



McIDAS-XCD Installation and Configuration Guide

Beta Version 0.1 (30 September 2023)

This document is available online at:

https://www.ssec.wisc.edu/mcidas/software/xcd/beta/Beta_McIDAS-XCD_Install_Procedure.pdf

Need Assistance?

Send your questions/requests to the McIDAS Help Desk:
mug@ssec.wisc.edu

Table of Contents

[Copyright Statement](#)

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

- [Overview](#)
- [Packaging](#)
- [Data Types](#)

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

- [System Requirements](#)
- [Configuring Accounts for McIDAS-XCD](#)
 - [Account Names and Uses](#)
 - [Configuring the Accounts](#)
- [Installing Containers in the *oper* Account](#)
- [Installing Custom LDM Configuration Files for the LDM Container](#)
- [Starting and Stopping McIDAS-XCD](#)
- [Installing McIDAS-X and McIDAS-XCD](#)
- [Installing the McIDAS-X ADDE Datasets](#)

Chapter 3 - [Configuring New-XCD](#)

- [Configuring the Data Directory](#)
- [Configuring the *start_xcd* Script](#)
 - [Default Container Configuration for the *start_xcd* Script](#)
 - [Default Data Parameters for the *start_xcd* script](#)
 - [Days of Data to Keep](#)
 - [Minute of Each Hour to Scour Data](#)
 - [Turning Off Containers from the *start_xcd* Script](#)
- [Turning On the CONDUIT Data Feed](#)

Chapter 4 - [Operations](#)

- [McIDAS-XCD Monitor](#)
- [McIDAS-XCD Log Files](#)

Appendix A - [Known Issues](#)

Appendix B - [Frequently Asked Questions](#)

Appendix C - [Optional Customization](#)

Appendix D - [Default ADDE Datasets](#)

Copyright Statement

Copyright© 2020-2023
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

McIDAS-XCD is a software package that enables Red Hat Enterprise Linux workstations receiving the NWSTG/NCEP channel of the National Weather Service NOAAPORT broadcast to process and serve the data in formats that are compatible with McIDAS-X. Most data are stored in their native file format, including GRIB Version 1, GRIB Version 2, BUFR, Radar imagery (a unique binary format), and raw text files. The data is available to McIDAS-X users via ADDE so they can display, analyze and interpret the data. McIDAS-X can also be used to convert compatible McIDAS-XCD data to McIDAS-X MD and GRID file formats.

See the topics below for more information:

- [Overview](#)
- [Packaging](#)
- [Data Types](#)

Overview

McIDAS-XCD, a software package that is currently in active development, replaces the legacy [McIDAS-XCD](#) software package that has reliably provided conventional data to McIDAS-X users for many years.

The redesign was necessitated due to the legacy package's complex install and upgrade procedures (primarily due to its mainframe-to-Unix history). The redesigned McIDAS-XCD package provides improvement in those areas by utilizing modern packaging solutions (Podman containers), and improvement in data filing and serving performance by utilizing SQLite, a simple open-source database.

Packaging

McIDAS-XCD utilizes component-based packaging using Podman containers. From <https://podman.readthedocs.io/en/latest/Introduction.html>:

Containers simplify the consumption of applications with all of their dependencies and default configuration files. Users test drive or deploy a new application with one or two commands instead of following pages of installation instructions.

This use of containers allows for the simplification of McIDAS-XCD installation and configuration, reducing the chance of implementation errors and allowing for smoother out-of-the-box performance.

[Chapter 2](#) contains the McIDAS -XCD installation instructions and configuration options.

Data Types

As with the legacy package, McIDAS-XCD uses a combination of text ingestors, binary ingestors, data monitors and decoders to define what actions are taken as data is received. The main data types, their default ADDE dataset names, and a description of how their data is processed is described in these sections:

- [Text Data](#)
- [Point Data](#)
- [GRIB Data](#)
- [Radar Imagery](#)

Text Data

Text data, which users access with McIDAS-X commands like WXTLIST, WWLIST, WWDISP and SYNRPRT, is available to users in the ADDE group RTWXTEXT (e.g., dataset RTWXTEXT/SYNOPTIC).

The LDM container files text data directly to disk as a daily file with extension .XCD. A bash daemon script in the McIDAS-XCD text container watches for new data and extracts metadata for insertion into a daily SQLite database on the local disk. McIDAS-X text servers (wxtgserv and obtgserv) query the daily SQLite databases to find the requested data and return it to the client.

The legacy McIDAS-XCD package used rapid-access text files to more quickly access certain types of text data. The upgraded McIDAS-XCD system does not create these files, but instead uses queries based on WMO headers and the daily station database table (described in the Point Data section below) to more quickly access the data.

Point Data

Unlike the legacy McIDAS-XCD package, the upgraded McIDAS-XCD does not create McIDAS MD files. However, it does create the structure needed to fulfill requests on demand from the Point servers for McIDAS-X client commands like PTLIST, PTDISP, SFCLIST, SFCPLOT and RAOBCON. The data is available to users in the ADDE group RTPTSRC (e.g., datasets RTPTSRC/SFCHOURLY and RTPTSRC/SHIPBUOY).

McIDAS-XCD uses certain text data identified by WMO headers filed in the daily SQLite text database (e.g., SA and SP for SFCHOURLY). At the start of each UTC day, a station table that is based on the current version of the McIDAS-X file STNDB.CORE is created in that day's database. Because McIDAS-XCD does **not** store the data in MD files, all of the METAR special observations are available (rather than the maximum of two specials per hour that are stored by the legacy system in MD files).

Note that MD files can still be created from the McIDAS-XCD data by using the McIDAS-X PTCOPY command.

GRIB Data

GRIB files, which can be in Version 1 or Version 2, are available to users in the ADDE group RTGRIDS (e.g., datasets RTGRIDS/GFS-GLME and RTGRIDS/NAM-USLC). The volume of GRIB data is typically more than 500 GB per day.

The GRIB messages are first filed to a temporary directory by the LDM container. A Python daemon within the xcdgrib container continually monitors the temporary directory. When it detects new data, the daemon extracts information and files metadata into a SQLite database while archiving the GRIB files on the local machine. There are separate SQLite databases for each GRIB version, model and date.

NEXRAD Radar Data

NEXRAD radar data, which can be Weather Surveillance Radar or Terminal Doppler Weather Radar products, is available to users in the ADDE groups RADAR, WSR and TDWR (e.g., datasets RADAR/WSR-BREF1, WSR/BREF1 and TDWR/CREF).

The LDM container directly files the radar data into a directory structure that includes the station name and products. The data is served by the NEXRAD radar data servers.

[Chapter 3](#) contains the McIDAS-XCD configuration options and instructions for each of the data types.

Chapter 2 - Installing McIDAS-XCD

This chapter consists of the instructions for installing the McIDAS-XCD package. The procedure consists of these tasks:

- [System Requirements](#)
- [Configuring Accounts for McIDAS-XCD](#)
 - [Account Names and Uses](#)
 - [Configuring the Accounts](#)
- [Installing Containers in the *oper* Account](#)
- [Installing Custom LDM Configuration Files in the LDM Container](#)
- [Starting and Stopping McIDAS-XCD](#)
- [Installing McIDAS-X Servers for McIDAS-XCD Data](#)
- [Installing the McIDAS-X ADDE Datasets](#)

(additional content, if any)

System Requirements

McIDAS-XCD requires a multi-processor system. A minimum requirement based on MUG testing machines would be **thirty-two** Intel(R) Xeon(R) Silver 4208 central processing units at 2.10 GHz. However, with the aforementioned system, text processing gets behind during times where the full LDM CONDUIT feed supplies a large amount of data at once. SSEC testing has shown that a machine with **forty-eight** Intel(R) Xeon(R) E5-2687W v4 central processing units at @ 3.00GHz is able to keep up at those times. Another option is to disable the CONDUIT feed by commenting out the CONDUIT lines in the SSEC-supplied LDM file **pqact.conf** and then run CONDUIT only on a separate multi-processor machine.

SSEC tests and runs McIDAS -XCD using the following system and software configuration. To aid in support, we recommend that your system be configured the same (or as close as possible):

- Red Hat Enterprise Linux (RHEL) 8
- Podman version 4.2 or higher.

Podman should be installed automatically with RHEL 8. If not or if the version recommended is not installed, podman needs to be installed on the machine by a system administrator or anyone with root privileges.

Note: Podman emulates the Docker command-line interface (CLI) with `alias docker=podman`. This allows Docker and Podman commands to be used interchangeably.

Podman also needs to be set up to allow network ports 80 and 388 to be used by the installed containers. Here is a description of these network ports:

- Default port 388 for LDM
- Default port 80 for httpd (webpage server)

For the McIDAS-XCD monitor web server to work properly, the system administrator needs to add a firewall rule for allowing http/https if the system does not have Apache managed via configuration management. This can be achieved by adding the Apache firewall class directly.

Optionally, if an administrator has access to the SSEC Gitlab repository and wants to obtain `mcidasxcdcontainer` source and configuration files, this requires the following software:

- Git

See [Appendix B - Optional Running McIDAS-XCD Containers Method](#) for this optional method.

Configuring Accounts for McIDAS -XCD

We recommend using separate accounts for the installation and operations of McIDAS-XCD. See the sections below for the recommended account names and required configuration changes.

- [Account Names and Uses](#)
- [Configuring the Accounts](#)

Account Names and Uses

oper - Technically, one can install the containers in any account on a machine. However, for consistency with the McIDAS-XCD legacy package, this document will show how to install the McIDAS-XCD containers as user *oper*.

mcidas - This is the account where McIDAS-X is compiled and installed. After McIDAS-X is installed, the McIDAS-XCD servers can be compiled and installed separately.

mcadde - This is the same as the McIDAS-XCD legacy package as this is the account from which the McIDAS-XCD servers are run and where the datasets are defined in RESOLV.SRV.

