

**Installation Instructions for the International MODIS/AIRS Processing Package (IMAPP) / Community Satellite Processing Package (CSPP)
Ozone Intrusion Air Quality Forecasting Application for Direct Broadcast
IDEA-I: Infusing satellite Data into Environmental Applications – International**

Version 1.0

University of Wisconsin-Madison, Space Science and Engineering Center (SSEC)
Supported by the NASA Science of Aqua and Terra Project and the NASA/NOAA Joint Polar Satellite System (JPSS) Project

Beta Version – 24 February 2014

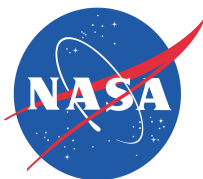
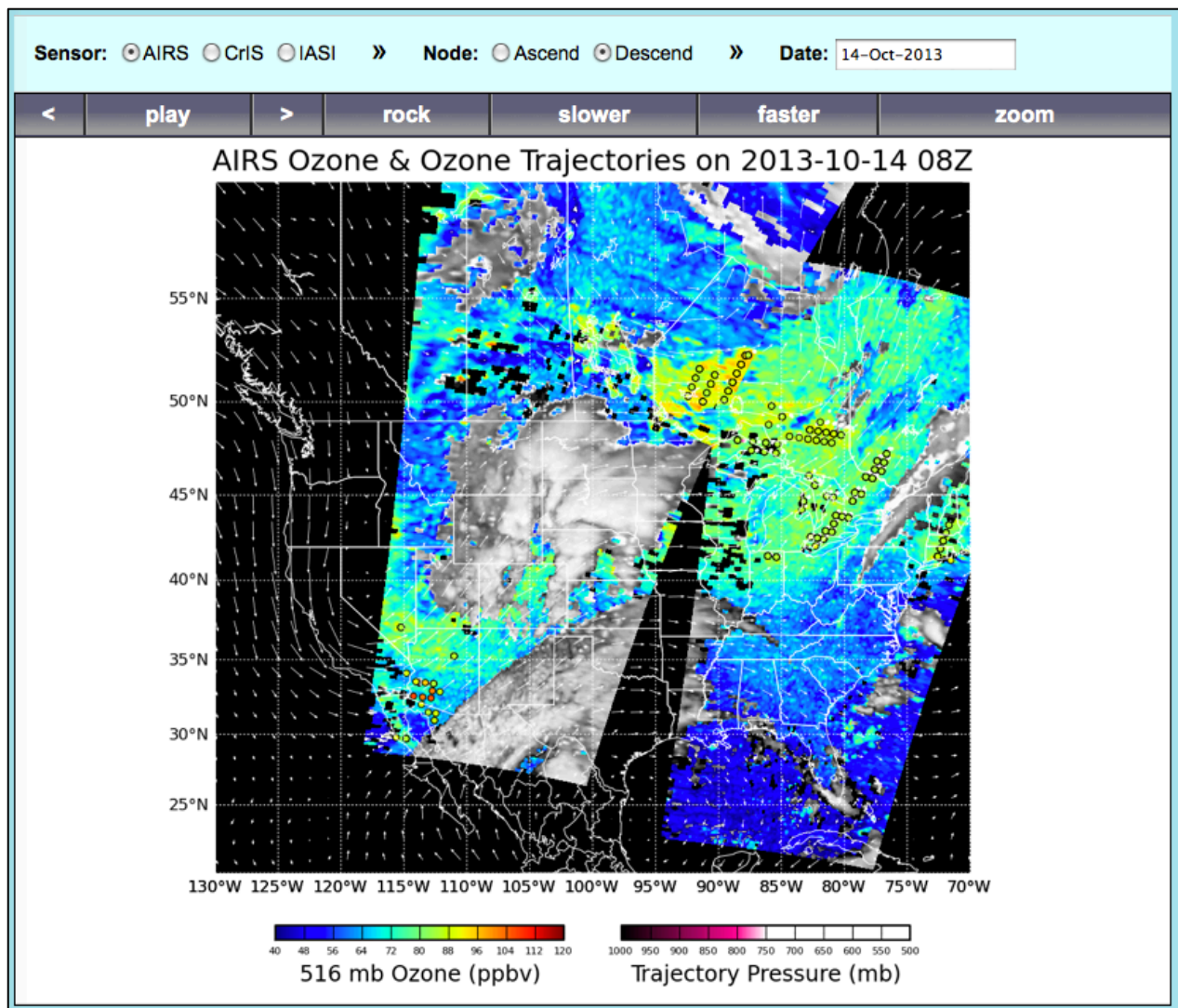


Table of Contents

SECTION 1: INTRODUCTION	3
1.1 OVERVIEW	3
1.2 HOW DOES THIS DIFFER FROM THE IDEA-I AEROSOL SOFTWARE?	3
1.3 SYSTEM REQUIREMENTS	3
1.4 INPUT DATA REQUIREMENTS	4
<i>AIRS, CrIS and IASI</i>	4
1.5 DISCLAIMER.....	4
SECTION 2: INSTALLATION AND CONFIGURATION	5
2.1 OVERVIEW	5
2.2 INSTALLATION OF IDEA-I SOFTWARE.....	5
2.3 SETTING UP CONFIGURATION FILES.....	5
2.3 ANCILLARY DATA	6
SECTION 3: CREATING HYPERSPECTRAL SOUNDER OZONE TRAJECTORIES	6
3.1 HOW CAN I GET STARTED QUICKLY?	6
3.2 IDEA-I OZONE DRIVER SCRIPT.....	7
3.3 IDEA-I OZONE OUTPUT FILES	9
3.3.1 <i>Output NetCDF Data files</i>	10
3.3.2 <i>Output Image files</i>	15
3.4 AUTOMATING EXECUTION OF IDEA-I OZONE TRAJECTORIES	16
SECTION 4: DISPLAYING IDEA-I OZONE FORECAST PRODUCTS	16
4.1 DISPLAYING OZONE TRAJECTORIES THROUGH A WEB BROWSER.....	16
4.2 DISPLAYING PRODUCTS USING MCIDAS-V	19
APPENDIX 1: IDEA-I TRAJECTORY MODEL AND REFERENCES	22
A1.1 TRAJECTORY MODEL	22
A1.2 REFERENCES.....	22

Section 1: Introduction

1.1 Overview

This document contains instructions for installation and operation of the IMAPP/CSPP Stratospheric Ozone Intrusion Forecast and Analysis software package, IDEA-I, in support of direct broadcast users. It is a globally configurable package that:

- **Creates 48 hour 3 dimensional trajectory forecasts of stratospheric ozone intrusions.**
 - Hyperspectral sounders **Aqua Atmospheric Infrared Sounder (AIRS)**, the **Suomi-NPP Cross-track Infrared Sounder (CrIS)** and/or the **MetOP Infrared Atmospheric Sounding Interferometer (IASI)** instrument Dual Regression Ozone Profile Retrievals are used to identify regions of high ozone concentrations (>80 ppb) below 500 hPa, and then a trajectory model is run to forecast the horizontal and vertical movement of the ozone over the next 48 hours. Retrieval software is freely distributed from either the IMAPP or CSPP websites.

The output products consist of both netCDF data files as well as hourly trajectory forecast images, which can be viewed as animations from a web browser using the PHP software that is included as part of this package.

The IMAPP/CSPP IDEA-I ozone Beta software is distributed through this ftp site:

ftp://ftp.ssec.wisc.edu/pub/IMAPP/beta/idea_ozone/

Software, test data, and documentation may be downloaded from this site.

The trajectories are predicted using the NASA Langley trajectory code [Pierce and Fairlie, 1993]. More detailed information about the model is provided in the Appendix.

1.2 How does this differ from the IDEA-I aerosol software?

The baseline trajectory forecast software is identical in both the IDEA-I aerosol and IDEA-I ozone packages, however the type of air quality event that is forecasted is quite different. The IDEA-I aerosol software identifies particulate matter pollution which can be caused by fires, smog or other natural or anthropogenic sources. The IDEA-I ozone intrusion software identifies regions of stratospheric intrusions, where stratospheric air descends into the troposphere due to tropopause folding events. These events are most likely to occur during the winter spring/months in the case of jet streams at the base of very strong low pressure systems and are most likely to influence surface ozone in regions of elevated terrain.

