

Getting Started with WSR-88D CODE

This document is an interim guide to getting started with the Common Operations Development Environment (CODE) for the WSR-88D. It covers basic topics at an introductory level and includes references to the CODE documentation. This guide does not include an introduction to algorithm development topics which will be included later in a CODE algorithm development tutorial.

Section 1 - Introduction to CODE

Section 2 - CODE Installation

Section 3 - Ingesting Archive 2 Data and Creating Products

Section 4 - Displaying WSR-88D Products

1. Introduction to CODE

1.1 Overview

CODE is primarily an algorithm development platform for the WSR-88D weather radar. However, CODE can also be used to ingest historical WSR-88D level 2 data and create valid WSR-88D products for various research and analysis purposes.

A CODE installation results in a stand-alone version of the WSR-88D Open RPG (ORPG). The term 'clone' is used because the RPG is identical to the operational system except that it is not connected to (nor does it control) a WSR-88D radar. The CODE ORPG is not connected to external systems and products are not distributed to users.

If the guidance contained in the CODE documentation is followed correctly, algorithms developed require very little effort to integrate into the operational weather radar. The CODE-developed algorithms are actually running on a 'clone' of the radar product generation (RPG) platform using the native application programming interface (API).

WSR-88D level 2 radar data can be ingested into a running CODE ORPG and the products produced are stored into a database. There are utilities that can be used to analyze the product contents, provide a graphic display of products, and export product data into image files and binary data files.

1.2 The CODE distribution archive

1.2.1 The CODE distribution web site

The CODE distribution is available for download at: <http://www.weather.gov/code88d/>

The distribution archive is obtained by following the link: [CODE Distribution](#) on the home page navigation bar. An extended set of archived WSR-88D level II data is provided with the link [Archive 2 Data](#).

1.2.2 Contents of the CODE Distribution

1.2.2.1 Software archives

The CODE distribution contains the source code for

- a. The WSR-88D ORPG
- b. CODE utility updates (if any)
- c. A set of sample algorithms

1.2.2.2 Documentation

The CODE guide is provided in 4 volumes:

Volume 1 - Guide to Setting Up the Development Environment

Installation instructions for the ORPG, CODE utility updates, and sample algorithms are provided. Procedures for starting / stopping the ORPG and initiating archive 2 data ingest are included.

Volume 2 - ORPG Application Software Development Guide

This part of the guide provides an overview of the ORPG software organization, instructions for integrating algorithm development software into the source tree, instructions for configuring the ORPG in order to add data stores and tasks for new algorithms, and compiling algorithm code. A description of major RPG data items is provided including the content and structure of the base data messages read by the algorithms, the definition and use of algorithm adaptation data, and the structure of final products. This guidance is intended to be independent of the algorithm API.

Volume 3 - WSR-88D Algorithm Programming Guide

This volume provides the reference for the algorithm API, the interface between the WSR-88D algorithms and the ORPG system services. Guidance for use of the API and the structure of well behaved algorithms is covered. The description and purpose of the sample algorithms provided with CODE is included.

Volume 4 - CODE Utility Guide

Contains documentation for major utilities: CODEview Text (**CVT**), CODEview Graphics (**CVG**), **play_a2**, and **change_radar**.

1.2.2.3 Major ORPG / CODE Utilities

The most useful utilities for algorithm development are:

- a. CODEview Text utility **cvT** -
Provides a text screen output of selected contents of WSR-88D products.
- b. CODEview Graphics utility **cvG** -
Provides a graphical display of WSR-88D products.
- c. Archive 2 data ingest utility **play_a2** -
Ingests WSR-88D archive 2 data into a running ORPG.
- d. Site Adaptation data utility **change_radar** -
Sets the site adaptation data corresponding to the source of the archive 2 data being fed into the ORPG.

2. CODE Installation

2.1 System Requirements

WSR-88D CODE is installed on an Intel Pentium PC running Red Hat Enterprise Workstation - specifically Enterprise 7. Alternatively, CentOS 7 has been tested and supported. Please note, the ORPG software will operate on a 64-bit or a 32-bit kernel. Detailed system requirements are listed on the CODE web site and in Volume 1 of the CODE Guide.