Configuring the Accounts

The *oper* user needs permissions to create containers on the chosen machine. A system administrator or anyone with root privileges will accomplish this by editing the `/etc/subuid` and `/etc/subgid` files for the *oper* user, adding fields in the format USERNAME:UID:RANGE. For example:

```
user1:100000:65536
oper:165536:65536
```

Note: If the McIDAS-XCD machine is monitored by a system administration tool such as Puppet, you may have to obtain additional permissions to change the subuid and subgid files so that the changes made above are not overwritten.

After the entries for the *oper* user have been added, type the following as the *oper* user:

```
podman system migrate
```

If executing the above command results in no error messages, the administrator can proceed to the next section.

Installing McIDAS-XCD Containers in the *oper* Account with .tar Files

By default, we recommend downloading .tar files that contain the McIDAS-XCD container images. Optionally, if an administrator has access to the SSEC gitlab repository, the method detailed in [Appendix B - Optional Running McIDAS-XCD Containers Method](#) may be used.

The following are the details for the installation of McIDAS-XCD using tar files that contain the McIDAS-XCD container images.

1. Log in to the McIDAS -XCD machine as user `oper`:
2. Download the `McIDAS-XCD.tar.Z` file using the link at <https://www.ssec.wisc.edu/mcidas/software/xcd/xcd-beta/> in the McIDAS-XCD Beta Release and Instructions section.
3. Set the `HOST_DATA_DIR` environmental variable and add it to the `/home/oper/.profile` file. For example:

```
export HOST_DATA_DIR=/data/xcd
```

This variable identifies where the McIDAS-XCD container software will store the data for use by the McIDAS-XCD servers. See Chapter 3, [Configuring the Data Directory](#) for more details.

4. Run the following tar command from the `/home/oper` directory, then go to the directory to where the images are extracted to:

```
tar xvf McIDAS-XCD.tar.gz
cd /home/oper/McIDAS-XCD
```

5. Load the images with the following command:

```
./load_image_files
```

6. List the installed Podman images:

```
[oper@<your machine> McIDAS-XCD]$ podman image ls
REPOSITORY          TAG          IMAGE ID      CREATED      SIZE
localhost/xcddecoders  latest      fa3ecf1ba571  25 hours ago  907
MB
docker.io/library/httpd  latest      c58ef9bfbb57  2 weeks ago  148
MB
docker.io/unidata/ldm-docker  latest    c76363b2ddab  6 weeks ago  640 MB
```

Installing Custom LDM Configuration Files for the LDM Container

Before running the McIDAS-XCD containers, the site operators may want to configure the files necessary for the LDM to be changed for customization. Predefined configuration files based on what the SSEC McIDAS-XCD system uses are located in the `/home/oper/McIDAS-XCD/etc` directory. Below are our recommendations and requirements:

registry.xml - We recommend changing the value of the queue size to at least 8G (8 Gigabytes)

ldmd.conf - We recommend replacing this with your pre-existing ldmd.conf file. Otherwise, the administrator can use the supplied file in the `/home/oper/McIDAS-XCD/etc` directory.

pqact.conf - We require our supplied pqact.conf to be appended to your existing pqact.conf file. This is because our version contains the directory structure expected by the McIDAS-XCD containers. Run a command like the following:

```
cat /home/oper/McIDAS-XCD/etc/pqact.conf >> <your pqact.conf>
```

Once you have the above three files specific for your site along with the pqact.conf file appended with the McIDAS -XCD-specific changes, copy them into the `/home/oper/McIDAS-XCD/etc` directory with the expected names before starting McIDAS-XCD.

Starting and Stopping McIDAS-XCD

1. From the current directory of `/home/oper/McIDAS-XCD`, start McIDAS-XCD with the following command:

```
./start_xcd
```

2. Wait until a command prompt returns before listing the running containers:

```
[oper@csppgamma McIDAS-XCD]$ podman ps
```

CONTAINER ID	IMAGE	COMMAND NAMES	CREATED	STATUS
bfbe85c8de7e	k8s.gcr.io/pause:3.5		14 seconds ago	Up 13
seconds ago	0.0.0.0:80->80/tcp, 0.0.0.0:388->388/tcp	5aclc7e39458-infra		
dca80934f71c	docker.io/unidata/ldm-docker:latest	runldm.sh	13 seconds ago	Up 13
seconds ago	0.0.0.0:80->80/tcp, 0.0.0.0:388->388/tcp	xcdldm		
bdbe6939a4c0	localhost/xcddecoders:latest	xcdadmin start gr...	12 seconds ago	Up 13
seconds ago	0.0.0.0:80->80/tcp, 0.0.0.0:388->388/tcp	xcdgrib		
591c62368b41	localhost/xcddecoders:latest	xcdadmin start te...	12 seconds ago	Up 12
seconds ago	0.0.0.0:80->80/tcp, 0.0.0.0:388->388/tcp	xcdtext		
0274415d4b9b	localhost/xcddecoders:latest	xcdadmin start mo...	12 seconds ago	Up 12
seconds ago	0.0.0.0:80->80/tcp, 0.0.0.0:388->388/tcp	xcdmonitor		
c49d60b5b3d6	localhost/xcddecoders:latest	crond	11 seconds ago	Up 11
seconds ago	0.0.0.0:80->80/tcp, 0.0.0.0:388->388/tcp	xcdscour		
c01b10b612af	docker.io/library/httpd:latest	httpd-foreground	11 seconds ago	Up 11
seconds ago	0.0.0.0:80->80/tcp, 0.0.0.0:388->388/tcp	xcdwebserver		

3. Let the containers run for about 5 minutes or so, checking again that the containers are running. This will allow the LDM to completely initialize, allowing for the pqact.conf.state file to be written upon shutdown.

4. Stop McIDAS-XCD with the following command:

```
./stop_xcd
```

Note: During McIDAS-XCD shutdown with the **stop_xcd** command, LDM will write the file `/home/oper/McIDAS-XCD/etc/pqact.conf.state`. This file keeps track of where in the queue file that LDM processing left off. It is highly recommended not to delete this file unless you want to reprocess the data in the entire queue file and delay current processing.

Installing McIDAS-XCD Servers

McIDAS-X must be built and installed on the workstation before the McIDAS-XCD servers can be built and installed. The McIDAS-XCD servers in the McIDAS-X package must be installed in order for a client machine to retrieve data from the McIDAS-XCD machine.

Install Procedure

McIDAS-XCD servers are included with the McIDAS-X 2022.1 packages/tar files and can be installed from the command line as user `mcidas`. The McIDAS-X package now comes with SQLite libraries and flags that are used to compile the McIDAS-XCD servers with SQLite.

Follow the instructions for downloading McIDAS-X at <https://www.ssec.wisc.edu/mcidas/software/x/download/> and installation at https://www.ssec.wisc.edu/mcidas/doc/users_guide/2022.1/install-2.html.

McIDAS-X Version 2022.1 or higher **MUST** be compiled and installed before executing the following steps. McIDAS-X does not need to be compiled with a flag like `-mysql` as with previous versions of McIDAS-XCD. This is because the McIDAS-XCD Beta servers utilize the SQLite libraries that are included by default in McIDAS-X Version 2022.1 and higher. If the machine you are using already has McIDAS-X 2022.1 compiled and installed with the `-mysql` flag, you **MUST** unmake `-X` and remake and reinstall without the `-mysql` flag. The following is that procedure from the `/home/mcidas` directory:

```
sh ./mcidas2022.1.sh unmake
sh ./mcidas2022.1.sh make
sh ./mcidas2022.1.sh install
```

The following is the procedure to install the McIDAS-XCD servers with McIDAS-X 2022.1 having been compiled and installed **without** the `-mysql` flag:

1. Log on to the workstation as the user `mcidas`. You should be in the `/home/mcidas` directory.
2. There have been updates to the McIDAS-XCD Server code since first being included in McIDAS-X 2022.1. To remedy this, copy the following two files into the `/home/mcidas` directory:
 - [build_McIDAS-XCD_Beta_XServer_Update.sh](#)

- [McIDAS-XCD_Beta_XServers_Update.tar](#)

3. Copy McIDAS-XCD Beta McIDAS-X server updates into the correct directory locations and compile and install the McIDAS-XCD Beta McIDAS-X Servers by running the following command from the **/home/mcidas** directory:

```
sh ./build_McIDAS-XCD_Beta_XServers_Update.sh
```

4. Log out of the **mcidas** account.
5. Install the McIDAS-X ADDE remote server with **mcadde** for the remote server account name. See [Installing the McIDAS-X ADDE Remote Server](#) in the *McIDAS-X User's Guide* for the instructions and more information.

Installing the McIDAS-X ADDE Datasets

The default ADDE datasets are created by running batch files provided with the McIDAS-XCD servers. 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). See [Appendix C - Default ADDE Datasets](#) for listings of the datasets created by these batch files.

The instructions below complete the procedure by running the XCDADDE.BAT batch file.

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

```
mcidas
```

3. In the McIDAS-X session, 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 was specified as the **HOST_DATA_DIR** environment variable in the Podman container procedure.

Type: **TE MCDATA "/data_directory_path**

For example: **TE MCDATA "/data/xcd**

4. In the McIDAS-X session, 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, if you

have a preexisting `~mcadde/mcidas/data/RESOLV.SRV` file, we recommend that you make a backup copy of that file before running the command below.

Type: **BATCH XCDADDE.BAT**

5. In the McIDAS-X session, register the McIDAS-X schemas with the following command:

Type: **BATCH XCDSHEMA.BAT**

6. Log out of the **mcadde** account

Chapter 3 - Configuring McIDAS-XCD

This chapter consists of the instructions for configuring the McIDAS-XCD package. The procedure consists of these tasks:

- [Configuring the Data Directory](#)
- [Configuring the `start_xcd` Script](#)
 - [Default Container Configuration for the `start_xcd` Script](#)
 - [Default Data Parameters for the `start_xcd` script](#)
 - [Days of Data to Keep](#)
 - [Minute of Each Hour to Scour Data](#)
 - [Turning Off Containers from the `start_xcd` Script](#)
- [Turning On the CONDUIT Data Feed](#)

(additional content, if any)

Configuring the Data Directory

The LDM and McIDAS-XCD containers use a volume mount (created when `./start_xcd` is run) to write data to a directory on the host machine. This directory must be specified using the `HOST_DATA_DIR` environmental variable. If the variable is not set, the containers will fail to start when `./start_xcd` is run.