1.3 System requirements

System requirements for the IDEA-I_ozone software are as follows:

- Intel or AMD CPU with 64-bit instruction support,
- 1 GB RAM,
- CentOS 5 64-bit Linux (or other compatible 64-bit Linux distribution),
- 2 GB disk space,
- Reliable and efficient internet connection for downloading the required ancillary data (approximately 400 MB per day).

Linux terminal commands included in these instructions assume the bash shell is used.

1.4 Input Data Requirements

AIRS, CrIS and IASI

IMAPP/CSPP dual regression retrieval files from the hyperspectral instrument AIRS, CrIS and/or IASI are required as input to the ozone trajectory forecast of IDEA-I_ozone. The University of Wisconsin-Madison dual regression retrieval software is freely available as part of the IMAPP and CSPP software packages. Please see:

http://cimss.ssec.wisc.edu/cspp/uwhsrtv_edr_v1.2.shtml.

1.5 Disclaimer

Original software scripts and automation included as part of this package are distributed under the GNU GENERAL PUBLIC LICENSE agreement version 3. Software included as part of this software package are copyrighted and licensed by their respective organizations, and distributed consistent with their licensing terms.

The University of Wisconsin-Madison Space Science and Engineering Center (SSEC) makes no warranty of any kind with regard to the IMAPP and/or CSPP IDEA-I software or any 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 IMAPP and/or CSPP IDEA-I software.

Section 2: Installation and Configuration

2.1 Overview

This software will generate stratospheric intrusion forecast products from input Aqua, Terra, Suomi-NPP and/or MetOP data. It relies on setting up configuration files that point to the location of the input satellite and ancillary data files, and execution can be automated by using the cron unix scheduler.

2.2 Installation of IDEA-I Software

Download the following files from the IMAPP Beta ftp site:

ftp://ftp.ssec.wisc.edu/pub/IMAPP/beta/idea_ozone/

IMAPP_CSPP_IDEA-I_Ozone_V1.0_BETA.tar.gz

Next, unpack the tarfile (a new directory named IDEA-I_ozone will be created):

```
tar xf IMAPP_CSPP_IDEA-I_Ozone_V1.0_BETA.tar.gz
```

2.3 Setting Up Configuration Files

Edit the Ozone.cfg configuration file located in the IDEA-I_ozone/config directory of your installation. All three hyperspectral instrument (AIRS, CrIS and IASI) IDEA-I configurations can be set through the Ozone.cfg file. We suggest that you copy the configuration file that you want to edit into a separate file, like my_ozone.cfg, and edit that file to use during your IDEA-I execution.

There are many local adjustments that can be made to your IDEA-I installation by setting variables in the configuration files. For instance, you can choose to download the required satellite data input files from a remote location, such as from the public University of Wisconsin ftp site, and you can choose how many days of data to keep on disk. Some important parameters are listed below.

Required parameters for all IDEA-I executions:

OUTPUT_DIR =	Output Directory name
REGION_NAME =	Domain name written into output netCDF file attribute
TOP_LEFT_LON_LAT =	IDEA-I domain upper left lon/lat (ex: -130, 60)
BOTTOM_RIGHT_LON_LAT =	IDEA-I domain lower right lon/lat (ex: -70, 20)

AIRS, CrIS and IASI Instrument Ozone specific parameters:

AIRSLOCAL =	Local directory where AIRS dual regression retrieval files are located
CrISLOCAL =	Local directory where CrIS dual regression retrieval files are located
IASILOCAL =	Local directory where IASI dual regression retrieval files are located

Other options allow users to download satellite data from a local server, and to fetch the ancillary data from the National Climatic Data Center (NCDC) when a user wishes to process satellite data that is more than 2 days old. Please see the Ozone.cfg file for a complete list of options.

2.3 Ancillary Data

National Centers for Environmental Prediction (NCEP) Global Forecast System (GFS) grib2 files are required as input to the trajectory forecast model. The required files are identified and downloaded as part of the IDEA-I execution. The site that the files are downloaded from can be set using the configuration files; the default is to fetch the files from the University of Wisconsin ancillary data site:

<ftp://ftp.ssec.wisc.edu/pub/eosdb/ancillary>.

GFS grib2 formatted files from 00 hour to 60 hour forecasts at 6 hourly intervals are required. If your satellite data collection for a given instrument includes observations prior to 12 UTC for a given IDEA-I.bash execution, you will also need to download forecast fields from the previous day. Each of the grib2 files is about 35MB each; therefore, each day will require a download of about 400 MB. Both the MODIS aerosol and hyperspectral sounder ozone trajectory forecasts require the same ancillary files; if you are using both software packages, this download is required only once per day.

For dates and times that you wish to process that are more than 2 days old, then you should use analysis fields from the Global Data Assimilation System (GDAS) instead of forecast fields. These are distributed by the National Climatic Data Center (NCDC); the configuration files have entries that point to these files if you choose to use them.

Section 3: Creating Hyperspectral Sounder Ozone Trajectories

This section describes how to run IDEA-I_ozone to create trajectories of ozone intrusions using hyperspectral sounder retrievals from AIRS, CrIS and/or IASI instruments. IDEA-I ozone uses the University of Wisconsin (UW) Dual Regression Hyperspectral Retrievals to identify regions of high ozone values (> 80 ppb) below 500 hPa and with dew point depressions > 15 C to identify regions of potential stratospheric ozone intrusions. A 48 hour trajectory model is run to forecast the movement of the ozone in 3 dimensions over the chosen domain.

UW Dual Regression software is freely distributed through the IMAPP and CSPP websites:

http://cimss.ssec.wisc.edu/cspp/uwhsrtv_edr_v1.2.shtml

3.1 How can I get started quickly?

This section provides the basics for running a standard installation of IDEA-I for the creation of routine ozone trajectory products. More detailed information can be found in the following sections.

- 1). Unpack the software as described in Section 2.2 and set the required parameters to match your domain and system in your configuration file (Section 2.3).
- 2). Set the IDEA_HOME variable to be your top level installation directory, for instance:

```
export IDEA_HOME=${HOME}/IDEA-I_ozone
```

- 3). Test the software on your own AIRS data Descending passes (nighttime) from yesterday:

```
    ${IDEA_HOME}/IDEA-I.bash AIRS ozone -vGXN Descend- -d YESTERDAY -f  
my_ozone.cfg
```

or if your domain is over the Contiguous United States (CONUS), you can download and use data from the Wisconsin MODIS 7 day archive using this command,

```
    ${IDEA_HOME}/IDEA-I.bash AIRS ozone -vGFXN Descend- -d YESTERDAY -f  
my_ozone.cfg
```

As a test, running for YESTERDAY is safer than running for TODAY, since the source data for TODAY may not yet be available; remember also that TODAY is defined in UTC time.

4). Add a daily run to crontab, using `-d=TODAY`, with scheduled executions times set to after all daily source data is available.

```
25 12 * * * /home/IDEA-I_ozone/IDEA-I.bash AIRS ozone -v -X -G -N Descend- -d TODAY -f  
my_ozone.cfg
```

This entry will process the descending passes of Aqua (nighttime) from today.

```
03 22 * * * /home/IDEA-I_ozone/IDEA-I.bash CrIS ozone -v -F -X -G -N Ascend+ -d TODAY -f  
my_ozone.cfg > /home/IDEA-I_ozone/logs/ideai_cris_$(date +%Y%m%d_%H%M).log 2>&1
```

This entry will process the daytime CrIS data (ascending orbit), and will include a download of the direct broadcast retrieval files staged at the Wisconsin Direct Broadcast direct broadcast ftp site (-F option). If your domain falls within the Contiguous US, you can use these files as input with this option. If you use this option, you must ensure the FETCH options in the config file are set to YES. It sends the output ascii text information to a date stamped log file in the form of:

```
ideai_cris_YYYYMMDD_HHMM.log (eg: ideai_cris_20130530_2257.log)
```

If your domain is wholly covered by satellite data from one UTC day, but the data is not all available until the next UTC day, run with `-d=YESTERDAY`

5). If you want to display your output files or images, follow the instructions in Section 4.

