLAGS[01] ERRORFLG : FLAGS[02] IDMONFLG : FLAGS[03]
DISPLAYNUM : FLAGS[04] MDF : FLAGS[05] NROWS : FLAGS[06] NCOLS : FLAGS[07] RAPIDACCESS :
CFLAGS[01] ERRORFILE : CFLAGS[02] OLDIDFILE : CFLAGS[03] NEWIDFILE : CFLAGS[04] IDTABLE :
CFLAGS[05] MASTERFILE : CFLAGS[06] PTRFILE # ----- End Of Cross Reference
List ----- # -----
----- # - You can modify any of the fields below - # -----
----- # decoder description DESCRIPTION="GFSMOS Decoder" # which
indices to search for this decoder INDEX=FO # which specific wmo headers to activate the
decoder for WMO=FOUS FOAK FOPA MINPRD=21 37 20 MAXPRD=26 39 20 # which specific station
origins to activate the decoder for ORIGIN=KWBC KWNO #
===== ERRORFLG=0 # error
output flag set to 1 to activate ERRORFILE=GFSMOS.ERR # error file name IDMONFLG=0 # station
id monitoring activation flag # set to 1 or 3 to monitor new stations # set to 2 or 3 to
monitor old stations OLDIDFILE=OLDGFS.IDM # old station id file used for monitoring
NEWIDFILE=NEWGFS.IDM # new station id file used for monitoring DISPLAYNUM=15 # decoder
number on status display MDF=81 # first real-time md file number to use for decoder NROWS=84
# number of rows to make for md file NCOLS=2000 # number of columns to make for md file
RAPIDACCESS=1 # rapid access system activation flag, used for # MOSRPT command. set to 1 to
activate, 0 to # inactivate IDTABLE=GFSMOS.IDT # ID file to build when creating md file
MASTERFILE=STNDB.CORE # master ID table file to use to build IDTABLE PTRFILE=GFSMOS.RAP #
rapid access pointer file name
```


McIDAS-XCD Status Display Window

The McIDAS-XCD Status Display window lists information about the data that is being processed by the ingestors and data monitors. The Status Display window can be started with the [statdisp](#) command.

A sample Status Display window is shown in Figure 3; the table below it defines each field in the display. The top section of the display shows the status of the ingestors. In the sample display the text ingestor INGETEXT last filed data at byte 1996218720 in the raw text file and last filed data in the index file FE19247.IDX at byte location 39104. The bottom section of the display shows the status of the data monitors. In the sample display the surface observation data monitor SAODEC has decoded data from the raw text file at byte offset 794512. You can tell that SAODEC is up to date because the *Begptr* and *Lasptr* fields are identical. Once SAODEC has read up to byte 794512 it rereads the index file SA19247.IDX to determine if there is more data to be decoded. If no more data is available then the data monitor will wait 30 seconds and then try again. In the sample display, SAODEC last wrote to MD file MDXX0007 at row 52 and column 5941. If a decoder becomes inactive for any reason, the status line will turn red. This indicates that the administrator should investigate to see if something is wrong.

Figure 3. Sample Status Display Window

```

McIDAS-XCD Status (pollux)
Wed Sep 4 17:22:30 2019
## Src Ingestor Time Byte Index Index file Origin Wmo Product
-----
1 NTXT INGETEXT 172229 1996218720 39104 FE19247.IDX KUWO FEUS 23
2 NDIR INGEBIN 172229 495069211 HRS.SPL
3 NEXR INGEBIN 172230 629377868 NEXR.SPL

## Decoder Time Begptr Lasptr Gridf Grid MD Row Col Text Index
-----
1 SAODEC 172227 794512 794512 7 52 5941 SR19247
2 RABDEC 172222 77232 77232 17 6 1244 UJ19247
3 SYNDEC 172227 4445472 4445472 57 18 1078 SM19247
4 SHPDEC 172227 4445472 4445472 37 17 2815 SM19247
7 PIRDEC 172225 186352 186352 67 18 662 UA19247
8 TERDEC 172225 182736 182736 FT19247
12 GRBFILER 172225 0 0 0
13 NEXRFILE 172227 0 0 0 N2XL0T 0417
14 NAMMOS 172225 90208 90208 77 1 1279 F019247
15 GFSMOS 172225 90208 90208 87 1 1413 F019247
17 TSHPDEC 172222 77232 77232 0 0 0 UJ19247
  
```

The table below defines each field in the Status Display window.

Field	Definition
##	ingestor
Src	circuit receiving the data
Ingestor	ingestor command name
Time	time the data was last received
Byte	last byte number the ingestor wrote
Index	last directory location the ingestor filed; not used by INGEBIN
Index file	index file name last written to; for INGEBIN this field displays the spool name
Origin	origin of the last block filed; the value is extracted from the WMO header; not used for INGEBIN
Wmo	product header of the last block filed; not used by INGEBIN
Product	WMO product number of the last block filed; not used by INGEBIN
##	data monitor number
Decoder	decoder name

Time	time data was last processed
Begptr	current index location being decoded
Lasptr	last index location decoder processes before checking for more data
Gridf/MD	last GRID or MD file the decoder wrote to
Grid/Row	last GRID number or ROW number written to
Col	last MD column number written to
Text	text description of the decoder process
Index	name of index file used to keep track of data in the raw text file

Chapter 2

Installing and Configuring McIDAS-XCD

This chapter consists of the McIDAS-XCD installation and configuration instructions. There are seven separate sections:

- [System Requirements](#)
- [Obtaining McIDAS-XCD 2020.1](#)
- [Installing McIDAS-XCD 2020.1](#)
- [Configuring McIDAS-XCD](#)
- [Starting and Stopping McIDAS-XCD](#)
- [Checking the Filing Status of McIDAS-XCD Data](#)
- [Uninstalling the Previous Version of McIDAS-XCD](#)

Check the system requirements before installing McIDAS-XCD on your workstation. Then use the instructions that follow to obtain, install and configure McIDAS-XCD.

System Requirements

- The McIDAS-XCD 2020.1 software package was tested on Red Hat Enterprise Linux 8.2 Intel and is supported on RHEL 7.x and 8.x.
- McIDAS-X 2020.1 (released in pair with McIDAS-XCD 2020.1 in July 2020) must be installed in the mcidas account according to the specifications in Chapter 1, *Installing and Configuring McIDAS-X*, in the [McIDAS User's Guide](#). Be sure to include the directory ~mcidas/bin in the environment variable PATH.
- The Unix workstation running the McIDAS-XCD software must have the user account oper. This account must be configured with the appropriate directories, links, and paths to run McIDAS-X. For more information, see the section titled *Configuring a New User Account* in Chapter 1 of the *McIDAS User's Guide*. If you already have an oper account on your workstation and do not want to run the -XCD package under this account, contact the McIDAS Help Desk at (608) 262-2455 or by [e-mail](#).
- The Unix workstation running the McIDAS-XCD software must have the group name mcdata which contains the user oper.
- McIDAS-XCD is designed to process data from the NOAAPORT data stream that is broadcast by the National Weather Service. The standard method for receiving this data stream is through the use of the [LDM software from Unidata](#).

In addition, the workstation must have enough disk space to run the McIDAS-XCD software package and store its data files. The table below lists the approximate disk space requirements per day for each data type decoded with the -XCD package. The space requirements were derived from the NOAAPORT broadcast in June 2020.

Data type	Daily space requirements
* ASCII text stream	2,700 MB
* MD files (all types/schemas)	198 MB
All GRIB1 and GRIB2 data	450,000 MB
All radar (WSR-88D and TDWR) products and stations	55,000 MB
All BUFR data	3,100 MB
* These data are generated by default for first-time installations. The GRIB, GRID and NEXRAD decoders are inactive by default.	
NOAAPORT approximate total	511,000 MB (511 GB)

In addition, there is a static requirement of 500 to 1,000 MB for other -XCD files. If enabled, the MySQL/MariaDB database (60 MB) and LDM product queue (default=400 MB) also consume significant disk space. Note: The LDM product queue can be set as large as 4 GB on a 32 bit Enterprise Linux workstation.

Remember that the space requirements above are approximates and are *per day*. If you save multiple days of data you must multiply the figures by the number of days for your disk space requirements.

Obtaining McIDAS-XCD 2020.1

The McIDAS-XCD 2020.1 package contains the following files.

File name	Description
xcd2020.1.sh	shell script that installs the McIDAS-XCD software
xcd2020.1.tar.Z	compressed tar file that contains all source and data files
xcd_README_2020.1	lists information to review before installation

Complete the instructions below to obtain the McIDAS-XCD 2020.1 installation files from the McIDAS Website.

1. Use your Web browser to download the files listed in the table above. Access the McIDAS Website at <http://www.ssec.wisc.edu/mcidas>, and follow the link for McIDAS-XCD software. Each site has its own login and password for downloading files. Have your site coordinator contact the McIDAS Help Desk if you can't remember yours.
2. Log on to the McIDAS-XCD workstation as user oper and move the downloaded files to the `~oper/mcidas` directory.
3. List the files and check the ownership.

Type: `ls -l ~oper/mcidas`

If the -XCD files are owned by user oper, skip steps 4-6 and proceed to the next section, [Installing McIDAS-XCD 2020.1](#).

4. Change the ownership to user oper, if needed. You must have root permission to do this. Switch to user root.

Type: `su root`

5. Change to the `~oper/mcidas` directory.

Type: `cd ~oper/mcidas`

6. Run the command below for each of the downloaded -XCD files.

Type: `chown oper file`

Installing McIDAS-XCD 2020.1

The McIDAS-XCD 2020.1 software installation process makes the directories `~oper/mcidas/xcd2020.1`, `~oper/mcidas/xcd2020.1/src` and `~oper/mcidas/xcd2020.1/data`, places the source and data files in them, and builds the software. When the build is complete, the source, helps, data, and binaries are copied to the directories `~oper/mcidas/src`, `~oper/mcidas/help`, `~oper/mcidas/data`, and `~oper/mcidas/bin`, respectively.

This section contains the instructions for installing McIDAS-XCD 2020.1. There are two sets of instructions. The instructions you use depend on whether you are doing an **upgrade** (installing -XCD 2020.1 on a workstation that's running a previous version of -XCD) or an **initial install** (installing the -XCD software for the first time on a workstation).

- Go to [Upgrading to McIDAS-XCD 2020.1](#) if the workstation already has a previous version of McIDAS-XCD installed.
 - Go to [Installing McIDAS-XCD 2020.1 for the First Time](#) if you are installing McIDAS-XCD for the first time on the workstation.
-

Upgrading to McIDAS-XCD 2020.1

The instructions for upgrading McIDAS-XCD on a workstation that already has a previous version of McIDAS-XCD installed are now entirely on the McIDAS Website (rather than in this manual). Go to the [Download McIDAS-XCD page](#) for the instructions.

Installing McIDAS-XCD 2020.1 for the First Time

If you are installing McIDAS-XCD on a workstation that already has a previous version of McIDAS-XCD installed, go to the previous section, [Upgrading to McIDAS-XCD 2020.1](#).

Installing McIDAS-XCD for the first time on a workstation consists of the three tasks in this section ([Adding the mcdata Group and Disabling Binary Logging](#), [Assigning Directory Permissions](#), [Loading the McIDAS-XCD Software](#)), followed by additional configuration steps in the section [Configuring McIDAS-XCD](#).

Before proceeding with the instructions below, verify that McIDAS-X 2020.1 has been installed on the workstation. If it hasn't you must do so **before** you install McIDAS-XCD 2020.1.

Adding the mcdata Group and Disabling Binary Logging

The workstation running the McIDAS-XCD software must have the group name mcdata which contains the users oper and mcidas. It also should disable the writing of binary log files associated with the database. This is needed because MySQL/MariaDB 8.0 and later enable binary logging by default, which fills large amounts of disk (possibly an entire file system) due to it logging millions of *INSERT into file_list* entries per day. Use this procedure to add the mcdata group, if needed, and disable binary logging.

1. Log on to the root account.
2. Add the following line to the `/etc/group` file. Replace *groupid* with a unique group ID number.

```
mcdata::groupid:oper,mcidas
```

For example: `mcdata::23000:oper,mcidas`

3. Disable the writing of binary log files associated with the database.

Type: `/home/oper/mcidas/bin/gribadmin disablelog`

When the command is finished, you should see a message like that below.

```
/etc/my.cnf.d/disable-log-bin.cnf successfully written!
```

4. Log out of the root account.

Assigning Directory Permissions

Use the steps below to assign write privileges to the directory `~mcidas/data`.

1. Log on to the workstation as user mcidas and change the group for the `~mcidas/data` directory to the group mcdata.

Type: `chgrp mcdata ~mcidas/data`

2. Change the permissions for the directory `~mcidas/data` so only the user mcidas and members of the group mcdata have write permission.

Type: `chmod 775 ~mcidas/data`

3. Log out of the mcidas account.

Loading the McIDAS-XCD Software

Complete the steps below to load the McIDAS-XCD software.

1. Logon to the workstation as user oper and change to the `~oper/mcidas` directory.

Type: `cd ~oper/mcidas`

2. Verify that the PATH environment variable contains the `~mcidas/bin` and `~oper/mcidas/bin` directories, and the operating system-specific modifications listed for the mcidas account in the [McIDAS User's Guide](#).

Type: `echo $PATH`

3. Verify that the `~oper/mcidas/data` directory exists (use command `mkdir ~oper/mcidas/data` to make it if it doesn't) and that the MCPATH environment variable is set and includes the `~oper/mcidas/data`, `~oper/mcidas/help`, `~mcidas/data`, and `~mcidas/help` directories.

Type: echo \$MCPATH

4. Build and install McIDAS-XCD 2020.1 using the same settings for optional flags `-noopt` and `-mysql` as were used to build McIDAS-X 2020.1. You must use the same compilers and optional flag settings for both the McIDAS-X and McIDAS-XCD builds.

Type: sh ./xcd2020.1.sh *flags* install all

where *flags* is zero, one or both of the two optional flags described below.

- The **-noopt** flag is used to build McIDAS-XCD with no optimization. If you don't specify this **-noopt** flag, McIDAS-XCD is built using the default optimization: `-O2` for builds with vendor compilers and `-O1` for builds with `gcc/g77`.
- The **-mysql** flag is used to build McIDAS-XCD using the MySQL/MariaDB database libraries. Use this flag if this workstation will be serving real-time GRIB files from the NOAAPORT data stream and you have installed and want to use a MySQL/MariaDB database to make the GRIB data access faster for your McIDAS-X clients/users. If the database is in a non-standard install location, specify the flag as **-mysql=directory**, where *directory* is the root MySQL/MariaDB install directory. Remember, the **-mysql** flag must be specified when building both McIDAS-X and McIDAS-XCD.

This script performs the following steps:

- creates the subdirectories `xcd2020.1`, `xcd2020.1/src`, and `xcd2020.1/data` in the `~oper/mcidas` directory
- uncompresses the file `xcd2020.1.tar.Z`
- compiles the source code and copies the binaries to the directory `~oper/mcidas/bin`; approximately 130 modules are compiled, so this step takes a few minutes to complete; your compiler may generate some warnings while the macro commands are compiling
- copies data files to the `~oper/mcidas/data` and the `~mcidas/data` directories
- copies the help files to the `~oper/mcidas/help` directory

When the script is finished, you will see the message below.