Set the `HOST_DATA_DIR` environmental variable with the path to the directory containing McIDAS-XCD data and configuration files. If the directory and XCD data do not already exist then they will be created when the containers start up.

An export command should also be added to the `~/.profile` file of the `oper` user.

As an example:

```
export HOST_DATA_DIR=/data
```

Configuring the `start_xcd` Script

Each of the decoders (`grib` and `text`), `monitor`, and the `scour` scripts packaged in the `xcddecoder` image must be run in its own separate container. Additionally, an Apache web server (`httpd`) must be run to serve the McIDAS-XCD monitor webpage. The `start_xcd` script file defines the images and configuration used for the multiple container setup. Combined with the `stop_xcd` script file, this allows the handling of the containers' startup and shutdown.

The configurations that the `start_xcd` script file handles include:

- Images/containers to run
- Network ports
 - Default port 388 for LDM
 - Default port 80 for `httpd` (webpage server)
- Volume mounts
- Default number of days of data for the GRIB, `text`, and radar containers.
- Default minute of each hour that GRIB, `text`, and radar data are scoured from the local machine.

For the most part, the default `start_xcd` script should be sufficient to start the McIDAS-XCD containers. If a network port is already in use on the host, the containers will fail to start until the port is no longer in use or is changed to an unused port in the script.

Default Container Configuration for the `start_xcd` Script

Here are some variables to be aware of at the top of the `start_xcd` script. It is recommended not to edit these values.

- `USERNAME`: Username inside the `xcddecoders` containers. Defaults to 'root'.
- `USER_ID`: User ID inside `xcddecoders` containers. Defaults to '0'.
- `GROUP_ID`: User ID inside `xcddecoders` containers. Defaults to '0'. Recommended to match `USER_ID`.

The following are the default settings for the Access Control List (ACL) for users `oper` and `mcadde` as set up on the host machine for the `HOST_DATA_DIR`.

- `setfacl -R -d -m u:mcadde:rwX ${HOST_DATA_DIR}`
- `setfacl -R -d -m u:oper:rwX ${HOST_DATA_DIR}`

The above `setfacl` commands are done because the McIDAS-XCD servers are run from the `mcadde` account and need permissions to access the SQLite databases and files that are saved on the local -XCD machine.

Default Data Parameters for the `start_xcd` script

Days of Data to Keep

In the `start_xcd` script, the operator can specify a predefined amount of time for data to keep of each data type. Because GRIB data coming across NOAAPORT is in such high volumes, we recommend retaining 2 days of data. For text, the default is 7 days of data. For radar, the default is the last 3 hours of data. The operator can change these values at any time, but, after editing, must run `stop_xcd` followed by `start_xcd`.

Here are the default values as contained in the `start_xcd` script:

```
# How many days of GRIB/TEXT data to save
DAYS_GRIB_DATA=2
DAYS_TEXT_DATA=7
# How many hours of RADAR data to save
HOURS_RADAR_DATA=3
```

Minute of Each Hour to Scour Data

McIDAS-XCD utilizes a `xcdscour` container that is responsible for deleting data based on the parameters in the Days of Data to Keep section above. A cron daemon is run each hour within the container for GRIB and text data together while a separate daemon does the same for radar data. The operator can change these values at any time, but, after editing, must run `stop_xcd` followed by `start_xcd`.

Here are the default values as contained in the `start_xcd` script:

```
# What minute of every hour to scour the data
MINUTE_GRIBTEXT_SCOUR=40
MINUTE_RADAR_SCOUR=45
```

For example, a value for `MINUTE_GRIBTEXT_SCOUR` of 40 means that scouring for that kind of data will start at 40 minutes past the top of every hour the container is running.

Turning Off Containers from the `start_xcd` Script

There may be instances where you want to run individual McIDAS-XCD containers on separate machines. For example, the McIDAS-XCD GRIB container requires quite a bit of disk space and processing resources compared to the McIDAS-XCD text container. If this is the case, you can edit the `start_xcd` script by commenting out the line after the “Run <name> container:” line.

For example, after the “####Run GRIB container line:”, add ‘#’ before the following line as so:

```
#podman run -dt --ipc private --shm-size 512m --pod xcd -v ${HOST_DATA_DIR}:/data
```

```
--env XCD_USERNAME=$USERNAME --env XCD_USER_ID=$USER_ID --env XCD_GROUP_ID=$GROUP_ID
--env HOST_DATA_DIR=$HOST_DATA_DIR --name xcdgrib xcdcoders:latest xcdadmin start
grib
```

After editing the file, be sure to run `stop_xcd` and `start_xcd` scripts from the `/home/oper/McIDAS-XCD` directory.

Turning On the CONDUIT Data Feed

By the default `~oper/McIDAS-XCD/etc/ldmd.conf` file, the LDM CONDUIT data feed is not active. This is because the CONDUIT data feed has a large amount of GRIB data that comes across it and could overwhelm a system at certain times of the day.

If you have a powerful enough system, the McIDAS-XCD administrator can add the CONDUIT feed by uncommenting out the following lines (removing the '#') in the `~oper/McIDAS-XCD/etc/ldmd.conf` file.

```
#request CONDUIT "[0]$" idd.unidata.ucar.edu
#request CONDUIT "[1]$" idd.unidata.ucar.edu
#request CONDUIT "[2]$" idd.unidata.ucar.edu
#request CONDUIT "[3]$" idd.unidata.ucar.edu
#request CONDUIT "[4]$" idd.unidata.ucar.edu
#request CONDUIT "[5]$" idd.unidata.ucar.edu
#request CONDUIT "[6]$" idd.unidata.ucar.edu
#request CONDUIT "[7]$" idd.unidata.ucar.edu
#request CONDUIT "[8]$" idd.unidata.ucar.edu
#request CONDUIT "[9]$" idd.unidata.ucar.edu
```

After editing the file, be sure to run `stop_xcd` and `start_xcd` scripts from the `~oper/McIDAS-XCD` directory to get the CONDUIT data feed running.

Chapter 4 - Operations

At any time on the McIDAS-XCD machine from the `oper` account, you can see the current status of the containers by typing

```
podman ps
```

McIDAS-XCD Monitor

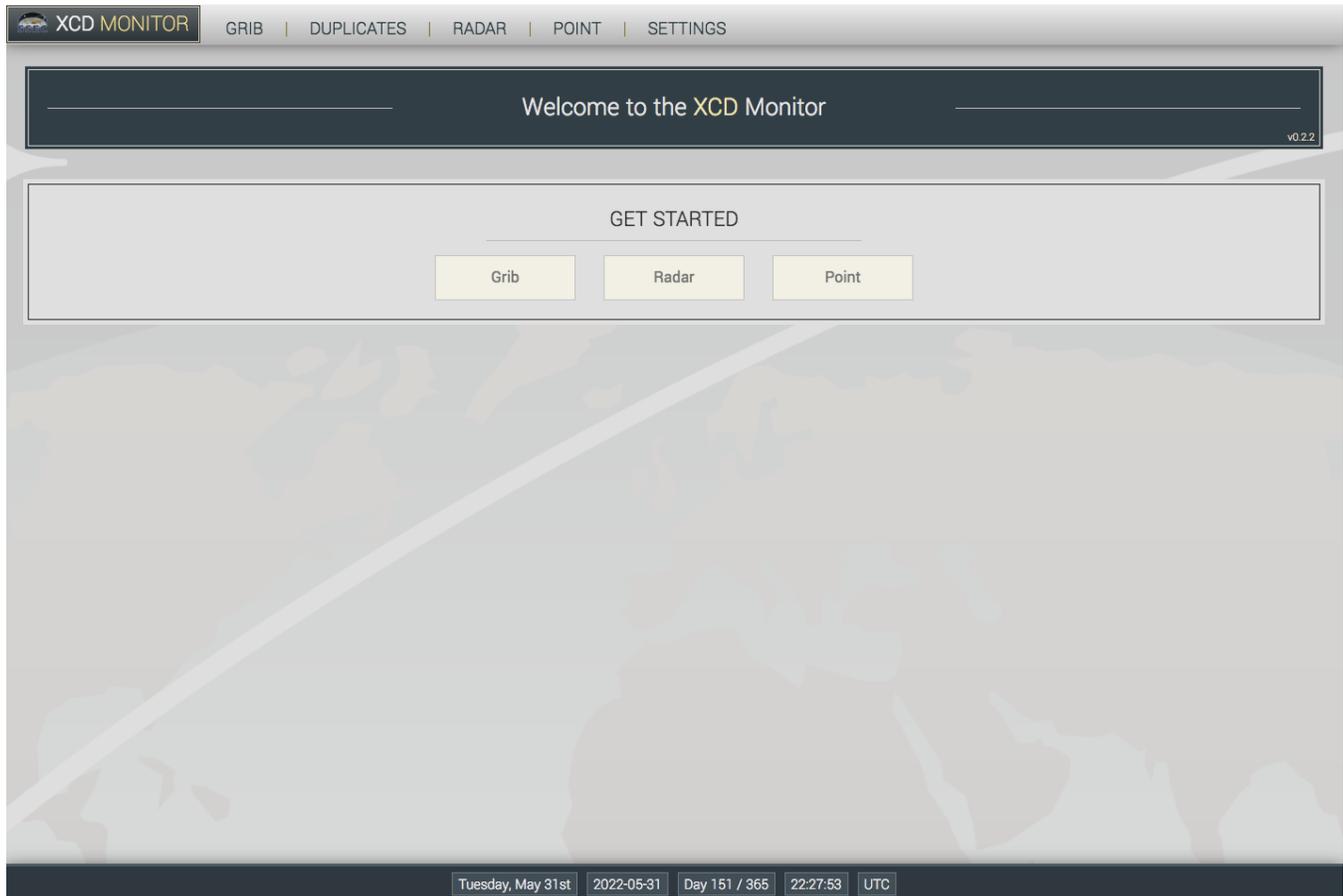
The McIDAS-XCD data monitor is used to identify the kind and amount of expected data coming in. There are separate monitors for Text/Point, GRIB, and RADAR. Port 1080 on the machine is the default

for viewing the monitor. The monitor should be viewable with computers on the same network as the McIDAS-XCD machine depending on how the networking rules have been set up.