3.2 IDEA-I Ozone driver script

The principal driver script for the hyperspectral sounder ozone product trajectories is `IDEA-I_ozone/IDEA-I.bash`, which wraps the perl processing scripts, `IDEA-I-AIRSozone.pl`, `IDEA-I-CrISozone.pl` and `IDEA-I-IASIozone.pl` located in the `IDEA-I_ozone/bin` directory. The `IDEA-I.bash` script requires two inputs, the sensor name and the product you wish to produce:

Usage: `./IDEA-I.bash <SENSOR> <product> <SENSORproduct-specific-arguments>`

Valid `<SENSOR>` `<product>` combinations are:
AIRS ozone | CrIS ozone | IASI ozone

It operates by default in a temporary directory within `IDEA-I_ozone/tmp`. This location can be set in the configuration file (`TMPDIR`). It creates intermediate files, including binary ancillary files (converted from grib2), final trajectory NetCDF (`.nc`) files as well as image files (`.png`) and places

them in a date stamped output file directory as defined in the configuration file that you are using. The default output file prefix is IDEA-I_ozone/products/myRegion. The software requires the following inputs, whose naming conventions, remote and local locations are set through the ozone configuration files:

- a) Dual regression retrieval files in HDF5 format. The naming conventions are:

```
<Instrument>_dYYYYMnDD_tHHMMSS<_info>.atm_prof_rtv.h5
```

Where

Instrument is either CrIS, AIRS or IASI

YYYY is the data year

Mn is the data month

DD is the data day of the month

HH is the data hour

MM is the data minut

SS is the data second

Info is either

granule ID (AIRS only)

MetOP satellite identification number (either M01 (A) or M02 (B) (IASI only))

- b) Dual regression h5 data file name examples:

```
S-NPP CrIS -      CrIS_d20131110_t180945.atm_prof_rtv.h5
Aqua AIRS -      AIRS_d20131110_t084123_g087.atm_prof_rtv.h5
MetOP-B IASI -   IASI_d20121026_t170300_M02.atm_prof_rtv.h5
```

To execute the ozone IDEA-I processing script for any of the hypespectral soundings, first set the top level IDEA-I_ozone directory, for example:

```
export IDEA_HOME=${HOME}/IDEA-I_ozone
```

Execute the run script using `-h` to print out a complete list of options. A subset of options is shown:

```
$IDEA_HOME/IDEA-I.bash AIRS ozone -h
```

```
-v run in verbose mode. Default is quiet.
-f configuration file - no default.
-F Allow fetching of Satellite Data remotely (Must also set FETCH option in
  config file to YES). Default is don't allow satellite data fetching.
-N Satellite orbit and data time. No defaults. Options are:
  Ascend+ Data orbit is ascending, and all data observations occur after
  12 UTC for the given day. This lets the IDEA-I software know how much
  ancillary data to gather, one day or two.
  Ascend- Data orbit is ascending, and at least some data observations
  occur before 12 UTC for the given day.
  Descend+ Data orbit is descending, and all data observations occur
  after 12 UTC for the given day. This lets the IDEA-I software know how
  much ancillary data to gather, one day or two.
  Descend- Data orbit is Descending, and at least some data observations
  occur before 12 UTC for the given day.
-G update GFS data from remote FTP site. Default is don't update at runtime.
-X execute trajectory model. Default is don't run.
-d Date - Can use either YYYYMMDD, TODAY, YESTERDAY
```


For example, executing the script with this set of options:

```
`${IDEA_HOME}/IDEA-I.bash AIRS ozone -v -G -X -N Descend- -d YESTERDAY -f  
my_ozone.cfg
```

or

```
`${IDEA_HOME}/IDEA-I.bash AIRS ozone -vGXN Descend- -d YESTERDAY -f my_ozone.cfg
```

will create a 48 hour ozone trajectory forecast based upon the Aqua AIRS retrievals from yesterday's descending orbits, which included data collected before 12 UTC, covering the domain defined in ``${IDEA_HOME}/config/my_ozone.cfg`. It will print a stream of text (verbose mode) as it progresses, and will update the NWP (GFS) data sets as part of this execution. The output products will be placed in the default location defined in my configuration file (`{IDEA-I}/products/myRegion`) with the extension that includes AIRSozone/<Ascend or Descend>/Date (YYYYMMDD).

```
`${IDEA_HOME}/IDEA-I.bash CrIS ozone -vGN Ascend+ -d 20130714 -f cris_ozone.cfg
```

This command will update the ancillary GFS files required to create a 48 hour ozone trajectory forecast based upon the CrIS Ozone profiles retrieved on 14 July 2013, which consisted entirely of Ascending orbit (Daytime) data all observed after 12 UTC on that day. When processing data that is more than 2 days old, you must use GFS analysis files from NCDC. To use these grib2 files, uncomment the middle section of GFS options in the configuration file, set `GFSFETCH = YES` and comment out the other two GFS sections. You must also include a sub-directory called "Analysis" as part of the `GFSLOCAL` parameter in the configuration file (for example, `{IDEA-I}/ancillary/Analysis/%Y_%m_%d_%j_%H` which is the default).

If I then execute this command:

```
`${IDEA_HOME}/IDEA-I.bash CrIS ozone -vXN Ascend+ -d 20130714 -f cris_ozone.cfg
```

the trajectory model will be executed. You may wish to separate the execution of the trajectory model and the ancillary file downloads, as it requires about 400 MB of GFS grib2 files for each day's execution. Ancillary data files that are used for one instrument can be reused for all instruments each day.

```
`${IDEA_HOME}/IDEA-I.bash IASI ozone -vGXN Ascend- -d TODAY -f my_ozone.cfg  
`${IDEA_HOME}/IDEA-I.bash IASI ozone -vGXN Descend+ -d TODAY -f my_ozone.cfg
```

These commands would be used to create one typical day of IASI MetOP-B IDEA-I trajectory forecasts, depending on if your data was observed before or after 12 UTC. The first entry uses the Ascending MetOP-B (nighttime) passes, while the second entry utilizes the Descending (daytime) passes.

3.3 IDEA-I ozone output files

A successful execution of the IDEA-I hyperspectral sounding ozone trajectory forecast will result in a text log file containing details about the trajectory forecast model run (`GFS_traject_3d_v01.log`), as well as NetCDF data and .png image files covering your domain, placed in the output file directory.

3.3.1 Output NetCDF Data files

Two data files in NetCDF format are produced for each IDEA-I ozone run.

One contains both single Field-of-View (FOV) ozone retrieved values for each pass, as well as aggregated gridded ozone and GFS model grid data. The naming convention for these files is:

(Instrument)ozone(X)_grid_48hr_YYYYMMDD.nc (ex: **AIRSozoneD_grid_48hr_20131110.nc**)

Where Instrument is either AIRS, CrIS or IASI and X is either A for Ascending or D for Descending orbit. These files contain separate data arrays for each pass acquired in both native resolution (dependent upon the instrument) as well as aggregated gridded resolution (default is 3-10km pixels by 4-10km pixels – set using the SUPIX and NPIX variables in your configuration file). The GFS fields are also included that cover your domain for the 48 hour time step. These fields are dimensioned “west_east” and “north_south” and are presented in 11 discrete time steps, from model initialization until 60 hours later, in 6 hour increments. The first available ozone product file may be several hours after the trajectory model initialization time, depending upon your earth location and on the instrument that you choose to use. Thus the 48 hour duration of the trajectory forecast is typical, but not precise.