```
McIDAS-XCD package installation is now complete
```

5. Switch to user `mcidas` so you can install the McIDAS-XCD ADDE servers.

Type: su mcidas

6. Run the script below to install the new ADDE server executable code in the `mcidas` account.

Type: sh ./xcd2020.1.sh install addservers

7. Exit from user `mcidas`.

Type: exit

You should now be logged on as user `oper`.

8. Start a McIDAS-X session.

Type: mcidas

9. Determine the full Unix path of the directory in which you want -XCD to write data. The location where you store data is up to you; be sure to take disk space needs into account when making your decision. The data directory needs to exist, and also needs to contain `grib` and `bufr` subdirectories. Once you've determined the path of your data directory, create a McIDAS string named `MCDATA` that contains the full Unix path to the directory. From the McIDAS-X session,

Type: **TE MCDATA** *"/data_directory_path*

For example: **TE MCDATA** *"/home/oper/xcd_data*

In order for -XCD to correctly file data in this example, the `/home/oper/xcd_data`, `/home/oper/xcd_data/grib`, and `/home/oper/xcd_data/bufr` directories need to exist. Create the `#MCDATA`, `#MCDATA/grib`, and `#MCDATA/bufr` directories, if necessary.

10. Run the batch file `XCD.BAT` to redirect to data files in `#MCDATA` and to initialize the files `GROUPS.DAT` and `COUNTRY.DAT`.

Type: **BATCH XCD.BAT**

11. Run the batch file `XCDDEC.BAT` to register MD file schemas, create rapid access pointer files, and initialize several data files.

Type: **BATCH XCDDEC.BAT**

Proceed to the next section, [Configuring McIDAS-XCD](#), and complete the configuration instructions for each section.

Configuring McIDAS-XCD

This section contains the procedures for configuring seven components of McIDAS-XCD.

- [Configuring Data Communications](#)
- [Configuring the Data Monitors](#)
- [Configuring the NEXRAD Decoder](#)
- [Configuring the GRIB Decoders](#)
- [Configuring Data Removal](#)
- [Configuring the Default ADDE Datasets](#)
- [Configuring Additional GRIB2 ADDE Datasets](#)

If you are installing McIDAS-XCD for the first time on a workstation, you must review all the sections and complete the appropriate actions for your workstation's configuration. If your workstation has been running -XCD and you want to modify its configuration, go to the corresponding section(s) and complete the actions needed to implement the desired changes.

Configuring Data Communications

The method you use for configuring McIDAS-XCD communications depends on the source of your data. If you use the [NOAAPORT Receive System](#), the LDM is your data source and you must complete the instructions below.

1. From a McIDAS-X session running under the oper account, run the batch file NOAAPORT.BAT to configure WMO header routings for the NTXT circuit.

Type: **BATCH NOAAPORT.BAT**

2. Deactivate the text and binary circuits. This is necessary because the LDM will feed data directly into McIDAS-XCD's text and binary ingestors.

Type: **CIRCUIT SET NTXT INACTIVE**

Type: **CIRCUIT SET NBIN INACTIVE**

3. Edit the configuration files NTXT.CFG and NBIN.CFG in the ~oper/mcidas/data directory. In each file, if it's not already done, comment out the FILE=, PERM=, HOST=, and HOST_PORT= lines and add the following line.

```
FILE=STDIN
```

4. The lines in your /home/lbm/etc/pqact.conf file for DDS, PPS, IDS, HRS, NNEXRAD, etc., should be piped to either /home/oper/mcidas/bin/ingebin.k or /home/oper/mcidas/bin/ingetext.k, as shown in the example below (use tabs, not spaces, to separate items within the lines).

```
PPS|DDS|IDS ^.* PIPE /home/oper/mcidas/bin/ingetext.k NTXT HDS|NNEXRAD ^.* PIPE  
/home/oper/mcidas/bin/ingebin.k NBIN NGRID ^.* PIPE /home/oper/mcidas/bin/ingebin.k NBIN
```

Note: The LDM must be started/stopped from the **oper** account. Therefore the **~ldm/data** directory and the **~ldm/data/lbm.pq** file must be writable by user **oper**.

Configuring the Data Monitors

Each data monitor controls one or more decoders. You can activate or deactivate any data monitor or any decoder by using the DECINFO command. See Chapter 3, [McIDAS-XCD Administrative Commands](#), for more information about DECINFO.

From a McIDAS-X session running under the oper account, list the status of the available data monitors and decoders.

Type: DECINFO

The table below lists the default status of data monitors and decoders.

Data Monitor	Decoder	MD Files	Status	Description	Configuration File
DMSFC	SAODEC	1-10	Active	Surface hourly	ISFCDEC.CFG
DMRAOB	RABDEC	11-30	Active	Upper air	IRABDEC.CFG
	TEMPSHIPDEC	41-50	Active	TEMP SHIP	ITSHPDEC.CFG
DMSYN	SYNDEC	51-60	Active	Synoptic	SYNDEC.CFG
	SHPDEC	31-40	Active	Ship/Buoy	ISHPDEC.CFG
DMMISC	NAMMOS	71-80	Active	NAM MOS	NAMMOS.CFG
	GFSMOS	81-90	Active	GFS MOS	GFSMOS.CFG
	PIRDEC	61-70	Active	PIREP/AIREP	PIRPDEC.CFG
	TERDEC		Active	Terminal Fest	TERMDEC.CFG
	TIRDEC		Inactive	TIROS NAV	TIRDEC.CFG
	MDRDEC		Inactive	MDR grids	MDRDEC.CFG
DMGRID	GRIBDEC		Inactive	NCEP GRIDS	GRIBDEC.CFG
DMNEXR	NEXRFILE		Inactive	Radar products	NEXRFILE.CFG
DMBIN	GRBFILER		Inactive	GRIB & BUFR	GRBFILER.CFG
DMCOND	CONDUITFILER		Inactive	GRIB2	CONFILER.CFG

Configuring the NEXRAD Decoder

If you will be decoding radar (formerly NEXRAD; now WSR-88D and TDWR) products you must configure the file NEXRID.CFG (which defines the stations and products that will be decoded by the NEXRFILE decoder, and the amount and location of data to store) and then activate the data monitor.

1. Edit the file `~oper/mcidas/data/NEXRID.CFG` and set the values for the following variables:

- o `FILE=` and `EXCEPT=` variables in the STATIONS control line
- o `FILE=` and `EXCEPT=` variables in the PRODUCTS control line
- o `WSRFILE=`, `TDWRFILE=` and `KEEP=` variables in the DISPOSE control line

2. Activate the NEXRAD data monitor by running the following command in a McIDAS-X session running under the oper account.

Type: **DECINFO SET DMNEXR ACTIVE**

Note: The Unix warning message *component of path does not exist* will normally be displayed by the NEXRAD decoder each time it creates a subdirectory for incoming stations and products. The error message is normal, so can be ignored.

Creating a Separate Spool File for the NEXRAD Decoder (optional)

This procedure is optional and is available only to sites that receive data via the LDM. The procedure significantly reduces processor usage by creating a separate spool file for NEXRAD data. The gains in efficiency are due to the NEXRAD decoder no longer needing to scan through GRIB data looking for the next NEXRAD product and vice-versa for the GRIB decoder.

Complete the steps below to implement the procedure.

1. Run the command below from the McIDAS-X Text and Command window to stop the NEXRAD decoder.

Type: **DECINFO SET DMNEXR INACTIVE**

2. Run the command below from the Unix prompt to copy the NBIN circuit configuration file (no changes to the file need be made).

Type: **cp ~oper/mcidas/data/NBIN.CFG ~oper/mcidas/data/NEXR.CFG**

3. Run the command below from the McIDAS-X Text and Command window to add the new NEXR circuit.

Type: **CIRCUIT ADD NEXR INGESTOR=INGEBIN CONFIG=NEXR.CFG SPOOL=NEXR.SPL "NOAAPORT NWSTG NEXRAD Binary Circuit**

4. Modify the NEXRAD decoder configuration file with the name of the new circuit.

- a. Open the file `~oper/mcidas/data/NEXRFILE.CFG` for editing.
- b. Change the value of `SPLFILE=` to `NEXR.SPL`.
- c. Save and close the file.

5. Run the command below from the McIDAS-X Text and Command window to define the full Unix path of the `NEXR.SPL` file.

Type: **REDIRECT ADD NEXR.SPL "/data_directory_path**

For example: **REDIRECT ADD NEXR.SPL "/home/oper/xcd_data**

6. Modify the LDM configuration so that it pipes NEXRAD data to a second instance of the binary ingestor.

- a. Open the file `~ldm/etc/pqact.conf` for editing.
- b. Create a separate entry in the *Request Entries* section for NEXRAD data and set it to pipe the data to the binary ingestor using the NEXR circuit. If you had a line that contained NNEXRAD as well as other binary feeds such as HRS, remove NNEXRAD from this line. The resulting entry should look like that in the box below.

```
NNEXRAD ^.* PIPE /home/oper/mcidas/bin/ingebin.k NEXR
```

- c. Save and close the file.

7. Run the command below from the Unix prompt to remove the old pointer file for the NEXRAD decoder.

Type: **rm ~oper/mcidas/data/NEXRFILE.PRO**

8. Run the command below from the Unix prompt to stop and restart the LDM.

Type: **ldmadmin restart**

9. Run the command below from the McIDAS-X Text and Command window to start the NEXRAD decoder.

Type: **DECINFO SET DMNEXR ACTIVE**

You should notice very soon after starting the NEXRAD decoder that there is a third entry in the ingestor section of the [statdisp_display](#). Two instances of the binary ingestor (INGEBIN) will exist. The NEXRAD data should now be written to NEXR.SPL by the binary ingestor using the NEXR circuit.

Configuring the GRIB Decoders

As noted in [Configuring the Data Monitors](#), the DMGRID, DMBIN and DMCOND data monitors control the GRIBDEC, GRBFILER and CONDUITFILER decoders, respectively. If activated, the data monitors perform the following actions with real-time NCEP GRIB messages.

- The DMGRID data monitor decodes the GRIB1 messages then creates and stores them in McIDAS Grid files.
- The DMBIN data monitor files the GRIB1 and GRIB2 messages into GRIB files of the same format and populates the MySQL/MariaDB database.
- The DMCOND data monitor files the GRIB2 files from the CONDUIT LDM feed into a separate spool file. This optional procedure is useful for sites that receive their data via the LDM and helps ensure that data processing keeps up with the data during high volume periods.

You can activate the data monitors as needed and make their data available to your users in datasets defined in [Configuring the Default ADDE Datasets](#). Keep in mind, however, that SSEC tests and supports real-time -XCD grid datasets only in GRIB format, and that McIDAS Grid files consume much more disk space than GRIB files. See Chapter 5, [Decoding GRIB Messages](#), for more information.

The sections below contain instructions for configuring the data monitors.

DMGRID Data Monitor for McIDAS Grid Files

If you want to decode the GRIB1 messages into McIDAS Grid files, activate the DMGRID data monitor by running the command below from a McIDAS-X session running under the oper account.

Type: **DECINFO SET DMGRID ACTIVE**

DMBIN Data Monitor for GRIB1 and GRIB2 Files

If you want to file the GRIB1 and GRIB2 messages into GRIB files of the same format and populate the MySQL/MariaDB database with GRIB1/GRIB2 metadata, use the [gribadmin](#) and [DECINFO](#) commands below.

Create the MySQL/MariaDB database by running the command below from a Unix shell prompt. The command will prompt for a password. Enter the password for the MySQL/MariaDB root user, which was set by the person that installed MySQL/MariaDB.

Type: **gribadmin makedb**

If you have problems running this command, review the *Configuration Sections* section of the gribadmin script. It contains variables that change its behavior. In particular, the mysql_path and gcc_lib_path variables may need to be changed.

Activate the DMBIN data monitor by running the command below from a McIDAS-X session running under the oper account.

Type: **DECINFO SET DMBIN ACTIVE**

DMCOND Data Monitor for LDM CONDUIT Feed GRIB2 Files (Optional)

If you will be decoding CONDUIT products in a separate spool file, you must activate the data monitor. Otherwise, the CONDUIT data will be filed using the DMBIN data monitor. First, you must recreate the DECINFO.DAT file to add DMCOND to the list of valid data monitors.

1. Run the command below from the McIDAS-X Text and Command window to list the current active/inactive status of each of the data monitors. This information will be used in steps 5 and 6 below.

Type: **DECINFO LIST**

2. Run the command below from the Unix prompt to remove the existing DECINFO.DAT file.

Type: **rm ~oper/mcidas/data/DECINFO.DAT**

3. Run the command below from the McIDAS-X Text and Command window to recreate the DECINFO.DAT file.

Type: **DECINFO**

4. Run the command below from the McIDAS-X Text and Command window to activate the DMCOND data monitor.

Type: **DECINFO SET DMCOND ACTIVE**

5. Run the command below from the McIDAS-X Text and Command window to list the updated active/inactive status of each of the data monitors.

Type: **DECINFO LIST**

6. If the active/inactive status of any of the data monitors is different than it was in step 1 above, use the [DECINFO command](#) (**DECINFO SET *monitor action***) to reset them to the proper status.

Creating a Separate Spool File for the CONDUIT Decoder (Optional)

This procedure is optional and is available only to sites that receive data via the LDM. The procedure increases throughput for GRIB messages by creating a separate spool file for the CONDUIT feed. If you are doing both the CONDUIT and NGRID LDM feeds in the same spool file, the tendency for dropped messages is possible.

Complete the steps below to implement the procedure.

1. Run the command below from the McIDAS-X Text and Command window to stop the DMCOND data monitor.

Type: **DECINFO SET DMCOND INACTIVE**

2. Run the command below from the McIDAS-X Text and Command window to add the new NCON circuit.

Type: **CIRCUIT ADD NCON INGESTOR=INGEBIN CONFIG=CONDUIT.CFG SPOOL=CONDUIT.SPL "NOAAPORT NWSSTG CONDUIT Binary Circuit**

3. Run the command below from the McIDAS-X Text and Command window to define the full Unix path of the CONDUIT.SPL file.

Type: **REDIRECT ADD CONDUIT.SPL *"/data_directory_path***

For example: **REDIRECT ADD CONDUIT.SPL *"/home/oper/xcd_data***

4. Modify the LDM configuration so that it pipes CONDUIT data to another instance of the binary ingestor.

a. Open the file **~ldm/etc/pqact.conf** for editing.

b. Create a separate entry in the *Request Entries* section for CONDUIT data and set it to pipe the data to the binary ingestor using the NCON circuit. First check if there is an entry that contains CONDUIT as well as other binary feeds such as HRS; if there is, remove CONDUIT from that entry. Then add a new entry like that in the box below, replacing its spaces with tabs (a requirement for the pqact.conf file).