After opening a web browser such as Safari or Firefox, type:

```
http://<MACHINE NAME HERE>
```

The following screen will come up:



At this point, you can click one of the GRIB, Radar, or Point buttons to get started. Once arriving at a page, you can use the top menu to scroll between windows from then on.

McIDAS-XCD Log Files

Log files for the xcddecoders container are located in the following directory:

```
${HOST_DATA_DIR}/logs
```

For example if `${HOST_DATA_DIR}` is defined as `/data/xcd`, the log file directories are located in `/data/xcd/logs`.

Appendix A - Known Issues

1. The LDM's CONDUIT data feed is a veritable firehose of data at certain times of the day fed through the xcdgrib container. Certain multiprocessor machines are able to keep up with the processing, but others unfortunately are not. Mainly, this seems to interfere with the processing done through the xcdtext container. At peak CONDUIT times, we have observed xcdtext processing get behind by about a half-hour and be back to current processing within the hour. For a real-time processing system, this is not ideal.

As an alternate approach, the -XCD system by default does not have the CONDUIT system turned on as accomplished in the etc/ldmd.conf file with the CONDUIT lines commented out. This means that the default xcdgrib container processes only the NGRID feed and the few grids that still come across the WMO feed. We advise that if you want to run the CONDUIT feed that you do so on a separate system, uncommenting the CONDUIT lines in the etc/ldmd.conf file while commenting out WMO, NGRID, and NNEXRAD entries.

2. BUFR files are currently not supported. We have tested a xcdbufr container, but it is still in development. McIDAS-X also needs servers for the BUFR data. It is hoped that the ECMWF eccodes package will aid in the creation of the xcdbufr container and the McIDAS-X BUFR servers.

Appendix B - Frequently Asked Questions

Question: The load on the McIDAS-XCD machine is high when using tools such as top and htop. Is this expected?

Answer: The legacy McIDAS-XCD system was designed for use on a single processor machine. The current McIDAS-XCD system is designed to use multiple processors. If you see a load of 72 on the system, but there are 36 processors, it would be similar to a load of 2 on a single processor system. We mostly see very high loads when using the CONDUIT feed for the xcdgrib container as a high volume of data comes across corresponding to the 0/6/12/18 Z GFS forecasts.

Question: Daily queries of the data sometimes take much longer than the legacy McIDAS-XCD. For example, if I am searching for all the US stations with temperatures between 80 and 100 degrees Fahrenheit on a day in July, a command sent to the McIDAS-XCD system may take upwards of 2 or 3 minutes compared to a few seconds on the legacy McIDAS-XCD system. Why is this happening?

Answer: The legacy McIDAS-XCD system used MD files that contained various meteorological data based on the schema for the particular data. The current McIDAS-XCD system uses SQLite databases instead of MD files to store more general data and does not contain any meteorological data. A workaround would be to create MD files with the PTCOPY command and then run the -X command needed to do the above on that data.

MUG may eventually have SQLite databases that more closely resemble the data in MD files to eliminate this problem.

Question: Running the PTLIST (and other) commands in McIDAS-X with dataset definitions pointing to the McIDAS-XCD server machine results in several error messages.

1. PTLIST RTPTSRC/SFCHOURLY SEL='ID KMSN' NUM=ALL
Error Message: PTLIST: Point data server unable to resolve this dataset:
RTPTSRC/SFCHOURLY
2. PTLIST RTPTSRC/SFCHOURLY SEL='ID KMSN' NUM=ALL
Error Message: PTLIST: Error in SELECT clause for parameter ID – parameter specified is not in dataset

Answer: Here are answers in the order above:

1. This indicates that the RTPTSRC dataset is not defined on the McIDAS-XCD server machine. Proceed to [Installing the McIDAS-X ADDE Datasets](#) steps 3 and 4.
2. This indicates that the McIDAS-X schemas are not defined on the McIDAS-XCD server machine. Proceed to [Installing the McIDAS-X ADDE Datasets](#) step 5.

Question: We are still unable to access the McIDAS-XCD server machine data. What could be some other causes?

Answer: Depending on your system, you may need to set up a profile that makes exceptions for user mcadde in relation to the system's firewall. If this is done in something like puppet, a system administrator with higher credentials may need to set a mcadde profile up for you with the appropriate firewall exceptions.

Appendix C - Optional Running McIDAS-XCD Containers

Method

This method is only available to those who have access to the SSEC gitlab repository.

Using the SSEC GitLab Repository to Install and Run McIDAS-XCD Containers

Installing containers for the first time

Log in to the McIDAS -XCD machine as user `oper`:

```
sudo -u oper -i
```

Set the `HOST_DATA_DIR` environmental variable and add it to the `/home/oper/.profile` file. For example:

```
export HOST_DATA_DIR=/data/xcd
```

From the user's GitLab account at <https://gitlab.ssec.wisc.edu>:

Go to **Settings** → **Access Tokens**. Then, answer the questions and click the **Create project access token** button. Make a copy of the generated token in a file of the user's choosing. The token format will be something like `gop_sFhFsSHhTzMDreGRLjks3STrutrkruthwnps`

Configure the local git client with a username and email address

```
git config --global user.name "<your_gitlab_username>"
git config --global user.email "<your_gitlab_email>"
```

Clone the GitLab repository into the `/home/oper` directory with the following procedure:

```
git clone https://gitlab.ssec.wisc.edu/xcd/mcidasxcdcontainer.git
> Cloning into `mcidasxcdcontainer`...
Username: <type your GitLab username>
Password: <type your project access token generated above>
```

Cache the given record in your computer to remember the token:

```
git config --global credential.helper cache
```

Change to the `mcidasxcdcontainer` directory:

```
cd mcidasxcdcontainer
```

To create the McIDAS -XCD Podman pod and to start the McIDAS-XCD containers, run:

```
./start_xcd
```

To verify that the containers have started successfully, run:

```
podman container ls -a
```

Installing updated containers

Log in to the McIDAS -XCD machine as user `oper`:

```
sudo -u oper -i
```

Check for any pre-existing McIDAS -XCD containers currently running with the command:

```
podman container ls -a
```

If there are McIDAS-XCD containers currently running, shut them down and remove them:

```
cd mcidasxcdcontainer
./stop_xcd
podman rmi -a -f
```

Next, pull the podman images and restart the McIDAS-XCD containers

```
git pull
./pull_images
./start_xcd
```

Appendix D - Default ADDE Datasets

This appendix lists the ADDE datasets that are created by default by completing the instructions in the [Installing the ADDE Datasets](#) section above. As noted there, the datasets are created by running BATCH files provided in the McIDAS-XCD package.

The supplied version of the **XCDRTPT.BAT** batch file (which can be run separately, or as part of **XCDADDE.BAT**) is listed below. Each line begins with DSSERVE ADD, and the dataset name it creates is in bold. Note that the #MCDATA variable must be the same as the `HOST_DATA_DIR` variable used above.

```
DSSERVE ADD RTPTSRC/SFCHOURLY PTDB 1 10 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time SFC
Hourly
DSSERVE ADD RTPTSRC/UPPERMAND PTDB 11 20 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time Upper
Air (Mandatory Levels)
DSSERVE ADD RTPTSRC/UPPERSIG PTDB 21 30 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time Upper
Air (Significant Levels)
```

DSSERVE ADD **RTPTSRC/SHIPBUOY** PTDB 31 40 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time Ship and Buoy data

DSSERVE ADD **RTPTSRC/TEMPSHIP** PTDB 41 50 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time Mandatory Level TEMP SHIP (RAOB) data

DSSERVE ADD **RTPTSRC/SYNOPTIC** PTDB 51 60 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time SYNOPTIC data

DSSERVE ADD **RTPTSRC/AIRCRAFT** PTDB 61 70 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time Aircraft data

DSSERVE ADD **RTPTSRC/NAMMOS** PTDB 71 80 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time NAM MOS data

DSSERVE ADD **RTPTSRC/GFSMOS** PTDB 81 90 TYPE=POINT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time GFS MOS data

The supplied version of the **XCDRTWX.BAT** batch file (which can be run separately, or as part of **XCDADDE.BAT**) is listed below. Each line begins with DSSERVE ADD, and the dataset name it creates is in bold.

DSSERVE ADD **RTWXTEXT/SFCHOURLY** OBTX TYPE=TEXT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time SFC Hourly Text

DSSERVE ADD **RTWXTEXT/UPPERAIR** OBTX TYPE=TEXT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time Upper Air Text

DSSERVE ADD **RTWXTEXT/SYNOPTIC** OBTX TYPE=TEXT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time SYNOPTIC Text

DSSERVE ADD **RTWXTEXT/TERMFCS**T OBTX TYPE=TEXT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real-Time Terminal Forecast Text

DSSERVE ADD **RTWXTEXT/NAMMOS** OBTX TYPE=TEXT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real_time NAM MOS Text

DSSERVE ADD **RTWXTEXT/GFSMOS** OBTX TYPE=TEXT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real_time GFS MOS Text

DSSERVE ADD **RTWXTEXT/BLANK WTXG** TYPE=TEXT RT=Y
INFO=""DATABASE=#MCDATA/xcd/textdatabase/(YYYY)(DDD)/(YYYY)(DDD).sqlite"" "SQLite Real_time Weather Text Data

The supplied version of the **XCDRADAR.BAT** batch file (which can be run separately, or as part of **XCDADDE.BAT**) is listed below. Each line begins with DSSERVE ADD, and the dataset name it creates is in bold.