A header file dump of a typical output (in this case AIRS) NetCDF gridded file consisting of 2 passes for the given day descending orbits is shown below:

```
netcdf AIRSozoneD_grid_48hr_20131110 {
dimensions:
    time = UNLIMITED ; // (11 currently)
    west_east = 121 ;
    north_south = 81 ;
    top_bottom = 25 ;
    across_track = 90 ;
    along_track_001 = 211 ;
    along_track_002 = 239 ;
    across_aggr = 30 ;
    along_aggr_001 = 71 ;
    along_aggr_002 = 80 ;
variables:
    int Aggr_Counter_001(top_bottom, along_aggr_001, across_aggr) ;
        Aggr_Counter_001:long_name = "Valid AIRS pixels per aggregated pixel for swath 001" ;
        Aggr_Counter_001:_FillValue = -9999 ;
        Aggr_Counter_001:units = "dimensionless" ;
        Aggr_Counter_001:coordinates = "Plevs LonAggr_001 LatAggr_001" ;
    int Aggr_Counter_002(top_bottom, along_aggr_002, across_aggr) ;
        Aggr_Counter_002:long_name = "Valid AIRS pixels per aggregated pixel for swath 002" ;
        Aggr_Counter_002:_FillValue = -9999 ;
        Aggr_Counter_002:units = "dimensionless" ;
        Aggr_Counter_002:coordinates = "Plevs LonAggr_002 LatAggr_002" ;
    float Aggr_Dew_Point_Depression_001(top_bottom, along_aggr_001, across_aggr) ;
        Aggr_Dew_Point_Depression_001:long_name = "aggregated pixel dew point depression for swath 001" ;
        Aggr_Dew_Point_Depression_001:_FillValue = -9999.f ;
        Aggr_Dew_Point_Depression_001:units = "K" ;
        Aggr_Dew_Point_Depression_001:coordinates = "Plevs LonAggr_001 LatAggr_001" ;
    float Aggr_Dew_Point_Depression_002(top_bottom, along_aggr_002, across_aggr) ;
        Aggr_Dew_Point_Depression_002:long_name = "aggregated pixel dew point depression for swath 002" ;
        Aggr_Dew_Point_Depression_002:_FillValue = -9999.f ;
        Aggr_Dew_Point_Depression_002:units = "K" ;
        Aggr_Dew_Point_Depression_002:coordinates = "Plevs LonAggr_002 LatAggr_002" ;
    float Aggr_Ozone_Retrieval_001(top_bottom, along_aggr_001, across_aggr) ;
        Aggr_Ozone_Retrieval_001:long_name = "aggregated pixel ozone retrieval for swath 001" ;
        Aggr_Ozone_Retrieval_001:_FillValue = -9999.f ;
        Aggr_Ozone_Retrieval_001:units = "ppmv" ;
```

```

Aggr_Ozone_Retrieval_001:coordinates = "Plevs LonAggr_001 LatAggr_001" ;

float Aggr_Ozone_Retrieval_002(top_bottom, along_aggr_002, across_aggr) ;
  Aggr_Ozone_Retrieval_002:long_name = "aggregated pixel ozone retrieval for swath 002" ;
  Aggr_Ozone_Retrieval_002:_FillValue = -9999.f ;
  Aggr_Ozone_Retrieval_002:units = "ppmv" ;
  Aggr_Ozone_Retrieval_002:coordinates = "Plevs LonAggr_002 LatAggr_002" ;
int Aggr_inROI_001(along_aggr_001, across_aggr) ;
  Aggr_inROI_001:long_name = "aggregated pixel in region of interest for swath 001" ;
  Aggr_inROI_001:_FillValue = -9999 ;
  Aggr_inROI_001:units = "dimensionless" ;
  Aggr_inROI_001:coordinates = "LonAggr_001 LatAggr_001" ;
int Aggr_inROI_002(along_aggr_002, across_aggr) ;
  Aggr_inROI_002:long_name = "aggregated pixel in region of interest for swath 002" ;
  Aggr_inROI_002:_FillValue = -9999 ;
  Aggr_inROI_002:units = "dimensionless" ;
  Aggr_inROI_002:coordinates = "LonAggr_002 LatAggr_002" ;
float Aggr_psf_001(along_aggr_001, across_aggr) ;
  Aggr_psf_001:long_name = "aggregation psfc time-interpolated for swath 001" ;
  Aggr_psf_001:_FillValue = -9999.f ;
  Aggr_psf_001:units = "hPa" ;
  Aggr_psf_001:coordinates = "LonAggr_001 LatAggr_001" ;
float Aggr_psf_002(along_aggr_002, across_aggr) ;
  Aggr_psf_002:long_name = "aggregation psfc time-interpolated for swath 002" ;
  Aggr_psf_002:_FillValue = -9999.f ;
  Aggr_psf_002:units = "hPa" ;
  Aggr_psf_002:coordinates = "LonAggr_002 LatAggr_002" ;
float Cloud_Effective_Emissivity_001(along_track_001, across_track) ;
  Cloud_Effective_Emissivity_001:long_name = "cloud effective emissivity for swath number 001" ;
  Cloud_Effective_Emissivity_001:_FillValue = -9999.f ;
  Cloud_Effective_Emissivity_001:units = "dimensionless" ;
  Cloud_Effective_Emissivity_001:coordinates = "Lon_001 Lat_001" ;
float Cloud_Effective_Emissivity_002(along_track_002, across_track) ;
  Cloud_Effective_Emissivity_002:long_name = "cloud effective emissivity for swath number 002" ;
  Cloud_Effective_Emissivity_002:_FillValue = -9999.f ;
  Cloud_Effective_Emissivity_002:units = "dimensionless" ;
  Cloud_Effective_Emissivity_002:coordinates = "Lon_002 Lat_002" ;
float Cloud_Optical_Depth_002(along_track_002, across_track) ;
  Cloud_Optical_Depth_002:long_name = "cloud optical depth for swath number 002" ;
  Cloud_Optical_Depth_002:_FillValue = -9999.f ;
  Cloud_Optical_Depth_002:units = "dimensionless" ;
  Cloud_Optical_Depth_002:coordinates = "Lon_002 Lat_002" ;
float Dew_Point_Depression_001(top_bottom, along_track_001, across_track) ;
  Dew_Point_Depression_001:long_name = "dew point depression for swath 001" ;
  Dew_Point_Depression_001:_FillValue = -9999.f ;
  Dew_Point_Depression_001:units = "K" ;
  Dew_Point_Depression_001:coordinates = "Plevs Lon_001 Lat_001" ;
float Dew_Point_Depression_002(top_bottom, along_track_002, across_track) ;
  Dew_Point_Depression_002:long_name = "dew point depression for swath 002" ;
  Dew_Point_Depression_002:_FillValue = -9999.f ;
  Dew_Point_Depression_002:units = "K" ;
  Dew_Point_Depression_002:coordinates = "Plevs Lon_002 Lat_002" ;
float LatAggr_001(along_aggr_001, across_aggr) ;
  LatAggr_001:long_name = "latitude for aggregated pixel for swath 001" ;
  LatAggr_001:_FillValue = -9999.f ;
  LatAggr_001:units = "degrees_north" ;
  LatAggr_001:valid_max = 90.f ;
  LatAggr_001:valid_min = -90.f ;
float LatAggr_002(along_aggr_002, across_aggr) ;
  LatAggr_002:long_name = "latitude for aggregated pixel for swath 002" ;
  LatAggr_002:_FillValue = -9999.f ;
  LatAggr_002:units = "degrees_north" ;
  LatAggr_002:valid_max = 90.f ;
  LatAggr_002:valid_min = -90.f ;
float Lat_001(along_track_001, across_track) ;