```
CONDUIT ^.* PIPE /home/oper/mcidas/bin/ingebin.k NCON
```

c. Save and close the file.

5. Run the command below from the Unix prompt to remove the old pointer file for the CONDUIT decoder.

Type: **rm ~oper/mcidas/data/CONDUIT.PRO**

6. Run the command below from the Unix prompt to stop and restart the LDM.

Type: **ldmadmin restart**

7. Run the command below from the McIDAS-X Text and Command window to activate the DMCOND data monitor.

Type: **DECINFO SET DMCOND ACTIVE**

You should notice very soon after starting the CONDUIT decoder that there is another entry in the ingestor section of the [statdisp display](#). Three instances of the binary ingestor (INGEBIN) will exist if you are doing the optional NEXRAD decoder as well. The CONDUIT data should now be written to CONDUIT.SPL by the binary ingestor using the NCON circuit.

Configuring Data Removal

As new data is decoded and filed by McIDAS-XCD your disk will begin to fill up. Complete the steps below to free up disk space by removing old data and rotating the -XCD log files.

Removing MD, GRID and Text Files

There are two ways to configure the removal of MD, GRID and text files. If you have a continuous McIDAS session running in the oper account, you may use the McIDAS-X scheduler. If you prefer to configure the data to be removed in the background without a McIDAS session running, you may use the Unix cron command. Command [QRTMDG](#) deletes old MD and grid files generated by the decoders; command [DELWXT](#) deletes old text files generated by the ingestors.

Using the McIDAS Scheduler

1. From a McIDAS-X session running under the oper account, verify that there is an active SKED process.

Type: ?

If the resulting list of processes does **not** include a line containing "SKED SKEDFILE", start a SKED process with the command below.

Type: **SKED SKEDFILE**

2. Enter the four commands below in the McIDAS-X local time scheduler to delete old data files. The variable *nn* represents the number of days of data to keep online.

Type: **SKE #Y 00:01:00 999999 24 "QRTMDG MD 1 90 nn DEV=NCN**

Type: **SKE #Y 00:01:00 999999 24 "DELWXT nn DEV=NCN**

Type: **SKE #Y 00:01:00 999999 24 "QRTMDG GRID 50001 100000 nn DEV=NCN**

Type: **SKU ON**

Using the Unix cron Command

1. From a shell prompt under the oper account, edit the current cron job list.

Type: crontab -e

2. Enter the three commands below on new lines in the editor. The variables *mm* and *hh* represent the minute and hour of the day to run the [QRTMDG](#) and [DELWXT](#) commands. The variable *nn* represents the number of days of data to keep online.

Type: *mm hh* * * * /home/oper/mcidas/bin/qrtdmg.k MD 1 90 nn DEV=NCN

Type: *mm hh* * * * /home/oper/mcidas/bin/delwxt.k nn DEV=NCN

Type: *mm hh* * * * /home/oper/mcidas/bin/qrtdmg.k GRID 50001 100000 nn DEV=NCN

Removing GRIB and BUFR Files

1. From a shell prompt under the oper account, edit the current cron job list.

Type: crontab -e

2. Enter the two commands below on new lines in the editor. The variables *mm* and *hh* represent the minute and hour of the day to run the [xcdscour](#) command. The variable *nn* represents the number of days of data to keep online.

Type: *mm hh* * * * /home/oper/mcidas/bin/xcdscour GRIB *nn path_to_grib_files*

Type: *mm hh* * * * /home/oper/mcidas/bin/xcdscour BUFR *nn path_to_bufr_files*

The xcdscour command deletes old GRIB1, GRIB2 and BUFR files generated by the decoders. If the GRIB decoder is configured to use a MySQL/MariaDB database the corresponding GRIB1 and GRIB2 database entries will be removed.

Rotating the -XCD Log Files

1. From a shell prompt under the oper account, edit the current cron job list.

Type: crontab -e

2. Enter the two commands below on new lines in the editor. The first entry runs the [xcdadmin](#) command hourly; the -t option prints the current time, disk usage, and load averages to the xcd.log file. The second entry runs the xcdadmin command once per day; the -r option rotates the log file. See [Using xcdadmin from a Unix Shell Prompt](#) to confirm that you are using the most recent version of xcdadmin.

Type: 0 * * * * /home/oper/mcidas/bin/xcdadmin log -t

Type: 0 0 * * * * /home/oper/mcidas/bin/xcdadmin log -r

Configuring the Default ADDE Datasets

This section describes how to configure the McIDAS-XCD workstation to serve -XCD data using McIDAS ADDE. McIDAS-X includes ADDE servers to provide users with data that has been decoded and stored by McIDAS-XCD.

1. Log on to the workstation as the user mcadde.
2. Start a McIDAS-X session.

Type: mcidas

3. Create a McIDAS string named MCDATA that contains the fully expanded path to the directory that you configured -XCD to store data. This is the same directory that you specified in step 9 of [Loading the McIDAS-XCD Software](#).

Type: **TE MCDATA** *"/data_directory_path*

For example: **TE MCDATA** *"/home/oper/xcd_data*

In order for -XCD to correctly file data in this example, the */home/oper/xcd_data*, */home/oper/xcd_data/grib*, and */home/oper/xcd_data/bufr* directories need to exist. Create the *#MCDATA*, *#MCDATA/grib*, and *#MCDATA/bufr* directories, if necessary.

4. Create McIDAS strings named WSRDATA and TDWRDATA that contain the fully expanded paths to the directories that you configured -XCD to store WSR-88D and TDWR data. These are the same directories that you specified for the values of WSRFILE= and TDWRFILE=, respectively, in step 1 of [Configuring the NEXRAD Decoder](#).

Type: **TE WSRDATA** *"/wsr_data_directory_path*

Type: **TE TDWRDATA** *"/tdwr_data_directory_path*

Example: **TE WSRDATA** *"/home/oper/xcd_data/wsr*

Example: **TE TDWRDATA** *"/home/oper/xcd_data/tdwr*

5. Run the batch file XCD.BAT to create file redirections and initialize the files GROUPS.DAT and COUNTRY.DAT. Running this batch file may overwrite non-default file redirections that you previously created. Therefore, we recommend that you make a backup copy of the file *~mcadde/mcidas/data/LWPATH.NAM* before running the command below.

Type: **BATCH XCD.BAT**

6. Run the batch file XCDADDE.BAT to create the real-time ADDE datasets listed in the tables below. Running this batch file creates all of the default datasets for the groups RTGRIDS, RTPTSRC, RTWXTEXT, RADAR, TDWR and WSR. It will therefore overwrite any locally-created datasets with names matching those in the batch file. Thus, we recommend that you make a backup copy of the file *~mcadde/mcidas/data/RESOLV.SRV* before running the command below.

Type: **BATCH XCDADDE.BAT**

Notes:

- o SSEC tests and supports real-time -XCD grid datasets only in GRIB format. Thus, the default for the RTGRIDS datasets is to contain GRIB1 and GRIB2 files, which are defined in XCDNOAAPORTGRID.BAT. If you want the RTGRIDS datasets to also contain McIDAS GRID files (in addition to GRIB1 and GRIB2) you must activate the GRID data monitor with command **DECINFO SET DMGRID ACTIVE** and create and run a batch file that defines different dataset names for the McIDAS GRID files. The batch file XCDGRID.BAT [last updated April 2012] can be used as a template since it shows the correct format for DSSERVE commands of McIDAS GRID datasets. If you choose to use it as a template, make sure you change the dataset names so that none of them are the same as the existing RTGRIDS datasets. As noted in [Creating McIDAS Grid Files with DMGRID](#), you can only make McIDAS GRID format datasets from GRIB1 messages (not GRIB1 and GRIB2 messages).
- o The command **BATCH XCDCONDUITGRID.BAT**, which is in XCDADDE.BAT but is not run because it is commented out, creates ADDE datasets for the subset of CONDUIT LDM GRIB2 grids that the SSEC Data Center provides for its users. If you want to make those datasets for your users, you can uncomment the line (remove the "REM" at the beginning of the line) in XCDADDE.BAT or run the command (**BATCH XCDCONDUITGRID.BAT**) by hand. If you want to make additional GRIB2 dataset changes (i.e., change or add datasets), see [Configuring Additional GRIB2 ADDE Datasets](#).

The default datasets listed in the five tables below are created by running batch files provided with McIDAS-XCD. They can be created either by running the XCDADDE.BAT batch file or the set of batch files that XCDADDE.BAT runs (XCDRTPT.BAT, XCDRTWX.BAT, XCDRADAR.BAT, XCDTDWR.BAT, XCDWSR.BAT, and XCDNOAAPORTGRID.BAT; XCDCONDUITGRID.BAT is commented out but can be optionally run).

The datasets below are created by the XCDRTPT.BAT batch file (which can be run separately, or as part of XCDADDE.BAT).

ADDE Dataset Name	Data Type	MD Files	Description
RTPTSRC/SFCHOURLY	POINT	1-10	real-time surface hourly data

RTPTSRC/UPPERMAND	POINT	11-20	real-time mandatory level RAOB data
RTPTSRC/UPPERSIG	POINT	21-30	real-time significant level RAOB data
RTPTSRC/SHIPBUOY	POINT	31-40	real-time ship and buoy surface reports
RTPTSRC/TEMPSHIP	POINT	41-50	real-time mandatory level TEMP SHIP (RAOB) data
RTPTSRC/SYNOPTIC	POINT	51-60	real-time synoptic data
RTPTSRC/AIRCRAFT	POINT	61-70	real-time AIREP and PIREP data
RTPTSRC/NAMMOS	POINT	71-80	real-time NAM MOS data
RTPTSRC/GFSMOS	POINT	81-90	real-time GFS MOS data

The datasets below are created by the XCDRTWX.BAT batch file (which can be run separately, or as part of XCDADDE.BAT).

ADDE Dataset Name	Data Type	Files	Description
RTWXTEXT/NAMMOS	TEXT (OBTX)	--	default dataset name used by the MOSRPT command
RTWXTEXT/GFSMOS	TEXT (OBTX)	--	default dataset name used by the MOSRPT command
RTWXTEXT/SFCHOURLY	TEXT (OBTX)	--	default dataset name used by the SFCRPT command
RTWXTEXT/SYNOPTIC	TEXT (OBTX)	--	default dataset name used by the SYNRP command
RTWXTEXT/TERMFCST	TEXT (OBTX)	--	default dataset name used by the TAFRPT command
RTWXTEXT/UPPERAIR	TEXT (OBTX)	--	default dataset name used by the RAOBRPT command

The datasets below are created by the XCDRADAR.BAT batch file (which can be run separately, or as part of XCDADDE.BAT).

ADDE Dataset Name	Data Type	Files	Description
RADAR/TDWR-BREF1	IMAGE	--	TDWR Base Reflec - tilt 1 (0.1-0.8deg) 48nmi (TZ0)
RADAR/TDWR-BREF2	IMAGE	--	TDWR Base Reflec - tilt 2 (1.0deg) 48nmi (TZ1)
RADAR/TDWR-BREF3	IMAGE	--	TDWR Base Reflec - tilt 3 (1.6-3.7deg) 48nmi (TZ2)
RADAR/TDWR-BVEL1	IMAGE	--	TDWR Base Rad Veloc - tilt 1 (0.1-0.8deg) 48nmi (TV0)
RADAR/TDWR-BVEL2	IMAGE	--	TDWR Base Rad Veloc - tilt 2 (1.0deg) 48nmi (TV1)
RADAR/TDWR-BVEL3	IMAGE	--	TDWR Base Rad Veloc - tilt 3 (1.6-3.7deg) 48nmi (TV2)
RADAR/TDWR-CREF	IMAGE	--	TDWR Composite Reflectivity - 124nmi (NCR)
RADAR/TDWR-DHR	IMAGE	--	TDWR Digital Hybrid Scan Reflectivity (DHR)
RADAR/TDWR-DPA	IMAGE	--	TDWR Digital Precipitation Array (DPA)
RADAR/TDWR-DSP	IMAGE	--	TDWR Digital Storm Total Precipitation (DSP)
RADAR/TDWR-LRREF	IMAGE	--	TDWR Long Range Reflectivity - 225nmi (TZL)
RADAR/TDWR-PCP1	IMAGE	--	TDWR Surface Rainfall - 1 hour running total (N1P)
RADAR/TDWR-PCPT	IMAGE	--	TDWR Surface Rainfall - storm total (NTP)
RADAR/TDWR-TOPS	IMAGE	--	TDWR Echo Tops (NET)

RADAR/TDWR-VIL	IMAGE	--	TDWR Vertical Integrated Liquid (NVL)
RADAR/WSR-BREF1	IMAGE	--	WSR Base Reflec - tilt 1 (0.5deg) 248nmi (N0Q)
RADAR/WSR-BREF2	IMAGE	--	WSR Base Reflec - tilt 2 (0.9deg) 248nmi (NAQ)
RADAR/WSR-BREF3	IMAGE	--	WSR Base Reflec - tilt 3 (1.3 1.5deg) 248nmi (N1Q)
RADAR/WSR-BREF4	IMAGE	--	WSR Base Reflec - tilt 4 (1.8deg) 248nmi (NBQ)
RADAR/WSR-BREF5	IMAGE	--	WSR Base Reflec - tilt 5 (2.4 2.5deg) 248nmi (N2Q)
RADAR/WSR-BREF6	IMAGE	--	WSR Base Reflec - tilt 6 (3.1 3.4 3.5deg) 248nmi (N3Q)
RADAR/WSR-BVEL1	IMAGE	--	WSR Base Rad Vel - tilt 1 (0.5deg) 162nmi (N0U)
RADAR/WSR-BVEL2	IMAGE	--	WSR Base Rad Vel - tilt 2 (0.9deg) 162nmi (NAU)
RADAR/WSR-BVEL3	IMAGE	--	WSR Base Rad Vel - tilt 3 (1.3 1.5deg) 162nmi (N1U)
RADAR/WSR-BVEL4	IMAGE	--	WSR Base Rad Vel - tilt 4 (1.8deg) 162nmi (NBU)
RADAR/WSR-BVEL5	IMAGE	--	WSR Base Rad Vel - tilt 5 (2.4 2.5deg) 162nmi (N2U)
RADAR/WSR-BVEL6	IMAGE	--	WSR Base Rad Vel - tilt 6 (3.1 3.4 3.5deg) 162nmi (N3U)
RADAR/WSR-CREF	IMAGE	--	WSR Composite Reflectivity - 124nmi (NCR)
RADAR/WSR-DHR	IMAGE	--	WSR Digital Hybrid Scan Reflectivity (DHR)
RADAR/WSR-DPA	IMAGE	--	WSR Digital Precipitation Array (DPA)
RADAR/WSR-DSP	IMAGE	--	WSR Digital Storm Total Precipitation (DSP)
RADAR/WSR-DVIL	IMAGE	--	WSR Digital Vertical Integrated Liquid (DVL)
RADAR/WSR-ENHTOPS	IMAGE	--	WSR Enhanced Echo Tops (EET)
RADAR/WSR-PCP1	IMAGE	--	WSR Surface Rainfall - 1 hour running total (N1P)
RADAR/WSR-PCPT	IMAGE	--	WSR Surface Rainfall - storm total (NTP)
RADAR/WSR-SRVEL1	IMAGE	--	WSR Storm-Rel Mean Rad Vel - tilt 1 (0.5deg) (N0S)
RADAR/WSR-SRVEL2	IMAGE	--	WSR Storm-Rel Mean Rad Vel - tilt 2 (1.3 5deg) (N1S)
RADAR/WSR-SRVEL3	IMAGE	--	WSR Storm-Rel Mean Rad Vel - tilt 3 (2.4 5deg) (N2S)
RADAR/WSR-SRVEL4	IMAGE	--	WSR Storm-Rel Mean Rad Vel - tilt 4 (3.1 3.4 3.5deg) (N3S)
RADAR/WSR-TOPS	IMAGE	--	WSR Echo Tops (NET)
RADAR/WSR-VIL	IMAGE	--	WSR Vertical Integrated Liquid (NVL)
RADAR/WSR-DREF1	IMAGE	--	WSR Diff Reflec - tilt 1 (0.5deg) 248nmi (N0X)
RADAR/WSR-DREF2	IMAGE	--	WSR Diff Reflec - tilt 2 (0.9deg) 248nmi (NAX)
RADAR/WSR-DREF3	IMAGE	--	WSR Diff Reflec - tilt 3 (1.3 1.5deg) 248nmi (N1X)
RADAR/WSR-DREF4	IMAGE	--	WSR Diff Reflec - tilt 4 (1.8deg) 248nmi (NBX)
RADAR/WSR-DREF5	IMAGE	--	WSR Diff Reflec - tilt 5 (2.4 2.5deg) 248nmi (N2X)

RADAR/WSR-DREF6	IMAGE	--	WSR Diff Reflec - tilt 6 (3.1 3.4 3.5deg) 248nmi (N3X)
RADAR/WSR-COR1	IMAGE	--	WSR Correlation Coef - tilt 1 (0.5deg) (N0C)
RADAR/WSR-COR2	IMAGE	--	WSR Correlation Coef - tilt 2 (0.9deg) (NAC)
RADAR/WSR-COR3	IMAGE	--	WSR Correlation Coef - tilt 3 (1.3 1.5deg) (N1C)
RADAR/WSR-COR4	IMAGE	--	WSR Correlation Coef - tilt 4 (1.8 deg) (NBC)
RADAR/WSR-COR5	IMAGE	--	WSR Correlation Coef - tilt 5 (2.4 2.5deg) (N2C)
RADAR/WSR-COR6	IMAGE	--	WSR Correlation Coef - tilt 6 (3.1 3.4 3.5deg) (N3C)
RADAR/WSR-SDP1	IMAGE	--	WSR Spec Diff Phase - tilt 1 (0.5deg) (N0K)
RADAR/WSR-SDP2	IMAGE	--	WSR Spec Diff Phase - tilt 2 (0.9deg) (NAK)
RADAR/WSR-SDP3	IMAGE	--	WSR Spec Diff Phase - tilt 3 (1.3 1.5deg) (N1K)
RADAR/WSR-SDP4	IMAGE	--	WSR Spec Diff Phase - tilt 4 (1.8 deg) (NBK)
RADAR/WSR-SDP5	IMAGE	--	WSR Spec Diff Phase - tilt 5 (2.4 2.5deg) (N2K)
RADAR/WSR-SDP6	IMAGE	--	WSR Spec Diff Phase - tilt 6 (3.1 3.4 3.5deg) (N3K)
RADAR/WSR-HMC1	IMAGE	--	WSR Hydrometeor Class - tilt 1 (0.5deg) (N0H)
RADAR/WSR-HMC2	IMAGE	--	WSR Hydrometeor Class - tilt 2 (0.9deg) (NAH)
RADAR/WSR-HMC3	IMAGE	--	WSR Hydrometeor Class - tilt 3 (1.3 1.5deg) (N1H)
RADAR/WSR-HMC4	IMAGE	--	WSR Hydrometeor Class - tilt 4 (1.8 deg) (NBH)
RADAR/WSR-HMC5	IMAGE	--	WSR Hydrometeor Class - tilt 5 (2.4 2.5deg) (N2H)
RADAR/WSR-HMC6	IMAGE	--	WSR Hydrometeor Class - tilt 6 (3.1 3.4 3.5deg) (N3H)
RADAR/WSR-HHC	IMAGE	--	WSR Hybrid Scan Hydrometeor Classification (HHC)
RADAR/WSR-OHA	IMAGE	--	WSR One Hour Accumulation (OHA)
RADAR/WSR-PTA	IMAGE	--	WSR Storm Total Accumulation (PTA)
RADAR/WSR-DAA	IMAGE	--	WSR Digital Accumulation Array (DAA)
RADAR/WSR-DTA	IMAGE	--	WSR Digital Storm Total Accumulation (DTA)
RADAR/WSR-DOD	IMAGE	--	WSR Digital One Hour Difference (DOD)
RADAR/WSR-DSD	IMAGE	--	WSR Digital Storm Total Difference (DSD)
* The 15 datasets created by the XCDTDWR.BAT file are the same as the RADAR/TDWR-* datasets in the XCDRADAR.BAT table above, except that the XCDTDWR.BAT datasets are named TDWR/* rather than RADAR/TDWR-*.			
* Likewise, the 57 datasets created by the XCDWSR.BAT file are the same as the RADAR/WSR-* datasets in the XCDRADAR.BAT table above, except that the XCDWSR.BAT datasets are named WSR/* rather than RADAR/WSR-*.			

The datasets below are among the GRIB1 datasets created by the XCDNOAAPORTGRID.BAT batch file (which can be run separately, or as part of XCDADDE.BAT).

ADDE Dataset Name	Data Type	Files	Description
RTGRIDS/ALL	GRID	--	all GRIB1 model grids decoded by McIDAS-XCD

RTGRIDS/ALL-GRIB1	GRID	--	all GRIB1 model grids decoded by McIDAS-XCD
RTGRIDS/AWC-ALL	GRID	--	all National Convective Weather Diagnostic grids
RTGRIDS/AWR-ALL	GRID	--	all Alaska Waters Regional Wave Model grids
RTGRIDS/ENP-ALL	GRID	--	all Eastern North Pacific Regional Wave Model grids
RTGRIDS/FFG-ALL	GRID	--	all NWS Flash Flood Guidance System grids
RTGRIDS/GFS-ALL*	GRID	--	all Global Forecast System GRIB1 grids
RTGRIDS/ICA-ALL*	GRID	--	all Ice Concentration Analysis grids
RTGRIDS/MDR-ALL	GRID	--	all Manually Digitized Radar grids
RTGRIDS/NAM-ALL*	GRID	--	all North American Mesoscale model GRIB1 grids
RTGRIDS/NCE-ALL	GRID	--	all NCEP quantitative precipitation forecast grids
RTGRIDS/RAP-ALL*	GRID	--	all Rapid Refresh model GRIB1 grids
RTGRIDS/UKM-ALL*	GRID	--	all UKMT Coastal Ocean Circulation grids
RTGRIDS/WHG-ALL	GRID	--	all significant height of wind waves grids
RTGRIDS/WNA-ALL	GRID	--	all Western North Atlantic Regional Wave Model grids
RTGRIDS/WWF-ALL*	GRID	--	all Global Wind-Wave Forecast grids
RTGRIDS/OVERFLOW	GRID	--	all miscellaneous grids for which no dataset has been defined
* These RTGRIDS datasets also contain subset datasets which make the task of locating and using specific grids easier and more efficient. See the file XCDNOAAPORTGRID.BAT for a complete listing of the RTGRIDS datasets and their contents.			

The datasets below are among the GRIB2 datasets created by the XCDNOAAPORTGRID.BAT batch file (which can be run separately, or as part of XCDADDE.BAT).

ADDE Dataset Name	Data Type	Files	Description
RTGRIDS/ALL-GRIB2	GRID	--	all GRIB2 grids decoded by McIDAS-XCD
RTGRIDS/ETSS-ALL*	GRID	--	all Extratropical Storm Surge GRIB2 grids
RTGRIDS/GFS-ALLGRIB2*	GRID	--	all Global Forecast System GRIB2 grids
RTGRIDS/HREF-ALL*	GRID	--	all High Resolution Ensemble Forecast GRIB2 grids
RTGRIDS/ICI-ALL*	GRID	--	all Current Icing Potential GRIB2 grids
RTGRIDS/LMP-ALL*	GRID	--	all Local AWIPS MOS Program GRIB2 grids
RTGRIDS/MGWM-ALL*	GRID	--	all Global Multi-Grid Wave Model GRIB2 grids
RTGRIDS/MOS-ALL*	GRID	--	all Model Output Statistics GRIB2 grids
RTGRIDS/NAM-ALLGRIB2*	GRID	--	all North American Mesoscale model GRIB2 grids
RTGRIDS/NBM-ALL*	GRID	--	all National Blended Model GRIB2 grids
RTGRIDS/NDF-ALL*	GRID	--	all National Digital Forecast Database GRIB2 grids
RTGRIDS/NMMB-ALL*	GRID	--	all Non-hydrostatic Mesoscale Model GRIB2 grids

RTGRIDS/PETS-ALL*	GRID	--	all Probabilistic Extratropical Storm surge GRIB2 grids
RTGRIDS/PSS-ALL*	GRID	--	all Probabilistic Storm Surge GRIB2 grids
RTGRIDS/RAP-ALLGRIB2*	GRID	--	all Rapid Refresh model GRIB2 grids
RTGRIDS/RTM-ALL*	GRID	--	all Real-Time Mesoscale analysis GRIB2 grids
RTGRIDS/RTOF-ALL*	GRID	--	all Real-Time Ocean Forecast system GRIB2 grids
RTGRIDS/SREF-ALL*	GRID	--	all Short Range Ensemble Forecast GRIB2 grids
RTGRIDS/SST-ALL*	GRID	--	all Sea Surface Temperature model GRIB2 grids
RTGRIDS/URMA-ALL*	GRID	--	all Unrestricted Mesoscale Analysis model GRIB2 grids
RTGRIDS/WRFE-ALL*	GRID	--	all Eulerian Weather Research and Forecasting model GRIB2 grids
RTGRIDS/WRFN-ALL*	GRID	--	all Weather Research and Forecasting model GRIB2 grids

* These RTGRIDS datasets also contain subset datasets which make the task of locating and using specific grids easier and more efficient. See the file XCDNOAAPORTGRID.BAT for a complete listing of the RTGRIDS datasets and their contents.

Configuring Additional GRIB2 ADDE Datasets

Adding a new GRIB2 dataset that accesses data in the MySQL/MariaDB database is a multi-step process. You must first access the navigation parameters of the GRIB2 file, then add these parameters to the `~mcidas/data/grib2NavLookup.txt` file, use the **DSSERVE** command to add the dataset to your `RESOLV.SRV` file, and then stop and restart `McIDAS-XCD` for your changes to take effect.

1. From a `McIDAS` session as user **oper**, run the [GRIB2NAV](#) command to list the navigation parameters of the GRIB2 file.

Type: **GRIB2NAV GRIB2Filename**

where *GRIB2Filename* is the fully qualified path of a valid GRIB2 file on your system.

Example output is:

```
Nav parameters for message 66, field 0 are NAM, 6, -4564482, 2089981, 20318, 950000, 250000, 250000
```

2. If the navigation parameters already exist in the `~mcidas/data/grib2NavLookup.txt` file, then you can either edit the first column to match your new dataset descriptor, leave the dataset as is and do nothing, or change the **DATAFILE=DESCRIPTOR** option of the **DSSERVE** command in step 3 below to match the first column of the navigation entry.

If the navigation parameters do not exist in the `~mcidas/data/grib2NavLookup.txt` file, then you will need to add them by logging in as user **mcidas** and updating the file. The format of this data file is as follows:

Column 1 is the filename prefix for the GRIB2 file created by `McIDAS-XCD` (12 character maximum). It is best if this is also the dataset descriptor name for the new dataset.

Column 2 is the grid source name (4 character maximum).

Column 3 is the projection type (1 and 4 are Mercator, 2 is Polar Stereographic, 6 is Lambert Conformal).

Columns 4-9 are the navigation parameters calculated by the **GRIB2NAV** command.

From the example above, the new navigation entry you should add to the `~mcidas/data/grib2NavLookup.txt` file would be:

```
NAM-USLCAW20 | NAM | 6 | -4564482 | 2089981 | 20318 | 950000 | 250000 | 250000
```

Note: If you update the `~mcidas/data/grib2NavLookup.txt` file, make a backup copy of the updated file so you can restore your local changes if the original file is overwritten (e.g., by installing a new version of `McIDAS-X`).

3. From a `McIDAS` session in the **mcadde** account, use the **DSSERVE** command to add the dataset to the `RESOLV.SRV` file.

Type: **DSSERVE ADD SAMPLE/NAM-USLCAW20 GRIB 1 999999 TYPE=GRID RT=Y INFO="" DATABASE=mcrtgrib DATAFILE=DESCRIPTOR GRIBTYPE=2" "NAM - CONUS/LC; AWIPS 20 km**

DATABASE=mcrtgrib defines the database where the meta data is stored.

DATAFILE=DESCRIPTOR indicates that this dataset will contain only the data files that match the descriptor name (in this case, `NAM-USLCAW20*` files). If you have a different name in column 1 of your `~mcidas/data/grib2NavLookup.txt` file, then you can include that name here in place of the word **DESCRIPTOR**.

GRIBTYPE=2 will only search for GRIB2 data files.

4. Stop and restart `McIDAS-XCD` for your changes to the `~mcidas/data/grib2NavLookup.txt` file to take effect in filing and in the database. If you need instructions, see the [Starting and Stopping McIDAS-XCD](#) section.
-

Starting and Stopping McIDAS-XCD

There are two ways to start and stop running McIDAS-XCD. If you prefer to have a continuous McIDAS session running in the oper account, you may use the McIDAS-XCD [STARTXCD](#) command in your McIDAS-X session. If you prefer to configure McIDAS-XCD without a McIDAS-X session running, you may use the [xcdadmin](#) command from a Unix shell prompt. If your data source is the LDM, you should start LDM before starting McIDAS-XCD, and stop LDM after stopping McIDAS-XCD.

Using STARTXCD in a McIDAS-X Session

Use the STARTXCD command as described below to start the McIDAS-XCD software. For more information about STARTXCD, see Chapter 3, [McIDAS-XCD Administrative Commands](#).

From a McIDAS-X session running under the oper account, start the McIDAS-XCD software with the command below.

Type: **STARTXCD**

Only **one** STARTXCD command can be run at a time. Do not include the command STARTXCD in your STARTUP.SYS file.

The STARTXCD command runs continuously in your McIDAS-X session, starting and stopping data monitors and ingestors as needed. If a data monitor or ingestor stops, STARTXCD automatically restarts it. If you cancel STARTXCD, also cancel the associated data monitors and ingestors. If you exit McIDAS-X, your decoders and ingestors will stop running.

Use the DECINFO command to activate and deactivate the data monitors. The STARTXCD process (which is continuously running) will check the status of the monitors and start or stop them accordingly.

To activate or deactivate an individual decoder within a data monitor, you must first deactivate the data monitor. Once the data monitor is no longer active, the status of the decoder(s) can be changed using the DECINFO command. Then use DECINFO to reactivate the data monitor using the new settings.

When you need to stop McIDAS-XCD, the best method is to exit the McIDAS-X session and wait for all -XCD related processes to exit.

Using xcdadmin from a Unix Shell Prompt

From a shell prompt under the oper account, start the McIDAS-XCD software with the xcdadmin command below.

Type: xcdadmin start

The xcdadmin start command runs the STARTXCD -XCD command in a mcenv session and behaves exactly as described in the section above, [Using STARTXCD in a McIDAS-X Session](#).

When you need to stop McIDAS-XCD, run the command below from a shell prompt under the oper account to stop all -XCD related processes.

Type: xcdadmin stop

The xcdadmin stop command kills the startxcd.k process. It then waits 30 seconds for all of the -XCD data monitors to exit. At the end of the 30 seconds, it kills the remaining data monitors. You should verify that all of the data monitors were stopped, and kill the processes, if necessary.

Type: ps -ef | grep DM

Type: kill -9 *process_ID*

Checking the Filing Status of McIDAS-XCD Data

To check the status of the McIDAS-XCD ingestors and data monitors, use the McIDAS-XCD [statdisp](#) command. To check that grib file information is being filed to the real-time GRIB MySQL/MariaDB database (mcrtgrib), use the McIDAS-XCD [gribadmin](#) command.

Starting the Status Display Window

The Status Display window gives up to the minute data flow information for the McIDAS-XCD ingestors and data monitors. To start the Status Display window use the `statdisp` command as described below. For more information about `statdisp`, see Chapter 3, [McIDAS-XCD Administrative Commands](#).

To display the Status Display window from the McIDAS-X Text and Command Window,

Type: **OS "statdisp &**

To display it from an Unix shell prompt,

Type: `statdisp &`

Note: The ampersand (&) runs the `statdisp` process in the background.

Checking the Real-time GRIB Database

If you are filing GRIB1 and GRIB2 messages into GRIB files of the same format and populating the MySQL/MariaDB database, the `gribadmin` script can list the most recent record filed. For more information about `gribadmin`, see Chapter 3, [McIDAS-XCD Administrative Commands](#).

To list the most recent record filed in the `mcrtgrib` MySQL/MariaDB database from a Unix shell prompt,

Type: `gribadmin latest`

The `modtime` is the last field listed. If the GRIB file information is being filed correctly, this should be today's date and a very recent time.

Uninstalling the Previous Version of McIDAS-XCD

Use the steps below to uninstall the previous version of McIDAS-XCD from your workstation. The instructions below use 2019.1 as an example of the version being uninstalled. If you are uninstalling a different version, specify it in place of "2019.1" in the instructions below.

1. Open a Unix window and log on to the workstation as user oper.

2. Change to the ~oper/mcidas directory.

Type: `cd ~oper/mcidas`

3. Run the shell script `xcd_init` to initialize the environment variables needed to remove the McIDAS-XCD 2019.1 package.

Type: `./xcd2019.1/xcd_init`

4. Run the shell script `xcd2019.1.sh` option to remove the McIDAS-XCD 2019.1 source files, data files and help files.

Type: `sh ./xcd2019.1.sh uninstall all`

5. Remove the client redirections by entering the command below from the McIDAS-X Text and Command Window.

Type: **BATCH RMXCDDEC.BAT**

6. Switch to user mcidas.

Type: `su mcidas`

7. Run the shell script `xcd2019.1.sh` option to remove the weather text and observation servers.

Type: `sh ./xcd2019.1.sh uninstall addservers`

8. Exit from user mcidas.

Type: `exit`

9. Remove the McIDAS-XCD 2019.1 installation files.

Type: `rm xcd2019.1.tar.Z`

Type: `rm xcd_README_2019.1`

Type: `rm xcd2019.1.sh`

Chapter 3

McIDAS-XCD Administrative Commands

This chapter contains command documentation for the system configuration of McIDAS-XCD, including administrative commands for file management and data availability. All of the user commands for accessing conventional data are distributed with McIDAS-X. See the [McIDAS User's Guide](#) for more information.

Only authorized administrative staff should use these commands. To run them, you must be logged on as oper. If the error message "Permission Denied" is displayed, your logon does not correspond to the logon in the installation procedure. See Chapter 2, [Installing and Configuring McIDAS-XCD](#), for more information.

The administrative commands are listed in alphabetical order below with a short description of their function.

BILDTEXT	builds the rapid access pointer and text files
CHKERR	lists the output from an error file
CIRCUIT	data circuit utility
DATACQ	plots data availability from MD files
DATARECV	plots MD file data on a multiple-panel display
DECINFO	decoder utility
DELWXT	deletes weather text and index files
GRIB2NAV	lists the navigation parameters necessary for entering new GRIB2 datasets
IDGROUP	ID group utility
IDMON	station ID monitoring utility
NMCAMT	lists the number of real-time grids received
QRTMDG	deletes real-time grid or MD files
SIGCO	significant level upper air storage utility
STARTXCD	starts the ingestor and decoder programs
STAT	lists the decoder and ingestor status
SUBGRD	creates geographic subsectors of Mercator grids
UPDIDS	updates the station reporting list
WMORTE	maintains a data routing table of WMO headers
gribadmin	Unix command for managing the MySQL/MariaDB database used by the GRIB server
statdisp	Unix command for displaying the status window
xcdadmin	Unix command for managing the XCD status and the log file
xcdscour	Unix command for deleting real-time GRIB or BUFR files, and their associated database entries.

BILDTEXT

Builds the rapid access pointer and text files for observational data.

Formats

BILDTEXT ADD *id pfile*
BILDTEXT DEL *id pfile*
BILDTEXT INIT *pfile tfile maxsta maxreps idtype maxobs minhrs nbytes decnam idfile maxtxt [keyword]*
BILDTEXT LIST *pfile*

Parameters

ADD adds a station to an existing pointer file

DEL deletes a station from an existing pointer file

INIT initializes the *pfile* and deletes the existing *pfile* and *tfile*

LIST lists the configuration of a pointer file

id station ID to add or delete

pfile pointer file name (no default)

tfile text file name (no default)

maxsta maximum number of stations to store

maxreps maximum number of reports to store per observation time per station (default=1)

idtype **C4** 4-character station ID
C8 8-character station ID
IDN station block number

maxobs maximum number of observation periods per station to store online (default=2)

minhrs minimum number of hours between observation blocks (default=1)

nbytes number of bytes necessary to store each line of an observation (default=80)

decnam decoder name for building the initial station ID list (no default)

idfile station ID file to use to build the initial station pointer list (default=**STNDB.CORE**)

maxtxt maximum number of megabytes to store in a text file (default=32)

Keyword

CIR= list of defined circuits in *idfile* to build the initial station pointer list (default=all)

Remarks

BILDTEXT creates a pointer file and text file for observational data used by rapid access routines.

The INIT option is typically run only once per observation type to initialize the file structure. Running INIT deletes the existing versions of *pfile* and *tfile*. This command is run automatically for SAO, RAOB, SYN, and terminal forecasts when the McIDAS-XCD server software package is installed.

To list the valid circuit and decoder names from which to build your ID tables, type:

CIRCUIT LIST

DECINFO LIST

When a station is added or deleted from a pointer file, the change does not take effect until the data monitor is restarted.

Examples

BILDTEXT INIT RAOB.RAP RAOB.RAT 1500 5 IDN 4 3 80 RAOB

This entry builds the pointer file RAOB.RAP which stores five reports for every 3-hourly observation for up to 1500 stations. Four observation periods are stored online for use with rapid access text applications. The raw text is stored in the file RAOB.RAT. The IDs are stored as station block numbers. The ID list built for the RAOB.RAP file is generated from the same ID list used by the RAOB decoder.

BILDTEXT ADD UES SAOMETAR.RAP

This entry adds the station UES to the pointer file SAOMETAR.RAP.

BILDTEXT INIT TERMFCST.RAP TERMFCST.RAT 2500 4 C4 6 1 80 TERMFCST X 8

This entry builds the pointer file TERMFCST.RAP which stores up to four reports per observation time and keeps up to six observation times available. The TERMFCST decoder builds the station list; the maximum size of the text file generated is eight megabytes. The raw text is stored in the file TERMFCST.RAT. The IDs are stored as character IDs.

CHKERR

Lists the output from an error file.

Format

CHKERR *file day time [keyword]*

Parameters

file file name (no default)

day Julian day, YYDDD (no default)

time time, HH (no default)

Keyword

NUM= number of lines to output (default=20)

Remarks

CHKERR lists the errors generated by a data monitor. User-written data monitors must call the subroutine ERMESS to write a file readable by CHKERR.

You can use CHKERR to isolate system problems such as periodic aborts caused by corrupt pointer files.

When you install the McIDAS-XCD server software, error messaging is not active for decoders. To activate error messaging, edit the .CFG file appropriate for the decoder. SSEC recommends keeping the error messaging inactive unless there is a problem.

Examples

CHKERR DMSFC.ERR

This entry lists the last 20 lines written to the file DMSFC.ERR.

CHKERR DMSFC.ERR 93025 NUM=30

This entry lists the 30 lines preceding day 93025 in the file DMSFC.ERR.

CIRCUIT

Data circuit utility.

Formats

CIRCUIT ADD *circuit* [*keywords*] "*description*
CIRCUIT DEL *circuit*
CIRCUIT EDIT *circuit* [*keywords*] "*description*
CIRCUIT LIST *circuit*
CIRCUIT SET *circuit* *action*

Parameters

ADD adds a circuit to the configuration file

DEL deletes a circuit from the configuration file

EDIT edits an existing circuit in the configuration file

LIST lists the specified circuit configuration (default=lists all circuits)

SET sets circuit processing to active or inactive

circuit circuit name; four characters maximum (no default)

action **ACTIVE** activates a circuit

INACTIVE deactivates a circuit

"*description* 80-character circuit description

Keywords

CONFIG= circuit configuration file name

INGESTOR= name of the ingestor to use; for example, **INGETEXT** or **INGEBIN**

SPOOL= spool file name; used for the **INGEBIN** ingestors

Remarks

CIRCUIT is an operational utility that adds, deletes, edits, activates and deactivates circuits. All other configuration information about the circuit is entered in the circuit's configuration file using a text editor.

If you change any parameters in the configuration file, you must inactivate the circuit for associated ingestors, wait for the circuit to stop, and then activate the circuit for the associated ingestors.

Examples

CIRCUIT LIST

This entry lists the circuit configurations for all circuits.

CIRCUIT ADD DDS INGESTOR=INGETEXT CONFIG=DDS.CFG "Domestic Data Service

This entry adds DDS to the list of circuits. Data from the circuit is processed when the circuit is activated. The configuration file name for this circuit is DDS.CFG.

CIRCUIT SET DDS ACTIVE

This entry activates the DDS circuit. The next time the STARTXCD program checks the circuit list, the DDS ingestion is started.

CIRCUIT EDIT DDS CONFIG=DDS01.CFG