2.2 Installation Procedures

2.2.1 Workstation Preparation

Volume 1 of the CODE guide provides instructions for the preparation of the Linux platform prior to the installation of CODE. These procedures require system administrative privileges. Major preparation actions are the installation of the Linux software packages, the TCP/IP configuration, and the creation of user accounts for the installation of the CODE ORPG. See *Preparation for Installation, the Linux PC Platform in Volume 1 Document 1* of the CODE guide for complete instructions.

2.2.1.1 Linux Software Packages

The major preparation act is the installation of Red Hat Enterprise Workstation Linux. The easiest method is to select all software packages during Linux installation. If this is not desired guidance is provided for the selection of software packages. It should be noted that the OEM Red Hat installations typically do not include all software packages required for CODE.

2.2.1.2 TCP/IP Network

The ORPG software requires that TCP/IP networking be configured (even if not connected to an actual network). The ORPG is not compatible with DHCP, a static IP address must be configured.

2.2.1.3 User Accounts

Each ORPG is installed into a specific Unix user account. Beginning with Build 10, the default location of the ORPG data directory is within each ORPG account home directory.

All accounts must be configured for and use the C shell (**cs**h) for both the compilation of software and the operation of the ORPG and utilities. Other shells including **ba**sh are not currently supported.

2.2.2 ORPG Installation

The installation of the ORPG requires the compilation of the ORPG source code provided with the CODE distribution. System administration privileges are not required for installation if all preparation procedures including the creation of user accounts and data directories have been accomplished. Detailed instructions are provided in the section *Installation Instructions* in **Volume 1 Document 1** of the CODE guide.

NOTE: It is recommended that the complete installation instructions in Volume 1 of the CODE guide be used for first-time installation. The CODE web site also provides a set of 'Concise Installation Instructions' that are useful as an example of instructions that have been tailored to provide uniform multiple installations.

Significant actions are:

- a. Extracting the ORPG source code archive.
- b. Replacing selected ORPG and Unix account configuration files (script installation).
- c. Installing ORPG patches (if provided with CODE).
- d. Compiling the ORPG source code.
- e. Installing customized ORPG configuration files.

All installation actions must be accomplished while logged into the account created for the ORPG installation. This account must use the C shell (**cs**h).

2.2.3 CODE Software Installation

2.2.3.1 CODE Utility Updates

The major CODE utilities **CVT** and **CVG** are integrated into and compiled with the ORPG software. Updates are often supplied with the CODE distribution and should be installed. See *Instructions for CODE Utilities* in **Volume 1 Document 2** of the CODE guide for installation instructions and special configuration options. All installation actions must be accomplished while logged into the account created for the ORPG installation.

2.2.3.2 CODE Sample Algorithms

If using CODE for algorithm development, a set of sample algorithms has been provided to demonstrate the use of algorithm API functions and guidance for the structure of algorithms. The installation of the sample algorithms is optional. See *Instructions for Sample Algorithms* in **Volume 1 Document 2** of the CODE guide. All installation actions must be accomplished while logged into the account created for the ORPG installation.

3. Ingesting Level 2 Data and Creating Products

Once installation is complete the CODE ORPG account can be used for algorithm development or for ingesting radar data and creating radar products for analysis. Algorithm development topics are not covered in this guide.

3.1 Archive Level 2 Data Ingest

Without a direct connection to a radar, the typical source of input data for a CODE ORPG is archived WSR-88D Level 2 data obtained in one of two ways. The compressed volume data files are input into the CODE ORPG using tools supplied with the ORPG distribution.

3.1.1 Sources of Level 2 Data

3.1.1.1 Archive 2 Data Included with the CODE Distribution.

When the ORPG is installed following the procedures in procedures in Volume 1, three volume files of archive 2 data from Melbourne, FL are installed in order to support immediate testing of the ORPG installation. In addition, 15 data sets of compressed archive 2 volume files are available on the CODE distribution CD and on the CODE web site. These data sets range from several volumes to over 4 hours of data. The data contains samples using the various volume scanning strategies and include various forms of precipitation (summer and winter storms, hurricanes, tornadoes, etc.). Data also include cases containing anomalous propagation, switches between radar velocity modes, and spot blanking.