```

```

    Lat_001:long_name = "latitude for swath number 001" ;
    Lat_001:_FillValue = -9999.f ;
    Lat_001:units = "degrees_north" ;
    Lat_001:valid_max = 90.f ;
    Lat_001:valid_min = -90.f ;
float Lat_002(along_track_002, across_track) ;
    Lat_002:long_name = "latitude for swath number 002" ;
    Lat_002:_FillValue = -9999.f ;
    Lat_002:units = "degrees_north" ;
    Lat_002:valid_max = 90.f ;
    Lat_002:valid_min = -90.f ;
float Latitude(north_south) ;
    Latitude:long_name = "latitude" ;
    Latitude:_FillValue = -9999.f ;
    Latitude:units = "degrees_north" ;
    Latitude:valid_max = 90.f ;
    Latitude:valid_min = -90.f ;
float LonAggr_001(along_aggr_001, across_aggr) ;
    LonAggr_001:long_name = "longitude for aggregated pixel for swath 001" ;
    LonAggr_001:_FillValue = -9999.f ;
    LonAggr_001:units = "degrees_east" ;
    LonAggr_001:valid_max = 180.f ;
    LonAggr_001:valid_min = -180.f ;
float LonAggr_002(along_aggr_002, across_aggr) ;
    LonAggr_002:long_name = "longitude for aggregated pixel for swath 002" ;
    LonAggr_002:_FillValue = -9999.f ;
    LonAggr_002:units = "degrees_east" ;
    LonAggr_002:valid_max = 180.f ;
    LonAggr_002:valid_min = -180.f ;
float Lon_001(along_track_001, across_track) ;
    Lon_001:long_name = "longitude for swath number 001" ;
    Lon_001:_FillValue = -9999.f ;
    Lon_001:units = "degrees_east" ;
    Lon_001:valid_max = 180.f ;
    Lon_001:valid_min = -180.f ;
float Lon_002(along_track_002, across_track) ;
    Lon_002:long_name = "longitude for swath number 002" ;
    Lon_002:_FillValue = -9999.f ;
    Lon_002:units = "degrees_east" ;
    Lon_002:valid_max = 180.f ;
    Lon_002:valid_min = -180.f ;
float Longitude(west_east) ;
    Longitude:long_name = "longitude" ;
    Longitude:_FillValue = -9999.f ;
    Longitude:units = "degrees_east" ;
    Longitude:valid_max = 180.f ;
    Longitude:valid_min = -180.f ;
float Ozone_Retrieval_001(top_bottom, along_track_001, across_track) ;
    Ozone_Retrieval_001:long_name = "retrieved ozone for swath 001" ;
    Ozone_Retrieval_001:_FillValue = -9999.f ;
    Ozone_Retrieval_001:units = "ppmv" ;
    Ozone_Retrieval_001:coordinates = "Plevs Lon_001 Lat_001" ;
float Ozone_Retrieval_002(top_bottom, along_track_002, across_track) ;
    Ozone_Retrieval_002:long_name = "retrieved ozone for swath 002" ;
    Ozone_Retrieval_002:_FillValue = -9999.f ;
    Ozone_Retrieval_002:units = "ppmv" ;
    Ozone_Retrieval_002:coordinates = "Plevs Lon_002 Lat_002" ;
float Plevs(top_bottom) ;
    Plevs:long_name = "AIRS pressure levels" ;
    Plevs:_FillValue = -9999.f ;
    Plevs:units = "hPa" ;
    Plevs:valid_max = 1100.f ;
    Plevs:valid_min = 500.f ;
int Qflag1_001(along_track_001, across_track) ;
    Qflag1_001:long_name = "Quality flag 1 for swath 001" ;

```

```

    Qflag1_001:_FillValue = -9999 ;
    Qflag1_001:units = "dimensionless" ;
    Qflag1_001:coordinates = "Lon_001 Lat_001" ;
int Qflag1_002(along_track_002, across_track) ;
    Qflag1_002:long_name = "Quality flag 1 for swath 002" ;
    Qflag1_002:_FillValue = -9999 ;
    Qflag1_002:units = "dimensionless" ;
    Qflag1_002:coordinates = "Lon_002 Lat_002" ;
int Qflag2_001(along_track_001, across_track) ;
    Qflag2_001:long_name = "Quality flag 2 for swath 001" ;
    Qflag2_001:_FillValue = -9999 ;
    Qflag2_001:units = "dimensionless" ;
    Qflag2_001:coordinates = "Lon_001 Lat_001" ;
int Qflag2_002(along_track_002, across_track) ;
    Qflag2_002:long_name = "Quality flag 2 for swath 002" ;
    Qflag2_002:_FillValue = -9999 ;
    Qflag2_002:units = "dimensionless" ;
    Qflag2_002:coordinates = "Lon_002 Lat_002" ;
int Qflag3_002(along_track_002, across_track) ;
    Qflag3_002:long_name = "Quality flag 3 for swath 002" ;
    Qflag3_002:_FillValue = -9999 ;
    Qflag3_002:units = "dimensionless" ;
    Qflag3_002:coordinates = "Lon_002 Lat_002" ;
float Skin_Temperature_001(along_track_001, across_track) ;
    Skin_Temperature_001:long_name = "retrieved skin temperature for swath number 001" ;
    Skin_Temperature_001:_FillValue = -9999.f ;
    Skin_Temperature_001:units = "K" ;
    Skin_Temperature_001:coordinates = "Lon_001 Lat_001" ;
float Skin_Temperature_002(along_track_002, across_track) ;
    Skin_Temperature_002:long_name = "retrieved skin temperature for swath number 002" ;
    Skin_Temperature_002:_FillValue = -9999.f ;
    Skin_Temperature_002:units = "K" ;
    Skin_Temperature_002:coordinates = "Lon_002 Lat_002" ;
float Surface_Pressure_GDAS_001(along_track_001, across_track) ;
    Surface_Pressure_GDAS_001:long_name = "surface pressure from GDAS for swath number 001" ;
    Surface_Pressure_GDAS_001:_FillValue = -9999.f ;
    Surface_Pressure_GDAS_001:units = "hPa" ;
    Surface_Pressure_GDAS_001:coordinates = "Lon_001 Lat_001" ;
float Surface_Pressure_GDAS_002(along_track_002, across_track) ;
    Surface_Pressure_GDAS_002:long_name = "surface pressure from GDAS for swath number 002" ;
    Surface_Pressure_GDAS_002:_FillValue = -9999.f ;
    Surface_Pressure_GDAS_002:units = "hPa" ;
    Surface_Pressure_GDAS_002:coordinates = "Lon_002 Lat_002" ;
float Total_Column_Ozone_001(along_track_001, across_track) ;
    Total_Column_Ozone_001:long_name = "total column ozone for swath number 001" ;
    Total_Column_Ozone_001:_FillValue = -9999.f ;
    Total_Column_Ozone_001:units = "DU" ;
    Total_Column_Ozone_001:coordinates = "Lon_001 Lat_001" ;
float Total_Column_Ozone_002(along_track_002, across_track) ;
    Total_Column_Ozone_002:long_name = "total column ozone for swath number 002" ;
    Total_Column_Ozone_002:_FillValue = -9999.f ;
    Total_Column_Ozone_002:units = "DU" ;
    Total_Column_Ozone_002:coordinates = "Lon_002 Lat_002" ;
float apcp(time, north_south, west_east) ;
    apcp:long_name = "accumulated precipitation" ;
    apcp:_FillValue = -9999.f ;
    apcp:units = "kg per m^2" ;
    apcp:coordinates = "Longitude Latitude" ;
float hpbl(time, north_south, west_east) ;
    hpbl:long_name = "planetary boundary layer height" ;
    hpbl:_FillValue = -9999.f ;
    hpbl:units = "m" ;
    hpbl:coordinates = "Longitude Latitude" ;
float psfc(time, north_south, west_east) ;
    psfc:long_name = "surface pressure" ;

```

```

psfc:_FillValue = -9999.f ;
psfc:units = "hPa" ;
psfc:coordinates = "Longitude Latitude" ;
float ptrp(time, north_south, west_east) ;
ptrp:long_name = "tropopause pressure" ;
ptrp:_FillValue = -9999.f ;
ptrp:units = "hPa" ;
ptrp:coordinates = "Longitude Latitude" ;
double time(time) ;
time:long_name = "Model Time" ;
time:units = "hours since 2013-11-09 12:00:00 GMT" ;
float uwind500(time, north_south, west_east) ;
uwind500:long_name = "zonal wind at 500 mb" ;
uwind500:_FillValue = -9999.f ;
uwind500:units = "m/s" ;
uwind500:coordinates = "Longitude Latitude" ;
float uwind700(time, north_south, west_east) ;
uwind700:long_name = "zonal wind at 700 mb" ;
uwind700:_FillValue = -9999.f ;
uwind700:units = "m/s" ;
uwind700:coordinates = "Longitude Latitude" ;
float uwind850(time, north_south, west_east) ;
uwind850:long_name = "zonal wind at 850 mb" ;
uwind850:_FillValue = -9999.f ;
uwind850:units = "m/s" ;
uwind850:coordinates = "Longitude Latitude" ;
float vwind500(time, north_south, west_east) ;
vwind500:long_name = "meridional wind at 500 mb" ;
vwind500:_FillValue = -9999.f ;
vwind500:units = "m/s" ;
vwind500:coordinates = "Longitude Latitude" ;
float vwind700(time, north_south, west_east) ;
vwind700:long_name = "meridional wind at 700 mb" ;
vwind700:_FillValue = -9999.f ;
vwind700:units = "m/s" ;
vwind700:coordinates = "Longitude Latitude" ;
float vwind850(time, north_south, west_east) ;
vwind850:long_name = "meridional wind at 850 mb" ;
vwind850:_FillValue = -9999.f ;
vwind850:units = "m/s" ;
vwind850:coordinates = "Longitude Latitude" ;
float zsfc(time, north_south, west_east) ;
zsfc:long_name = "surface geopotential height" ;
zsfc:_FillValue = -9999.f ;
zsfc:units = "gpm" ;
zsfc:coordinates = "Longitude Latitude" ;