This entry changes the name of the DDS circuit configuration file to DDS01.CFG.

DATAcq

Plots data availability from MD files.

Format

DATAcq *map mdf time [keywords]*

Parameters

map map for the data plot; use any of the predefined maps used by the MAP command (default=WORLD)

mdf MD file number (no default)

time time of the data, HH (default=0)

Keywords

COL= *rep miss* reporting and missing data color levels (default=7 5)

ELE= *min max* TV element range for the data plot

GRA= graphics frame number (default=current)

LAT= *min max* latitude range to define the map

LIN= *min max* TV line range for the data plot

LON= *min max* longitude range to define the map

MDC= *min max* column range from the MD file to plot (default=all)

MDR= row from the MD file to plot (default=row containing the time determined by the time parameter)

SIZE= size of the plot points, in pixels (default=2)

Remarks

To plot the MD file data availability, the TIME, MOD, LAT and LON keys must be in the following locations in the MD file.

Key	Location
TIME	row header
MOD	data section
LAT	column header or data section
LON	column header or data section

To draw multiple plots in the same frame, use the McIDAS command PANEL to set up frame panels. Use the global keyword PAN to specify the frame panel where the plot is to be drawn.

Examples

DATAcq USA 4 12

This entry plots the data availability over the United States for 12 UTC from MD file 4.

DATAcq SAT 13 12

This entry plots the data availability for 12 UTC from MD file 13 over the currently displayed satellite image.

DATAcq X 33 X PAN=2

This entry plots the data availability over a world map for 0 UTC from MD file 33. The plot is drawn in panel two of a multipanel frame.

DATAECV

Plots acquired MD file data on a multiple-panel display.

Format

DATAECV *time* [*keywords*]

Parameter

time valid time (default=current hour)

Keywords

DAY= Julian day, YYDDD (default=current)

DEC= source decoder for the data: **ISFC**, **IRAB**, **ISHP**, **SYN**, **PIRP** (no default)

GRA= graphics frame number for the plot (default=current)

MAP= map for the data plot (default=world)

SIZE= height of the plotted characters, in pixels (default=2)

TIME= time for the plot (default=current)

Remarks

DATAECV is a macro that repeatedly calls the command DATAQC to plot acquired MD file data in a multiple-panel display.

The table below lists the default setting for each decoder.

Decoder	Default plotting time	MD files	Map
ISFC	Nearest hour observation	1-10	World
IRAB	Nearest 12-hour observation	11-20	World
ISHP	Nearest hour observation	31-40	World
SYN	Nearest 6-hour observation	51-60	World
PIRP	Nearest hour observation	61-70	World

Examples

DATAECV DEC=ISFC SYN

This entry creates a two-panel global plot of surface hourly and synoptic data received for the current hour and synoptic time.

DATAECV DEC=ISFC IRAB ISHP SYN PIRP MAP=CA X X USA

This entry creates a five-panel display and plots the current ISFC data over California, and the SYN data over the United States. It uses the default maps to plot the current data for the IRAB, ISHP and PIRP decoders.

DECINFO

Decoder utility.

Formats

DECINFO ADD *monitor* [*keywords*]
DECINFO DEL *type process* [*keywords*]
DECINFO EDIT *monitor decoder action* [*keywords*] "*description*"
DECINFO LIST *monitor decoder*
DECINFO SET *monitor action*

Parameters

ADD adds data monitors and decoders

DEL deletes data monitors and decoders

EDIT edits data monitors and decoders

LIST lists the current data monitor/decoder configurations

SET activates or deactivates data monitors

monitor data monitor name

decoder decoder name

type **DM** deletes a data monitor
DEC deletes a decoder

process data monitor or decoder to delete

action **ACTIVE** activates data monitors and decoders
INACTIVE deactivates data monitors and decoders

"description" 32-character description of the decoder

Keywords

CONFIG= configuration file name for the decoder

DEC= decoders to add with the ADD option

DM= data monitor from which the decoder is deleted; use with the DEL option

FORM= ALL lists decoder configuration information

Remarks

DECINFO is an operational utility that adds, deletes, edits, lists, activates and deactivates data monitors and decoders.

If you add, delete, activate or deactivate a decoder, you must restart the decoder's data monitors for the action to take effect.

Examples

DECINFO LIST

This entry lists all the current data monitor/decoder configurations. If no configuration file exists, one is initialized with the settings in the table in Chapter 2's [Configuring the Data Monitors](#).

DECINFO ADD LOCAL DEC=FOUS67 TORNADO

This entry adds the decoders FOUS67 and TORNADO to the data monitor LOCAL.

DECINFO EDIT LOCAL TORNADO ACTIVE CONFIG=TORN.CFG "Tornado Warning Decoder

This entry activates the TORNADO decoder running under the data monitor LOCAL and attaches the label Tornado Warning Decoder. The configuration information for this decoder is in the TORN.CFG file.

DECINFO EDIT LOCAL FOUS67 ACTIVE "FOUS67 Decoder

This entry activates the FOUS67 decoder running under the data monitor LOCAL and attaches the label FOUS67 Decoder.

DECINFO EDIT LOCAL FOUS67 CONFIG=FOUS67.CFG

This entry changes the name of the configuration file for the FOUS67 decoder to FOUS67.CFG.

DECINFO SET LOCAL ACTIVE

This entry activates the data monitor LOCAL. The next time the STARTXCD program checks the data monitor, LOCAL is started.

DECINFO DEL DEC NGMMOS DM=DMMISC

This entry deletes the decoder NGMMOS from the data monitor DMMISC. To delete the decoder and also remove it from the Decoder section of the [Status Display Window](#), complete the instructions below.

1. Logon to the workstation as user **oper**.
 2. Run command **xcdadmin stop** from the **oper** xterm.
 3. Run command **ldmadmin stop** from the **oper** xterm.
 4. Exit any preexisting Status Display Window by closing the window.
 5. Start a McIDAS-X session as user **oper**.
 6. Run command **LWU DEL DECOSTAT.DAT** from the McIDAS-X session to delete the preexisting file that has the information for the statdisp program.
 7. Run command **DECINFO DEL DEC NGMMOS DM=DMMISC** from the McIDAS-X session to delete the decoder NGMMOS from the data monitor DMMISC. (Note: If you want to make the decoder inactive [rather than delete it entirely] and want it removed from the Status Display Window, run this command instead: **DECINFO EDIT DMMISC NGMMOS INACTIVE**.)
 8. Run command **EXIT** from the McIDAS-X session to end it.
 9. Run command **ldmadmin start** from the **oper** xterm.
 10. Run command **xcdadmin start** from the **oper** xterm.
 11. Run command **xcdadmin statdisp** from the **oper** xterm to open a Status Display Window. Note that NGMMOS is no longer listed in the Decoder section.
-

DELWXT

Deletes weather text and index files.

Format

DELWXT *days* [*keyword*] "*path*"

Parameters

days number of days before today to save text data, maximum of 10 (default=1)

"*path*" path name to search for data or index files to delete

Keyword

DAY= deletes the specified day's files, YYDDD (no default)

Remarks

DELWXT deletes weather text and index files for a specified number of days. It should run from the system time scheduler once per day. Scheduling DELWXT to run daily frees up a considerable amount of file space by deleting old weather text and index files. At SSEC, DELWXT runs at 00:05 UTC and deletes files older than three days.

To delete a specific day's data, use the keyword DAY.

Examples

SKE 93003 00:05 999999 24 "DELWXT 3

This entry schedules DELWXT to run every 24 hours at 00:05 UTC from the system time scheduler. DELWXT saves weather text and index files containing data for the current day plus the three previous days. For more information about command SKE, see the [McIDAS User's Guide](#).

DELWXT DAY=95017

This entry deletes the text and index files for 17 January 1995.

GRIB2NAV

Lists the navigation parameters necessary for entering new GRIB2 datasets.

Formats

GRIB2NAV *GRIB2Filename*

Parameters

GRIB2Filename fully qualified path of a valid GRIB2 file on your system

Remarks

When adding a new GRIB2 dataset that accesses data in the MySQL/MariaDB database, the dataset's navigation parameters must be added into the `~mcidas/data/grib2NavLookup.txt` file. These navigation parameters can be listed by the **GRIB2NAV** command.

Example output is:

```
Nav parameters for message 66, field 0 are NAM, 6, -4564482, 2089981, 20318, 950000, 250000, 250000
```

The fields are defined as follows:

- Field 1 (**NAM** in this example) is the grid source name (4 character maximum)
 - Field 2 (**6** in this example) is the projection type (1 and 4 are Mercator, 2 is Polar Stereographic, 6 is Lambert Conformal)
 - Fields 3-8 (**-4564482, 2089981, 20318, 950000, 250000, 250000** in this example) are the calculated navigation parameters for the grib2 file
-

Example

GRIB2NAV *sampleGFS.grb2*

This command lists the navigation parameters of the **sampleGFS.grb2** GRIB2 file, which can then be entered in the `grib2NavLookup.txt` file. See [Configuring Additional GRIB2 ADDE Datasets](#) for more information.

IDGROUP

ID group utility.

Formats

IDGROUP ADD *name* [*keywords*]
IDGROUP COMP *name* [*keywords*]
IDGROUP DEL *name* [*keywords*]
IDGROUP LIST *name* [*keywords*]
IDGROUP SAVE *name* [*keywords*]

Parameters

ADD adds stations to an existing group or creates a new group

COMP compresses the file after many groups are altered; deletes groups not marked as permanent

DEL deletes stations from an existing group or deletes an entire group

LIST lists stations in a group or all defined groups

SAVE sets the save flag for a group or country

name name of the group to update; 12 characters maximum (no default)

Keywords

DEC= decoder types; use with the ADD option

GROUP= group containing stations to add to the group specified in the *name* parameter; use with the ADD option

ID= *id1* . . . *idN* stations to add to or delete from a group

LAT= *min max* latitude boundaries of a group

LON= *min max* longitude boundaries of a group

SAVE= **P** creates a permanent group
T creates a temporary group

TYPE= **COUNTRY** performs an operation on a country
GROUP performs an operation on a group (default)

Remarks

IDGROUP is a utility for creating and editing groups of stations used with rapid text accessing applications.

You can only delete stations from a group you created.

The LAT and LON keywords are only valid when creating ID groups for the first time.

If the ID or GROUP keyword is not specified with the ADD option, IDGROUP scans the entire station database (STNDB.* files) to find stations matching either the state or country header. Valid decoder names for the DEC keyword are SAOMETAR, RAOB, FOUS14, SYNOPTIC and TERMFCST.

You can add up to 20 stations at a time using the ADD option with the ID keyword.

IDGROUP appends a zero to all WMO station block numbers with five digits.

Examples

IDGROUP ADD NH

This entry creates the group NH which contains all the stations in New Hampshire.

IDGROUP ADD GM TYPE=COUNTRY

This entry creates the country GM which contains all stations in Gambia.

IDGROUP ADD FRED ID=MSN MEM MSP 72645 72532 SEA SAC SFO

This entry creates a group named FRED containing eight stations.

IDGROUP ADD FRED ID=EGLL UUEE 26216 GROUP=NH

This entry adds stations EGLL, UUEE, 26216 and all the stations in group NH to the group FRED.

IDGROUP LIST

This entry lists all defined groups.

IDGROUP LIST FRED

This entry lists all the stations in the group FRED.

IDGROUP DEL FRED ID=UUEE 72645

This entry deletes Moscow and Green Bay from the group FRED.

IDGROUP ADD MIDWESTSFC LAT=35 50 LON=85 100 DEC=SAOMETAR SYNOPTIC

This entry creates the group named MIDWESTSFC which contains all stations between 35° and 50° N and 85° and 100° W reporting either surface hourly data or synoptic reports.

IDMON

Station ID monitoring utility.

Formats

IDMON COMP *file idfile type*

IDMON HIST *file days station*

IDMON LIST *file*

Parameters

COMP compares the active reporting stations with the stations being decoded

HIST lists the history of a station

LIST lists the station data from the old or new station file

file file name containing the list of old or new stations

idfile file containing the master table of the station in the MD file

type data type: **ISFC**, **IRAB** or **SYN**

days number of days before today to summarize the station reporting status (default=file creation date)

station station ID or WMO header

Remarks

IDMON monitors the status of stations and generates lists to inform operations of new stations and stations that stopped reporting. With this information, operations can remove or add data to the station database (STNDB.CORE).

To activate or deactivate station ID monitoring, edit the appropriate decoder configuration file. Decoders developed at SSEC create two station files: *OLDtype.IDM* and *NEWtype.IDM* where *type* is one of the following data types.

Data type	Decoder
ISFC	SAO/METAR
IRAB	upper air
SYN	synoptic
PIRP	pilot report
TERM	terminal forecast
GFS	GFSMOS
NAM	NAMMOS

File *OLDtype.IDM* contains a record of all stations that are reporting data and are included in the master table of stations in the decoder's MD file.

File *NEWtype.IDM* contains all stations reporting data, but are not included in the master table of stations in the decoder's MD file.

To add new stations reporting data or delete stations no longer reporting, use command UPDIDS. SSEC recommends updating station files monthly for locally developed decoders. If you update your own master ID table, please document the changes and notify SSEC.

SSEC sends updated ID files with each McIDAS upgrade.

Examples

IDMON HIST OLDISFC.IDM 50

This entry lists the stations filed in OLDISFC.IDM that have reported since station monitoring was activated, but have not reported in 50 days or more.

IDMON HIST OLDISFC.IDM X MSN

This entry lists the last date and time data was reported from Madison, Wisconsin, and filed in OLDISFC.IDM.

IDMON HIST NEWIRAB.IDM

This entry lists the new RAOB stations that are reporting data but are not currently being filed in the MD file.

IDMON COMP OLDSYN.IDM SYNDEC.IDT

This entry lists the old synoptic stations that exist in the column headers of the MD file but have never reported.

IDMON LIST NEWGFS.IDM

This entry lists the station location of new stations that did not previously report for the GFSMOS decoder.

NMCAMT

Lists the number of real-time grids received for a specified day.

Format

NMCAMT [*keywords*]

Keywords

DAY= Julian day, YYDDD (default=current)

FORM= STD lists a brief description of the grids; see the Remarks (default)

ALL lists detailed information about the grids; see the Remarks

GRIDF= grid file numbers to check (default=lists all real-time grid files for the specified day)

MOD= ALL searches all model types (default)

m1 . . mn searches for one or more specific models, valid options are:

GFS Global Forecast System model grids

NAM North American Mesoscale model grids

NGM Nested Grid Model grids

TIME= ALL searches all run times (default)

t1 . . tn searches the specified range of run times, H, HH:MM or HH:MM:SS

Remarks

NMCAMT lists both complete and partial real-time grids. A partial grid contains incomplete grid sections. Currently, the GFS model is the only model transmitted in pieces; thus, it is the only model containing partial grids.

Note: The NGM options are still valid in this command even though the NWS terminated the NGM model and its associated products beginning 03 March 2009 (NWS TIN08-90).

The FORM=STD option provides the following information:

- grid file number
- grid number
- parameter
- level
- run time
- forecast time
- model name of partial grids

The screen below shows an example of the FORM=STD output.