DSSERVE ADD **RADAR/TDWR-BREF1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/TZ0/* "TDWR Base Reflec - tilt 1 (0.1-0.8deg) 48nmi (TZ0)

DSSERVE ADD **RADAR/TDWR-BREF2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/TZ1/* "TDWR Base Reflec - tilt 2 (1.0deg) 48nmi (TZ1)

DSSERVE ADD **RADAR/TDWR-BREF3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/TZ2/* "TDWR Base Reflec - tilt 3 (1.6-3.7deg) 48nmi (TZ2)

DSSERVE ADD **RADAR/TDWR-BVEL1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/TV0/* "TDWR Base Rad Veloc - tilt 1 (0.1-0.8deg) 48nmi (TV0)

DSSERVE ADD **RADAR/TDWR-BVEL2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/TV1/* "TDWR Base Rad Veloc - tilt 2 (1.0deg) 48nmi (TV1)

DSSERVE ADD **RADAR/TDWR-BVEL3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/TV2/* "TDWR Base Rad Veloc - tilt 3 (1.6-3.7deg) 48nmi (TV2)

DSSERVE ADD **RADAR/TDWR-CREF** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/NCR/* "TDWR Composite Reflectivity - 124nmi (NCR)

DSSERVE ADD **RADAR/TDWR-DHR** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/DHR/* "TDWR Digital Hybrid Scan Reflectivity (DHR)

DSSERVE ADD **RADAR/TDWR-LRREF** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/TZL/* "TDWR Long Range Reflectivity - 225nmi (TZL)

DSSERVE ADD **RADAR/TDWR-PCP1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/N1P/* "TDWR Surface Rainfall - 1 hour running total (N1P)

DSSERVE ADD **RADAR/TDWR-PCPT** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/NTP/* "TDWR Surface Rainfall - storm total (NTP)

DSSERVE ADD **RADAR/TDWR-TOPS** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/NET/* "TDWR Echo Tops (NET)

DSSERVE ADD **RADAR/TDWR-VIL** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/tdwr\ID/NVL/* "TDWR Vertical Integrated Liquid (NVL)

DSSERVE ADD **RADAR/WSR-BREF1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/N0Q/* "WSR Base Reflec - tilt 1 (0.5deg) 248nmi (N0Q)

DSSERVE ADD **RADAR/WSR-BREF2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/NAQ/* "WSR Base Reflec - tilt 2 (0.9deg) 248nmi (NAQ)

DSSERVE ADD **RADAR/WSR-BREF3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/N1Q/* "WSR Base Reflec - tilt 3 (1.3|1.5deg) 248nmi (N1Q)

DSSERVE ADD **RADAR/WSR-BREF4** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/NBQ/* "WSR Base Reflec - tilt 4 (1.8deg) 248nmi (NBQ)

DSSERVE ADD **RADAR/WSR-BREF5** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/N2Q/* "WSR Base Reflec - tilt 5 (2.4|2.5deg) 248nmi (N2Q)

DSSERVE ADD **RADAR/WSR-BREF6** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/N3Q/* "WSR Base Reflec - tilt 6 (3.1|3.4|3.5deg) 248nmi (N3Q)

DSSERVE ADD **RADAR/WSR-BVEL1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/N0U/* "WSR Base Rad Vel - tilt 1 (0.5deg) 162nmi (N0U)

DSSERVE ADD **RADAR/WSR-BVEL2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/NAU/* "WSR Base Rad Vel - tilt 2 (0.9deg) 162nmi (NAU)

DSSERVE ADD **RADAR/WSR-BVEL3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/N1U/* "WSR Base Rad Vel - tilt 3 (1.3|1.5deg) 162nmi (N1U)

DSSERVE ADD **RADAR/WSR-BVEL4** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/NBU/* "WSR Base Rad Vel - tilt 4 (1.8deg) 162nmi (NBU)

DSSERVE ADD **RADAR/WSR-BVEL5** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr\ID/N2U/* "WSR Base Rad Vel - tilt 5 (2.4|2.5deg) 162nmi (N2U)

DSSERVE ADD **RADAR/WSR-BVEL6** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N3U/*
 "WSR Base Rad Vel - tilt 6 (3.1|3.4|3.5deg) 162nmi (N3U)

DSSERVE ADD **RADAR/WSR-CREF** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NCR/* "WSR
 Composite Reflectivity - 124nmi (NCR)

DSSERVE ADD **RADAR/WSR-DHR** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DHR/* "WSR
 Digital Hybrid Scan Reflectivity (DHR)

DSSERVE ADD **RADAR/WSR-DPA** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DPA/* "WSR
 Digital Precipitation Array (DPA)

DSSERVE ADD **RADAR/WSR-DSP** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DSP/* "WSR
 Digital Storm Total Precipitation (DSP)

DSSERVE ADD **RADAR/WSR-DVIL** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DVL/* "WSR
 Digital Vertical Integrated Liquid (DVL)

DSSERVE ADD **RADAR/WSR-ENHTOPS** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/EET/*
 "WSR Enhanced Echo Tops (EET)

DSSERVE ADD **RADAR/WSR-PCP1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N1P/* "WSR
 Surface Rainfall - 1 hour running total (N1P)

DSSERVE ADD **RADAR/WSR-PCPT** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NTP/* "WSR
 Surface Rainfall - storm total (NTP)

DSSERVE ADD **RADAR/WSR-SRVEL1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N0S/*
 "WSR Storm-Rel Mean Rad Vel - tilt 1 (0.5deg) (N0S)

DSSERVE ADD **RADAR/WSR-SRVEL2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N1S/*
 "WSR Storm-Rel Mean Rad Vel - tilt 2 (1.3|5deg) (N1S)

DSSERVE ADD **RADAR/WSR-SRVEL3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N2S/*
 "WSR Storm-Rel Mean Rad Vel - tilt 3 (2.4|5deg) (N2S)

DSSERVE ADD **RADAR/WSR-SRVEL4** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N3S/*
 "WSR Storm-Rel Mean Rad Vel - tilt 4 (3.1|3.4|3.5deg) (N3S)

DSSERVE ADD **RADAR/WSR-TOPS** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NET/* "WSR
 Echo Tops (NET)

DSSERVE ADD **RADAR/WSR-VIL** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NVL/* "WSR
 Vertical Integrated Liquid (NVL)

DSSERVE ADD **RADAR/WSR-DREF1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N0X/*
 "Diff Reflec - tilt 1 (0.5deg) 248nmi (N0X)

DSSERVE ADD **RADAR/WSR-DREF2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NAX/*
 "Diff Reflec - tilt 2 (0.9deg) 248nmi (NAX)

DSSERVE ADD **RADAR/WSR-DREF3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N1X/*
 "Diff Reflec - tilt 3 (1.3|1.5deg) 248nmi (N1X)

DSSERVE ADD **RADAR/WSR-DREF4** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NBX/*
 "Diff Reflec - tilt 4 (1.8deg) 248nmi (NBX)

DSSERVE ADD **RADAR/WSR-DREF5** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N2X/*
 "Diff Reflec - tilt 5 (2.4|2.5deg) 248nmi (N2X)

DSSERVE ADD **RADAR/WSR-DREF6** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N3X/*
 "Diff Reflec - tilt 6 (3.1|3.4|3.5deg) 248nmi (N3X)

DSSERVE ADD **RADAR/WSR-COR1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N0C/*
 "Correlation Coef - tilt 1 (0.5deg) (N0C)

DSSERVE ADD **RADAR/WSR-COR2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NAC/*
 "Correlation Coef - tilt 2 (0.9deg) (NAC)

DSSERVE ADD **RADAR/WSR-COR3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N1C/*
 "Correlation Coef - tilt 3 (1.3|1.5deg) (N1C)

DSSERVE ADD **RADAR/WSR-COR4** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NBC/*
 "Correlation Coef - tilt 4 (1.8 deg) (NBC)

DSSERVE ADD **RADAR/WSR-COR5** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N2C/*
 "Correlation Coef - tilt 5 (2.4|2.5deg) (N2C)

DSSERVE ADD **RADAR/WSR-COR6** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N3C/*
 "Correlation Coef - tilt 6 (3.1|3.4|3.5deg) (N3C)

DSSERVE ADD **RADAR/WSR-SDP1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N0K/* "Spec
 Diff Phase - tilt 1 (0.5deg) (N0K)

DSSERVE ADD **RADAR/WSR-SDP2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NAK/* "Spec
 Diff Phase - tilt 2 (0.9deg) (NAK)

DSSERVE ADD **RADAR/WSR-SDP3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N1K/* "Spec
 Diff Phase - tilt 3 (1.3|1.5deg) (N1K)

DSSERVE ADD **RADAR/WSR-SDP4** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NBK/* "Spec
 Diff Phase - tilt 4 (1.8 deg) (NBK)

DSSERVE ADD **RADAR/WSR-SDP5** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N2K/* "Spec
 Diff Phase - tilt 5 (2.4|2.5deg) (N2K)

DSSERVE ADD **RADAR/WSR-SDP6** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N3K/* "Spec
 Diff Phase - tilt 6 (3.1|3.4|3.5deg) (N3K)

DSSERVE ADD **RADAR/WSR-HMC1** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N0H/*
 "Hydrometeor Class - tilt 1 (0.5deg) (N0H)

DSSERVE ADD **RADAR/WSR-HMC2** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NAH/*
 "Hydrometeor Class - tilt 2 (0.9deg) (NAH)

DSSERVE ADD **RADAR/WSR-HMC3** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N1H/*
 "Hydrometeor Class - tilt 3 (1.3|1.5deg) (N1H)

DSSERVE ADD **RADAR/WSR-HMC4** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/NBH/*
 "Hydrometeor Class - tilt 4 (1.8 deg) (NBH)