```

The second file contains trajectory forecast information created from running the trajectory model. It includes both the number of trajectories that were initialized as well as the location of the trajectories through the 48 hour time period. It is matched up with the gridded output NetCDF file through the xtraj and ytraj fields that contain latitude and longitude. Both NetCDF files are used to create the output file .png images. The naming convention for these files is:

(Instrument)ozone(X)_traj_48hr_YYYYMMDD.nc (ex: **AIRSozoneD_traj_48hr_20131110.nc**)

where again, Instrument is either AIRS, CrIS or IASI and X is either A for Ascending or D for Descending orbit. A header file dump of a typical trajectory output NetCDF file is shown below:

```

netcdf AIRSozoneD_traj_48hr_20131110 {
dimensions:
    time = UNLIMITED ; // (42 currently)
    traj = 647 ;

```

variables:

```
float age_traj(time, traj) ;
    age_traj:long_name = "age of trajectory" ;
    age_traj:_FillValue = -9999.f ;
    age_traj:units = "hours" ;
    age_traj:coordinates = "ytraj xtraj" ;
float ao3_traj(traj) ;
    ao3_traj:long_name = "initial ozone volume mixing ratio" ;
    ao3_traj:_FillValue = -9999.f ;
    ao3_traj:units = "ppmv" ;
    ao3_traj:coordinates = "ytraj xtraj" ;
float ptraj(time, traj) ;
    ptraj:long_name = "pressure level of trajectory" ;
    ptraj:_FillValue = -9999.f ;
    ptraj:units = "hPa" ;
    ptraj:coordinates = "ytraj xtraj" ;
float qtraj(time, traj) ;
    qtraj:long_name = "water vapor mixing ratio of trajectory" ;
    qtraj:_FillValue = -9999.f ;
    qtraj:units = "kg/kg" ;
    qtraj:coordinates = "ytraj xtraj" ;
double time(time) ;
    time:long_name = "Model Time" ;
    time:units = "hours since 2013-11-09 12:00:00 GMT" ;
float ttraj(time, traj) ;
    ttraj:long_name = "temperature of trajectory" ;
    ttraj:_FillValue = -9999.f ;
    ttraj:units = "Kelvin" ;
    ttraj:coordinates = "ytraj xtraj" ;
float xtraj(time, traj) ;
    xtraj:long_name = "longitude" ;
    xtraj:_FillValue = -9999.f ;
    xtraj:units = "degrees_east" ;
    xtraj:valid_max = 180.f ;
    xtraj:valid_min = -180.f ;
float ytraj(time, traj) ;
    ytraj:long_name = "latitude" ;
    ytraj:_FillValue = -9999.f ;
    ytraj:units = "degrees_north" ;
    ytraj:valid_max = 90.f ;
    ytraj:valid_min = -90.f ;
```

3.3.2 Output Image files

Image product files showing the hourly progression of ozone trajectories are produced for each execution of IDEA-I ozone. The hourly output .png format files are named xxframe.png, where xx is the hour of the forecast since 12 UTC. The last .png file will always be the 60 hour trajectory forecast, 60frame.png; the first file will depend upon the retrieval data files that you are using, and their relationship to 12 UTC of your forecast day. The size of the image files is by default 1280x960, but can be set using the IMGRES parameter in your configuration file. These files can be displayed through a web browser as an animation with controls using the PHP software that is included with this package. Please see the section on Displaying IDEA-I Ozone Products.

If you have the ImageMajik convert command available, it will create an animated GIF file composite of all of the .png files in the directory. This file naming convention is: (Instrument)ozone(X)_traj_48hr_YYYYMMDD.gif (ex: AIRSozoneD_traj_48hr_20131110.gif), where Instrument is either AIRS, CrIS or IASI and X is either A for Ascending or D for Descending orbit.

3.4 Automating Execution of IDEA-I Ozone Trajectories

Execution of the IDEA-I Hyperspectral Sounder Ozone Retrievals software can be automated using the unix cron scheduler. For example, to schedule a daily execution for Aqua AIRS, a daily crontab entry would look something like this:

```
25 12 * * * /home/IDEA-I_ozone/IDEA-I.bash AIRS ozone -v -X -G -N Descend- -d TODAY -f my_ozone.cfg
```

```
0 23 * * * /home/IDEA-I_ozone/IDEA-I.bash AIRS ozone -v -X -G -N Ascend+ -d TODAY -f my_ozone.cfg > /home/IDEA-I_ozone/logs/ideai_airs_$(date +%Y%m%d_%H%M).log 2>&1
```

The first entry will execute daily at 14:00 UTC (dependent upon your computer system time), using all of the Descending passes found in the directory identified in the my_ozone.cfg file. The descending orbits represent the nighttime passes, and the (-) means that at least some of those data observations occurred before 12 UTC today. The second entry will create a trajectory forecast using all of the Ascending passes (daytime observations). If your domain is wholly covered by AIRS data from one UTC day, but the data is not all available until the next UTC day, run with -d YESTERDAY. Add the -F flag if you want to download data from the University of Wisconsin Direct Broadcast 7 day data archive ftp site covering the Contiguous United States domain. To do this, you must also set your config file AIRSFETCH, CrISFETCH and/or IASIFETCH parameters to YES.

```
14 12 * * * /data3/kathys/idea_2013/ozone/IDEA-I_ozone/IDEA-I.bash CrIS ozone -v -F -X -G -N Descend- -d TODAY -f my_ozone.cfg > /home/IDEA-I_ozone/logs/ideai_cris_$(date +%Y%m%d_%H%M).log 2>&1
```

```
03 22 * * * /data3/kathys/idea_2013/ozone/IDEA-I_ozone/IDEA-I.bash CrIS ozone -v -F -X -G -N Ascend+ -d TODAY -f my_ozone.cfg > /home/IDEA-I_ozone/logs/ideai_cris_$(date +%Y%m%d_%H%M).log 2>&1
```

These options both show how you execute the IDEA-I ozone software for both nighttime (Descending) and daytime (Ascending) CrIS instrument passes including fetching the data from the Wisconsin Direct Broadcast data ftp site. It will place the text output generated by standard output and standard error in a date/time stamped file in /home/IDEA-I_ozone/logs.

Section 4: Displaying IDEA-I Ozone Forecast Products

4.1 Displaying Ozone Trajectories Through a Web Browser

The output image .png files can be displayed as animations in a web browser using the PHP software that is included in the IDEA-I/www directory. You must have a web server running on the computer you are serving data from.

If your web server is configured so that "http://www.your-web-site.com/" points at "/www/html/" then linking /www/html/idea-i-ozone -> /home/IDEA-I_ozone/www will display graphics available at http://www.your-web-site.com/idea-i-ozone

Example: Your IDEA-I_ozone/www package location: /home/kathys/IDEA-I_ozone/www

Your web server configured: /var/www/html/

Then link www to /var/www/html:

```
cd /var/www/html
```

```
ln -s /home/kathys/IDEA-I_ozone/www idea-i-ozone
```


View at: <http://www.your-web-site.com/idea-i-ozone>

In the `www/` directory, you can change the settings for your website IDEA-I display by editing the `config.php` file.

The defaults as they are set will result in any data that is located within the `IDEA-I_ozone/products/myRegion` directory to be displayed with radio buttons for any sensors found in the directory (our example includes AIRSozone and CrISozone), and a radio button display for the Nodes found in subdirectories (Ascend or Descend) along with a selectable Date Calendar on the webpage. AIRS is the default sensor display.