3.1.1.2 Archive 2 Data from NCEI (formerly NCDC)

The National Centers for Environmental Information (NCEI) collects Level II data from over 150 NEXRAD sites and makes the data available at no cost to the user via the NCEI radar resources web page: <https://www.ncdc.noaa.gov/data-access/radar-data>.

3.1.2 Starting the ORPG and data ingest

The section ***Running the ORPG in Volume 1 Document 1*** of the CODE guide contains the instructions for starting and stopping the ORPG, for initiating data ingest using archive 2 data files and troubleshooting start problems. The ORPG must be run and all development activities accomplished while logged into the account in which the ORPG is installed.

3.1.2.1 The ORPG is started with the `mrpg` command:

```
mrpg -v -p startup
```

The `-v` option provides verbose output and the `-p` option selects a startup with a cleanup of existing ORPG data stores.

With CODE, the ORPG should be shutdown using both of the following commands:

```
mrpg shutdown
mrpg cleanup
```

3.1.2.2 Archive 2 data files are ingested into the ORPG with the `play_a2` command

```
play_a2 -d $HOME/data/ar2data
```

The `-d` option designates the location of the archive 2 data files.

Data ingest can be stopped by entering `<Ctrl> C` in the terminal screen used to start `play_a2`.

3.2 LDM Level 2 Data Ingest

If you already have a live feed or source of LDM data, the data can be ingested into a running ORPG clone using the `read_ldm` tool. Setting up an LDM data source is beyond the scope of CODE.

3.3 Generating the Desired Products

Starting the ORPG and ingesting base data does not result in all WSR-88D products being generated.

The WSR-88D system does not actually generate a specific product unless it is requested by a user or designated for default generation. In addition some products will not be produced even when requested if the input data does not result in the detection of certain meteorological phenomena.

The easiest method to request a product using CODE is to configure the desired product for default generation using the `product_generation_tables` configuration file located in the `$HOME/cfg` directory.

Excerpt from the `product_generation_tables` file:

```
Default_prod_gen {
```

#prod_id	wx_modes	gen_pr	stor_reten	p1	p2	p3	p4	p5	p6
2	6	1	180	UNU	UNU	01-45	UNU	UNU	UNU
4	2	1	180	UNU	UNU	11-3	UNU	UNU	UNU
4	4	1	180	UNU	UNU	11-3	UNU	UNU	UNU
7	4	1	180	UNU	UNU	11-1	UNU	UNU	UNU
.
42	2	1	180	UNU	UNU	UNU	UNU	UNU	UNU
44	6	1	180	UNU	UNU	UNU	UNU	UNU	UNU
49	2	1	180	UNU	UNU	UNU	UNU	UNU	UNU
51	2	1	180	UNU	UNU	UNU	UNU	UNU	UNU
.
151	6	1	180	UNU	UNU	UNU	UNU	UNU	UNU
152	7	1	360	12	24	UNU	UNU	UNU	UNU
298	6	1	180	UNU	UNU	10-0	UNU	UNU	UNU
301	6	1	180	UNU	UNU	10-0	UNU	UNU	UNU
}									

The first column is the product id (which is not the same as the external product code for legacy algorithms). Columns **p1** through **p6** represent the product dependent parameters in a request message for the product. The elevation parameter for elevation products is always represented in the **p3** column. The value **UNU** means the parameter is not used. The section ***ORPG Configuration for Application Developers in Volume 2 Document 2*** of the CODE guide contains instructions for this configuration file.

The product id can be obtained from the **product_attr_table** configuration file also in the **\$HOME/cfg** directory. Other attributes include an identification of use of the 6 product dependent parameters for each product to include range of values, default value, encoding, and unit of measure. The section ***ORPG Configuration for Application Developers in Volume 2 Document 2*** of the CODE guide documents this configuration file.