DSSERVE ADD **RADAR/WSR-HMC5** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N2H/*
 "Hydrometeor Class - tilt 5 (2.4|2.5deg) (N2H)

DSSERVE ADD **RADAR/WSR-HMC6** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/N3H/*
 "Hydrometeor Class - tilt 6 (3.1|3.4|3.5deg) (N3H)

DSSERVE ADD **RADAR/WSR-HHC** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/HHC/* "Hybrid
 Scan Hydrometeor Classification (HHC)

DSSERVE ADD **RADAR/WSR-OHA** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/OHA/* "One
 Hour Accumulation (OHA)

DSSERVE ADD **RADAR/WSR-PTA** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/PTA/* "Storm
 Total Accumulation (PTA)

DSSERVE ADD **RADAR/WSR-DAA** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DAA/* "Digital
 Accumulation Array (DAA)

DSSERVE ADD **RADAR/WSR-DTA** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DTA/* "Digital
 Storm Total Accumulation (DTA)

DSSERVE ADD **RADAR/WSR-DOD** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DOD/* "Digital
 One Hour Difference (DOD)

DSSERVE ADD **RADAR/WSR-DSD** NEXR TYPE=IMAGE RT=Y DIR=#MCDATA/xcd/wsr/ID/DSD/* "Digital
 Storm Total Difference (DSD)

Notes:

- The 13 datasets created by the **XCDTDWR.BAT** file are the same as the RADAR/TDWR-* datasets in the **XCDRADAR.BAT** listing 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** listing above, except that the **XCDWSR.BAT** datasets are named WSR/* rather than RADAR/WSR-*.

The supplied version of the **XCDNOAAPORTGRID.BAT** batch file (which can be run separately, or as part of **XCDADDE.BAT**) is hundreds of lines longer than the batch files above. For that reason the entire listing is not included here. Instead, the subset of entries with "ALL" in their name are listed below because they give a good idea of the numerical models included in the RTGRIDS datasets. As before, each line begins with DSSERVE ADD, and the dataset name it creates is in bold.

```
DSSERVE ADD RTGRIDS/ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_(SRC).sqlite
GRIBTYPE=1" "All GRIB1 grids
DSSERVE ADD RTGRIDS/ALL-GRIB1 GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_(SRC).sqlite
GRIBTYPE=1" "All GRIB1 grids
DSSERVE ADD RTGRIDS/AWC-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_AWC.sqlite GRIBTYPE=1
GRIBMDL=190" "AWC - All National Convective Weather Diagnostic grids
DSSERVE ADD RTGRIDS/AWR-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_AWRW.sqlite GRIBTYPE=1
GRIBMDL=122" "AWR - All Alaska Waters Regional Wave Model grids
DSSERVE ADD RTGRIDS/ECM-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_ECMF.sqlite GRIBTYPE=1
SOURCE=ECMF" "ECMWF - All European Cntr for Med range Weather Fcst
DSSERVE ADD RTGRIDS/ENP-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_ENPW.sqlite GRIBTYPE=1
GRIBMDL=124" "ENP - All Eastern North Pacific Regional Wave Model grids
DSSERVE ADD RTGRIDS/FFG-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_NWFF.sqlite GRIBTYPE=1
GRIBGEO=255 GRIBMDL=151" "FFG - All NWS Flash Flood Guidance System grids
DSSERVE ADD RTGRIDS/GFS-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_GFS.sqlite GRIBTYPE=1
GRIBMDL=80 81 82 94 96" "GFS - All Global Forecast System GRIB1 grids
DSSERVE ADD RTGRIDS/GFS-ALLGRIB1 GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_GFS.sqlite GRIBTYPE=1
GRIBMDL=80 81 82 94 96" "GFS - All Global Forecast System GRIB1 grids
DSSERVE ADD RTGRIDS/ICA-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_ICA.sqlite GRIBTYPE=1
GRIBMDL=120" "ICA - All Ice Concentration Analysis grids
DSSERVE ADD RTGRIDS/ICA-ALLGRIB1 GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_ICA.sqlite GRIBTYPE=1
GRIBMDL=120" "ICA - All Ice Concentration Analysis grids
DSSERVE ADD RTGRIDS/MDR-ALL GRID 69001 70000 RT=Y "MDR - Manually Digitized Radar grids
```

DSSERVE ADD **RTGRIDS/MOS-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_MOS.sqlite GRIBTYPE=1
GRIBGEO=236 GRIBMDL=0"" "MOS - Model Output Statistics from the Model Development Laboratory

DSSERVE ADD **RTGRIDS/NAH-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_NAHW.sqlite GRIBTYPE=1
GRIBMDL=123"" "NAH - All North Atlantic Hurricane Wave Model grids

DSSERVE ADD **RTGRIDS/NAM-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_NAM.sqlite GRIBTYPE=1
GRIBGEO=207 211 212 215 216 217 236 242 GRIBMDL=84 89"" "NAM - All North American Mesoscale
numerical model GRIB1 grids

DSSERVE ADD **RTGRIDS/NAM-ALLGRIB1** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_NAM.sqlite GRIBTYPE=1
GRIBGEO=207 211 212 215 216 217 236 242 GRIBMDL=84 89"" "NAM - All North American Mesoscale
numerical model GRIB1 grids

DSSERVE ADD **RTGRIDS/NCE-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_NCEP.sqlite GRIBTYPE=1
GRIBGEO=218 GRIBMDL=180"" "NCEP - All NCEP quantitative precip forecast grids

DSSERVE ADD **RTGRIDS/NPH-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_NPHW.sqlite GRIBTYPE=1
GRIBMDL=125"" "NPH - All North Pacific Hurricane Wave Model grids

DSSERVE ADD **RTGRIDS/RAP-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_RAP.sqlite GRIBTYPE=1
GRIBMDL=86 105"" "RAP - All Rapid Refresh model GRIB1 grids

DSSERVE ADD **RTGRIDS/RAP-ALLGRIB1** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_RAP.sqlite GRIBTYPE=1
GRIBMDL=86 105"" "RAP - All Rapid Refresh model GRIB1 grids

DSSERVE ADD **RTGRIDS/UKM-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_UKMT.sqlite GRIBTYPE=1
GRIBGEO=37 38 39 40 41 42 43 44 255 GRIBMDL=15 19 45 89"" "UKMT - All Coastal Ocean Circulation grids
global/Merc

DSSERVE ADD **RTGRIDS/WHG-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_TEST.sqlite GRIBTYPE=1
GRIBGEO=255 GRIBMDL=99"" "WHG - All sig. wind wave height grids Great Lakes/LC

DSSERVE ADD **RTGRIDS/WNA-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_WNAW.sqlite GRIBTYPE=1
GRIBMDL=121"" "WNA - All Western North Atlantic Reg. Wave Model grids

DSSERVE ADD **RTGRIDS/WWF-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib1_WWFM.sqlite
GRIBTYPE=1 GRIBMDL=10 SOURCE=WWFM"" "WWF - All Global Wind-Wave Forecast grids

DSSERVE ADD **RTGRIDS/ALL-GRIB2** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_(SRC)*.sqlite
GRIBTYPE=2"" "All GRIB2 grids

DSSERVE ADD **RTGRIDS/GFS-ALLGRIB2** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_GFS*.sqlite GRIBTYPE=2""
"GFS - All Global Forecast System GRIB2 grids

DSSERVE ADD **RTGRIDS/ICA-ALLGRIB2** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_ICA-GLME.sqlite
GRIBTYPE=2 GRIBMDL=120"" "ICA - All Ice Concentration Analysis grids

DSSERVE ADD **RTGRIDS/ICI-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_ICI-USLC13KM.sqlite
GRIBTYPE=2"" "ICI - All Current Icing grids

DSSERVE ADD **RTGRIDS/ICN-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_ICN-USLC13KM.sqlite
GRIBTYPE=2"" "ICN - All Current Icing Potential grids

DSSERVE ADD **RTGRIDS/LMP-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_LMP*.sqlite GRIBTYPE=2""
"LAMP - All Local AWIPS MOS Program grids

DSSERVE ADD **RTGRIDS/MGWM-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_MGWM*.sqlite
GRIBTYPE=2"" "MGWM - All Global Multi-Grid Wave Model files

DSSERVE ADD **RTGRIDS/NAM-ALLGRIB2** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_NAM*.sqlite GRIBTYPE=2""
"NAM - All North American Mesoscale numerical model GRIB2 grids

DSSERVE ADD **RTGRIDS/NBM-ALLGRIB2** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_NBM*.sqlite GRIBTYPE=2""
"NBM - All North American Blended numerical model GRIB2 grids

DSSERVE ADD **RTGRIDS/NDF-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_NDF*.sqlite GRIBTYPE=2""
"NDFD - All National Digital Forecast Database grids

DSSERVE ADD **RTGRIDS/NMMB-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_NMMB*.sqlite
GRIBTYPE=2"" "NMMB - All Non-hydrostatic Mesoscale Model grids

DSSERVE ADD **RTGRIDS/PSS-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_PSS*.sqlite GRIBTYPE=2""
"PSS - All Probabilistic Storm Surge grids

DSSERVE ADD **RTGRIDS/RAP-ALLGRIB2** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_RAP*.sqlite GRIBTYPE=2""
"RAP - All Rapid Refresh GRIB2 grids

DSSERVE ADD **RTGRIDS/RTM-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_RTM*.sqlite GRIBTYPE=2""
"RTMA - All Real-Time Mesoscale Analyses grids

DSSERVE ADD **RTGRIDS/RTOF-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_RTOF*.sqlite
GRIBTYPE=2"" "RTOF - All Real-time Ocean Forecast system grids

DSSERVE ADD **RTGRIDS/SREF-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_SREF*.sqlite
GRIBTYPE=2"" "SREF - All Short Range Ensemble Forecast model grids