```
McDAS0: rch01\outfile.d [T=1]
Grid# Grid Data Level M1 V1 S04
5109 207 5 1000 0 0 STD
5109 207 1 150 0 0 STD
5119 07 02 000 0 00 STD
5120 1 1 1000 0 00 STD
5120 11 1 0000 0 00 STD
Total number of grids: 5
Total number of partial grids: 0
JULCALF: done
```

The projection of an GFS grid determines how it is sent. Grids that are 145 rows by 289 columns are sent in the following eight sectors:

Sector	Latitude range	Longitude range
1	0°N-90°N	60°E-30W
2	0°N-90°N	150°E-60°E

QRTMDG

Deletes real-time grid or MD files.

Formats

QRTMDG GRID *bfile efile numdays*

QRTMDG MD *bfile efile numdays*

Parameters

GRID deletes grid files

MD deletes MD files

bfile beginning file in the range to delete; must end with the number 1

efile ending file in the range to delete; must end with a zero (default=*bfile* + 9)

numdays number of days of real-time data to store online; the range is 2 to 9 (default=4)

Remarks

The QRTMDG command is used to maintain the "cylinders" of real-time MD and grid files. The cylinders are groups of ten files whose file number is based on the last digit of the Julian date. For example, data from day 135 are stored in files 5, 15, 25, etc., and data from day 310 are stored in files 10, 20, 30, etc. Since each cylinder consists of ten files you can keep up to nine days of data online. QRTMDG is used to delete all but the *numdays* most recent files in each cylinder. For example, if you run QRTMDG with a value of 4 in *numdays* on day 135, the real-time files for days 135, 134, 133 and 132 are retained.

SSEC recommends scheduling QRTMDG to run at 00:01 UTC daily to delete the old MD and grid files from your real-time cylinders.

QRTMDG works during the transition to a new year by determining the Julian date of the next day and deleting the files for that day. For example, running QRTMDG on day 002 will delete files 3, 13, 23,... (which contained day 363 data) so they can be written when data for day 003 arrives.

Due to the cylinder structure and day numbering during the transition to a new year (from days ...,364, 365 to 001, 002,...) there are limits to the number of days of data you can keep online between days 365 and 008 because, for example, MD file 4 can store data for day 364 or 004, but not both. During a non-leap year transition (e.g., 2005-2006), the constraints limit you to a maximum of four files on day 004; five files on days 003 and 005; six files on days 002 and 006; seven files on days 001 and 007; and eight files on days 365 and 008. During a leap year transition (e.g., 2008-2009), the constraints limit you to five files on days 003, 004 and 005; six files on days 002 and 006; seven files on days 001 and 007; and eight files on days 366 and 008.

If you store four or less days of data (five or less days during leap year transitions), QRTMDG works as expected because there are no day numbering conflicts. For example, running QRTMDG each day with a *numdays* value of 4 retains surface hourly MD files 5, 4, 3, 2 on day 365; files 1, 5, 4, 3 on day 001; files 2, 1, 5, 4 on day 003, etc.

Example

SKE 2006129 00:01:00 999999 24 "QRTMDG MD 1 40 4

This example stores the current day's MD data plus the previous three days for MD files in the range 1 to 40. When this command runs on day 2006307, all MD files in the range 1 to 40 are deleted except: 4, 5, 6, 7, 14, 15, 16, 17, 24, 25, 26, 27, 34, 35, 36 and 37.

SIGCO

Significant level upper air storage utility.

Formats

SIGCO ADD *country*

SIGCO DEL *country*

SIGCO LIST *country*

Parameters

ADD adds a country to the list

DEL deletes a country from the list

LIST lists countries for which significant level upper air data is saved (default)

country 2-character country code

Remarks

SIGCO specifies the countries for which significant level upper air data is decoded and filed. The list of countries is stored in the file SIGCO.DAT. Changes to the list are implemented when the upper air decoder is restarted. Use the McIDAS-X command CCODE to obtain a list of valid two-letter country codes. See the [McIDAS User's Guide](#) for more information.

Examples

SIGCO ADD VE

This entry adds Venezuela to the list of countries for which significant level data is saved.

SIGCO DEL MX

This entry deletes Mexico from the list of countries for which significant level data is saved.

STARTXCD

Starts the ingestor and decoder programs.

Format

STARTXCD *dtime*

Parameter

dtime number of seconds to pause between programs
(default=30)

Remarks

STARTXCD is the parent program that automatically starts and stops the McIDAS-XCD ingestors and decoders.

Never have more than one STARTXCD command running at a time.

STAT

Lists McIDAS-XCD decoder and ingestor status.

Format

STAT [*keyword*]

Keyword

TOL= warning tolerance in minutes; if an ingestor or decoder does not process data within the tolerance, an asterisk (*) appears next to the time stamp

Remark

STAT provides a snapshot of the bulletin board status display.

SUBGRD

Creates geographic subsectors of Mercator grids.

Format

SUBGRD *sgridf* *bgrid* *egrid* *dgridf* [*keywords*]

Parameters

sgridf source grid file

bgrid beginning grid number to subsect (default=1)

egrid ending grid number to subsect (default=all)

dgridf destination grid file number

Keywords

LAT= *slat* *nlat* destination latitude extents (no default)

slat southern latitude extent

nlat northern latitude extent

LON= *elon* *wlon* destination longitude extents (no default)

elon eastern longitude extent

wlon western longitude extent

MAX= maximum number of grids in destination grid file (default=*egrid-bgrid* +1)

Remarks

The command SUBGRD only creates geographic subsectors of Mercator projection grids.

Example

SUBGRD 1000 1 10 1200 LAT=20 60 LON=40 150

This entry creates grid subsectors with the geographic domain 20° to 60° North and 40° to 150° West from grids 1 through 10 in grid file 1000. The grid subsectors are stored in grid file 1200, which stores 10 grids maximum.

UPDIDS

Updates the station reporting list for decoders.

Formats

UPDIDS ACT *decoder source minnum type [keywords]*

UPDIDS INACT *decoder source cutday type idtable [keywords]*

Parameters

ACT activates decoding for a station list

INACT deactivates decoding for a station list

decoder decoder name

source source ID file written by IDNEW

minnum minimum number of station references needed to activate the decoder

cutday inactivates stations that have not reported in the past number of *cutdays* (default=100 days)

type **CID** character ID (default)

IDN station block number

idtable current ID table used for comparison

Keywords

CIR= activates stations for specified circuits (default=**ALL**)

FILE= file name to update (default=**STNDB.CORE**)

Remarks

UPDIDS activates or deactivates decoding for specified stations. Stations must exist in the station dictionary to successfully activate a decoder for a station. The *source* station list used with UPDIDS is generated in the decoders. These file names are stored in the .CFG file associated with each decoder.

Examples

UPDIDS ACT SAOMETAR NEWISFC.IDM 10 CID

This entry activates stations that have reported 10 or more times in the file NEWISFC.IDM for the decoder SAOMETAR.

UPDIDS INACT RAOB OLDIRAB.IDM 50 IDN IRABDEC.IDT

This entry deactivates stations for the RAOB decoder that have not reported in the last 50 days.

WMORTE

Maintains a data routing table of WMO headers.

Formats

```
WMORTE ADD CIR circuit
WMORTE ADD INDEX index ALIAS=
WMORTE ADD WMO [keywords]
WMORTE DEL CIR circuit
WMORTE DEL INDEX index ALIAS=
WMORTE DEL WMO [keywords]
WMORTE EDIT INDEX index ALIAS=
WMORTE EDIT WMO [keywords]
WMORTE LIST circuit
```

Parameters

ADD adds a circuit, index, or WMO header

DEL deletes a circuit, index, or WMO header

EDIT edits an index or WMO header

LIST lists the WMO headers, indices and aliases for a circuit

CIR circuit

INDEX index

WMO WMO header specified with keyword **HEADER**

circuit circuit name

index 2-character index name

Keywords

ALIAS= alias file name used with the *index* options; eight characters maximum (default=*index*)

CIRCUIT= circuit name for adding, editing, and deleting the WMO specification

HEADER= WMO headers to add, edit or delete; two characters maximum

INDEX= index for adding or editing WMO headers; two characters maximum; you can specify more than one index when adding multiple WMO headers (default=WMO header specified)

Remarks

WMORTE creates a routing table describing the location of text data. It is created by assigning one or more WMO headers, ingested from each circuit, to an index name. The file that stores the text data is defined by the index name. For example, if the index specified is CS and the date of the data contained in the file is 93002, the file name for the index file is CS93002.IDX.

The WMO headers and index names can only be two characters. If data is ingested with a WMO header that is not on the list of defined headers, the data is filed in the miscellaneous index file ZZ.

Each index file can have an associated alias name. The alias name or index name can then be used by the ADDE weather text server. For example, since data ingested with the WMO header CS is climatological information, you could assign an alias name CLIMATE to the index CS. A user could then specify either CS or CLIMATE when using the WXTLIST command.

You can define a maximum of 32 circuits, 512 indices and 1024 WMO headers using this routing system.

If you alter the contents of a circuit's routing table, other than the alias name, you must restart the circuit to activate the new or updated routing table.

The first time WMORTE is run, it initializes the file IDXALIAS.DAT to a predefined routing table for the DDS, IDS, PPS and Carswell circuits. This must be done before starting the circuits with STARTXCD.

Each WMO header is stored as either a primary or secondary index. Use the command **WMORTE LIST *circuit*** (where *circuit* is a valid circuit name, for example, DDS) to list your system's indexing. Primary indices are listed under the INDEX column; secondary indices are listed under the WMO HEADERS column.

Examples

WMORTE LIST DDS

This entry lists the aliases, indices, and WMO headers for the DDS circuit.

WMORTE ADD CIR CDS

This entry adds the circuit CDS to the defined circuit names.

WMORTE ADD INDEX RW ALIAS=RIVER

This entry adds the index name RW to the list of defined indices and gives it the alias name RIVER.

WMORTE ADD WMO HEADER=RR RW CIRCUIT=CDS INDEX=RW RW

This entry adds the WMO headers RR and RW, which are filed in the index RW to the CDS circuit.

WMORTE EDIT INDEX SA ALIAS=SURFACE

This entry changes the alias name of the index SA to SURFACE.

WMORTE DEL CIR CARS

This entry deletes the circuit CARS from the list of defined circuit names.

WMORTE DEL WMO HEADER=WF WU CIRCUIT=IDS DDS

This entry deletes the WMO headers WF and WU from the IDS and DDS circuits.

WMORTE DEL INDEX SM

This entry deletes the index SM from the routing table.

gribadmin

Unix command for managing the MySQL/MariaDB database used by the GRIB server.

Formats

```
gribadmin makedb
gribadmin removedb
gribadmin disablelog
gribadmin fields
gribadmin num model
gribadmin latest model
```

Parameters

makedb makes the database from scratch with the following attributes:

- the database is named `mcrtgrib`
- the read-only user is `gribread` and has no password
- the read-write user is `gribwrite` with password `gribwrite`

removedb removes the database completely from MySQL/MariaDB

disablelog disables the writing of binary log files associated with the database; this overrides the default in MySQL/MariaDB 8.0 and later, which enable binary logging by default; we recommend disabling the logging because it is not useful and it fills large amounts of disk (possibly an entire file system) due to it logging millions of *INSERT into file_list* entries per day; this option must be run as **root**, see the Remarks for instructions

fields prints the fields in the database and their data types

num displays the number of records in the database; if *model* is not specified then the number of records for each model will be displayed; if **total** is specified then the total number of records in the database will be displayed

latest displays the latest record in the database; if *model* is not specified then the latest record in the database will be displayed regardless of the model

model model/source name appearing in the GRIB files

Remarks

The *Configuration Sections* section of the `gribadmin` script appears in the box below. It contains variables that change its behavior. These variables must be reviewed before using the script. In particular, the `mysql_path` and `gcc_lib_path` variables may need to be changed.

```
##### # Configuration Section: # # The
variables in this section may need to be modified # to conform to your system. # Database
and user names for MySQL calls database_name="mcrtgrib" database_user="gribread" # Path to
the MySQL executable mysql_path=/usr/local/mysql/bin # Path to gcc shared libraries
gcc_lib_path=/usr/local/lib #####
```

The `makedb` and `removedb` options will prompt for a password. Enter the password for the MySQL/MariaDB root user, which was set by the person that installed MySQL/MariaDB.

The `disablelog` option disables the writing of binary log files associated with the database. This is needed because MySQL/MariaDB 8.0 and later enable binary logging by default, which fills large amounts of disk (possibly an entire file system) due to it logging millions of *INSERT into file_list* entries per day. To run the `disablelog` option, run the command below as **root**:

Type: `/home/oper/mcidas/bin/gribadmin disablelog`

When the command is finished, you should see a message like that below.

```
/etc/my.cnf.d/disable-log-bin.cnf successfully written!
```

Once its successful completion is verified, exit from **root**.

Examples

`gribadmin fields`

This entry prints the fields in the database and their data types.

`gribadmin latest GFS`

This entry displays the latest record among GFS GRIB files in the database.

statdisp

Unix command that starts the McIDAS-XCD status display.

Format

statdisp [*flags*]

Flags

-bg <i>color</i>	background color (default=black)
-display <i>display</i>	workstation name and window manager to use for display
-fg <i>color</i>	foreground color (default=white)
-font <i>font</i>	font size to use (default=6 x 12)
-geometry+ <i>xoffset</i> + <i>yoffset</i>	position of status window
-resize	stops automatic window resizing
-sample <i>seconds</i>	screen refresh sampling time (default=5 seconds)
-threshold <i>minutes</i>	warning threshold time (default=5 minutes)
-warn <i>color</i>	warning color (default=red)

Remarks

This command starts an X window to display the status of McIDAS-XCD decoders and ingestors. Active decoders and ingestors are displayed in the foreground color. Decoders and ingestors that are inactive longer than the warning threshold time are displayed in the warning color. See the [McIDAS-XCD Status Display Window](#) section for more information on the output from statdisp.

To cancel the status display, click on Quit the Window in the Title bar. To display a window in a smaller size than is necessary to view the entire bulletin board, use the -resize option.

To force statdisp to get data from a file other than `~/oper/mcidas/data/DECOSTAT.DAT`, set the environmental variable `XCD_disp_file` to the fully expanded file name.

To start statdisp from the McIDAS-X command window, precede it with **OS "**. Run it in the background using the **&** (ampersand) shell option.

Use the -geometry option to specify the screen position of the status window. Specify the horizontal and vertical offsets in pixels. Offsets must be preceded by plus signs (+).

Example

```
statdisp -bg white -fg black -warn magenta &
```

This entry starts the status display with a white background, black foreground, and magenta warning messages from the Unix command window. The **&** (ampersand) shell option runs the command in the background.

xcdadmin

Unix command for managing the XCD status and the log file.

Formats

```
xcdadmin start
xcdadmin stop
xcdadmin status
xcdadmin statdisp
xcdadmin log flag
```

Parameters

```
start    starts XCD in a background mcenv session

stop     kills any background XCD sessions

status   prints XCD status (on or off) as well as useful statistics including XCD version, log file location, data directory, disk space usage, and processor load averages

statdisp runs the statdisp program

log      accesses the log file, performing the action specified in flag

flag   -f   follows the log file (using tail -f)

        -l   prints the contents of the XCD log file (default)

        -r   rotates the log file, creating a new xcd.log file named xcd.log.mondd_hhmm_GMT

        -t   prints the current time, disk usage, and load averages to the log file
```

Remarks

The *Configurable Options* section of the xcdadmin script appears in the box below. It contains variables that change its behavior. These variables must be reviewed before using the script. In particular, the xcd_data_path variable must be set.