You can modify the PHP software to display different regions, satellites and products by editing the `IDEA-I_ozone/www.config.php` file. Here are a few parameters that can be adjusted:

- **dataDir** - default: `myRegion` (this directory is soft linked from `../products`, and includes 3 days of example data of a region over the United States)
This is the top level directory where the images being produced by your IDEA-I_ozone executions should reside. This `myRegion` directory is currently a soft link into the `IDEA-I_ozone/www` directory. The website will display a single image on the front page from the most recently processed day. By clicking on the image or on the "View Latest" button it will display the trajectory forecast imagery as an animation that you can control through a web browser. The images for an animation must be in a date sub-folder that following the following naming convention,

`yyyymmdd` where,

`yyyy` is the year
`mm` is the month
`dd` is the day of month

Optionally, you can use additional sub-folders to organize the date folders under the top level folder specified by "dataDir". Example:

```
region_1
  sensor_1
    node_1 (Ascend or Descend)
      20120105
        images ...
      20120107
        images ...
  sensor_2
    node_2 (Ascend or Descend)
      20120105
        images ...
      20120107
        images ...
region_2
  and so on ...
```

In the example data that is included with the release, `sensor_1` is AIRS, `sensor_2` is CrIS, `node_1` is Ascend and `node_2` is Descend.

The date folders must be the last folders in your directory structure - they are the folders that must contain the actual images that will be animated.

- **params** - default: array('date');
You must specify the structure of the directory tree that your images are stored in. This is done via the params array.

Example:

If dataDir = 'myRegion';

and the directory tree structure example shown above, then params should be

```
public static $params = array('sensor', 'node', 'date');
```

This is specifying the structure of your directory tree where you store the images and it is also giving a parameter name to each of the directory tree levels. The last "parameter" or level in your directory tree must be your date folders (of the form yyyyymmdd) and must be called date.

- **param_defaults** - default: array('latest')
In addition to specifying the structure of the directory tree in which you store images you want to animate, you need to specify the default folder to use for each "parameter" using the param_defaults array.

So using the example above, you could specify something like this:

```
public static $param_defaults = array('sensor' => 'sensor_2',  
                                     'node' => 'node_1',  
                                     'date' => 'latest');
```

This would mean that if the URL is not specifying a particular sensor/node/date/ folder to animate, then the default date folder to look for images in would be

```
sensor_2/node_1/{latest}/
```

where {latest} is the latest (chronologically) date folder in

```
sensor_2/node_1/
```

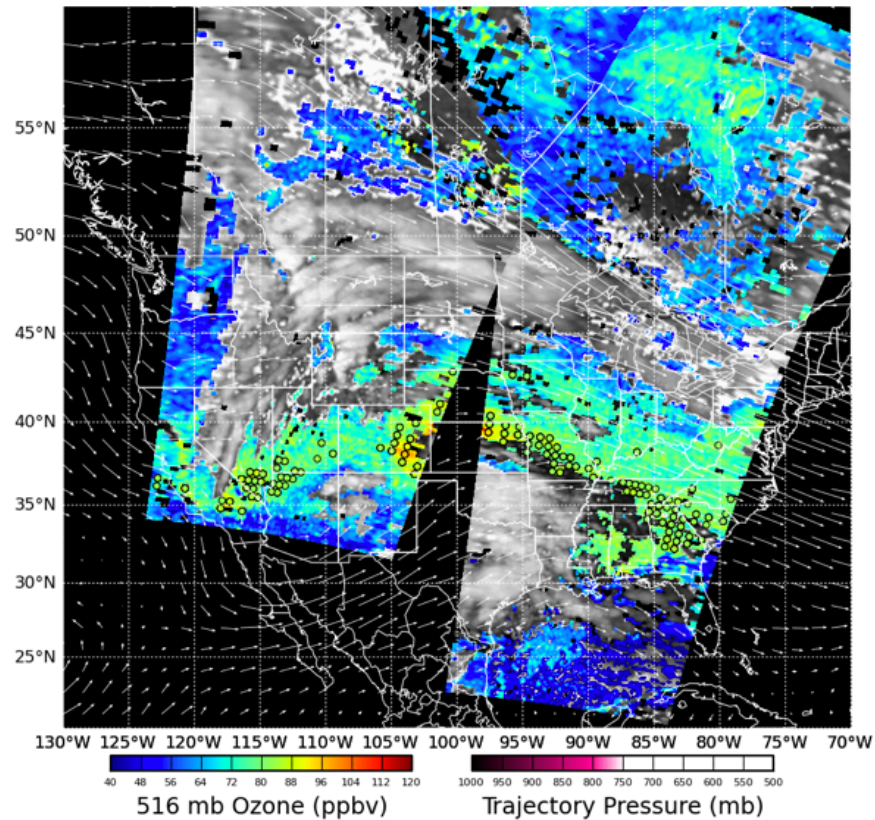
- **logoFilename** - Default is my_logo.png
You can copy your logo into the /www/templates/default/images director and it will display in the top right corner of the web page.

Other options exist that allow you to set the control buttons to be either dropdown menus or radio buttons. For a list of all the options, please see the README_IDEA_I_WWW.txt file located in the IDEA-I_ozone/www directory. The figure below is a screen capture of a website display where both AIRS and CrIS are available for both Ascending and Descending Nodes.

Ozone Forward Trajectory Forecast

Sensor: AIRSozone CrISozone » Node: Ascend Descend » Date:

AIRS Ozone & Ozone Trajectories on 2013-12-13 08Z
AIRS swath start times: 7.78Z 9.41Z