DSSERVE ADD **RTGRIDS/SST-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_SST-GLME.sqlite
GRIBTYPE=2"" "SST - All Sea Surface Temperature model grids

DSSERVE ADD **RTGRIDS/URMA-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_URMA*.sqlite
GRIBTYPE=2"" "URMA - All Unrestricted Mesoscale Analysis model grids

DSSERVE ADD **RTGRIDS/WRFN-ALL** GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_WRFN*.sqlite
GRIBTYPE=2"" "WRF - All Weather Research and Forecasting model grids

```
DSSERVE ADD RTGRIDS/WRFE-ALL GRIBDB 1 999999 TYPE=GRID RT=Y
INFO=""DATABASE=#MCDATA/xcd/gribdatabase/(YYYY)(DDD)/(YYYY)(DDD)_grib2_WRFE*.sqlite
GRIBTYPE=2" "WRF Eulerian - All Weather Research and Forecasting model grids
```

[The previous Table of Contents and full content of the “old” format of this document is below.](#)

Notes/tips:

1. To move content from the old section below into the new section above:
 - a. mark the section using the cursor
 - b. use Ctrl+X to cut the section
 - c. find the correct location in the new format above and then use Ctrl+V to paste it in with the existing formatting or Ctrl+Shift+V to paste it without the formatting
 - d. if necessary, update content and do any cleanup needed to make its format work in the new section

2. If you need to add a second-level bullet (e.g., “Account Names and Uses” and “Configuring the Accounts” in Chapter 2 above), put the cursor at the end of the previous bullet then hit Enter, and then Tab.

Table of Contents

[Table of Contents](#)

[Prerequisites](#)

[Required](#)

[Optional](#)

[Obtaining docker-compose Files](#)

[Configuring the Data Directory](#)

[Setting the Data Directory](#)

[Required Permissions](#)

[Configuration Files](#)

[docker-compose.yml](#)

[xcd.env](#)

[ldmd.conf and pqact.conf](#)

[Loading Docker Images](#)

[Method 1: Using docker pull](#)

[Method 2: Using docker load](#)

[Running Containers with docker-compose](#)

[Startup Process of Containers](#)

[Listing Running Containers](#)

[Running Commands in Containers](#)

[Running an Interactive Bash Shell in a Container](#)

[Installing Custom Idmd.conf and pqact.conf Files in LDM Container](#)

[Removing Containers with docker-compose](#)

[Stopping Containers with docker-compose](#)

[Removing Images with docker](#)

[Removing Containers with docker](#)

[Procedure for Updating Running Containers](#)

[More Information](#)

[Docker User Namespace at SSEC](#)

[Unidata LDM Container Usage](#)

[Singularity](#)

[Installation](#)

[Notes](#)

[Podman](#)

Prerequisites

Required (Ask TC for these to be installed)

- CentOS 7 or 8
- Docker version 19.03.4
- Docker Compose version 1.24
- Git (for obtaining `xcddecoder` source and `docker-compose` files)

Recommended

- Network access to Docker Hub (for pulling Apache web server image and Unidata LDM image)
 - Alternatively, pull these images from Docker Hub on a different machine and save as a `.tar` to be copied

Obtaining `docker-compose` Files

GitLab

Clone and download the `mcidasxcdcontainers` Git project containing the `docker-compose` files. The Git project also includes files for building the XCD image.

You must have access to the <https://gitlab.ssec.wisc.edu/clpenne/mcidasxcdcontainer> Git repository.

CD to your home directory. Run `git clone` and enter your SSEC username and password:

```
git clone https://gitlab.ssec.wisc.edu/clpenne/mcidasxcdcontainer.git
```

Optional: Remove all unused Docker images from system

```
docker system prune -a
```

Website

TBD

Docker User Namespace

On SSEC machines, TC installs Docker with a user namespace enabled. The user namespace prevents processes within the container from running as 'root' by applying an offset of 100000 to every user within the container. For example, if a process is running as 'root' or UID 0 inside a container, outside it will appear as UID 100000.

Since by default container processes are run as 'root' or UID 0, TC sets up a user named 'dockmap' which is assigned UID 100000.

Setup 'dockmap' user and group (assign UID and GID 100000)
Setup 'dockmap_oper' group (assign GID 101000)
Add the user that is used to run Docker with to the 'dockmap_oper' group

Loading Docker Images

Method 1: Using `docker pull`

Log into the SSEC GitLab container registry. The server is `gitlab.ssec.wisc.edu:5555`.

```
[oper@csppgamma Docker_Building_Tools]$ docker login
gitlab.ssec.wisc.edu:5555
Username: robo
Password:
WARNING! Your password will be stored unencrypted in
/home/oper/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-s
tore

Login Succeeded
[oper@csppgamma Docker_Building_Tools]$
```

Now pull the image using `docker pull`:

```
docker pull
gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/xcddecoders:latest
```

Next, tag the image:

```
docker tag gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/xcddecoders:latest
xcddecoders:latest
```

There is also a short script in the `Docker_Building_Tools` directory (located in `/home/oper/mcidasxcdcontainer/Docker_Building_Tools`) that contains the pull and tag commands:

```
cd /home/oper/mcidasxcdcontainer/Docker_Building_Tools
./gitlab_registry_pull_stable
```

List installed Docker images:

```
[oper@csppgamma Docker_Building_Tools]$ docker image ls
```

REPOSITORY	SIZE	TAG	IMAGE ID
xcddecoders		latest	5a002becbc4e
4 hours ago	800MB		
gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/xcddecoders		latest	5a002becbc4e
4 hours ago	800MB		

Method 2: Using `docker load`

Download or copy `xcddecoders.tar` from wherever it is hosted or distributed.

CD to the directory containing the image `.tar` file and load the image with the `docker load` command:

```
docker load -i ./xcddecoders.tar
```

List installed Docker images:

```
[oper@csppgamma ~]$ docker image ls
```

REPOSITORY	SIZE	TAG	IMAGE ID	CREATED
xcddecoders		latest	f19efa431017	About an
hour ago	800MB			

Running Containers with `docker-compose`

Check the version of `docker-compose`. You should be running `docker-compose` version 1.17 or greater (1.24 is recommended).

```
[oper@csppgamma ~]$ docker-compose --version
docker-compose version 1.24.1, build 4667896b
[oper@csppgamma ~]$
```

SSEC Technical Computing installs `docker-compose` version 1.9 which has compatibility issues with the `docker-compose.yml` format used by this project (<https://docs.docker.com/release-notes/docker-compose/#1170>).

Start `docker-compose` with the `docker-compose up` command. Docker Compose will pull two additional Docker images that are specified in `docker-compose.yml`. These images are `unidata/lm-docker:latest` and `httpd:2.4` and are built and maintained separately from the `xcddecoders` image.

```
docker-compose up
```

Important: If you are running `docker-compose` from outside the directory containing the `docker-compose.yml` file, then the path to `docker-compose.yml` must be included using the `-f` argument.

```
docker-compose -f /home/oper/mcidasxcdcontainer/Tools/docker-compose.yml
up
```

Optional: Run `docker-compose` in detached mode using `--detach` or `-d`. This will put `docker-compose` into the background.

```
docker-compose --detach up
```

Listing Running Containers

Get a list of running containers and their names with the following commands:

```
docker container ls
```

```
[oper@cspgamma Docker_Compose]$ docker-compose ps
      Name                                Command                                State      Ports
-----
-----
xcdgrib      /usr/local/bin/entrypoint    ...      Up
xcdldm      /entrypoint.sh runldm.sh      Up
0.0.0.0:388->388/tcp
xcdmonitor  /usr/local/bin/entrypoint    ...      Up
xcdmonitorweb  httpd-foreground              Up      0.0.0.0:80->80/tcp
xcdscour    /usr/local/bin/entrypoint    ...      Up
xcdtext     /usr/local/bin/entrypoint    ...      Up
```

Listing Docker Volumes

Get a list of Docker volumes and their names with the following commands:

```
docker volume ls
```

```
[oper@remus ~]$ docker volume ls
DRIVER          VOLUME NAME
local          tools_xcdmonitorvolume
```

Running Commands in Containers

To run commands in containers, use the `docker exec` command:

```
docker exec xcdldm ldmadmin config
```

Running an Interactive Bash Shell in a Container

To run a Bash shell in a container, use the `docker exec` command with the `-i` and `-t` arguments:

```
docker exec -it xcdgrib bash
```

Startup Process of XCD Containers

- Dockerfile (not run on startup)
- Entrypoint (run as root)
- Command (run as oper user using gosu)
 - Command defined in Dockerfile, Docker command argument, or in docker-compose.yml
 - Xcdadmin
 - XCD decoder process or daemon
 - XCD monitor daemon
 - Crond

Description and Roles of Containers and Volumes

Containers

xcdbufr

Runs BUFR decoder. Created from `xcddecoders` image. **Work in progress.**

xcdgrib

Runs GRIB decoder. Created from `xcddecoders` image. Container startup command is `xcdadmin start grib`.

xcdldm

Runs LDM inside a Docker container. Created from `unidata/l dm-docker` image (<https://hub.docker.com/r/unidata/l dm-docker/>) and is pulled from Docker Hub in `docker-compose.yml`. The source image is developed by Unidata (<https://github.com/Unidata/l dm-docker>) and is modified during startup by copying or mounting custom LDM configuration files into the LDM container.

xcdmonitor

Run the XCD monitor backend. Writes the XCD monitor page to the `xcdmonitorvolume` Docker volume mount. Does not host the monitor webpage (`xcdwebmonitor` takes care of that). Container startup command is `xcdadmin start monitor`.

xcdscour

Runs XCD scour scripts as cron jobs. Created from `xcddecoders` image. The crontab file within the `xcddecoders` container is setup to run the scour scripts. Container startup command is `crond`.

xcdtext

Runs text decoder. Created from `xcddecoders` image. Container startup command is `xcdadmin start text`.

xcdweb

Runs web server for hosting XCD monitor page. Reads the XCD monitor page from the `xcdmonitorvolume` Docker volume mount. Created from official Docker `httpd` image (https://hub.docker.com/_/httpd/) and is pulled from Docker Hub in `docker-compose.yml`.