```
##### # Configurable Options: # # The
variables in this section hold information that is # subject to varying McIDAS-XCD
configurations. Please # check the values of each of these variables and make # any
necessary changes to accommodate your system's # configuration. Note: The value of
'xcd_data_path' # must be modified in order for the xcdadmin script to # work.
xcd_admin="oper" xcd_admin_home="/home/oper" xcd_admin_profile="$xcd_admin_home/.profile" #
The 'xcd_data_path' holds the fully qualified path to # the directory in which XCD data will
be written. This # variable must be changed for the 'xcdadmin' script to # work.
xcd_data_path="Insert path to XCD data directory here" # The 'xcd_log_file' variable holds
the fully qualified path # to the file in which output from 'startxcd' is written. # This
file must be writable by the McIDAS-XCD admin # account. It is advisable to set up a daily
cron job which # calls 'xcdadmin log -r' to rotate the log file. Otherwise # it is possible
for the log file to grow until it fills the # disk. xcd_log_file=$xcd_admin_home/xcd.log #
The 'seconds_to_wait' variable holds the number of # seconds to wait for data monitors to
exit gracefully # before a 'kill' command is used. It is beneficial to # wait for data
monitors to exit on their own accord # because data can be lost if/when 'kill' is used.
seconds_to_wait=30 # The 'statdisp_refresh' variable holds the refresh rate # in seconds for
the statdisp window. statdisp_refresh=5 # Number of old log files to keep on disk. When #
'xcdadmin log -r' is called the name of the current log # file is appended with the current
date and time, e.g. # xcd.log.Apr12_1200_GMT and a new log file 'xcd.log' is # created. Then
the number of old log files is checked # and if it exceeds 'num_logs_to_save' then the older
# log files will be deleted. For example; If you rotate # the logs daily then setting
'num_logs_to_save' to 7 # results in saving one week's worth of log files in # addition to
the current log file on disk. num_logs_to_save=7
#####
```

Examples

```
xcdadmin status
```

This entry prints XCD status (**on** or **off**) as well as useful statistics including XCD version, log file location, data directory, disk space usage, and processor load averages.

```
xcdadmin log -t
```

This entry prints the current time, disk usage, and load averages to the log file.

xcdscour

Unix command for deleting real-time GRIB1, GRIB2, or BUFR files, and their associated database entries.

Formats

xcdscour GRIB *numdays*
xcdscour BUFR *numdays*

Parameters

GRIB deletes GRIB1 and GRIB2 files

BUFR deletes BUFR files

numdays number of days of real-time data to store online; the range is 1 to 9 (no default)

Remarks

The xcdscour command is used to maintain the real-time GRIB1, GRIB2, and BUFR files by deleting all but the *numdays* most recent files. If xcdscour is not run, GRIB1, GRIB2, and BUFR data files will not be removed and will continue to accumulate and use disk space.

If the GRIB decoder is configured to use a MySQL/MariaDB database the corresponding entries will be removed and the database will be optimized before the data files are removed.

SSEC recommends scheduling xcdscour to run at a time when the load on the -XCD workstation is low because the MySQL/MariaDB database cleaning and optimizing step can cause data access delays while it is running.

Example

```
5 1 * * * /home/oper/mcidas/bin/xcdscour GRIB 3 /data/grib
```

This entry creates a cron job that, at 1:05 a.m. local machine time, removes any GRIB1 and GRIB2 files in the /data/grib directory from more than three days ago. Data from the current day as well as three full previous days is retained.

Chapter 4

Troubleshooting

This chapter lists problems that may occur with McIDAS-XCD. Under each symptom or error message, possible solutions are given. The problems discussed here are listed below.

- [-XCD is not receiving real-time data - users report no real-time data, or the ingestor status display is red](#)
- [-XCD is not creating McIDAS Grid files](#)
- [-XCD is not writing GRIB files](#)
- [-XCD is not writing to the real-time GRIB MySQL/MariaDB database](#)
- [-XCD data is garbled or missing](#)
- [xcdadmin script is not working](#)
- [directory `***NEXRAD_PATH***` exists on the workstation](#)

-XCD is not receiving real-time data - users report no real-time data, or the ingestor status display is red

- Verify with your LDM Administrator that the LDM is ingesting data. Verify that `startxcd.k` is running under the user `oper`. If it is not, see [Starting and Stopping McIDAS-XCD](#) for information on starting -XCD. If you are filing GRIB information into a real-time MySQL/MariaDB database, verify with your MySQL/MariaDB Administrator that `mysql` is running. Contact your System Administrator to add all three of these to your machine's start-up process.
- The file system may be full. You can use the McIDAS-XCD commands [DELWXT](#), [QRTMDG](#), and [xcdscour](#) to delete older text, point, grid, and grib files and to clean the real-time GRIB database. Do not delete any files for the current day. See [Configuring Data Removal](#) for information on configuring cron and the McIDAS-X Scheduler to remove McIDAS-XCD data files.
- If the file system is not full, check the LDM configuration file `ppact.conf` to make sure it is calling `ingebin.k` and `ingetext.k`. If this is OK, check the LDM configuration file `ldmd.conf` to see that it is making the proper requests.
- Check the `/home/oper/xcd.log` file for any useful information that may help to resolve the problem.
- Check for an obstruction in the antenna and verify that all receiving hardware is working properly.
- Contact your source provider to see if they are having a problem with the broadcast.
- If running MySQL/MariaDB 8.0 or later, check whether the writing of binary log files associated with the database is disabled. As noted in the [gribadmin](#) command, we recommend disabling it because it is not useful and it fills large amounts of disk (possibly an entire file system) due to it logging millions of `INSERT into file_list` entries per day.

-XCD is not creating McIDAS Grid files

The DMGRID GRIB decoder can't file grids if it can't find `RTMODELS.CFG`, which contains information about real-time grid file locations.

The file `RTMODELS.CFG` should reside in `~oper/mcidas/data` when McIDAS-XCD is installed correctly. Either the decoder can't reach the file or it is missing. If it's missing, recreate the file or copy a new version of the default file from `~oper/mcidas/xcdversion/data/RTMODELS.CFG` (where `version` is the version number, e.g., 2020.1).

-XCD is not writing GRIB files

From a McIDAS-X session running under the `oper` account, run the `REDIRECT LIST` command to check for any file redirections.

Type: **REDIRECT LIST**

Look for the `*.gr*` and `*.bu*` directory entries and verify that these directories exist on your workstation. If these directories do not exist, create them. Once this is done, -XCD will automatically start writing GRIB1 and GRIB2 files to this directory, which in turn will allow for data to be written to the database. Then users should be able to access the GRIB data.

-XCD is not writing to the real-time GRIB MySQL/MariaDB database

From a Unix shell prompt under the `oper` account, run the `gribadmin` command to check the most recent record filed in the `mcrtrgrib` database.

Type: **gribadmin latest**

- If MySQL/MariaDB is installed properly and the `mcrtrgrib` database was created, the most recent record filed in the database will be listed. The `modtime` is the last line of the record. If the GRIB file information is being filed correctly, this should be today's date and a very recent time.

- If no record is returned, or if the modtime is not recent, this means that the mcrtgrib database exists, but no data is being written to it. Verify with your -X **and** -XCD Administrator that both packages were compiled with the -mysql flag.
- If an error message containing "Can't connect to local MySQL server" is displayed, this means that MySQL/MariaDB is installed on your machine, but that it was not started. Contact your MySQL/MariaDB Administrator to restart MySQL/MariaDB.
- If an error message from gribadmin containing "mysql: not found" is displayed, this means that the gribadmin command can not find the mysql executable. Confirm that mysql is in the PATH of the user oper.
- If an error message from gribadmin containing "Access denied for user '@localhost' to database 'mcrtgrib'" is displayed, this means that MySQL/MariaDB is running, but the mcrtgrib database was never created. See [Configuring the GRIB Decoders](#) for more information on creating the mcrtgrib database.

-XCD data is garbled or missing

When more than one ingestor is trying to read the same circuit, text data may be missing or text output garbled. If you are decoding grids, grids may be missing.

Only one ingetext.k process should be running for each text circuit, and only one ingebin.k process should be running for each binary circuit (e.g., one ingebin.k process running for GRIB data while another is running for NEXRAD data). Check the number of ingetext.k and ingebin.k processes and -XCD data monitors running. At a Unix shell prompt, as user oper,

Type: `ps -ef | grep DM`

Type: `ps -ef | grep inge`

If there are multiple processes running for one of the circuits or one of the data monitors, stop and restart -XCD and the LDM. Some data may be lost while the processes are stopped.

1. Stop -XCD and kill any remaining data monitors (DM*), if necessary. At a Unix shell prompt, as user oper,

Type: `xcdadmin stop` (you will need to wait for the data monitors to cleanly exit)

Type: `ps -ef | grep DM`

Type: `kill -9 process_id` (if necessary)

2. Stop the LDM and kill any remaining processes, if necessary. At a Unix shell prompt, as user oper,

Type: `ldmadmin stop` (stops the INGEBIN and INGETEXT processes)

Type: `ps -ef | grep inge`

Type: `kill -9 process_id` (if necessary)

3. Restart the LDM and -XCD by running the two commands below, in order.

Type: `ldmadmin start`

Type: `xcdadmin start`

If this process doesn't work, check for an obstruction in the receiving antenna.

xcdadmin script is not working

- Verify that you are logged in as user oper.
- Run command which xcdadmin to verify the location of the xcdadmin script.
- Verify that the oper account has a .profile file.
- Open the xcdadmin file and verify that its `xcd_admin_profile` variable is set to the oper account's .profile file.

directory *NEXRAD_PATH*** exists on the workstation**

The ***NEXRAD_PATH*** directory is created if the NEXRAD data monitor is enabled and the `DIRFILE=` value is not set in the file `~oper/mcidas/data/NEXRID.CFG`. See Chapter 2's [Configuring the NEXRAD Decoder](#) section for instructions to set the `DIRFILE=` value and other NEXRAD options.

Chapter 5

Decoding GRIB Messages

The McIDAS-XCD GRIB decoder converts the binary grid data in the NOAAPORT data stream sent by the National Weather Service (NWS) into McIDAS grid files. The McIDAS-XCD GRBFILER decoder puts this same data into GRIB1 and GRIB2 files referenced by a MySQL/MariaDB database and accessed by the GRIB server. This section describes the steps required to ingest and decode the data stream into McIDAS grid files and/or GRIB files.

- [Processing the GRIB message](#)
 - [Converting GRIB codes](#)
 - [Creating McIDAS Grid Files with DMGRID](#)
 - [Creating GRIB1 and GRIB2 Files with DMBIN](#)
-

Processing the GRIB message

When DMGRID data monitor or the DMBIN data monitor successfully reads a complete message, it calls the McIDAS-XCD GRIB decoder. The decoder first decodes the Product Definition Section (PDS) to determine the type of data contained in the message. After processing this section of the message, the decoder has enough information to determine whether to continue processing. The administrators at your site can configure the decoder to process or discard messages based on various criteria.

GRIB1 and GRIB2 messages may be discarded based on several attributes: GRIB1 based on the model (specified as the value stored in byte 6 of the PDS), the model run time, the valid time of the forecast fields, the geographic location the message represents, the level the data represents, and the meteorological parameter; GRIB2 based on the model and domain (specified as the model prefix assigned in file `grib2NavLookup.txt`), the model run time, and the valid time of the forecast fields.

If disk space is a concern, you can use this discard feature to save only the models and data that you typically use: 500 and 1000 mb height and temperature fields, for example. Several models are sent in more than one projection. If you only need one of them, configure the decoder so the other projections are discarded.

The configuration files where this information is stored are `NOGRIB.CFG` (for GRIB1 messages) and `NOGRIB2.CFG` (for GRIB2 messages). These files are read when the data monitor is started. If you change the values in either file, you must restart the data monitor. If the decoder cannot find the files, all messages are decoded. The `NOGRIB.CFG` file has eleven positions separated by the pipe character (`|`). Below is a description of each position in `NOGRIB.CFG`. The format of `NOGRIB2.CFG` is the same but it has only five positions, so is not described below. See the `NOGRIB2.CFG` file for its format and an example entry.

NOGRIB.CFG format

Position# Description

- | 1 | Model number to discard. If this value is -1, the model number is not used as selection criteria. This is the value stored in byte 6 of the PDS. The commonly used values are listed below. The <code>~mcidas/data/gbtbpds001.av1</code> file contains a complete list of the known values. | | | | | | | | | | | | | | | | | | |
|---------------|---|---------------|-------------|----|---------------------------------|----|-----------------------------------|----|--|----|------------------|----|------------------|----|------------------|----|------------------|-----|----------------------------|
| | <table><thead><tr><th>Common values</th><th>Description</th></tr></thead><tbody><tr><td>10</td><td>Global Wind Wave Forecast Model</td></tr><tr><td>80</td><td>GFS model, 62-Wave Spectral Model</td></tr><tr><td>81</td><td>GFS model, Spectral Statistics, Aviation Run</td></tr><tr><td>83</td><td>NAM model, 80 km</td></tr><tr><td>84</td><td>NAM model, 12 km</td></tr><tr><td>86</td><td>MAPS (RUC) model</td></tr><tr><td>89</td><td>NAM model, 48 km</td></tr><tr><td>120</td><td>Ice Concentration Analysis</td></tr></tbody></table> | Common values | Description | 10 | Global Wind Wave Forecast Model | 80 | GFS model, 62-Wave Spectral Model | 81 | GFS model, Spectral Statistics, Aviation Run | 83 | NAM model, 80 km | 84 | NAM model, 12 km | 86 | MAPS (RUC) model | 89 | NAM model, 48 km | 120 | Ice Concentration Analysis |
| Common values | Description | | | | | | | | | | | | | | | | | | |
| 10 | Global Wind Wave Forecast Model | | | | | | | | | | | | | | | | | | |
| 80 | GFS model, 62-Wave Spectral Model | | | | | | | | | | | | | | | | | | |
| 81 | GFS model, Spectral Statistics, Aviation Run | | | | | | | | | | | | | | | | | | |
| 83 | NAM model, 80 km | | | | | | | | | | | | | | | | | | |
| 84 | NAM model, 12 km | | | | | | | | | | | | | | | | | | |
| 86 | MAPS (RUC) model | | | | | | | | | | | | | | | | | | |
| 89 | NAM model, 48 km | | | | | | | | | | | | | | | | | | |
| 120 | Ice Concentration Analysis | | | | | | | | | | | | | | | | | | |
| 2 | Beginning of the model run time range to discard. If this value is -1, the model run time is not used as selection criteria. This value is stored in byte 16 of the PDS. | | | | | | | | | | | | | | | | | | |
| 3 | End of the model run time range to discard. | | | | | | | | | | | | | | | | | | |
| 4 | Beginning of the model valid time range to discard. If this value is -1, the valid time of the model is not used as selection criteria. This value is stored in bytes 19 and 20 of the PDS. | | | | | | | | | | | | | | | | | | |
| 5 | End of the model valid time range to discard. | | | | | | | | | | | | | | | | | | |
| 6 | Beginning of the geographic ID range to discard. If this value is -1, the geographic ID is not used as selection criteria. This value is stored in byte 7 of the PDS. The file <code>~mcidas/data/gbtbpds001.bv1</code> contains a list of the IDs and their corresponding geographic coverage and projections. | | | | | | | | | | | | | | | | | | |

- 7 End of the geographic ID range to discard.
- 8 Beginning of the pressure level range to discard. If this value is -1, the pressure level is not used as selection criteria.
- 9 End of the pressure level range to discard.
- 10 Beginning of the parameter number range to discard. If this value is -1, the parameter number is not used as selection criteria. This value is stored in byte 9 of the PDS. The file ~mcidas/data/gbtbpds001.2v2 contains a complete list of the known values. Below is a table of commonly used values.

Common values	Description
1	Pressure
2	Pressure reduced to MSL
7	Geopotential Height
11	Temperature
33	u-component wind
34	v-component wind
52	Relative Humidity

- 11 End of the parameter number range to discard.

Examples of NOGRIB.CFG entries

The entry below discards all fields of the aviation run (77) from projections 37 through 44. These projections are associated with the high resolution "thinned" grid format.