Animate

Last updated: 3 February 2012



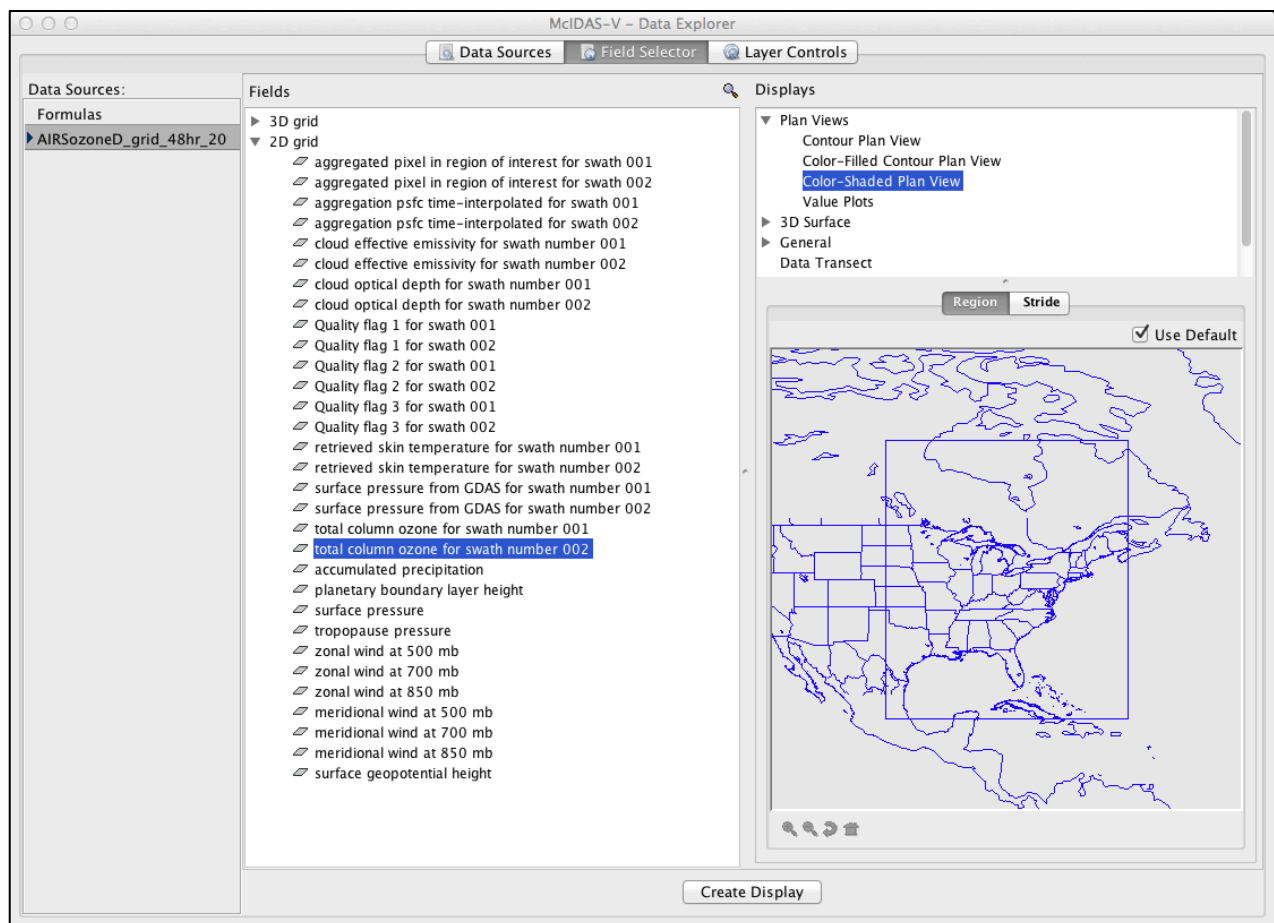
4.2 Displaying Products using McIDAS-V

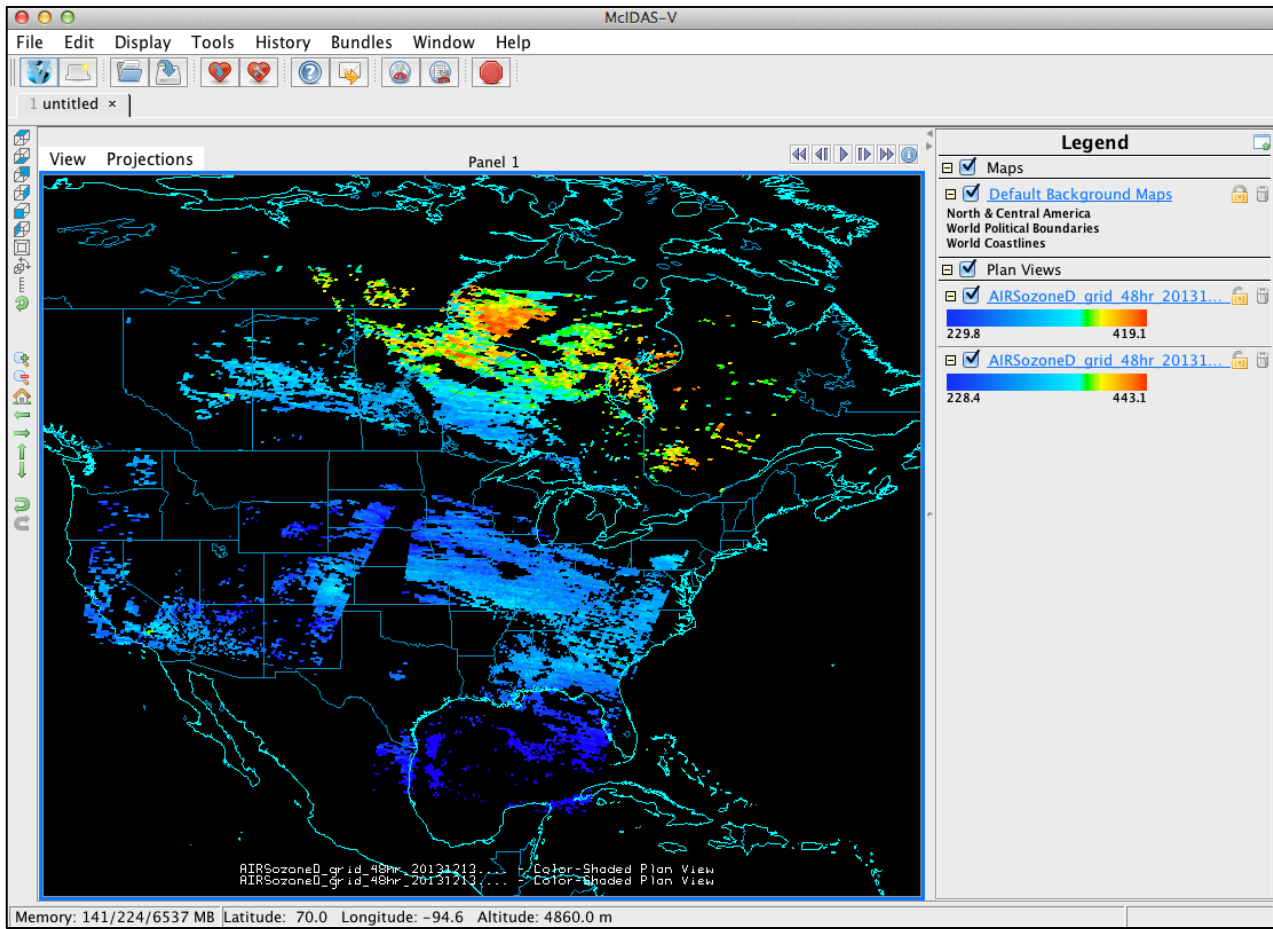
A free visualization and analysis tool that can be used to display the output NetCDF gridded file is the Man computer Interactive Data Access System – V (McIDAS-V) available via download from:

<http://www.ssec.wisc.edu/mcidas/software/v/>

To display your IDEA-I ozone grid files using McV, download and install the McV software onto your local machine (Linux, Mac and Windows are supported). Once the IDEA-I output gridded NetCDF file (ex: **CriSozoneD_grid_48hr_20131213.nc**) is on your local machine, launch McV. In the Data Explorer Window, choose **Gridded Data->Local** from the left navigation options. In the **Files:** window, navigate to your NetCDF file, select the file, and click on the **Add Source** button at the bottom. A new window will appear, with sections labeled Fields and Displays. Under **Fields**, select **2D grid**. This will expand the list of fields available for display in the file. As an example, select “total column ozone for swath number 001”. Once you select this parameter, the Displays section will be active. Select **Color-Shaded Plan View**, then click on the **Create Display** button at the bottom of the window. The McIDAS-V image window will display the retrieved total column ozone for the first pass in the Image Window. If there is more than one pass stored in the file, you can display them all in the same window, by returning to the **Data Explorer-> Field Selector** window and repeating the steps taken to display the first total column ozone retrieved pass. The two following Figures show the Data Explorer->Field Selector and the McIDAS-V image window display of an IDEA-I ozone file containing 2 passes captured and processed over the US from Aqua AIRS on 13 December 2013. Once your data is loaded into the McV display, you can change the data range, enhancement, overlay other products, change projections, interrogate data, etc. Please see the documentation that is available for McIDAS-V for more information:

<http://www.ssec.wisc.edu/mcidas/software/v/documentation.html>





Screen capture of the McIDAS-V Data Explorer listing of a IDEA-I ozone gridded output NetCDF file (top), and the image window display (bottom) showing the AIRS retrieved Total Ozone for 2 Aqua passes from observations gathered and processed on 13 December 2013.

Appendix 1: IDEA-I Trajectory Model and References

A1.1 Trajectory Model

The trajectories are predicted using the NASA Langley trajectory code [Pierce and Fairlie, 1993] which solves 3 ordinary differential equations:

$$\begin{array}{rcl} Dx/Dt = U(x,y,p,t) & & 1 \\ Dy/Dt = V(x,y,p,t) & & 2 \\ Dp/Dt = W(x,y,p,t) & & 3 \end{array}$$

at each time step. Here x , y , and p are the zonal, meridional and pressure of the trajectories. U and V are the zonal and meridional components of the velocity field evaluated on a pressure surface and W is the pressure vertical velocity. A 4th order Runge-Kuta implicit integration scheme [Austin and Tuck, 1985] was used to integrate equations 1-3 with three-dimensional winds which are linearly interpolated in space and time from GFS forecasts which are stored at 6 (3)-hour intervals.

A1.2 References

Al-Saadi, J. A. et al., Improving National Air Quality Forecasts with Satellite Aerosol Observations, BAMS, DOI:10.1175/BAMS-86-9-1249, 2005.

Austin, J. and A . F. Tuck, The calculation of stratospheric air parcel trajectories using satellite data., Q. J. R. Meteorol Soc., 111, 279-307, 1985.

Pierce, RB Fairlie, TDA, Chaotic Advection in the Stratosphere - Implications for the Dispersal of Chemically Perturbed air From the Polar Vortex, J. Geophys. Res., 98, 18589-18595, 1993.