The product dependent parameters for each product are also listed in Table IIA in the RPG to Class 1 User Interface Control Document. A copy of a recent version of this document is provided with CODE (e.g., **2620001j_rpg_class1_draft.pdf**).

For a meteorological description of WSR-88D products see the Product Specification Interface Control Document. A copy of a recent version of this document is provided with CODE (e.g., **2620003j_prod_spec_draft.pdf**) . Some examples:

- a. The basic reflectivity data (surveillance data) is contained in the digital reflectivity or DR product (id 94, product code 94). This product is described in Section 1 of the product specification.
- b. The basic velocity data (Doppler data) is contained in the digital velocity or DV product (id 99, code 99) and is described in Section 2 of the product specification.
- c. The mesocyclone related products: M (id 42, code 60), MD (id 141, code 141), etc. are described in Section 20.

For individuals not familiar with WSR-88D products, the best way to associate products configured in the ORPG and described in the product specification is to use the short name or mnemonic (e.g., DV, DR, M, MD, SRM, etc) which can be found at the beginning of the description attribute **desc** in the **product_attr_table** configuration file.

4. Displaying WSR-88D Products

All final products (products intended for distribution) are stored in a special product database file. The CODE utilities **CVT** and **CVG** can access this database as well as load single product binary files. Both utilities can identify certain inconsistencies in the structure of products. This document provides a very high level overview of their capabilities.

4.1 CODEview Graphics (CVG)

CVG is documented in **Volume 4 Document 2** of the CODE guide. It is an interactive application with a graphical user interface (X-windows / Motif).

4.1.1 Graphical Display of a Product

The primary purpose of CVG is to provide a graphical display of final products. CVG does not emulate any operational display. Its capabilities are aimed at product evaluation and comparison rather than meteorological forecasting. However if configured properly the graphic display provided by CVG can give an indication of the product's utility in a forecast office.

Since there is no standard format for intermediate products, they are not displayed by CVG.

Basic display capabilities include:

- **cvg** provides a list of all final products currently in the ORPG product database and allows the user to select a product for display. **cvg** provides the capability to switch to a different product database and to display products stored as individual binary files.
- Once a product is selected, **cvg** either
 - displays the entire contents of the product
 - displays a list of all available data layers in the symbology block and the presence of a tabular alphanumeric block (TAB) and a graphic alphanumeric block (GAB) for selection by the user.
- **cvg** detects and reports errors in product structure. The types of errors detected are being expanded.
- The capability to display multiple layers / products overlaid on top of one another.
- A $\pm 32X$ Zoom capability.
- Manual and automatic animation of a selected product. This can be either a time sequence animation (displaying a product from the previous or subsequent volume scan) or an elevation sequence animation (displaying elevations within a volume scan in either order).

- **CVG** provides information about the data point of a displayed product with respect to the cursor location. This information includes the coordinates of the point and the data value being displayed.
- **CVG** can display two products simultaneously. This includes a basic linking between the displayed products during animation and with respect to displaying information about the data point at the cursor location.

A complete description of using CVG to display products is contained in **Section I of Volume 4 Document 2**.

4.1.2 Configuring New Product for Display by CVG

CVG can be configured to display new products as long as the products use existing data packets and components in a manner consistent with existing products. A text configuration file must be modified and possibly a new legend display configuration file and new color palette file defined. A complete description of the configuration process is contained in **Section II of Volume 4 Document 2**.

4.2 CODEview Text (CVT)

CVT is documented in **Volume 4 Document 1** of the CODE guide. It is a command-line tool and displays text output in the terminal window. A shell script can use combinations of CVT commands to automate analysis activities.

4.2.1 Displaying an ASCII output of selected portions of a product

CVT provides the capability of inspecting selected portions of a final product.

- The user can display the header portion of a product, data packets within a layer of the symbology block, and the contents of other optional blocks.
- Product data packets containing two-dimensional data arrays can be output in whole or in part.
- CVT can provide a basic inventory of the contents of the product database.

4.2.2 Exporting a product to a binary file

CVT can export a selected product message in the product database to an individual binary file or accomplish an ASCII output of the contents in Hexadecimal format.