```
77 | -1 | -1 | -1 | -1 | 37 | 44 | -1 | -1 | -1 | -1 |
```

The entry below discards relative humidity fields (52) from 500 to 700 millibars for the 12 km NAM (84).

```
84 | -1 | -1 | -1 | -1 | -1 | -1 | 500 | 700 | 52 | 52 |
```

The entry below discards any field from a 12 UTC model run with a valid time between 36 and 42 hours, inclusive.

```
-1 | 12 | 12 | 36 | 42 | -1 | -1 | -1 | -1 | -1 | -1 |
```

Recommended NOGRIB.CFG entries

SSEC recommends that the following entries be added to your NOGRIB.CFG file.

The entry below is for ICA grids known by Unidata to be incorrectly formatted.

```
120 | -1 | -1 | -1 | -1 | 218 | 218 | -1 | -1 | -1 | -1 |
```

The entry below is for ICA grids known by Unidata to be incorrectly formatted.

```
120 | -1 | -1 | -1 | -1 | 242 | 242 | -1 | -1 | -1 | -1 |
```

The entry below is for WWF grids with bad geographic data.

```
10 | -1 | -1 | -1 | -1 | 233 | 233 | -1 | -1 | -1 | -1 |
```

Converting GRIB codes

For users to understand GRIB messages, the decoder must change portions of them into meteorological values. For example, a value of 11 in the ninth byte of the PDS is meaningless until it is converted to temperature in degrees Kelvin. Other attributes that must be converted include the geographic location, the forecast time units, and the generating model name and originating location.

GRIB1

Currently, five types of ASCII file lookup tables are included with the McIDAS-XCD GRIB1 decoder for this purpose. These files are found in ~mcidas/data and begin with the characters gbtbpds as shown below.

Attribute	Section/Byte	Files
processing center	PDS/5	gbtbpds001.0v1
parameter/unit	PDS/9	gbtbpds001.2v1, gbtbpds001.2v2, gbtbpds001.2v3, gbtbpds001.2v131
forecast time	PDS/18	gbtbpds001.4v1
model	PDS/6	gbtbpds001.av1, gbtbpds001.a34v1, gbtbpds001.a60v1, gbtbpds001.a74v1, gbtbpds001.a98v1
geographic location	PDS/7	gbtbpds001.bv1, gbtbpds001.b98v1

When the decoder finds a value for one of these attributes, it checks the appropriate lookup table for information about the value. If it cannot find the information, the message is discarded.

GRIB2

Three ASCII file lookup tables are included with the McIDAS-XCD GRIB2 decoder for this purpose. These files are found in ~mcidas/data and begin with the characters grib2 as shown below.

Attribute	File
geographic location	grib2NavLookup.txt
parameter/unit	grib2Parameters.txt
model	grib2NCEPModels.txt

The McIDAS-XCD [GRIB2NAV](#) command lists McIDAS navigation values that can be added to grib2NavLookup.txt. See [Configuring Additional GRIB2 ADDE Datasets](#) for more information.

Creating McIDAS Grid Files with DMGRID

When the unpacking process is complete and the entire GRIB message is successfully decoded, the decoder passes the GRIB structures to DMGRID and/or DMBIN to be reformatted for McIDAS. DMGRID, which converts GRIB1 messages to McIDAS grid files, is described below. DMBIN, which files GRIB1 and GRIB2 messages into GRIB files of the same format, is described in the [next section](#).

Once the message is converted to McIDAS format, DMGRID uses the `mcrtrgrdf` function to determine the grid file for storing the message. The correct grid file is determined by using stored grid header information, consisting of the model, the runtime of the model, and the forecast time, geographic coverage, and the configuration file `RTMODELS.CFG`. If the model information is not explicitly described in `RTMODELS.CFG`, the grid is filed in a scratch grid file. The format of `RTMODELS.CFG` is described below, and is followed by an example.

Once the correct grid file is determined, DMGRID checks if this grid can be filed as is, or if the grid must be pieced together with a previously filed grid. Piecing together is often necessary because most of the gridded fields that cover the globe are sent in four or eight pieces. When a partial grid is received, DMGRID checks if a similar grid has recently been filed. If so, DMGRID pieces the two fields together, refiling the new grid into the same location. If no match is found, the grid is filed as the first grid of this type.

Finally, DMGRID updates the Status Window, telling the administrator that a new grid has been filed and its location. DMGRID then checks the spool file for new data to process.

RTMODELS.CFG format

SCRATCH= a group of 10 grid files for storing grids based on models not specified in `RTMODELS.CFG`; if `SCRATCH=99000`, the range of grid files used is 99001 through 99010 based on the Julian day of the model run time

model= *ftype fgridf runint vtint maxvt* **GEO=** *min max*

information for filing a particular *model*, for example, **NAM** or **GFS**

ftype filing format for this model

0 everything from the model is stored in one grid file per model run time; if this value is used, *vtint* and *maxvt* are not necessary

1 grids are filed based on model run time and valid forecast time

2 all grids from a model run are filed in the same grid file regardless of run time or forecast time; if this value is used, *runint*, *vtint*, and *maxvt* are not necessary

3 same as 1 except no grids are assumed beyond the *maxvt* forecast time

fgridf first grid file in the range to use for this model

runint interval between model run times (hhmmss)

vtint forecast period interval to separate forecast grids

maxvt maximum forecast time, after which all grids are stored in the same grid file

GEO= *min max* range of geographic IDs to store in this grid file range; the file `~mcidas/data/gbtbpd001.bv1` contains a list of the IDs and their corresponding geographic coverage and projections (default=1 255)

ID= generating process ID from `~mcidas/data/gbtbpd001.av1`

RTMODELS.CFG defaults

The `RTMODELS.CFG` file supplied in McIDAS-XCD creates the grid files listed below.

Grid File	Model	Run	Forecast Range	Coverage
50001-50010	AWC-ALL	00Z - 2359Z	00hr	All
51001-51010	AWRW-ALL	00Z,06Z,12Z,18Z	00hr <= Forecast Time <= 120hr	All
52001-52010	ECMF-EQME	00Z,12Z	00hr <= Forecast Time <= 144hr	Equatorial, ECMWF 5-8
52101-52110	ECMF-GLME	00Z,12Z	00hr <= Forecast Time <= 240hr	Global, ECMWF 1-4,9-12
52201-52210	ECMF-NWME	00Z,12Z	00hr	All

53001-53010	NAM-AKPS	0Z	00hr <= Forecast Time <= 60hr	All
53021-53030	NAM-AKPS	12Z	00hr <= Forecast Time <= 60hr	All
53101-53110	NAM-USLC	0Z	00hr <= Forecast Time <= 60hr	All
53121-53130	NAM-USLC	12Z	00hr <= Forecast Time <= 60hr	All
53201-53210	NAM-USLC2	0Z	00hr <= Forecast Time <= 60hr	All
53211-53220	NAM-USLC2	6Z	00hr <= Forecast Time <= 48hr	All
53221-53230	NAM-USLC2	12Z	00hr <= Forecast Time <= 60hr	All
53231-53240	NAM-USLC2	18Z	00hr <= Forecast Time <= 48hr	All
53301-53310	NAM-USLCAWI4	0Z	00hr <= Forecast Time <= 60hr	All
53311-53320	NAM-USLCAWI4	6Z	00hr <= Forecast Time <= 48hr	All
53321-53330	NAM-USLCAWI4	12Z	00hr <= Forecast Time <= 60hr	All
53331-53340	NAM-USLCAWI4	18Z	00hr <= Forecast Time <= 48hr	All
53501-53510	NAM-AKPSAWIP	0Z	00hr <= Forecast Time <= 60hr	All
53521-53530	NAM-AKPSAWIP	12Z	00hr <= Forecast Time <= 60hr	All
53601-53610	NAM-AKPSAWI2	0Z	00hr <= Forecast Time <= 60hr	All
53621-53630	NAM-AKPSAWI2	12Z	00hr <= Forecast Time <= 60hr	All
54001-54010	FFG-ALL	00Z - 2359Z	00hr <= Forecast Time <= 24hr	All
55001-55010	GFS-AKPS	00Z	00hr <= Forecast Time <= 120hr	All
55011-55020	GFS-AKPS	06Z	00hr <= Forecast Time <= 120hr	All
55021-55030	GFS-AKPS	12Z	00hr <= Forecast Time <= 120hr	All
55031-55040	GFS-AKPS	18Z	00hr <= Forecast Time <= 120hr	All
55101-55110	GFS-AKPSLRES	00Z, 18Z	00hr <= Forecast Time <= 240hr	All
55201-55210	GFS-AKPSSSIF	00Z	00hr	All
55301-55310	GFS-GLME	00Z	00hr <= Forecast Time <= 240hr	Global, GEO 37-44
55311-55320	GFS-GLME	06Z	00hr <= Forecast Time <= 240hr	Global, GEO 37-44
55321-55330	GFS-GLME	12Z	00hr <= Forecast Time <= 240hr	Global, GEO 37-44
55331-55340	GFS-GLME	18Z	00hr <= Forecast Time <= 240hr	Global, GEO 37-44
55401-55410	GFS-GLMELRES	00Z	00hr < Forecast Time <= 168hr	All
55601-55610	GFS-NWME62	00Z, 12Z	00hr < Forecast Time <= 84hr	All
55701-55710	GFS-NWME170	00Z	>= 96hr Forecast Time	All
55801-55810	GFS-HIME	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 120hr	All
55901-55910	GFS-HIMELRES	00Z	00hr < Forecast Time <= 240hr	All
56001-56010	GFS-HIMESSIF	0Z	00hr	All
56101-56110	GFS-NHPS	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 120hr	All
56201-56210	GFS-NHPSLRES	00Z	00hr < Forecast Time <= 240hr	All
56301-56310	GFS-NHPSSSIF	00Z	00hr	All
56401-56410	GFS-PRPSLRES	00Z	00hr < Forecast Time <= 240hr	All
56501-56510	GFS-PRPSSSIF	00Z	00hr	All
56601-56610	GFS-USLC	00Z	00hr < Forecast Time <= 240hr	All
56611-56620	GFS-USLC	06Z	00hr < Forecast Time <= 240hr	All
56621-56630	GFS-USLC	12Z	00hr < Forecast Time <= 240hr	All
56631-56640	GFS-USLC	18Z	00hr < Forecast Time <= 240hr	All
56701-56710	GFS-USPS	00Z	00hr < Forecast Time <= 120hr	All
56711-56720	GFS-USPS	06Z	00hr < Forecast Time <= 120hr	All

56721-56730	GFS-USPS	12Z	00hr < Forecast Time <= 120hr	All
56731-56740	GFS-USPS	18Z	00hr < Forecast Time <= 120hr	All
56801-56810	GFS-USPSLRES	00Z	00hr < Forecast Time <= 240hr	All
56901-56910	GFS-USPSSSIF	00Z	00hr	All
57001-57010	GFS-USPS2	00Z	00hr < Forecast Time <= 120h	All
57011-57020	GFS-USPS2	06Z	00hr < Forecast Time <= 120h	All
57021-57030	GFS-USPS2	12Z	00hr < Forecast Time <= 120h	All
57031-57040	GFS-USPS2	18Z	00hr < Forecast Time <= 120h	All
59001-59010	ICA-GLME	00Z	00hr	All
59101-59110	ICA-NHPS	00Z	00hr	All
59201-59210	ICA-NWME	00Z	00hr	All
60001-60010	ICN-ALL	Every hour	00hr,01hr,02hr,03hr,06hr,09hr,12hr	All
62001-62010	NAH-ALL	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 72h	All
63001-63010	NCE-ALL	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 72h	All
65101-65110	NGM-AKPS	00Z,12Z	00hr < Forecast Time <= 48h	All
65201-65210	NGM-USLC	00Z,12Z	00hr < Forecast Time <= 48h	All
65301-65310	NGM-USPS	00Z,12Z	00hr < Forecast Time <= 48h	All
65401-65410	NGM-USPS2	00Z,12Z	00hr < Forecast Time <= 48h	All
66001-66010	RCM-ALL	00Z - 2359Z	00hr	All
67001-67010	RFC-ALL	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 72h	All
68001-68010	MAPS (RUC-USLC)	00Z<=Run Time<03Z	00hr,03hr,06hr,09hr,12hr	All
68011-68020	MAPS (RUC-USLC)	03Z<=Run Time<06Z	00hr,03hr,06hr,09hr,12hr	All
68021-68030	MAPS (RUC-USLC)	06Z<=Run Time<09Z	00hr,03hr,06hr,09hr,12hr	All
68031-68040	MAPS (RUC-USLC)	09Z<=Run Time<12Z	00hr,03hr,06hr,09hr,12hr	All
68041-68050	MAPS (RUC-USLC)	12Z<=Run Time<15Z	00hr,03hr,06hr,09hr,12hr	All
68051-68060	MAPS (RUC-USLC)	15Z<=Run Time<18Z	00hr,03hr,06hr,09hr,12hr	All
68061-68070	MAPS (RUC-USLC)	18Z<=Run Time<21Z	00hr,03hr,06hr,09hr,12hr	All
68071-68080	MAPS (RUC-USLC)	21Z<=Run Time<00Z	00hr,03hr,06hr,09hr,12hr	All
68101-68110	MAPS (RUC-USLC2)	00Z<=Run Time<03Z	00hr through 09hr,12hr	All
68111-68120	MAPS (RUC-USLC2)	03Z<=Run Time<06Z	00hr through 09hr,12hr	All
68121-68130	MAPS (RUC-USLC2)	06Z<=Run Time<09Z	00hr through 09hr,12hr	All
68131-68140	MAPS (RUC-USLC2)	09Z<=Run Time<12Z	00hr through 09hr,12hr	All
68141-68150	MAPS (RUC-USLC2)	12Z<=Run Time<15Z	00hr through 09hr,12hr	All
68151-68160	MAPS (RUC-USLC2)	15Z<=Run Time<18Z	00hr through 09hr,12hr	All
68161-68170	MAPS (RUC-USLC2)	18Z<=Run Time<21Z	00hr through 09hr,12hr	All
68171-68180	MAPS (RUC-USLC2)	21Z<=Run Time<00Z	00hr through 09hr,12hr	All
68201-68210	MAPS (RUC-USPS)	00Z<=Run Time<03Z	00hr,03hr,06hr,09hr,12hr	All
68211-68220	MAPS (RUC-USPS)	03Z<=Run Time<06Z	00hr,03hr,06hr,09hr,12hr	All
68221-68230	MAPS (RUC-USPS)	06Z<=Run Time<09Z	00hr,03hr,06hr,09hr,12hr	All
68231-68240	MAPS (RUC-USPS)	09Z<=Run Time<12Z	00hr,03hr,06hr,09hr,12hr	All
68241-68250	MAPS (RUC-USPS)	12Z<=Run Time<15Z	00hr,03hr,06hr,09hr,12hr	All
68251-68260	MAPS (RUC-USPS)	15Z<=Run Time<18Z	00hr,03hr,06hr,09hr,12hr	All
68261-68270	MAPS (RUC-USPS)	18Z<=Run Time<21Z	00hr,03hr,06hr,09hr,12hr	All
68271-68280	MAPS (RUC-USPS)	21Z<=Run Time<00Z	00hr,03hr,06hr,09hr,12hr	All
69001-69010	MDR-ALL	-----	All Observational Hours	All
72001-72010	NPH-ALL	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 72h	All
75001-75010	SST-GLME	00Z	00hr	All
75101-75110	SST-GLME2	00Z	00hr	All

75201-75210	SST-GLMETHIN	00Z	00hr	All
76001-76010	UKM-GLME	00Z,12Z	00hr < Forecast Time <= 120h	All
76101-76110	UKM-GLMETHIN	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 72h	All
77001-77010	WHG-ALL	00Z,12Z	All Forecast Hours out to 60 hours	All
78001-78010	WNA-ALL	00Z,06Z,12Z,18Z	00hr < Forecast Time <= 120h	All
79101-79110	WWF-GLMETHIN	00Z,12Z	12hr,24hr,48hr,72hr	All

Creating GRIB1 and GRIB2 Files with DMBIN

When the unpacking process is complete and the entire GRIB message is successfully decoded, the decoder passes the GRIB structures to DMBIN, which stores metadata about the GRIB messages in the MySQL/MariaDB database. DMBIN also writes raw GRIB1, GRIB2, and BUFR data to files to which the database points. DMBIN does not use the RTMODELS.CFG file.

Once the correct GRIB1 or GRIB2 file is determined, DMBIN files the metadata in the database as is. It is up to the GRIB server to determine if the grid must be pieced together with other corresponding grids of the same field, time, etc. Piecing together is often necessary because most of the gridded fields that cover the globe are sent in four or eight pieces, as in the GFS-GLME dataset.