Volumes

xcdmonitorvolume

Contains the XCD monitor webpage that is served by the `xcdwebmonitor` container. The monitor statistics are updated by the `xcdmonitor` container writing to the volume. The volume is configured in `docker-compose.yml`.

Removing Containers with `docker-compose`

To take down all the containers running under `docker-compose`, use `docker-compose down`. The command will first stop the containers and then remove the stopped containers from the Docker system:

```
docker-compose down
```

To take down just one container, add the name of the container or service listed in `docker-compose.yml`:

```
docker-compose down xcdmonitor
```

Stopping Containers with `docker-compose`

To stop all the containers running under `docker-compose`, use `docker-compose stop`. The command will stop the containers and leave them in a state where they can be restarted using `docker-compose up`:

```
docker-compose down
```

Removing Images with docker

List the images:

```
[oper@csppgamma Docker_Compose]$ docker image ls
```

REPOSITORY	CREATED	SIZE	TAG	IMAGE ID
xcddecoders	19 hours ago	708MB	latest	add1ea062d00
xcdldm	19 hours ago	638MB	latest	53d6bec46768
httpd	13 days ago	165MB	2.4	d3017f59d5e2

Use the `docker rmi` command to remove images by the image ID or image name.

Removing container by image ID:

```
docker rmi add1ea062d00
```

Removing container by image name:

```
docker rmi xcddecoders
```

Removing Containers with docker

List the container IDs of running containers:

```
docker container ls
```

List the container IDs of exited or stopped containers:

```
[oper@csppgamma Docker_Compose]$ docker ps -aq -f status=exited
```

a2120a2b464b
9f392421efc1
daacb3a83f2b

Remove the container using its ID with the `docker rm` command. If the container is still running, it must be stopped first.

```
docker rm 9f392421efc1
```

Procedure for Updating Running Containers

1. Shut down container with `docker-compose down [service-name]`.
 - a. Service names are listed in `docker-compose.yml` under the `services` section (ex. `xcdgrib`)
2. Pull or load the new image into Docker system using `docker pull` or `docker load`
3. Start up the new container with `docker-compose up [service-name]`

Building `xcddecoders` Image

Using Docker

Files in `./Tools` directory (draft)

Using Podman

Building `xcdmcidas` Image

Using Docker

Files in `./Tools` directory

Using Podman

Managing Images on GitLab Repository

Pushing Images

Managing Images as .tar Files

More Information

Docker User Namespace at SSEC

[TC Docker Info](#)

Unidata LDM Container Usage

The `xcdldm` image is built using the Unidata LDM Docker image as the base image. More information is available at <https://github.com/Unidata/ldm-docker>.

Docker Logs

Checking Docker logs using the ``docker logs`` command
Useful for troubleshooting containers that have crashed/exited

Singularity

Installation

NOTE: make sure to use latest stable versions for Go and Singularity installation. Commands in Quick Start are slightly out of date.

https://sylabs.io/guides/3.5/user-guide/quick_start.html#quick-installation-steps

Notes

Singularity is intended to be an alternative to Docker yet still compatible with existing Docker images (some issues with this, read on).

Images can be build from Singularity definition files. Definition files are analogous to Dockerfiles but have a different format meaning they are not directly compatible. Singularity definition files combine both the Dockerfile and Entrypoint script into a single file.

Building/pulling images with Singularity saves them as `.sif` files. These files are roughly equivalent to a Docker `.tar` file. Does not need to be unpacked to be run. Docker images can be pulled from registries and saved to `.sif` files allowing them to be run using Singularity.

Singularity does not have access to the Docker images loaded into the Docker system (in other words the images listed using ``docker image ls``) meaning they must be obtained from a Docker registry with a pull command.

When Singularity images are run, the current working directory and the current user's \$HOME directory are mounted in the container by default (TODO: how to disable this if unnecessary?)

Singularity containers run as the current user on the host. The commands `useradd` and `groupadd` do not work in the container entry script since the container can't change users on the host

- Issue with Unidata LDM Docker container since it creates a LDM user in the entrypoint script
- Issue with the xcddecoders container since it creates users and groups
- Existing containers must be adapted to be run in Singularity
 - Version for Docker (use Dockerfiles)
 - Version for Singularity (use definition files)
- Alternative solution (maybe?): create user/groups manually on the host

Docker Live Restore

In normal Docker configurations, if the Docker daemon is stopped or restarted then any running containers are shut down. This can occur unexpectedly such as when Docker is updated. Docker live restore keeps containers alive and running if the Docker daemon is stopped. **Important:** Docker service and containers must be restarted after enabling live restore.

<https://docs.docker.com/config/containers/live-restore/>

Notes

What is needed:

- Files for building and running the XCD and McIDAS containers. These are hosted on the SSEC GitLab page at <https://gitlab.ssec.wisc.edu/clpenne/mcidasxcdcontainer/>. The git project is viewable by anyone with a login on the SSEC GitLab site. Email penne@wisc.edu for edit access.
- CentOS 7.6 or greater. Docker cannot run on CentOS 6. Containers have not been tested on MacOS or Windows.
- Docker CE. Ask TC to install Docker as they will setup Docker to be correctly configured. The Docker install should have user namespaces and live restore enabled. Your user on the machine also needs to be added to the Docker group to run Docker without sudo.
- Unidata's LDM Docker image. The LDM container is distributed separately from the XCD and McIDAS containers and must be pulled from Unidata's Docker registry or built using the files at <https://github.com/Unidata/ldm-docker> (more on this later).
- HTTPD (web server) Docker image.

Terms:

- Dockerfile: <https://docs.docker.com/engine/reference/builder/>
- Docker Image: Read-only template used to create a Docker container
- Docker Container
- Docker vs Docker-Compose

Building Images:

Run these commands from the `mcidasxcdcontainer/Tools/` directory. The 'docker build' commands build the images from the Dockerfiles located in `../xcdmcidas` and `../xcddecoders`. The '-t' argument assigns a name and version tag to each image (latest is the default tag).

- `docker build -t xcddecoders:latest ../xcddecoders/`
- `docker build -t xcdmcidas:latest ../xcdmcidas/ (OPTIONAL)`

Listing Images:

- `docker image ls`

Removing images:

- `docker rmi <image-name-or-id>`

Using GitLab Container Registry:

The SSEC GitLab site has private Docker container registries enabled. The registries can only be accessed after logging in with SSEC GitLab credentials. Follow the instructions for setting up and logging in to the GitLab registry:

https://gitlab.ssec.wisc.edu/help/user/packages/container_registry/index

The Docker login command should look like this:

```
docker login gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/
```

Enter your SSEC GitLab username and password when prompted.

Pushing Images to Container Registry:

Tag the images with the GitLab registry address and version tag:

```
docker tag xcdmcidas:latest gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/xcdmcidas:latest
```

```
docker tag xcddecoders:latest
```

```
gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/xcddecoders:latest
```

Push the images to the private GitLab registry:

```
docker push gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/xcdmcidas:latest
```

```
docker push gitlab.ssec.wisc.edu:5555/clpenne/mcidasxcdcontainer/xcddecoders:latest
```

Check that the images were successfully pushed to the registry at https://gitlab.ssec.wisc.edu/clpenne/mcidasxcdcontainer/container_registry. The list of images show when they were last updated.

Pulling Images

Obtaining LDM container

Running containers (follow procedure above)

Checking status of Docker service and uptime

Docker containers cannot start or run if the Docker service/daemon isn't running

```
sudo service docker status
```

Compose Spec

<https://github.com/compose-spec>

Will this work with Podman?

New-XCD Podman Setup Instructions

Note: Podman emulates the Docker CLI with `alias docker=podman` which allows Docker and Podman commands to be used interchangeably.

Log in to `csppgamma` as user `oper`:

```
sudo -u oper -i
```

Check for any New-XCD containers currently running with the command:

```
podman container ls -a
```

If there are New-XCD containers running, shut them down and remove them:

```
podman pod stop xcd
podman rm xcdgrib xcdldm xcdscour xcdmonitor xcdwebserver
podman pod rm xcd
podman volume rm xcdmonitorvolume
```

Clone the GitLab repo:

```
git clone https://gitlab.ssec.wisc.edu/clpenne/mcidasxcdcontainer.git
cd mcidasxcdcontainer/Tools
```

If the project already exists, run a Git pull:

```
git pull
```

Set the `HOST_DATA_DIR` environmental variable and add it to `~/.profile`

```
export HOST_DATA_DIR=/data/xcd
```

In the `mcidasxcdcontainer/Tools` directory, run the commands in `gitlab_registry_pull_podman` to pull the New-XCD image.

Next, run `run_xcd_podman` to create the New-XCD Podman pod and to start the containers. The `run_xcd_podman` script contains the Podman commands for creating an XCD Pod and running the New-XCD containers.

To verify that the containers have started successfully, run:

```
podman container ls -a
```

Config Files

This method requires stopping the LDM process in a running LDM container and copying the files into the container before starting LDM again.

First, stop LDM in the `xcdldm` container:

```
podman exec xcdldm ldadmin stop
```

Next, use the `podman cp` command to copy files from the host filesystem to the container filesystem:

```
podman cp ./ldmd.conf xcdldm:/home/ldm/etc/ldmd.conf
podman cp ./pqact.conf xcdldm:/home/ldm/etc/pqact.conf
podman cp ./registry.xml xcdldm:/home/ldm/etc/registry.xml
```

It is also possible to copy a directory containing the configuration files:

```
podman cp ./etc/. xcdldm:/home/ldm/etc/.
```

Finally, restart LDM in the container:

```
podman exec xcdldm ldmadmin start
```

Running Podman with Docker-Compose

<https://www.redhat.com/sysadmin/podman-docker-compose>

Installing McIDAS-XCD Servers